

English



Fujitsu Software BS2000

IMON

Installation Monitor

User Guide

Valid for:
IMON 3.4A

Edition June 2026

Comments... Suggestions... Corrections...

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to: bs2000.info@fujitsu.com.

Certified documentation according to DIN EN ISO 9001:2015

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2015.

Copyright and Trademarks

Copyright © 2026 Fujitsu

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

Table of Contents

imon_en_250305	8
1 Preface	9
1.1 Brief product description	10
1.2 Target group	11
1.3 Notational conventions	12
1.4 Summary of contents	13
1.5 Changes in IMON V3.4	15
1.5.1 Changes compared to the previous edition	16
2 Delivery, parking, installation	18
2.1 Prerequisites and defaults	19
2.2 Delivery	21
2.3 Parking	25
2.4 Installation	27
2.4.1 Installation preparation	28
2.4.2 Installation procedure	31
2.5 Postprocessing (product-dependent)	33
2.6 Structure and attributes of the supply elements	34
2.6.1 Delivery	35
2.6.2 Supply unit	37
2.6.3 Supply group, release unit	39
2.6.4 Supply component, release item	40
2.6.5 Installation unit (IU)	41
2.6.6 Installation item (II)	45
2.6.7 Path name, logical ID and type of an installation item	48
2.7 Install on the basis of customer approval IDs	51
2.8 Deinstallation	54
2.9 Undo - undoing an installation	57
2.10 Dynamic activation	59
2.11 Request correction delivery	65
3 Installation sequences under BS2000	67
3.1 Default installation BS2000	68
3.1.1 Installation on the home pubset	72
3.1.1.1 Example in menu mode	73
3.1.1.2 Example with SDF statements	85
3.1.2 Installation on the imported pubset	91
3.1.2.1 Example in menu mode	92
3.1.2.2 Example with SDF statements	104

- 3.1.3 Placement and activation in the default installation 109
- 3.2 Customer-specific installation under BS2000 114**
 - 3.2.1 Installation on the home pubset with customizing 119
 - 3.2.1.1 Example in menu mode 120
 - 3.2.1.2 Example with SDF statements 138
 - 3.2.2 Installation with prior parking of the software 148
 - 3.2.2.1 Example in menu mode 149
 - 3.2.2.2 Example with SDF statements 163
 - 3.2.3 Multiple installation of parked software 171
 - 3.2.3.1 Example in menu mode 172
 - 3.2.3.2 Example with SDF statements 183
 - 3.2.4 Multiple installations of software, which is already installed 186
 - 3.2.4.1 Example in menu mode 187
 - 3.2.4.2 Example with SDF statements 198
 - 3.2.5 Placement and activation in the customer-specific installation 201
- 4 IMON functions and interfaces (IMON-BAS) 207**
 - 4.1 Starting and terminating IMON-BAS 210**
 - 4.2 Working with the menu interface of IMON-BAS 212**
 - 4.3 Description of the IMON-BAS menu options 216**
 - 4.3.1 File menu 218
 - 4.3.2 Edit menu 229
 - 4.3.3 Show menu 264
 - 4.3.4 View menu 283
 - 4.3.5 Options menu 290
 - 4.4 The IMON-BAS statements 292**
 - 4.4.1 ACTIVATE-UNITS Dynamically activating installed software 295
 - 4.4.2 ADD-INSTALLATION-UNITS Register installed software 300
 - 4.4.3 CHECK-UNITS Check installed software 304
 - 4.4.4 DEINSTALL-SUPPLY-UNITS Deinstall software 309
 - 4.4.5 GENERATE-IDF Export SCI entries using IDF 316
 - 4.4.6 INSTALL-UNITS Install and register software 323
 - 4.4.7 MODIFY-IMON-OPTIONS Modify IMON parameter settings 350
 - 4.4.8 PARK-UNITS Park Software 353
 - 4.4.9 PRINT-DOCUMENTATION Print delivery documentation 362
 - 4.4.10 REMOVE-INSTALLATION-UNITS Remove installation units from SCI 367
 - 4.4.11 REMOVE-PACKAGES Deleting a software delivery from an SCI 370
 - 4.4.12 REMOVE-SUPPLY-UNITS Remove supply units from SCI 373
 - 4.4.13 REQUEST-CORRECTION-DELIVERY Request correction deliveries for
registered supply units 376
 - 4.4.14 RESET-CUSTOMER-APPROVAL Reset customer approval ID of a supply unit
..... 381

4.4.15 SAVE-SOFTWARE-INVENTORY Copy current SCI	383
4.4.16 SET-CUSTOMER-APPROVAL Set customer approval ID for a supply unit	384
4.4.17 SHOW-FORMATTED-FILE Show formatted file	386
4.4.18 SHOW-INSTALLATION-ITEMS Show installation items	388
4.4.19 SHOW-INSTALLATION-UNITS Show installation units	397
4.4.20 SHOW-PACKAGES Output information on software deliveries	403
4.4.21 SHOW-SUPPLY-UNITS Show supply units	410
4.4.22 SWITCH-TO-FHS Switch to FHS interface of IMON	419
4.4.23 UNDO-SUPPLY-UNITS Undo installation	420
4.5 The IMON-BAS macros	423
4.5.1 IMOSHFF Show formatted file	424
4.5.2 IMOSHII Show installation items	426
4.5.3 IMOSHIU Show installation units	432
4.5.4 IMOSHSU Show supply units	437
4.5.5 Example illustrating output with macros	443
5 Administering software	445
5.1 The files of IMON	446
5.1.1 Software Configuration Inventory (SCI)	447
5.1.2 IMON parameter files	449
5.1.2.1 IMON parameter file for optional installation functions	450
5.1.2.2 IMON parameter file for default installation parameters	454
5.1.2.3 Parameter file for mail configuration	456
5.1.3 Structure and Installation Information file (SYSSII file)	457
5.1.4 Installation Definition File (IDF file)	458
5.1.5 Installation definition file (IDF file, internal format)	461
5.1.6 IMON reference files	469
5.2 Administering path names and product versions (IMON-GPN)	471
5.2.1 Example with commands	474
5.2.2 The macros of IMON-GPN	476
5.2.3 GETINSP Get installation path	477
5.2.4 GETINSV Get version of installation unit	481
5.2.5 GETPROV Get selected product version	485
5.2.6 SELPROV Select product version	487
5.2.7 SETINSP Enter or modify installation path	490
5.2.8 Example with macros	494
5.3 Administering installation units	497
5.3.1 Registering installation units in the SCI	498
5.3.2 Exporting and importing SCI entries	499
5.3.3 Removing installation units from the SCI	500
5.3.4 Fetching information from the SCI	501
5.3.5 Removing obsolete information from the SCI	503

6 Working with IMON-SIC	505
6.1 Starting and terminating IMON-SIC	506
6.2 The menu functions of IMON-SIC	508
6.2.1 Mask layout	509
6.2.2 The sequence of masks	511
6.2.3 Description of the menu functions of the main menu	512
6.2.4 Open, display or check SYSSII file	513
6.2.5 Create SYSSII file	515
6.2.6 Example with masks from the main menu	517
6.2.7 Menu functions of the SYSSII menu	525
6.2.8 Modify release unit attributes	526
6.2.9 Modify release unit structure	527
6.2.10 Modify release item attributes	529
6.2.11 Add logging information	533
6.2.12 Erase logging	534
6.2.13 Show SYSSII file	535
6.2.14 Save SYSSII file	536
6.2.15 Return to main menu	537
6.3 IMON-SIC statements	538
6.3.1 CHECK-SII Check contents of SII file	540
6.3.2 SHOW-SII Output contents of SYSSII file	541
6.4 The macros of IMON-SIC	543
6.4.1 IMOKIT Read information from SYSSII file	544
6.4.2 IMOKIA Read comments, dependencies and parameter text	550
6.4.3 IMOKIS Output contents of SYSSII file	554
6.5 The C programming interface of IMON-SIC	559
6.5.1 IMOKIT Read information from SYSSII file	561
6.5.2 IMOKIA Read comments, dependencies and parameter text	563
6.5.3 IMOKIS Output contents of SYSSII file	565
7 Troubleshooting	567
7.1 The troubleshooting process	568
7.2 Contents and structure of the console log	572
7.3 Restarting an aborted installation	575
7.4 Error handling and activation restart	576
7.5 Troubleshooting at package installation of POSIX products (item types *PS, *NP)	579
8 Installing IMON	580
9 Appendix	584
9.1 SDF syntax representation	585
9.2 SDF command return codes	601
9.3 Macro syntax	602

9.4 Installation sequence plans **608**
9.4.1 Default installation on the home pubset 609
9.4.2 Default installation on an imported pubset 611
9.4.3 Customer-specific installation on the home pubset with customizing 613
9.4.4 Customer-specific installation with prior parking 615
9.4.5 Sequence of a customer-specific multiple installation of a parked delivery .. 617
9.4.6 Sequence of a customer-specific multiple installation of a previously installed
delivery 618
10 Glossary **619**
11 Abbreviations **624**
12 Related publications **625**

imon_en_250305

i Successful execution of the Conditional Content function!

1 Preface

The IMON (Installation MONitor) product installs software products (software for short), records it in a Software Configuration Inventory (SCI), and administers it. This manual describes the functional scope and method of operation of IMON and its three components:

- IMON-BAS (Basic System)
- IMON-GPN (Installation Path Manager)
- IMON-SIC (Structure Information Creator)

1.1 Brief product description

IMON installs and administers software configurations. The software products to be found on a typical mainframe system are many and varied – a situation not infrequently rendered even more complex by a multiplicity of versions and correction releases. IMON offers outstanding flexibility for configurations which change frequently and boosts the costefficiency of computer centers.

Thanks to SOLIS2 (software delivery and information system), the BS2000 software products to be installed have a unified structure so that they can be installed by IMON in a standard procedure.

Once installation is completed, IMON registers the installed software in a Software Configuration Inventory (SCI) containing information on the locations of the various software modules in the system. IMON administers the SCI.

IMON provides a range of related services:

- for automatic installation of software for BS2000
- automatic deinstallation of software that is no longer needed for BS2000
- an Undo function for the last installation performed
- to implement central administration of all information relating to installation in a SCI
- to “park” the software you receive, i.e. to temporarily store the files under a particular user ID prior to installation
- to print the delivery documentation
- to request information on available corrections or on a correction delivery
- to search the SCI and retrieve information for users and for program-related processing
- to decouple path names and logical IDs
- for parallel administration of coexistent product versions – including correction versions – in a system.

IMON is executable in interactive and batch modes.

IMON offers the following interfaces:

- an SDF interface for the IMON-BAS and IMON-GPN components for entering statements and commands
- a menu interface for the IMON-BAS and IMON-SIC components
- a subroutine interface for IMON-BAS, IMON-GPN and IMON-SIC.

The functional scope depends partly on the privileges assigned to the user.

1.2 Target group

This manual is intended for BS2000 system administrators who want to use IMON for software installation, to register software already installed in the system and to remove administrative information about software that is no longer needed.

Nonprivileged users can employ IMON to obtain information on the software to which they have access.

You should be familiar with BS2000 and the BS2000 components if you want to install software. See the “Introductory Guide to Systems Support” manual for more details [1].

You should also be familiar with the BS2000 command language SDF (System Dialog Facility) and know the major SDF commands. Your sources of more detailed information on SDF are the manuals “SDF Dialog Interface” [2] and “Commands” [3].

You should also be familiar with the FHS masks and a menu-driven interface. See the manual “FHS” [7] for details.

1.3 Notational conventions

The following notational conventions are used in this manual:

In the examples, the user input is represented as **demibold typewriter text** and system output in `typewriter text`.

The notational conventions of the SDF statements, the return codes of the SDF commands and the metasyntax of the macros are explained in detail in the appendix.

1.4 Summary of contents

This manual consists of 9 chapters, the Appendix, and various lists. This overview provides an approximate description of the contents of the individual chapters.

Chapter: *Delivery, parking, installation*

This chapter contains a general description of the sequence of a product installation:

from assembling and delivering the product components through parking, installing and postprocessing to starting the installed product.

It also describes the structure and attributes of the delivered elements. Installation based on customer approval IDs, the deinstallation procedure, the Undo function and the requesting of correction deliveries are described at the end.

Chapter: *Installation sequences under BS2000*

The chapter describes how to install software using IMON under BS2000.

The sequences of sample installations divided into default and customer-specific installations are documented.

Chapter: *IMON functions and interfaces (IMON-BAS)*

This chapter is an introduction to working with IMON-BAS. It shows how IMON is started and the functions offered by IMON-BAS via menu options, SDF statements, and macro calls.

Chapter: *Administering software*

This chapter describes the files used by IMON to administer the software. It also describes how IMON administers path names and installation units. Administration of the path names with IMON-GPN commands is illustrated by an example. The IMON-GPN macros are described.

Chapter: *Working with IMON-SIC*

This chapter contains an introduction to working with IMON-SIC. Here you find out how IMON-SIC is started and the menu functions and subroutine calls (Assembler and C interface) it offers.

Chapter: *Troubleshooting*

This chapter describes how IMON informs the user of errors that occur during installation.

It gives information on how to analyze the error and eliminate it as well as describing the most important files for installation (log files and backup files). It describes the contents and structure of the console log and the circumstances under which you can restart an aborted installation procedure.

Chapter: *“Installing IMON”*

This chapter describes the hardware and software required to run IMON and describes how IMON is installed.

Chapter: *Appendix*

The appendix contains the following tables and overviews:

- the syntax description for SDF statements and SDF commands
- the conventions for command return codes
- the metasyntax for macro calls
- the flow charts for the installation sequences for BS2000 (see chapter [\(Installation sequences under BS2000.\)](#))

Indexes

The rear of the manual contains these indexes in the following order:

- Glossary
- Abbreviations
- Related publications

The complete text of each publication that is referenced in the text by a number is listed in the “Related publications” section alongside its relevant number. This index also contains information on ordering publications.

1.5 Changes in IMON V3.4

This manual describes the functionality of the installation monitor, IMON V3.4, which can be used for installations with target versions BS2000 V21.0 and higher.

In the following section, the changes to the IMON components compared to IMON V3.2 and V3.3 are described.

1.5.1 Changes compared to the previous edition

IMON-BAS

The treatment of POSIX files has been enhanced.

- New item type for POSIX based products (*NP):

The item type *NP is a combination of the item types *PS and *NW. It has been introduced for products for which the deinstallation in POSIX must be run with the old installation libraries ("SINLIBs") also for the same main version.

As for the type *PS, the type *NP describes the POSIX-satellites that can be deinstalled and installed automatically in the frame of the IMON installation in POSIX (via "POSIX Processing").

The processing of the item types *PS and *NP differs only when the main version of the updated product remains the same.

In this case the following steps are schematically processed for *NP items:

1. Creation of the new product files ("SINLIB") with the suffix .NEW
2. POSIX package deinstallation on basis of the old product files
3. Copy of the .NEW files on the files without suffix
4. POSIX package installation on basis of the new product files

In contrast the deinstallation of *PS items for same main version is run on basis of the new product files.

- New values for POSIX processing parameter

"POSIX processing" now offers the new options "ALL" and "BY-CONFIGURATION-FILE" in addition to existing ones YES and NO.

This choice will determine further processing:

- NO=

no POSIX installation (as it is today)

- ALL=

POSIX installation for all POSIX products inside the delivered package (*PS and *NP), no matter whether they are already registered to \$SYSROOT.POSIX.CONFIGURATION or not.

- YES=

as ALL (for compatibility reasons)

- BY-CONFIGURATION-FILE=

POSIX installation only for those POSIX products inside the delivered package (*PS and *NP), which are already registered to \$SYSROOT.POSIX.CONFIGURATION.

The default value is BY-CONFIGURATION-FILE for products which contain POSIX installation items (i.e. with item type *PS or *NP). Products that are not yet installed in POSIX (with the same or other main version) are therefore not installed in POSIX per default.

It is recommended to select the specific products to install according to the needs and to install them e.g. in the guided dialog of the POSIX installation program.

A first installation can also be forced per supply unit by changing the default to "ALL". However the file `$$SYSROOT.IMON.ACTIONS.ADD` then contains all the POSIX packages of the supply unit.

If not all the packages must be installed in POSIX, they can be explicitly removed with EDT from the ADD file after the run of the IMON installation procedures (removal of the corresponding line).

- **Install BY-CONFIGURATION**

Install all supply units based on configuration.

All supply units from delivery will be installed if they are in higher or equal version than already installed in system.

If supply unit was not in system, then it's installation will be skipped.

- The meaning of "version" in the SDF statements and in the FHS masks was changed in IMON.

Instead of the OSD version, the BS2000 version is now used in all instructions. Entering the previously used OSD versions is supported in a compatible manner.

Overview of the corresponding version names:

OSD version	BS2000 version
10.0	19.0
11.0	20.0
-	21.0

IMON-GPN

GETINSV Get version of installation unit.

- Macro return code: Too many subsystem versions

2 Delivery, parking, installation

This chapter contains a general description of the product installation procedure: from the assembly and delivery of the product components through to their parking, installation, and postprocessing, and finally starting the installed product.

It also describes the structure and attributes of the distributed elements followed by a description of how to remove products not needed any more from the system (deinstallation) as well as of how the last installation of a product can be undone (Undo function).

General sequence

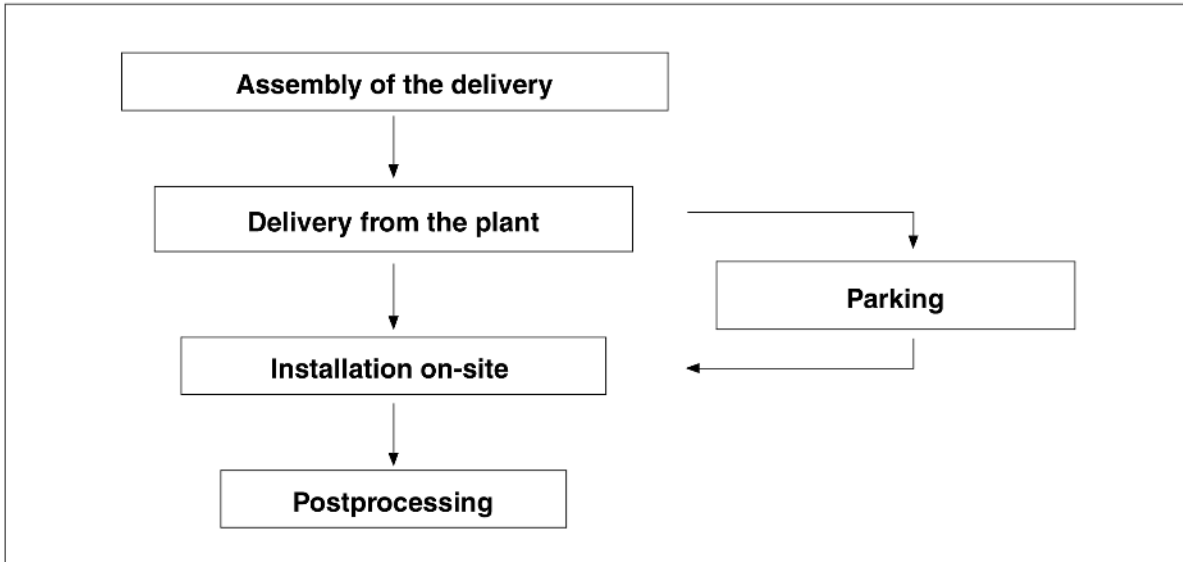


Figure 1: General flow

The product is assembled by Fujitsu in accordance with the customer's wishes.

It is then forwarded either on a data volume (MTC, CD, DVD) or without a medium (via file transfer or download) to the customer who parks the software in their system or installs it immediately.

Postprocessing may be required for some products.

2.1 Prerequisites and defaults

Product installation using IMON V3.4 is possible with BS2000 V21.0 or later.

Prerequisites for the supply unit to be installed can be found in the accompanying documentation.

The pubset that is to be the basis for the installation must be imported into the system.

Target system

The target system is the system on which the software product will run. This could be the most recent BS2000 version (installation on the home pubset) or another BS2000 version (installation on the imported pubset).

Identifiers required

Various user IDs (see below) are used during the installation. They must be set up already and have sufficient memory space. The number of PAM pages supplied is detailed in the supply information.

- Work file ID

All files that are needed temporarily in order to install the software are stored under this user ID. This includes files that are part of the delivery as well as files that are generated by IMON during installation.

The default is the user ID SYSSAG. This is also required when the product is distributed via file transfer.

The IMON options can also be used to define a different user ID or an additional file name prefix.

If the SYSSAG user ID is not set up, the TSOS user ID and the IMON file name prefix are assumed.

IMON automatically deletes some of the work files after installation.

As the SYSSAG work ID is not required in normal BS2000 operation, the default pubset of this ID can also be a switchable pubset. However, during parking and installation exclusive access of the system to the default pubset of the work ID must be guaranteed.

- Installation ID

The files of the software to be installed are stored on the system under this user ID (see "[Placement - specifying the definitive depot location](#)" ([Installation preparation](#))).

The installation ID is thus part of the path name under which the installation items are registered in the SCI.

- Park ID (is only required if the delivery is to be parked)

All files of the software to be installed later that are contained on the distribution medium are transferred to this user ID.

The files can be deleted automatically by IMON after a successful installation procedure. The automatic deletion option can be deactivated if you intend to carry out another installation. The files then remain under the park ID for other installations.

[:<catid>:][<userid>]

<catid> refers to the catalog ID of the pubset on which the software is parked. If no catalog ID is specified, the software is parked on the default pubset of the specific park ID.

<userid> refers to the park ID. The character string \$. refers to the system default ID (DEFLUID system parameter).

System privileges

The SUBSYSTEM-MANAGEMENT and USER-ADMINISTRATION privileges are needed for parking and installing the product. Moreover, you should work under the user ID TSOS as some privileged function calls are required (e.g. to store files under other user IDs).

Private software

IMON supports the installation of software that was delivered by Fujitsu using the SOLIS2 supply system. Private software cannot be installed using IMON, however the supply components can be registered and administered in the SCI. The registration of private software is dealt with in detail in the [section "Administering installation units"](#).

Language setting for the FHS interface

The FHS menu interface is available in English and German on data display terminals of the 3270, 9750, 9755, 9763, 8160 and equivalent types.

You can select either of the languages, and the setting is made in either a task-specific or user-specific manner. (The language can naturally also be set for the entire system using the MSGLPRI system parameter).

Task-specific: `MODIFY-MSG-ATTRIBUTES TASK-LANGUAGE={D/E}`

User-specific: `ADD-USER/MODIFY-USER-ATTR . . . ,DEFAULT-MSG-LANGUAGE={D/E}`

2.2 Delivery

Before a software product can be delivered and installed, all of the requisite files must be available on a distribution medium. For BS2000 products, this supply medium is usually a magnetic tape cartridge (MTC), a CD or a DVD in ARCHIVE format, or a PLAM library that is transferred to the customer's system by means of file transfer.

IMON extracts the files of the delivered software product from the data volume (using the ARCHIVE or SIR utilities) or from the PLAM library (using the LMSCONV utility) when you call one of the IMON "Park" or "Install" functions. If you call the IMON "Print documentation" function, IMON extracts the supply documentation from the data volume.

A delivery consists of one or more supply units (SU) and the accompanying supply information.

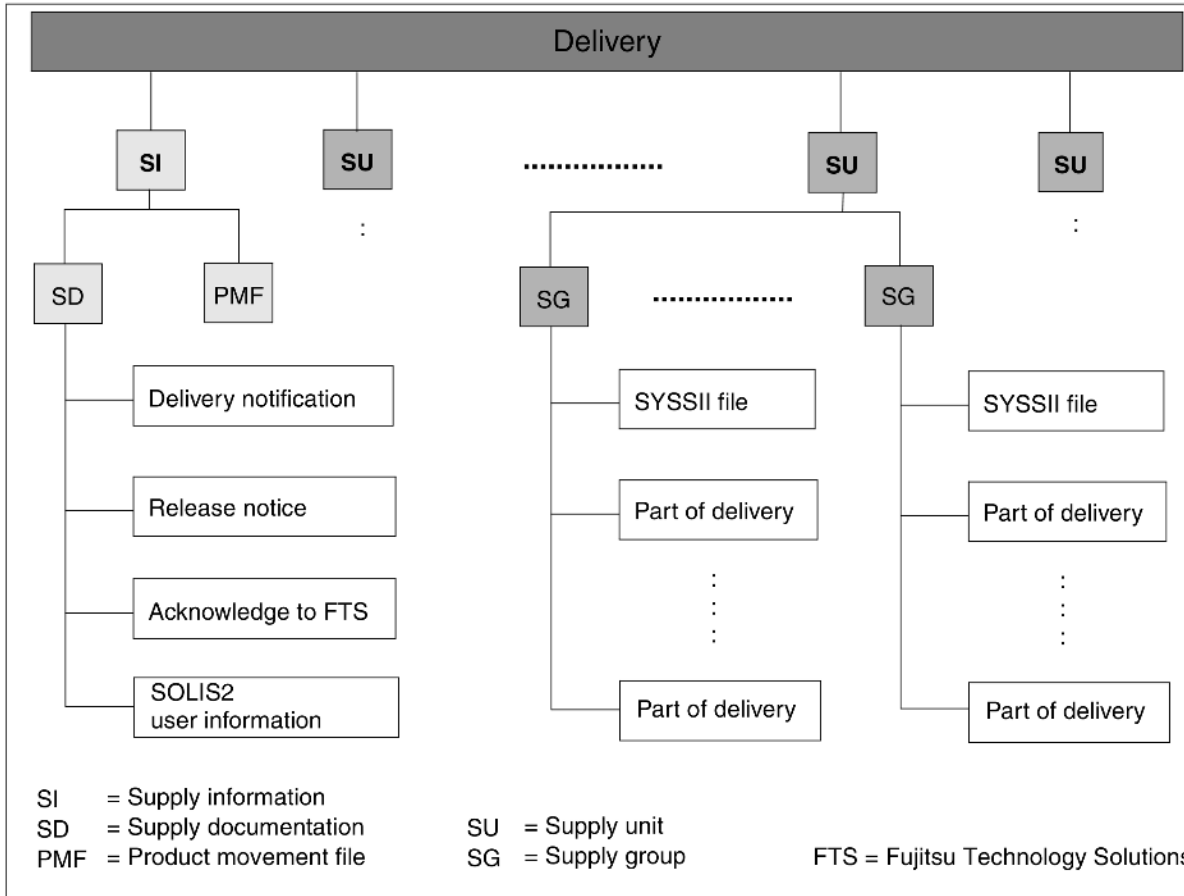


Figure 2: Delivery structure

The supply units

A supply unit consists of one or more supply groups (SG). A supply group can, in turn, consist of one or more supply components (SC).

These supply components are the individual files (e.g. the object module, the syntax files, the REP file, etc.) that make up a product. The product itself (e.g. SDF-CONV) is represented by the supply group. The supply unit in this case would be BS2OS.SDF.

The smallest unit that can be administered by IMON is the supply component.

The smallest unit that can be installed by IMON is the supply unit.

The term release unit (RU) is used as a synonym for supply group.

The term release item (RI) is used as a synonym for supply component.

The supply information

The supply information consists of the supply documentation and the so-called product movement file.

1. The supply documentation is made up of the following three printable documents:

the supply information, release notices and acknowledgement to Fujitsu.

- a. The **supply information** (accompanying information) contains important options relating to the software delivery and provides information on the following:

- the nature of the delivery (e.g. correction)
- the package name

This sort term is generated automatically for each software delivery from the day of the year and a serial number. The customer-specific code (customer or equipment ID) is part of the SOLIS2 file name of the delivery.

- VSN of the data volume if the software is supplied on MTC, CD or DVD
- list of the supply units and break-down into supply components
- number of PAM pages required per supply unit
- installation information and notes on technical dependencies
- one delivery note per supply unit

- b. All of the release notices for the supply units are combined in one library.

- c. The acknowledgement sent to Fujitsu Systems Support means that the customer confirms that the delivery was received. Please fill out the acknowledgement form and sent it to Fujitsu.

All of the documentation or parts thereof can be printed out using the “*File: Print documentation*” menu function or the PRINT-DOCUMENTATION statement.

The delivery must first be opened. The information on the package name, customer ID, VSN and device type of the data medium from the delivery notification are required.

If the delivery is opened for the first time when it is being printed, it is registered in the SCI, and the supply documentation it contains is transferred to the documentation library.

Note

Do not ignore the notes on earlier deliveries in the accompanying document. Failure to comply with these notes could result in inconsistencies that can only be rectified by reordering all of the affected products.

2. The **product movement file** is the second most important component of the supply information.

It is evaluated by IMON during installation. It contains information relating to the extraction of the supply components from the 032 or from the library as well as the information from the SYSSII files.

Fujitsu Technology Solutions creates one SYSSII file per supply group/release unit. It contains, among other things, information on the structure of the supply group/release unit, as well as information on the allocation of path names to the logical names of the product components.

The product movement file contains everything that appears in all of the SYSSII files. IMON only evaluates the product movement file to create the product structure and enter the data in the SCI. The SYSSII files are also included in the delivery and are transferred to the target system first. However, they are deleted from the target system after they have been merged in the PLAM library SOLLIB.IMON.SYSSII using the work file ID.

You can suppress transfer of the SYSSII files to the target system if you use an IMON parameter file.

Initial, correction and delta deliveries

A delivery containing units supplied for the first time is known as an initial delivery.

If the supply units are already at your disposal and you receive a current delivery which consists only of corrections or changes, the delivery is known as a correction delivery. Refer to [section "Request correction delivery"](#) for requesting information on available corrections or on a correction delivery.

A delta delivery is a special form of correction delivery containing only new or changed supply components differing from those you received in the last delivery.

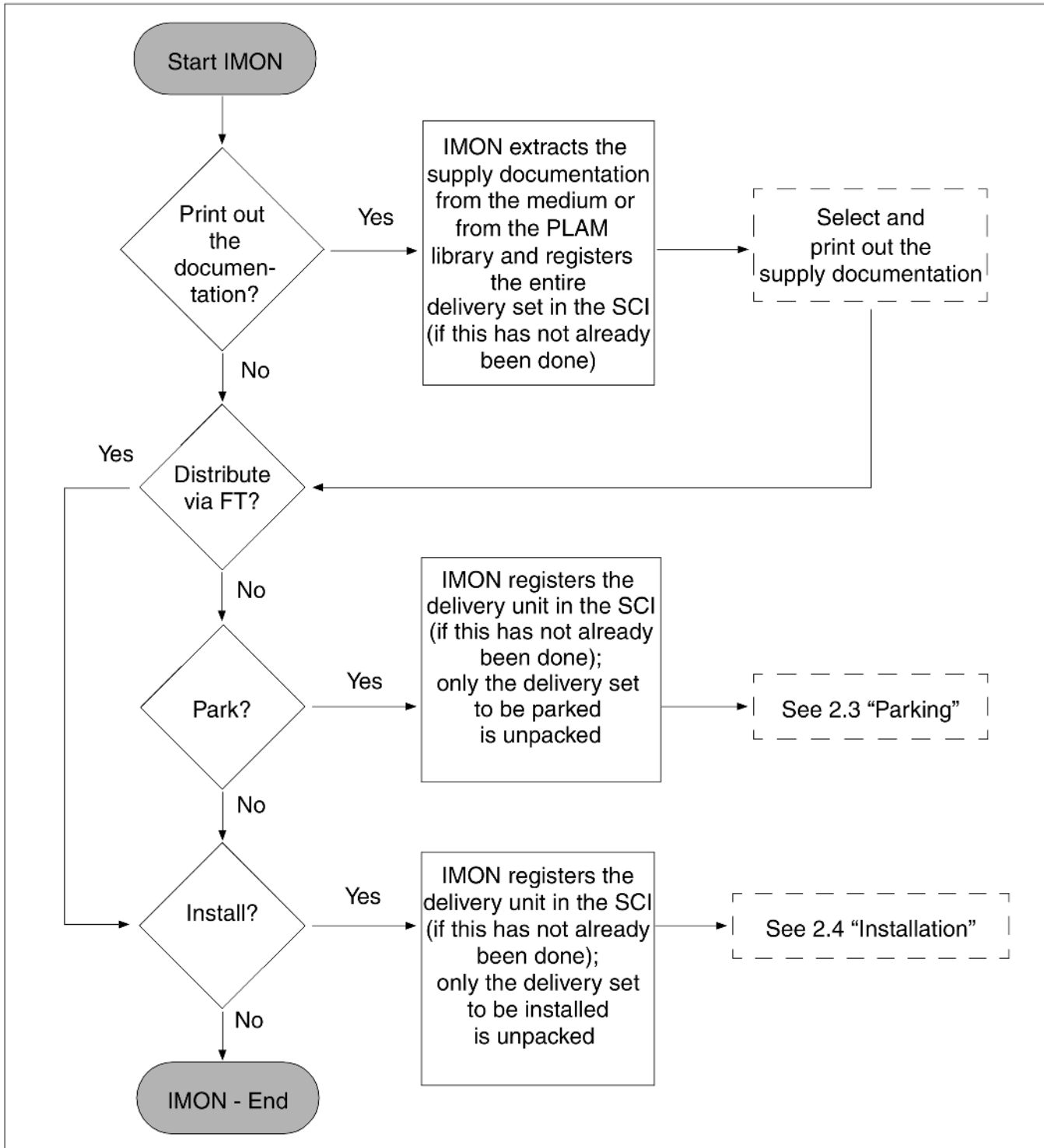


Figure 3: "Unpacking" the delivery

2.3 Parking

The software distributed on the can be stored temporarily ("parked") under a specified user ID (**park ID**) prior to installation. The target system (BS2000 version and pubset) must be specified. The defaults are the BS2000 version and the home pubset of the system that is running.

The park function is not offered if the software was distributed via file transfer, as in this case the entire delivery set is already on the customer's system.

The SOLIS2 delivery and the Software Configuration Inventory (SCI) must be opened. If an SCI was not explicitly opened, then IMON implicitly uses the standard SCI of the home pubset. If the delivery was already opened, it is already registered in the SCI and can be opened via the *REGISTERED-MEDIUM option.

The software is stored intermediately using the IMON function "Park" or the PARK-UNITS statement. This does not interrupt the IMON session.

- All of the files in the delivery or selected parts of the delivery are transferred to the park ID from the distribution medium.
- The supply documentation is stored in a library.
- At the same time, the transferred supply units are registered in the SCI as parked.
- Specifications relating to the target system, the backing up of existing files that are overwritten as a result of parking, and the updating of the RMS depot (transfer of system corrections) can be entered as early as during the parking stage.

The files remain under the park ID until they are installed using the IMON installation function. After successful installation, the files can be released by the park ID for automatic deletion or be retained for a further installation.

Any files that are no longer needed can be deleted with the cleanup procedure \$<park ID>.IMONDEI.PRK.<name of supply unit>.<package name> that is generated after a successful park. This procedure contains all DELETE-FILE commands required to delete the parked supply units.

For more information on the park ID, see also "[Prerequisites and defaults](#)".

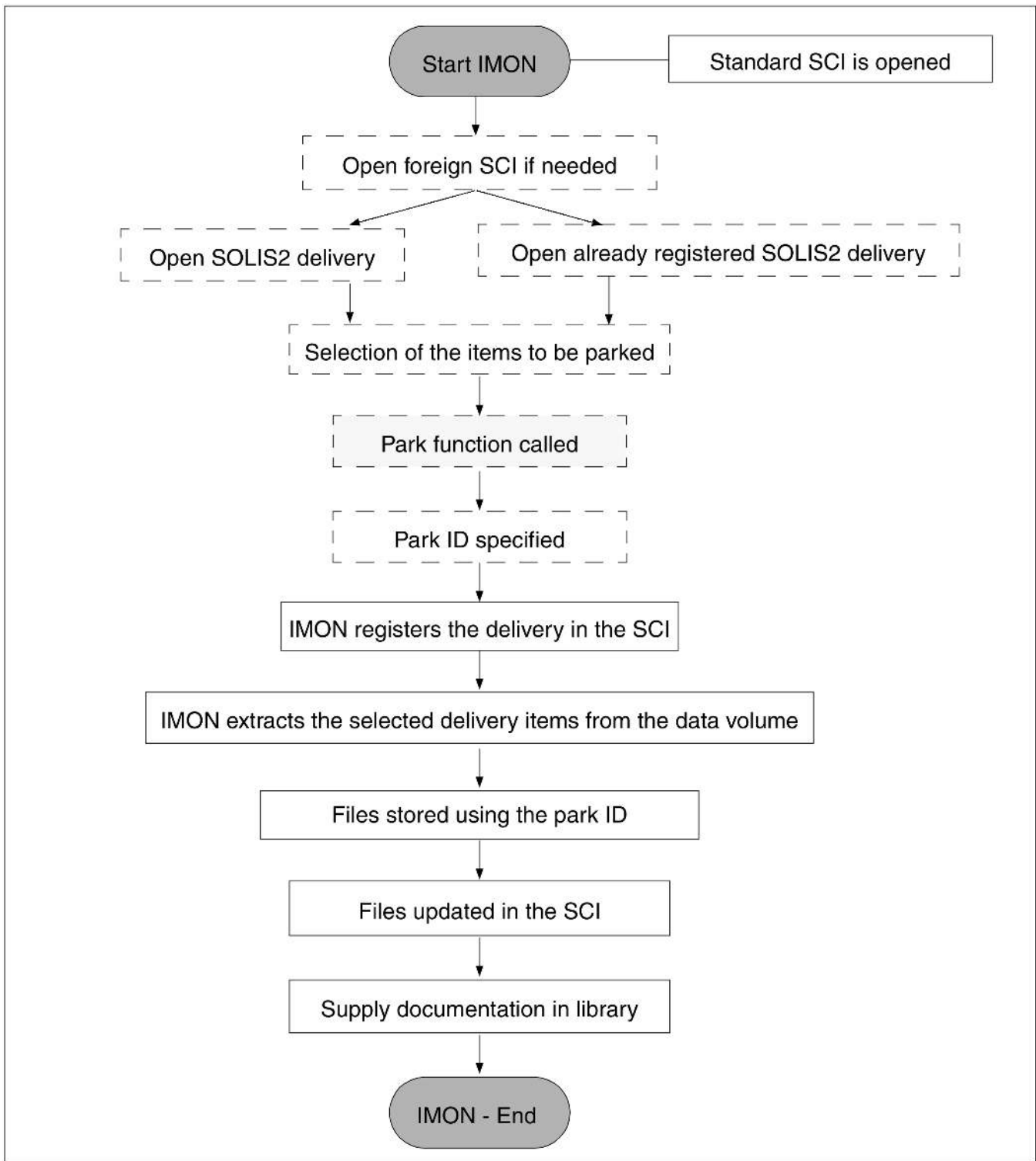


Figure 4: Parking

2.4 Installation

During installation, the supplied software is brought to the target system (depot location) from the distribution medium or the park ID. The depot locations provided can be customized to the configuration of the customer's system. During this process the release units and the release items they contain are converted to installation units and installation items and installed in the target system.

The SOLIS2 delivery and the Software Configuration Inventory (SCI) must be open. If an SCI was not explicitly opened, IMON implicitly uses the standard SCI of the home pubset.

There are two different installation types for BS2000 software:

- Default installation

With this type of installation, the entire delivery is stored on the home pubset or on an imported pubset under the user ID specified by Fujitsu. Installation on the home pubset is the installation variant most frequently used.

- Customer-specific installation

The following are examples of customer-specific installations: installations under different user IDs; installations where the software is first parked; multiple installations from a park ID or from software which is already installed.

The first time installation and migration of the operating system are described in the manual "System Installation" [4].

Information on installing private software is given in the [section "Structure and Installation Information file \(SYSSII file\)"](#).

The installation process is initiated by the IMON function "Edit: Install" or the INSTALL-UNITS statement. It is divided into two stages:

- > Installation preparation
- > Actual installation process

The result of the installation preparation stage is the generation of a procedure that also contains the special features of the delivery and the customer-specific requirements.

Execution of this procedure then initiates the actual installation process. The delivery documentation is stored in a library. This library is assigned the following name: \$<work file ID>.<package name>.<customer ID>.DOC

Installation procedures can be started selectively.

Software which is parked or already installed can be multiply installed if the erasing of the files is prevented by park and working file recognition.

Supplementary information on overwriting existing installation items, on what to do if the user ID conflicts, on deleting installation files that are no longer required, and on activating special installation items can also be entered.

Optional and special installation functions can be controlled and automated (if desired) using a customer-specific IMON parameter file (see [section "IMON parameter files"](#)).

IMON reads the structural format of the supply units from the product movement file. IMON extracts the release items of the processed supply units from the distribution medium and stores them in the target system as installation items. The file attributes and the path names (file names) of the installation items are formed in accordance with specific rules (see ["Path name, logical ID and type of an installation item"](#)).

2.4.1 Installation preparation

The following steps are part of the installation preparations:

1. Selecting the products to be installed from the delivery set
2. Carrying out system-specific adaptations (customizing)
 - placement – specification of the depot location of the files to be installed
 - activation – preparation for activation of the software to be installed
3. Controlling installation functions using the IMON parameter files
4. Checking the software configuration on the basis of the reference files
5. Generating the procedure

You select the supply units to be installed when calling the installation function from IMON.

IMON together use the other specifications relating to the target system, the depot location for files, and activation of the software to be installed to generate a procedure containing all of the process steps needed to install the selected supply units.

1. Selecting the products to be installed from the delivery set

When you open a delivery (menu mode: *File:Open:SOLIS2 delivery*, SDF statement:

INSTALL-UNITS), you specify whether the delivery should be fetched from the distribution medium itself or from the SCI.

If the delivery was already opened, it is registered in the SCI and can be opened via the “Registered delivery” option (in the *File:Open:SOLIS2 delivery:Registered* menu or using the SDF statement `INSTALL-UNITS ..., DISTRIBUTION=*REGISTERED-MEDIUM`).

If a number of deliveries are affected, they are installed in the order in which they were created. The order is taken from the package names of the individual deliveries. The files in a supply unit are installed together.

IMON will offer a complete or partial installation depending on the selection made when opening the delivery:

- Complete installation

All supply units contained in the delivery are offered by IMON for installation. If a supply unit from this delivery was already registered in the SCI, IMON outputs an appropriate warning when the data volume is opened.
- Partial installation

In this case, only those supply units that were not installed are offered for selection (e.g. when opening a delivery that is already registered and partially installed).

PLAM libraries can be used directly as an input medium for installing software (in the menu: *File:Open:SOLIS2 delivery:Library* and *Edit:Install* or using the SDF statement `INSTALL-UNITS ..., DISTRIBUTION-MEDIUM=*LIBRARY`).

2. System-specific adaptations (customizing)

System-specific adaptations can be made once the set of supply units to be installed has been defined. Global and local installation parameters can be specified for this purpose.

Global parameters apply to all selected supply units, whereas local parameters apply only to specific supply units.

Placement – specifying the definitive depot location

The depot location of the files (installation location) is defined by specifying a catalog ID, and/or user ID, and/or a prefix.

Fujitsu specifies a default depot location for each supply unit. If a default installation is carried out, the installation items are installed in these predefined depot locations (see "[Placement and activation in the default installation](#)"). To install to these predefined depot locations, it is enough to simply leave unchanged the default value 1 of the "Placement mode" in the "Global installation parameters" dialog box in menu mode or the *STD default of the PLACEMENT-MODE operand in the SDF statement INSTALL-UNITS.

However user-specific depot locations can also be selected. To do this, the value 2 is assigned to the "Placement mode" option of the "Global installation parameters" dialog box in menu mode. When working with SDF statements, INSTALL-UNITS ..., PLACEMENT=*BY-DIALOG is specified. IMON branches in both cases to a (further) dialog box in which specifications relating to the depot location can be entered locally, i.e. separately for each supply unit.

Activation – activating the software to be installed

Once the installation items have been placed, IMON performs measures to prepare the activation of special installation items. The relevant preparatory measures should ensure that the installed software runs correctly in the user system from the next system start.

Until then the installed software is locked for system users (see LOCK-PRODUCT-VERSION in the „Commands“ [4] manual).

Even before the actual installation, IMON:

- generates the subsystem catalog
- activates the syntax files
- activates the message files
- stores system corrections (reps) in the RMS depot and creates the corresponding loaders
- makes preparations for adding POSIX satellites to the POSIX system
- starts procedures

To do this, entries must be made in various files (parameter or configuration files). The measures depend on the installation item type and the general conditions of the BS2000 version.

If activation is to be carried out as prescribed by Fujitsu for the individual supply units, it is enough to simply leave unchanged the default value 1 of the "Activation preparation mode" option in the "Global installation parameters" dialog box in menu mode or the *STD default of the ACTIVATION-MODE operand in the SDF statement INSTALL-UNITS.

User-specific measures are introduced in menu mode by specifying the value 2 for the "Activation preparation mode" option. When working with SDF statements, INSTALL-UNITS ..., ACTIVATION-MODE=*PARAMETERS(...) is specified.

IMON branches in both cases to a (further) dialog box in which specifications relating to the activation mode can be entered locally, i.e. separately for each supply unit.

If an installed supply unit is already run in the current system without a break in system running, the user can execute a "dynamic activation". For this, the installation of the supply unit must have been successfully completed and the supply unit must support activation in the current system. A supply unit is dynamically activated with the SDF statement ACTIVATE-UNITS or through the menu *Edit: Activate*.

3. Controlling installation functions via the IMON parameter files

The user can also control installation functions via two different IMON parameter files:

- For optional installation functions a sample IMON parameter file is provided with IMON and is named SYSPAR.IMON-BAS.<version>. The user can copy this file and change the parameter settings in the file to meet his or her requirements. IMON evaluates the parameter file stored with the name SYSPAR.IMON.<user code> under the user ID from which the installation was started. If it does not exist IMON evaluates the file \$TSOS.SYSPAR.IMON.
- On request IMON can also save various installation parameters in a special parameter file (the default parameter file is SYSPAR.IMON.LAST under the user ID from which installation was started). From this parameter file IMON automatically takes parameter values for the corresponding installation parameters and uses them for the current installation. This function enables customers to automate non-standard installations because it is no longer necessary to reenter the same non-standard parameters in the various dialog boxes of the FHS menu interface.

For more detailed information on the IMON parameter file, see [section "IMON parameter files"](#).

4. Checking the software configuration on the basis of reference files

The check is carried out on the basis of the current reference file set in the IMON options (default is \$TSOS.SYS.IMON.SCI.REF on the home pubset). The dependencies of the supply units to be installed on the supply units already are listed in a protocol (see [section "IMON reference files"](#)).

The check can also be suppressed using (Operand CHECK-CONFIGURATION=*NO in the statement INSTALL-UNITS or *Check-Configuration =2 (No)* in the dialog box *Global Installation parameters*).

5. Generating the procedure

The procedure is generated by IMON. To do this, parameters that were assigned values in Step 1 (selection) and Step 2 (customizing) as well as the IMON parameter files (if they were provided in Step 3) are evaluated.

IMON generates the procedure. It contains the work steps needed to install the selected delivery. The structure of the procedure supports a restart/recovery mechanism in the event of errors (see ["Principle sequence of an installation step with error handling" \(The troubleshooting process\)](#)).

Installation procedures can be started selectively.

IMON outputs an appropriate message to confirm generation.

2.4.2 Installation procedure

The actual installation procedure is executed once the generated procedure is started. Only then are the supplied files transferred to their depot location in the system from the distribution medium or the park ID (or from the installation ID in the event of multiple installation of a product already installed). The preparatory measures for running the product are also made by IMON.

The procedure can be started immediately and automatically by IMON, or it can be started later manually by the user with the ENTER-PROCEDURE command. The name of the procedure is created as follows:

:<catid>:\$<work file ID>.<package name>.<time stamp>.IE

Example: :I29A:\$SYSSAG.B4.10MAI10617.MAY101242452010.IE

An installation is concluded successfully once IMON has recorded the installation units in the SCI (see [section "Registering installation units in the SCI"](#)).

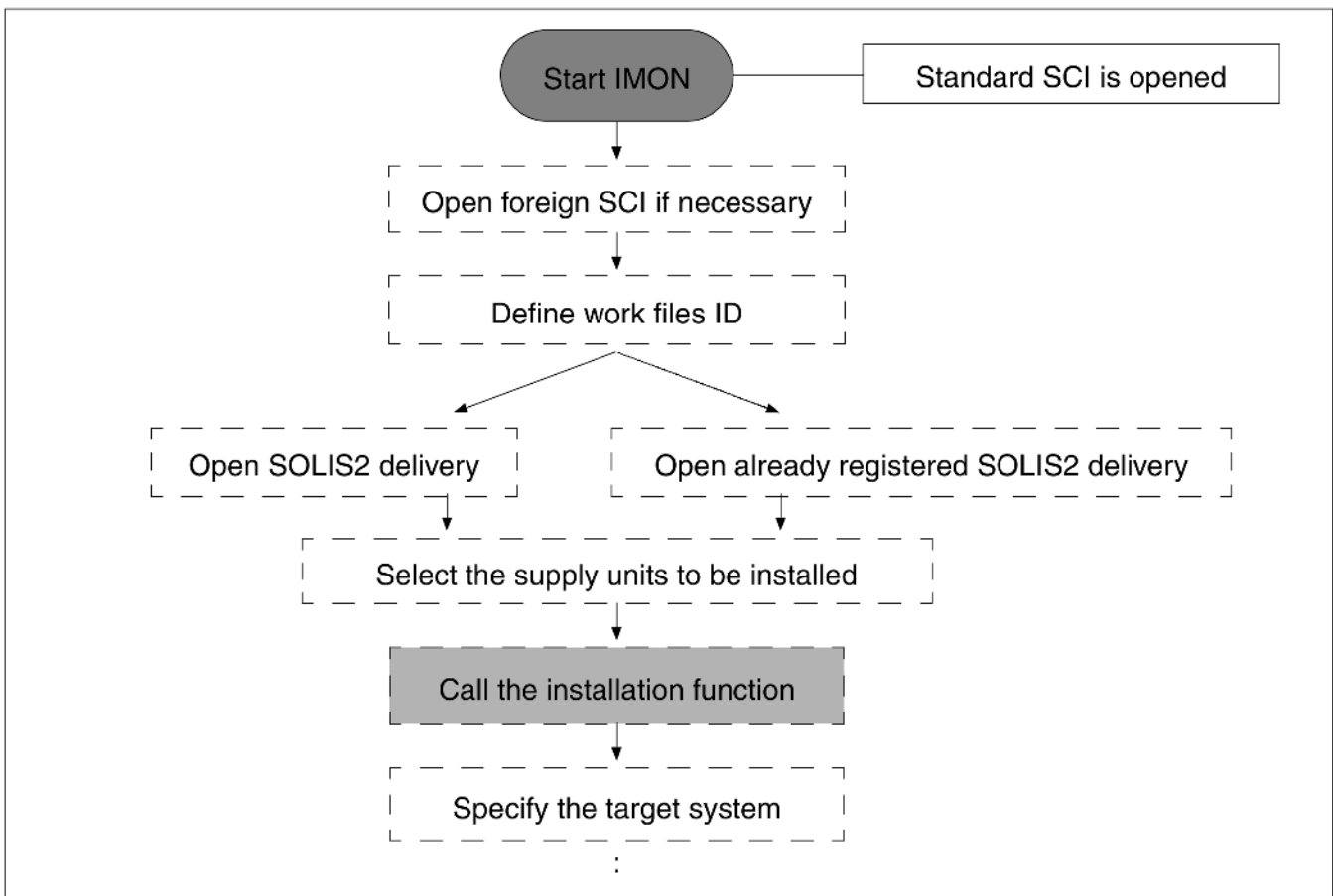


Figure 5: Installation (Part 1)

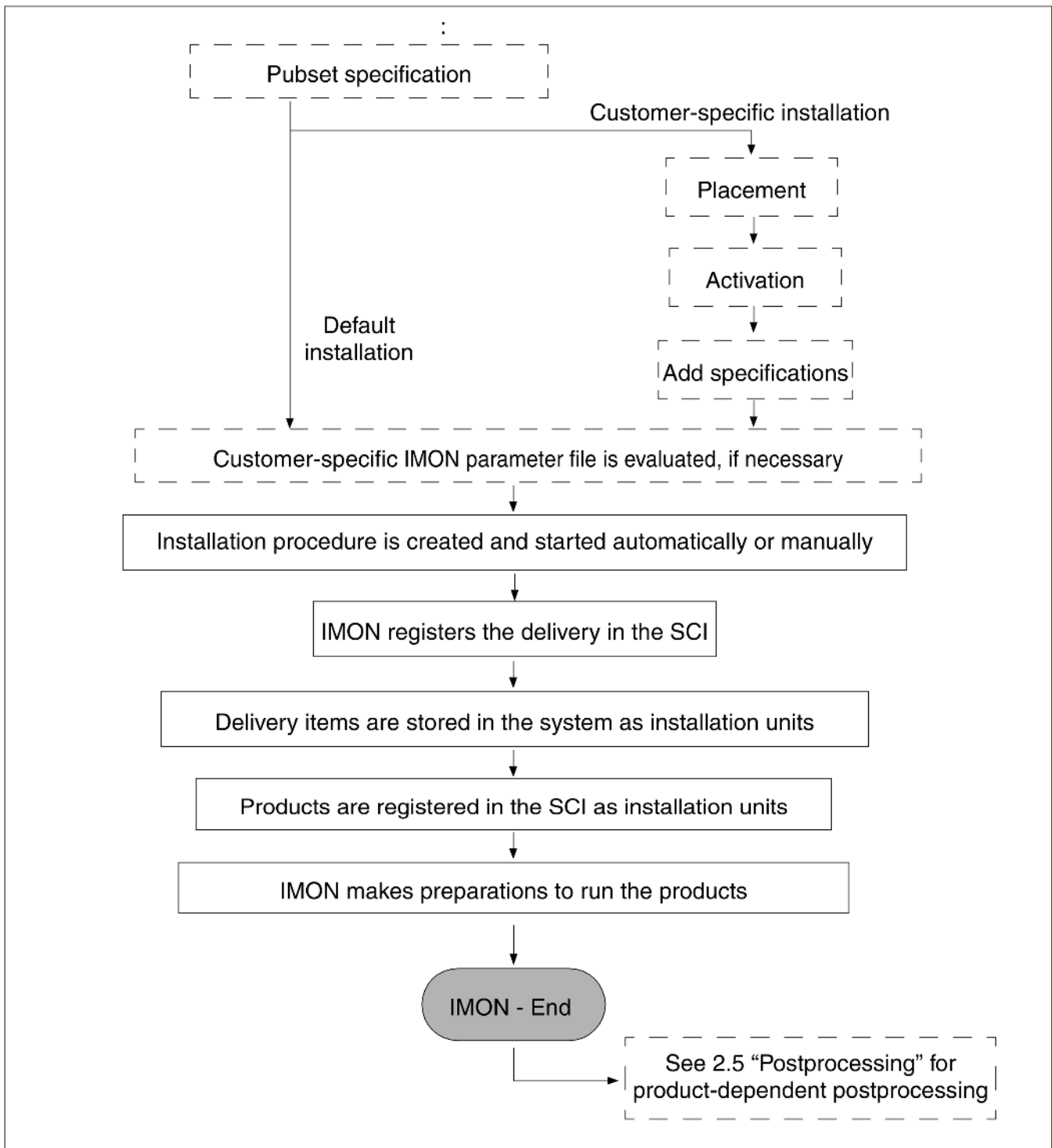


Figure 6: Installation (Part 2)

2.5 Postprocessing (product-dependent)

The files of a product can be modified, moved, and further adapted to the system configuration after installation.

The information in the supply information must be noted for product-dependent postprocessing. In the case of installations on the home pubset, postprocessing can be carried out after IMON is terminated.

A distinction must be made between the following cases when dealing with an installation on an imported pubset:

- The imported pubset is operated as a data pubset of the current system. In this case, the postprocessing can also be carried out immediately after termination of IMON.
- The imported pubset is to be used as the home pubset. In this case, the postprocessing may only be carried out if the system was loaded from this pubset.

Product-dependent postprocessing mostly consists of the processing of specified procedures. These procedures are referred to in the output to the console and in the installation protocol.

The following are other examples of postprocessing:

- customization of the CMDFILE for the BS2000 startup (insertion of new START-SUBSYSTEM commands, deletion of old ones, or modification of commands)
- modification of the subsystem declaration with regard to the address space and customization of the SHRSIZE system parameter
- customization of parameter files
- setup of special user IDs
- renaming of files

Note

Modification of the path name must be recorded by the user using the IMON-GPN functions in the SCI by reassigning the logical name to the path name.

2.6 Structure and attributes of the supply elements

- Delivery (imon_en_250305, #19) 1
- Supply unit (imon_en_250305, #20) 1
- Supply group, release unit (imon_en_250305, #21) 1
- Supply component, release item (imon_en_250305, #22) 1
- Installation unit (IU) (imon_en_250305, #23) 1
- Installation item (II) (imon_en_250305, #24) 1
- Path name, logical ID and type of an installation item (imon_en_250305, #25) 1

2.6.1 Delivery

A delivery consists of one or more supply units and the related delivery information. It is the largest delivered unit, which can be installed with IMON as a whole.

Description of a delivery

A delivery is clearly described by <package-name> and <user-code>.

<package-name> Package name of the delivery in the form of SDF data type <alphanum-name 1..12>, which is mechanically generated at the time of each software delivery from the date and a serial number.

<user-code> Customer specific code (customer or system ID) in the form of SDF data type <alphanum-name 1..8>.

The distribution medium, which contains all the files of the delivery, is clearly described by VSN and device type of the volume, or in the case of a delivery by file transfer, through the filename of the imported PLAM library.

Attributes of a delivery

- Description of the delivery (package name and customer ID)
- Information about the delivery registered in SCI
 - User ID, under which the delivery is registered in SCI
 - Path name, where the working files of the delivery are stored
 - Distribution medium (VSN and device type of the volume, or path name of the PLAM library imported by the file transfer)
 - Creation time of the SCI entry (Date and time)
 - Modification time of the SCI entry (Date and time)
 - Path name of the documentation library, in which the delivery documentation is saved.
- Information on the enclosed supply units (name, version, installation status)
- Number of enclosed supply units

Example of the attributes output of a delivery with IMON

2.6.2 Supply unit

The supply unit is the unit supplied using SOLIS2. It is the smallest supplied unit that can be installed in its entirety by IMON.

Designation of a supply unit

Each supply unit bears a unique designation formed as follows:

<unit-name> <version>

<unit-name> Name of the supply unit in the format of the SDF data type

<structured-name 1..15>

<version> Version of the supply unit. The format is "mm.naso".

mm Main version (01..99).

n Addition version (0..9)

aso Correction state (a=alpha, so=digits (00..99))

Example of the designation for a supply unit

PERCON 02.9C00

Note

For software not supplied by Fujitsu the <unit name> is the name of the supply unit in the form of the SDF data type <text 1..30 without-sep>.

Attributes of a supply unit

- Name and version
- Delivery information (package name and customer ID)
- Installation information (status, creation user ID, date and time of generation and last change)
- Installation counter, which goes up every time a supply unit is installed
- Activation information (shows if dynamic activation is enabled)
- Installation units (name, version)
- Number of installation units
- Customer approval ID

(Timestamp that is set manually using the SET-CUSTOMER-APPROVAL statement. It identifies the supply unit for special installation on the basis of the customer approval ID. The value of the timestamp is 0 if the customer approval ID is reset.)

Example of an attribute listing for a supply unit with IMON

//show-supply-units unit-name=percon,

information=*par(instal-units=*yes,report-level=*all-attr)

=====

Supply Unit = PERCON

Version = 02.9C00

----- Delivery Information -----

Package-Name = <package-name> User Code = SOL2P

----- Installation Information -----

Status = Installed Creation-User-Id = TSOS

Creation-Date = 2021-04-01 Change-Date = 2021-04-01

Creation-Time = 13:56:55 Change-Time = 14:01:18

Installation counter = 00011

----- Activation Information -----

Activable : Yes

----- Customer Approval Information -----

Customer appr.= No

----- Installation Units -----

Name = PERCON

Version = 02.9C00

(1 unit)

2.6.3 Supply group, release unit

The supply groups (SGs) are also known as release units (RUs). A particular release unit may be a component of more than one supply unit.

A release unit consists of one or more technical units, for example programs, DSSM subsystems, prelinked modules, etc.

A release unit may be supplied more than once; an initial delivery may be followed by correction deliveries. The process of installation turns a release unit into an installation unit, which is registered in the SCI.

The designation and attributes of a release unit correspond to those of an installation unit (see "[Installation unit \(IU\)](#)").

2.6.4 Supply component, release item

The supply components (SCs) are also known as release items (RIs). A release item is the smallest unit (file) supplied as a component of a product – for example a load module, syntax file, or the like. The file name includes the name and version of the release unit.

A product (release unit) can refer to its release items without using their names. The mechanism involved here employs an internal ID assigned to each release item and known as the logical ID.

Once it is installed, a release item is known as an installation item. The installation items are registered in the SCI.

2.6.5 Installation unit (IU)

Installation units (IUs) are release units that have been **installed**.

An installation unit is the smallest unit which can be activated with IMON. An installation unit comprises a set of installation items and is registered in the SCI by IMON. An installation unit has the same name and the same version designation as the release unit.

Designation of an installation unit

Each installation unit bears a unique designation formed as follows:

<unit-name> <version>

<unit-name> Name of the installation unit in the format of the SDF data type
<structured-name 1..15>

<version> Version of the installation unit. The format is "mm.naso".

mm Main version (01..99).

n Addition version (0..9)

aso Correction state (a=A..Z, so=00..99)

Example of the designation for an installation unit

EDT 17.0D30

Note

In the case of software other than that supplied by Fujitsu, <unit-name> is the name of the installation unit in the format of the SDF data type <text 1..30 without-sep>.

The following syntactical rules apply:

- The first character must be an uppercase alpha character, a digit or @
- The next character must be an uppercase alpha character, a digit or any of the following: @, +, -, / or . (period).

Attributes of an installation unit

- Designator of an installation unit:
name and version of the installation unit

-
- Installation information:
 - lost+Found (ascertained by IMON)
 - NO: the installation unit could be assigned
 - YES: there is no entry in the associated SYSSII file
 - BS2000 version for which the installation unit is installed
 - user ID under which installation was performed
 - functional level on which the installation unit executes
 - TU: if all the installation items of the installation unit execute in Task Unprivileged mode
 - TPR: all the installation items of the installation unit execute in Task Privileged mode
 - BOTH: if the installation unit includes installation items which execute in both TU and TPR modes
 - creation date and time: date and time of first registration in the SCI
 - change date and time: date and time of most recent registration in the SCI
 - Installation counter, which goes up every time a supply unit is installed
 - Activability information about the installation unit:
 - states if the installation unit can be dynamically activated or not (YES/NO)
 - activation stage of the dynamic activation
 - Information on the installation items in the installation unit:
 - name and version of each installation item
 - installation path of each installation item
 - number of installation items

Example of an attribute listing for an installation unit with IMON

```
//show-installation-units unit-name=edt,
```

```
information=*par(instal-items=*yes,report-level=*all-attr)
```

```
SCI Name = :I29A:$TSOS.SYS.IMON.SCI
```

```
=====
```

```
Installation Unit = EDT Version = 17.0D30
```

```
----- Installation Information -----
```

```
FU-Level = TU
```

```
BS2000 = 21.0
```

```
Crea-User-Id = TSOS
```

```
Creation-Date = 2021-04-01
```

```
Change-Date = 2021-04-01
```

```
Creation-Time = 12:41:53
```

```
Change-Time = 14:00:31
```

```
Installation counter = 00011
```

```
----- Activation Information -----
```

```
Activable : Yes
```

```
Level : 41
```

----- Installation Items -----

Name = EDT	Version	= 17.0
Instal. Path = :I29A:\$TSOS.EDT		
Name = EDTSTART	Version	= 17.0
Name = SINPRC.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SINPRC.EDT.170		
Name = SYSACF.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSACF.EDT.170		
Name = SYSFGM.EDT.170.D	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSFGM.EDT.170.D		
Name = SYSLIB.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSLIB.EDT.170		
Name = SYSLNK.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSLNK.EDT.170		
Name = SYSMES.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSMES.EDT.170		
Name = SYSNRF.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSNRF.EDT.170		
Name = SYSREP.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSREP.EDT.170		
Name = SYSRME.EDT.170.D	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSRME.EDT.170.D		
Name = SYSRMS.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSRMS.EDT.170		
Name = SYSSDF.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSSDF.EDT.170		
Name = SYSSII.EDT.170	Version	= 17.0
Initial Path = :I29A:\$TSOS.SYSSII.EDT.170		
Initial Path = :I29A:\$SYSSAG.SOLLIB.IMON.SYSSII		
Name = SYSSMB.EDT.170	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSSMB.EDT.170		
Name = SYSSSC.EDT.170.112	Version	= 17.0
Instal. Path = :I29A:\$TSOS.SYSSSC.EDT.170		

(16 items)

2.6.6 Installation item (II)

Installation items (IIs) are release items that have been installed. An installation item is the smallest **installed** object registered in the SCI by IMON. The designation (file name) of an installation item is derived from that of the associated release item on installation.

Designation of an installation item

An installation item bears the designation <filename 1..30 without-cat-user-gen-vers> in accordance with the following syntactical rules:

<aaabbb>.<unit name>[.<version>][.<extension>]

- <aaabbb> Specifies the usage (aaa) as well as the contents and the processing rule (bbb) for the file.
- <unit-name> Name of the installation unit to which the installation item belongs, in the format of the SDF data type <structured-name 1..15>.
- <version> The version of the installation item consists of three digits “mmn”, where:
- mm = is the main version (01..99) of the installation unit
 - n = is the addition version (0..9) of the installation unit.
- This parameter is not used if the file is independent of the version of the installation unit.
- <extension> The extension makes it possible to distinguish between two or more installation items having the same <aaabbb> string.

Example of the designation for an installation item

SYSLNK.EDT.170

Note

In the case of software other than that supplied by Fujitsu, the designator for an installation item can be in the format of the SDF data-type <filename 1..30 without-cat-user-gen-vers> without restrictive syntax rules.

Attributes of an installation item

- Designator of the installation item: Name and version of the installation item
- Designator for the installation unit to which the installation item belongs:
- Name and version of the installation unit
- Lost+Found (ascertained by IMON)
Indicates whether the installation item could be assigned (NO)
or if the entry in the associated SYSSII file is missing (YES).

-
- Installation information
 - user ID under which installation was performed
 - type of the installation item (e.g. DAT for data)
 - date and time of creation and most recent change
 - Logical information:
 - Identification of the hardware dependency (TARGET) of the installation item:
 - A the installation item is independent of the hardware variant.
 - S the installation item is part of the /390 variant of the subsystem.
 - K the installation item is part of the X86 variant of the subsystem.
 - Logical ID of the installation item (see ["Path name, logical ID and type of an installation item"](#))
 - Function level on which the installation item executes:
 - Task Unprivileged (TU), if the installation item is not privileged
 - Task Privileged (TPR), if the installation item is privileged
 - Information on the initial installation path
 - creation path, i.e. the initial installation path registered in the SCI
 - installation state
 - file attributes
 - the appropriate values are taken from the catalog entry on installation.
 - Information on the current installation path
 - path name which is the installation path currently registered in the SCI
 - installation state
 - file attributes (the appropriate values are taken from the current catalog entry)

Example of an attribute listing for an installation item with IMON

```
//show-installation-items item-name=syslnk.edt.170,  
information=*par(report-level=*all-attr)
```

```
SCI Name = :I29A:$TSOS.SYS.IMON.SCI
```

```
=====
```

```
Installation Item = SYSLNK.EDT.170          Version      = 17.0  
----- From Installation Unit
```

```
-----  
Name = EDT                               Version = 17.0B00  Lost-Found = NO  
----- Installation Information
```

```
-----  
Creation-User-Id = TSOS                   Type          = DAT  
Creation-Date    = 2021-04-01             Change-Date   = 2021-04-01  
Creation-Time    = 14:00:01               Change-Time   = 14:00:31  
----- Logical Information
```

```
-----  
Logical-ID      = SYSLNK                   FU-Level      = TU  
                                           Target        = A
```

```
----- Initial Installation Path
```

```
-----  
Filename = :I29A:$TSOS.SYSLNK.EDT.170    (Placed)  
Block-Control = No  
Creation-Date = 2021-04-01    Change-Date   = 2021-04-01  
Creation-Time = 13:58:14     Change-Time   = 13:59:13  
User-Access   = Special      Access        = Read  
ACL           = No           Basic-ACL      = -----  
Migrate       = Inhibit      Backup-Class   = A  
----- Current Installation Path
```

```
-----  
Filename = :I29A:$TSOS.SYSLNK.EDT.170    (Placed)  
Block-Control = No  
Creation-Date = 2021-04-01    Change-Date   = 2021-04-01  
Creation-Time = 13:58:14     Change-Time   = 13:59:13  
User-Access   = Special      Access        = Read  
ACL           = No           Basic-ACL      = -----  
Migrate       = Inhibit      Backup-Class   = A
```

2.6.7 Path name, logical ID and type of an installation item

The path name indicates where the installation item (file) is to be found in the BS2000 system.

Format: `:<cat-id>:$<userid>.<item-name>` e.g. `:HOME:$TSOS.SYSSDF.PERCON.029`

The path name is registered in the SCI when the installation item is installed with IMON. At this time, too, the path name is assigned to the logical ID of the installation item.

The depot location can be modified after installation. Note the following:

If you change the path name in this way, you must use the IMON-GPN functions to redefine the logical ID/path name assignment and update the SCI.

The logical ID of an installation item is derived from the designator and unequivocally describes the installation item within its installation unit.

The logical ID is independent of the location of the file in the BS2000 system.

Format: `<aaabbb>[.<extension>]`, e.g. `SYSFGM.E`

Note

In the case of software other than that supplied by Fujitsu, the designator for an installation item can be in the format of the SDF data-type `<filename 1..30 without-cat-user-gen-vers>` without restrictive syntax rules.

Each release item has a logical ID defined in the SYSSII file and assigned to the path name on installation.

The logical ID enables users and other programs to identify and process the path name with the aid of the IMON-GPN functions.

The type of a release item or installation item determines the processing steps that must be performed for the item in question on installation. Additional information is defined in customizing (see "[Installation preparation](#)"). The accompanying document lists the type for each release item, complete with explanatory notes on the processing steps.

Examples

- Release items of type "DAT" are installed without change in the system. Existing files are overwritten.
- Release items of type "SDF" (syntax file) are treated in the same way as release items of type "DAT". In addition, they can be activated or merged with the system syntax file (see [figure 29 \(Edit menu\)](#) and [figure 30 \(Edit menu\)](#)).
- Release items of type *NP is a combination of the item types *PS and *NW.

It has been introduced for products for which the deinstallation in POSIX must be run with the old installation libraries ("SINLIBs") also for the same main version.

Notes on POSIX-Processing

As for the type *PS, the type *NP describes the POSIX-satellites that can be deinstalled and installed automatically in the frame of the IMON installation in POSIX (via "POSIX Processing").

The processing of the item types *PS and *NP differs only when the main version of the updated product remains the same.

In this case the following steps are schematically processed for *NP items:

1. creation of the new product files ("SINLIB") with the suffix .NEW
2. POSIX package deinstallation on basis of the old product files
3. copy of the .NEW files on the files without suffix
4. POSIX package installation on basis of the new product files

In contrast the deinstallation of *PS items for same main version is run on basis of the new product files.

Remarks

1. The suffixes .NEW and .SAV are reserved for the processing of POSIX installation libraries ("SINLIBs") of type *NP by IMON. The use of these suffixes for the renaming or the copy of POSIX installation libraries (of type *PS or *NP) leads to unexpected results.

The following table shows the contents of the various item types:

Item type	Contents
DAT	Default file type
MES	Message file (new format; message file created with MSGMAKER)
SDF	SDF syntax file
REP	RMS file for system corrections (reps)
SSD	DSSM declaration file
SSC	DSSM declaration file (new format; declaration file created with SSCM)
SRC	Library with elements of the type S or M
PL*	Library with elements of any type
PLM	Library with elements of the type M
PLR	Library with elements of the type R
PLS	Library with elements of the type S
DO	Procedure that can be called during execution of the installation procedure
ENT	ENTER file that can be called during execution of the installation procedure and then deleted
NST	Supply component that is not a BS2000 file (e.g. publication or data volume) The installation item is ignored by IMON
%xx	Supply components for internal use, which have two alphanumeric characters xx, e.g. %CD is for a CD. The installation item is ignored by IMON
*DA	File of the type DAT, only of relevance for certain target versions
*DC	File of DAT or *DA type that must also be copied under a specific name during installation

*DF	Dummy installation item whose full installation path is specified
*FE	Release Notice (English)
*FG	Release Notice (German)
*NW	File is assigned the standard suffix .NEW if it already exists in the target system prior to the installation
*PS/*NP	POSIX item (mainly SINLIB) indicating that the linked item must be registered as a satellite in the POSIX system. For *NP the file is added the standard suffix .NEW if it already exists in the target system before the installation.

The following table illustrates some examples of the connection between the path name, the logical name, the code for hardware dependency (TARGET), the item type, and contents of release items or installation items.

Path name	Logical name	TARGET	Type	Contents
... .SYSLNK. ...	SYSLNK	A or S	DAT	Module library
... .SKMLNK. ...	SYSLNK	K	DAT	Module library
... .SKULNK. ...	SYSLNK	K	DAT	Module library
... .SYSSSC. ...	SYSSSC	A	SSC	Subsystem declarations
... .SYSSDF. ...	SYSSDF	A	SDF	SDF syntax file
... .SYSMES. ...	SYSMES	A	MES	Message file
... .SYSPRG. ...	SYSPRG	A or S	DAT	User program
... .SYSFGM.D	SYSFGM.D	A	*FG	Release Notice (German)
...				

Installation items differ from the previous SYS files of the /390 mode in terms of their SKU and SKM usage classes:

Contents	/390 variant TPR and TU (as before)	x86 variant for TPR	x86 variant for TU
Module library for dynamic reloading	SYSLNK.<product>	SKMLNK.<product>	SKULNK.<product>
Module library for static binding	SYSOML.<product>	SKMOML.<product>	SKUOML.<product>
Program file (phase)	SYSPRG.<product>	---	---

For the x86 variant the delivered and installed installation items are differentiated from each other using the usage classes SKU or SKM (e.g. SKULNK.<product> for a module library).

2.7 Install on the basis of customer approval IDs

This installation function is based on the assumption that the customer uses several pubsets as follows.

- All software deliveries that the customer receives are installed on a shared pubset. This pubset contains all deliveries received and is used exclusively as a source pubset for the installations on the other pubsets of the customer.
- At least one further pubset is used for a test system on which new correction statuses or versions of supply units can be tested independently of current production use. The supply units to be tested are installed from the SCI of the source pubset (INSTALL-UNITS statement with UNIT=*FROM-SCI(...)).
- The appropriate number of pubsets is used for the production systems. Only supply units that have previously been sufficiently tested on the test system and therefore guarantee a certain quality standard defined by the customer are installed on these pubsets.

Approve supply unit

The supply units to be tested are installed on the test system from the SCI of the source pubset (INSTALL-UNITS statement with UNIT=*FROM-SCI(...)). Once a supply unit has reached the desired quality standard, the tested version on the source pubset can be explicitly approved for production use. To do this, the corresponding version of the supply unit is given a customer approval ID (in the form of a timestamp) in the SCI of the source pubset using the SET-CUSTOMER-APPROVAL statement.

Install approved supply unit

During a scheduled interruption of a production system the “install on the basis of the customer approval IDs” function simplifies updating of the supply units used on the system.

Only supply units that have a customer approval ID are selected. A delta installation is then performed for the selected supply units; in other words, only those installation items that have an older installation timestamp in the target system are installed. RMS processing is unnecessary and only the REP loaders are copied from the source pubset.

Possible calls:

- > Call the INSTALL-UNITS statement with UNIT=*CUSTOMER-APPROVED(...). To obtain advance information, you can select information output exclusively for supply units with (or without) a customer approval ID.
- > Set output of supply units in the *View: Filter...* menu.
Set the selection criterion *Customer-Approved=Yes* in the *SU selection* dialog box and select the desired supply units. Install the selected units with *Edit: Install customer-approved LE(SU)*. Only the SORT supply unit is approved in the following example.

```

File Edit Show View Options
-----
IMON: SCI:6A0B::\$TSOS.SYS.IMON.SCI
-----
Unit name      Vers Corr Package name  User code  Inst. Status  CAP Act
X SORT         08.0 A30 <package-name> SOL2P    Installed    N    Y
              *** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 7: Select approved supply units (example)

```

File Edit Show View Options
-----
: 12 *. Search ... :
---- : *. Add ... : -----
: 3. Remove : gh 5 of 5
: 4. Installed : More:
: *. Park ... : Status CAP Act
X SOR : 6. Generate installation definition file ... : d N Y
: 7. Deinstall ... :
: 8. Undo ... :
: *. Activate ... :
: *. Check ... :
: *. Request correction delivery :
: 12. Customer-Approved Install ... :
:.....:
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 8: Call installation function for approved supply unit (example)

With the exception of supply unit selection (only approved supply units are accepted), the further course of installation is similar to multiple installation from the SCI (see example in section "Multiple installations of software, which is already installed").

Restrictions

Installation cannot be performed unless the following conditions are fulfilled for all selected supply units.

- The change date of the supply unit must be earlier than the approval time (timestamp of the customer approval ID).
- Installation on the source pubset must have taken place after installation on the target pubset.
- The correction status on the source pubset must be higher than that on the target pubset.
- The product movement files must be available on the work file ID.

The installation process is aborted if a non-approved supply unit is specified. This can happen if supply units are explicitly specified in the INSTALL-UNITS statement or if the customer approval ID is not used as a selection criterion in menu mode.

Reset customer approval ID

Because the customer approval ID is entered for the supply unit in the SCI only, the associated installation items must not be modified. If, during installation, an installation item of an approved supply unit is changed, IMON automatically resets the customer approval ID (the timestamp is set to 0). When a supply unit is released, the customer approval ID of an older correction status of the version is automatically reset. The ID can be reset manually using the RESET-CUSTOMER-APPROVAL statement.

2.8 Deinstallation

During deinstallation all installed or parked supply units that are not needed any more are removed. The deinstallation procedure consists of the following actions:

- activated files are deactivated, if necessary
- installed files are deleted, if necessary
- entries in the IMON-SCI are removed

If the original state that existed before the installation is to be restored and a supply unit is to be deinstalled, then this can be accomplished under certain circumstances using the Undo function (see [section "Undo - undoing an installation"](#)).

The deinstallation process is triggered using the IMON function "Edit: Deinstall" or the DEINSTALL-SUPPLY-UNITS statement. The process is divided into two phases:

- > deinstallation preparation
- > actual deinstallation process

Deinstallation preparation

In this phase IMON checks which objects are affected by the deinstallation and if the deinstallation can be performed without error:

1. Analysis of the target system and of the SCI to determine which files are affected:
IMON first creates a list of all affected files and then checks the target system (to see if the files exist) and the SCI (assignments to other installation units that will remain installed in the system)
Note
The target system to be processed for the deinstallation (i.e. the home subset of the target system) is determined using the catalog ID of the currently open SCIs.
2. Analysis of file access:
The access rights of the files to be deinstalled are checked and changed with the MODIFY-FILE-ATTRIBUTES command, if necessary.
3. Subsystems check:
If a supply unit to be deinstalled corresponds to one or more subsystems, then IMON checks if the affected subsystems are deactivated.

Test mode

The deinstallation can also be called in the Test mode (operand EXECUTION= *NO in the DEINSTALL-SUPPLY-UNITS statement or *Direct Execution = 2 (No)* in the menu mode). In this case only the analysis is performed and the results are recorded.

Actual deinstallation procedure

In this phase actual deinstallation is performed with the following actions:

- deactivation of files
- deletion of files

-
- cleanup of the SCI

Deactivating files

Some special files that were activated during installation (e.g. syntax files, message files), must be deactivated in the same manner for deinstallation:

1. Syntax files (SDF item type)

The entries for each syntax file to be deinstalled are removed from the following files:

- Default SDF parameter file of the target system
- SDF parameter file that is entered in the SCI for the syntax file

An activated syntax file is deactivated directly.

2. Message files (MES item type)

The entries for each message file to be deinstalled are removed from the following files:

- Default MIP parameter file of the target system
- MIP parameter file that is entered in the SCI for the syntax file

An activated message file is deactivated directly.

3. Subsystem declarations (SSC and SSD item types)

Subsystem declarations located in SYSSSC files to be deinstalled are removed from the static subsystem catalog (the default catalog and the subsystem catalog entered in the SCI).

The corresponding source files of the subsystem catalog (<dssm catalog>.SRC) are also updated.

The dynamic subsystem catalog is updated accordingly for the current system during deinstallation, and the affected subsystems are removed dynamically (REMOVE-SUBSYSTEM command). The affected subsystem must be stopped before this is done (STOP-SUBSYSTEM command).

4. POSIX files (item types *PS / *NP)

Deinstallation of installed POSIX items is executed before the removal of the files, synchronously in the processing of the "Edit: Deinstall" or the DEINSTALL-SUPPLY-UNITS statement. If POSIX is not present, the "Edit: Deinstall" or the DEINSTALL-SUPPLY-UNITS statement is aborted. Then the operator is asked at the console to choose among abort or continue the processing.

Deleting files

All files that belong only to the selected supply units are deleted in the target system.

Cleaning up the SCI

All affected supply units are removed from the SCI together with the installation units they contain (when they are not also assigned to another installed supply unit).

Consistency and error handling

To ensure consistent deinstallation, the deinstallation procedure is performed in several steps. Each step of the deinstallation must execute successfully before the next step can be started.

If an error arises during a step of the deinstallation, an error handling routine is initiated:

- In interactive mode, processing is stopped and a message requiring confirmation and containing information on how to deal with the error is output. The deinstallation is aborted, continued while ignoring the error or continued in the Test mode depending on the response from the user.

-
- In batch mode, execution of the DEINSTALL-SUPPLY-UNITS statement is continued in the Test mode when an error arises.

Restrictions

The following actions performed during installation are not undone during deinstallation:

- Library elements that were merged with alternative libraries during installation are not removed during deinstallation.
- The RMS delivery quantities are not cleaned from the RMS depot.
- For parked supply units, the entries are reset in the SCI to their original status. The parked files are erased using the clean-up procedure generated during parking (see the PARK-UNITS statement, "[PARK-UNITS Park Software](#)"). No saving of these erased file is available (operand FILE-SAVING is ignored).
- Message files that were merged into the default system message file (SYSMES.EKP.01) are not removed during deinstallation.

2.9 Undo - undoing an installation

To undo the last installation executed for a supply unit the supply unit is removed from the system (just like during deinstallation). In addition, the original state that existed before installation of the supply unit is restored based on the Undo files saved during installation.

A prerequisite for the Undo function is that a backup including the creation of the Undo files was performed during installation. An Undo file contains the metadata for the deinstallation of a supply unit. It is stored during installation under the SYSSAG work file ID (if not present, then under TSOS) of the target pubset with the following name:

:<catid of the target pubset>:\$SYSSAG.IMON.UNDO.<name of the supply unit>

Only one SCI with standard name \$TSOS.SYS.IMON.SCI (home pubset or imported pubset) can be processed by the Undo function.

The Undo process is triggered by the IMON function "Edit: Undo" or the UNDO-SUPPLY-UNITS statement. It is divided into two phases:

- > Preparation of the Undo function
- > Actual execution

Preparing the Undo function

Specifies if the actual Undo actions are to be executed after the execution of the preliminary analysis. The preliminary analysis is used to determine if the most important requirements have been fulfilled:

1. Subsystems that are to be removed must be stopped.
2. Files whose activation is to be undone and that are not to be reconstructed while the UNDO function is processing must be accessible.

Test mode

The Undo function can also be called in the Test mode (operand EXECUTION= *NO in the UNDO-SUPPLY-UNITS statement or *Direct Execution = 2 (No)* in the menu mode). In this case only the preliminary analysis is performed and the results are recorded.

Actual execution

The following actions are taken in this phase, just like for deinstallation (see "[Deinstallation](#) "):

- Deactivation of files
Subsystems, syntax and message files that were activated during the corresponding installation are deactivated. An activated POSIX item is removed from the POSIX system. The POSIX "remove" commands are saved in the \$SYSROOT.IMON.ACTIONS.REM file that is automatically executed the next time the POSIX subsystem is started.
- Deletion of files
Files that were created during the corresponding installation are deleted.
- Cleanup of the SCI
Entries that were created in the SCI during the corresponding installation are removed from the SCI.

In addition, the following actions are taken to restore the original state:

- Activation of files

Subsystems, syntax and message files that were deactivated during the corresponding installation are reactivated. In addition, each POSIX product that is removed during installation when adding a new product is restored in the POSIX system. The necessary POSIX “add” commands are stored in the \$SYSROOT.IMON.ACTIONS.ADD file that is executed the next time the POSIX subsystem is started.

- Reconstruction of files

Files that were changed or deleted during the corresponding installation are reconstructed.

Consistency and error handling

To ensure consistent processing, the Undo function is performed in several steps. Each processing step must execute successfully before the next step can be started.

If an error arises during a processing step, then an error handling routine is initiated:

- In interactive mode, processing is stopped and a message requiring confirmation and containing information on how to deal with the error is output. Execution of the Undo function is either aborted, continued while ignoring the error or continued in the Test mode depending on the response from the user.
- In procedure or batch mode, the UNDO-SUPPLY-UNITS statement is aborted abnormally when the first error arises.

2.10 Dynamic activation

When a supply unit is installed, activation is prepared for the next system run. For subsystems, apart from the activation of the message and syntax files, only the static subsystem catalog is updated. The dynamic subsystem catalog of the current system is not changed.

With the “dynamic activation” function, a newly installed supply unit (of the corresponding installation units) can be made available already in the current system, i.e. with no interruption. Dynamic activation includes:

- Activation of the syntax files in the current system
- Activation of the message files in the current system
- Activation of the POSIX files in the current system. The POSIX commands needed to directly register the activated unit in the POSIX system using the /START-POSIX-INSTALLATION command are generated in the activation procedure.
- Starting the subsystem from the subsystem catalog of the installation unit (this point is not executed in case of “non-subsystems”)

Dynamic activation can be carried out for every supply unit (or installation unit), which has the attribute “activable”. The choice of the supply units (or installation units) to be activated is determined either from the opened standard SCIs, or from an opened delivery from the last installation process.

Definition of activability

Supply units and installation units have the attribute “activable” or “not activable”, entered in SCI with “Activable=Yes” or “Activable=No”, and shown with the corresponding information output (see example of supply unit on "[Supply unit](#)" or installation unit on "[Installation unit \(IU\)](#)").

A supply unit is activable if it contains at least one activable installation unit.

Activability of an installation unit

When attributes are assigned, first of all the distinction is made for installation units between subsystems and non-subsystems.

“Non-subsystems” are in any case classified as “activable”.

Subsystems are divided into 5 levels with respect to their activability. The Activation level is entered in SCI as an additional attribute (information output in the field “Level”):

Level 1: the subsystem is activable

Level 2: the subsystem is activable but the creation-time subsystem attribute is changed

Level 3: the subsystem is only activable under certain conditions

Level 41: the subsystem has dependencies to other subsystems, but is activable

Level 42: the subsystem has dependencies to other subsystems and is only activable under certain conditions

The assignment of the activation level depends on the subsystem attributes defined by the subsystem catalog:

-
- In **Level 1** the subsystem attributes permit dynamic activation:

creation-time = at-creation-request / at-subsystem-call
version-exchange = allowed / forbidden
version-coexistence = allowed / forbidden
state-change-cmds = allowed
subsystem-hold = allowed

In the section *FUNCTIONAL DEPENDENCE WITH SUBSYSTEMS* of the catalog definitions there are no subsystem entries.

In the section *REFERENCED SUBSYSTEMS* of the catalog definitions, there are no further subsystem entries except for CP.

- In **Level 2** the subsystem attributes permit dynamic activation, if attributes are changed in advance:

creation-time = at-dssm-load / before-dssm-load / before-system-ready / after-system-ready / mandatory-at-startup

version-exchange = allowed / forbidden

version-coexistence = allowed / forbidden

state-change-cmds = allowed

subsystem-hold = allowed

In the section *FUNCTIONAL DEPENDENCE WITH SUBSYSTEMS* of the catalog definitions, there are no subsystem entries.

In the section *REFERENCED SUBSYSTEMS* of the catalog definitions, there are no subsystem entries except for CP.

- In **Level 3** the following subsystem attribute prevents a subsystem included in the selection from being started or stopped:

state-change-cmds = forbidden

- In **Level 41** the subsystem attributes permit dynamic activation:

creation-time = at-creation-request / at-subsystem-call
version-exchange = allowed / forbidden
version-coexistence = allowed / forbidden
state-change-cmds = allowed
subsystem-hold = allowed

There are subsystem entries in the section *FUNCTIONAL DEPENDENCE WITH SUBSYSTEMS* or in the section *REFERENCED SUBSYSTEMS* of the catalog definitions.

- In **Level 42** the subsystem attributes permit dynamic activation, if attributes are changed in advance:

creation-time	=	at-dssm-load / before-dssm-load / before-system-ready / after-system-ready / mandatory-at-startup
version-exchange	=	allowed / forbidden
version-coexistence	=	allowed / forbidden
state-change-cmds	=	allowed
subsystem-hold	=	allowed

There are subsystem entries in the section *FUNCTIONAL DEPENDENCE WITH SUBSYSTEMS* or in the section *REFERENCED SUBSYSTEMS* of the catalog definitions.

Dynamic activation is performed through the IMON function “Edit: activate” or the statement ACTIVATE-UNITS. It is divided into two phases:

- > Preparation of the dynamic activation
- > Actual execution

Preparation of dynamic activation

Choice of supply units or installation units to be activated

First the supply units or installation units to be dynamically activated are selected. The choice is made in menu mode using the IMON function “*Edit: Activate*”, in SDF mode directly with the statement ACTIVATE-UNITS. The choice can be made in the following ways:

- The user states the objects to be activated explicitly with the operand UNIT-NAME or marks them directly in the work area.
- The user only has the activable objects displayed (in the operand UNIT-NAME=*BY-DIALOG or in menu mode *View:Supply-Units* and *Activable=2 (Yes)* in the *Supply Units* dialog box. He then selects the desired objects directly in the work area. The amount of activable objects can be further limited by the following selection criteria:
 - Choice of one or more deliveries (package name and customer ID are shown in the operand SELECT=*SOLIS2-DELIVERY(...) or in the dialog box *Supply Unit View*)
 - Choice of the objects of the most recently executed installation (corresponds to the preset value: Operand SELECT=*LAST-INSTALLED or *Last Installation=1 (Yes)* in the dialog box *Supply Unit View*)

If a selection includes multiple occurrences of an installation unit, the unit will be activated once only. If there are several versions of the same installation unit, only the highest version will be activated.

After the selection has been made, IMON identifies the method of dynamic activation. There are three ways:

- A new subsystem is activated.
- The correction version of a subsystem is activated.
- The version of the subsystem to be activated replaces an existing subsystem.

IMON then generates the following two files.

- Log file \$SYSSAG.<prefix>.<time-stamp>.RP that contains all DSSM commands (STOP-, REMOVE-, ADD- and START-SUBSYSTEM) for the subsystems that can be activated

- Activation procedure \$SYSSAG.<prefix>.<time-stamp>.DA that contains all necessary commands and statements for dynamic activation of the selected objects

The default prefix is the string IMONACU (see also ["Important activation files" \(Error handling and activation restart\)](#)).

Once the two files have been generated, IMON interrupts the preparation phase with a prompt. The user can check the log file and then respond to the prompt with answer “1” or “2” to either abort or continue the activation process.

Response	Effect
1	Aborts the activation process. Log file and activation procedure are deleted.
2	Resumes the activation process (see also "Actual activation"). Depending on the call option selected, the activation procedure is started automatically or the activation process is terminated normally so that the activation procedure can be started manually at a later time.

Structure of the log file

The log file is divided into two parts.

- Part 1 lists all supply units or installation units selected for activation and, if necessary, any errors or warnings that occurred during their processing.
- Part 2 contains all DSSM commands to activate the subsystems for which no error was reported in the first part. This information is listed in columns.

Column	Meaning and possible values
1	Subsystem type; possible values <ul style="list-style-type: none"> • SS: Subsystem selected via the supply unit or installation unit • SS-DEP: Subsystem to which there is a dependency • SS-DEL: Subsystem to be removed (i.e. deletion from the subsystem catalog or, if necessary, deactivation of associated syntax files)
2	Name of the subsystem
3	Version of the subsystem
4	DSSM command (abbreviated name) that is to be executed for the subsystem (possible values: STOP, REMOVE, ADD, START)
5 - 8	Identification of the installation unit that contains the relevant subsystem

Structure of the activation procedure

The activation procedure breaks down into individual processing steps that support a restart/recovery mechanism in the event of an error (see [section "Error handling and activation restart"](#)). The following processing steps are possible.

1. Releasing an installation unit lock

This activation step groups all UNLOCK-PRODUCT-VERSION commands for the selected installation units or the installation units in the selected supply units.

2. Creating a subsystem catalog

This activation step groups all SSCM statements that are needed to create the catalogs for all relevant subsystems.

3. Stopping the subsystem to be deleted

This activation step groups all STOP-SUBSYSTEM commands that are needed to stop the subsystems that are to be deleted (if there are installation items of the type SSC).

4. Deactivating the syntax files of the subsystem to be deleted

This activation step groups all MODIFY-SDF-PARAMETERS commands that are needed to deactivate the syntax files of subsystems to be deleted (if there are installation items of the type SDF).

5. Stopping a subsystem

This activation step groups all STOP-SUBSYSTEM commands for the subsystems resulting from the selection.

To check whether the subsystems have been stopped, IMON generates a call of the internal `WAIT-SUBSYSTEM-STATUS` function for each of the subsystems. These calls are terminated without error if all subsystems have the `NOT CREATED` status. If one of the subsystems has a different status, the call fails and the activation process is interrupted.

Note

The subsystems to be stopped are divided into two subgroups.

- The subsystems in the selection that are divided further into subsystems of level 4 and subsystems of levels 1 and 2.
- Subsystems that have dependencies to the selected subsystems. These subsystems must be stopped before the selected subsystems are stopped.

6. Deleting a subsystem

This activation step groups all REMOVE-SUBSYSTEM commands for the subsystems resulting from the selection.

7. Activating a message file

This activation step groups all MODIFY-MSG-FILE-ASSIGNMENT commands for all message files included in the selection.

8. Activating a syntax file

This activation step groups all MODIFY-SDF-PARAMETERS commands for all syntax files included in the selection.

9. Adding a new subsystem catalog

This activation step groups all ADD-SUBSYSTEM commands for the subsystem catalogs created in activation step 2.

In order to check whether the subsystem catalogs have been added, IMON generates a call of the internal `WAIT-SUBSYSTEM-STATUS` function for each of the subsystems.

These calls are terminated without error if all subsystems have the `NOT CREATED` status. If one of the subsystems has a different status, the call fails and the activation process is interrupted.

10. Starting a subsystem

This activation step groups all START-SUBSYSTEM commands for the subsystems resulting from the selection.

In order to check whether the subsystem catalogs have been started, IMON generates a call of the internal `WAIT-SUBSYSTEM-STATUS` function for each of the subsystems.

These calls are terminated without error if all subsystems have the `NOT CREATED` status. If one of the subsystems has a different status, the call fails and the activation process is interrupted.

Note

The subsystems to be stopped are divided into two subgroups.

- The subsystems in the selection that are divided further into subsystems of level 4 and subsystems of levels 1 and 2.
- Subsystems that have dependencies to the selected subsystems. These subsystems must be stopped before the selected subsystems are stopped.

11. Performing POSIX processing

This activation step groups all POSIX processing processes for all installation items of type `*PS / *NP` included in the selection.

12. Resetting subsystem attributes for subsystems of level 2

This activation step groups all `MODIFY-SUBSYSTEM-PARAMETER` commands for the subsystems classified with level 2.

13. Restoring the status of a subsystem stopped due to dependencies

This activation step groups all `START-SUBSYSTEM` commands for the subsystems that were stopped due to existing dependencies (see activation step [5](#)).

Actual activation

The actual activation takes place when the generated activation procedure is run. The activation procedure cannot be started unless the user resumes the preparation phase interrupted by IMON after checking the log file by choosing response “2”. IMON deletes the generated files if the user selects response “1”.

As default, the procedure is run automatically. The procedure can also optionally be started manually with the command `ENTER-PROCEDURE` (operand `START=*BY-USER` in the statement `ACTIVATE-UNITS` or `Start = 2 (By user)` in the dialog box *Activation parameters*).

Refer to the [section "Error handling and activation restart"](#) for information on handling errors that may occur during the course of the activation procedure.

2.11 Request correction delivery

Customers can request information on corrections and/or correction deliveries using the new REQUEST-CORRECTION-DELIVERY statement or the *Edit: Request correction delivery* option. IMON creates a formatted request (with the file name \$TSOS.IMON.DELREQ.<timestamp>) that contains the desired data and sends this request by mail to the software distribution center of Fujitsu. IMON uses the following parameter file for mail configuration.

```
$TSOS.SYSPAR.IMON.<customer_ID>.MAIL.CONFIG
```

The scope of the request can be defined as follows.

- Delta delivery with only the modified release items of the supply unit (default)
- Correction delivery for the entire supply unit
- Only information on the corrections made

Using the parameter file for mail configuration (see "[Parameter file for mail configuration](#)") and the user input, IMON creates a correction request (with the file name \$TSOS.IMON.DELREQ.<timestamp>) that is sent by mail to the software distribution center of Fujitsu.

The following delivery methods can be selected.

- Delivery to the customer on a SOLIS2 data volume of the specified device type.
- Delivery in a PLAM library by direct transfer to the customer system using openFT. This option is available only to customers with a direct BS2000 network connection.
- WWW delivery made available on the Internet.

WWW delivery

This delivery method enables customers to request correction deliveries for their BS2000 systems at short notice and at no extra cost.

Once the request has been received, the software distribution center performs the following actions.

- The request is checked against the specific individual software configuration.
- The delivery package is custom generated as an FT delivery in a PLAM library and is assigned an identification key using the MD5 algorithm in order to ensure integrity.
- The delivery is compressed in BS2000 using the BS2ZIP utility, transferred in binary form to a POSIX system, and is made available on the Internet with password protection for a period of 14 days after the requested provision date. Binary transfer is made using the program specified by the user for collection from the Internet (FTP or openFT).
- The user is informed by mail that the delivery is available. The mail contains not only the delivery documentation but also the access data needed to identify the delivery (URL, ID, password, MD5 key). The mail is sent to the address of the system administrator registered with Fujitsu.

After receipt of the above information, the user performs the following actions.

- Using the access data sent with the mail, the user signs on to the WWW server of Fujitsu. On the page specified by the URL there is a link for downloading the BS2000 delivery and with information on how to proceed further.
- The delivery package can be downloaded onto any customer server on which the program for collecting the delivery as specified in the request (FTP or openFT) is available. It is essential to use this program so that data transfer is binary compatible.

-
- If necessary, the delivery can be unpacked and its integrity checked on the customer server before transferring it to BS2000.
 - Transfer to BS2000 must be in binary form.
 - The package is unpacked using the procedure supplied and a new MD5 key is created for the integrity check; this can be compared with the key in the mail.
 - Further processing of the delivery package is then performed using IMON.

Prerequisites for a WWW delivery

- Either FTP (part of interNet Services) or the chargeable supply unit openFT must be available.
- The following customer information must be supplied to Fujitsu in order to identify the WWW participant.
 - Short name of the system to which the delivery is to be sent (customer ID)
 - Name and mail address of the system administrator

3 Installation sequences under BS2000

This chapter describes how to install software using IMON under BS2000.

The procedure is demonstrated using sample installations for both standard and customer specific installations.

3.1 Default installation BS2000

In a default installation, the complete delivery is stored on the home pubset under the user ID stipulated by Fujitsu or on another, imported pubset.

System environment and requirements for the sample default installations

The following settings and requirements are valid for the default installations described on pages 71 ([Installation on the home pubset](#)) through 110 ([Example with SDF statements](#)). The first table lists the general system environment and the features of the SOLIS delivery.

Current system	MM.N
Target system of the installation	MM.N
Package name of the SOLIS2 delivery	<package-name>
Customer identifier	SOL2P
Supply units contained	EDT, OPENFT, OPENFT-CR, PERCON, SORT

MM.N - current version

The next two tables detail the differences between the IDs and file names for an installation on the home pubset and an installation on an imported pubset.

For an installation on the home pubset

Catalog ID of the home pubset	I29A
Standard SCI	:I29A:\$TSOS.SYS.IMON.SCI
Standard system ID	TSOS
Work file ID	SYSSAG on home pubset (preset)
Path for work files	:I29A:\$SYSSAG.
Library for the delivery documentation	:I29A:\$SYSSAG.<package-name>.SOL2P.DOC
Library for the backup	:I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P
Installation procedure	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IE
Log of important console messages	:I29A:\$SYSSAG.<package-name>.IC
Installation log	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IL
Log of the installation actions	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IA
Log of the installation parameters	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IP
Log of the registration in the SCI	:I29A:\$SYSSAG.<package-name>.<time-stamp>.II

For an installation on an imported pubset

Catalog ID of the imported pubset	6A0B
Foreign SCI	:6A0B:\$TSOS.SYS.IMON.SCI
Standard system ID for the installation on the imported pubset	TSOS
Work file ID	SYSSAG on imported pubset
Path name for work files	:6A0B:\$SYSSAG.
Library for the delivery documentation	:6A0B:\$SYSSAG.<package-name>.SOL2P.DOC
Library for the backup	:6A0B:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P
Installation procedure	:6A0B:\$SYSSAG.<package-name>.<time-stamp>.IE
Log of important console messages	:6A0B:\$SYSSAG.<package-name>.IC
Installation log	:6A0B:\$SYSSAG.<package-name>.<time-stamp>.IL
Log of the installation actions	:6A0B:\$SYSSAG.<package-name>.<time-stamp>.IA
Log of the installation parameters	:6A0B:\$SYSSAG.<package-name>.<time-stamp>.IP
Log of the registration in the SCI	:6A0B:\$SYSSAG.<package-name>.<time-stamp>.II

The <time-stamp> part of the name is the timestamp of the installation in the format <month><day><time><year>, where

<month>	is the first three letters of the month
<day>	is a one or two digit day of the month
<time>	is the time at which the installation started in the form hhmmss
<year>	is a four digit figure

General steps in a default installation

IMON generates and starts the installation procedure. It loads all of the files in the supplied products to the location specified by Fujitsu and registers them in the SCI as installation units.

Call IMON

Depending on the call, IMON branches to menu mode or waits for SDF statements.

Open SCI	The standard SCI \$TSOS.SYS.IMON.SCI on the home pubset is opened implicitly (in SDF mode when started, in menu mode when the delivery is opened). If a foreign SCI is to be created/written, it must be opened explicitly. An SCI opened beforehand must be closed in menu mode.
Define work file ID	IMON creates work files on the home pubset under the SYSSAG user ID by default.
Open SOLIS2 delivery	ARCHIVE reads the data volume. Information on the package name, user code, VSN, and volume type from the accompanying information is needed during this process. A delivery that is already open can be opened as a registered delivery.
Confirm supply units	<p>All supply units are offered for further processing by IMON, confirm this selection with [DUE].</p> <p>When a registered delivery is opened, all supply units that have not yet been installed are offered for further processing.</p>
Select install	Installation of the selected supply units is preset.
Select parameter file	The default parameter file SYSPAR.IMON.LAST of the current environment is preset. If this file exists, IMON uses the specific installation parameters stored there as defaults in the subsequent dialog boxes.
Specify target system	The operating system version and the home pubset of the active system are preset.

IMON generates and starts the installation procedure. It loads all of the files in the supplied products to the location specified by Fujitsu and registers them in the SCI as installation units.

**Close
SOLIS2
delivery**

The SOLIS2 delivery is closed explicitly if IMON is to be used again (e.g. to check the SCI). Otherwise IMON can be terminated directly.

**Terminate
IMON**

If the SOLIS2 delivery was not already closed, it is closed implicitly when IMON is terminated.

**Any
postprocessing**

Any necessary postprocessing and the order in which it should be carried out can be found in the delivery information.

IMON generates and starts the installation procedure. It loads all of the files in the supplied products to the location specified by Fujitsu and registers them in the SCI as installation units.

The schematic representation of the sequence can be found in ["Default installation on the home pubset"](#) (for the home pubset) or ["Default installation on an imported pubset"](#) (for the imported pubset).

3.1.1 Installation on the home pubset

The schematic representation of the sequence can be found on "[Default installation on the home pubset](#)".

3.1.1.1 Example in menu mode

<p>Call IMON</p>	<p>IMON is started once English is set as the task-wide language for messages and masks. After loading the program and entering [DUE], IMON switches to the menu mode. The IMON welcome screen appears first.</p> <pre data-bbox="326 457 1419 659"> /modify-msg-attributes task-language=e /start-imon % IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from fil e ':I29A:\$TSOS.SYSLNK.IMON-BAS.034.TU' % IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved PLEASE ACKNOWLEDGE </pre> <p>The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI is used.</p> <pre data-bbox="326 772 1419 1339"> File Edit Show View Options ----- IMON: Welcome screen ----- *** Welcome in Installation MONitor *** Use menu File: Open: Standard SCI to display standard SCI contents. Use menu File: Open: SOLIS2 delivery to install a delivery in standard SCI. Use menu File: Open: Foreign SCI and then menu File: Open: SOLIS2 delivery to install a delivery in a foreign SCI ----- Command ==> F1=Help F3=Exit F10=Menu F12=Cancel </pre>
<p>Open SCI</p>	<p>Since no SCI has been opened explicitly yet, the standard SCI on the home pubset is opened implicitly when the delivery is opened any time thereafter (see also the information on the welcome screen). The explicit opening of the standard SCI is displayed for the customer specific installation (see "Example in menu mode"). There you will also see that the SCI contains 163 entries, i.e. 163 installation items, before installation.</p>
<p>Specify work file ID</p>	<p>The preset for the work file ID is used in this example. The ability to set the work file ID through the <i>Options</i> menu is shown for the installation on an imported pubset, for example, on "Example in menu mode".</p> <p>When opening the delivery in the "SOLIS2 Delivery" dialog box the preset work file ID is displayed as a partially qualified path name with catalog ID (in this case with :I29A:\$SYSSAG., see "Example in menu mode").</p>
<p>Open SOLIS2 delivery</p>	<p>The delivery should then be opened: Option 1 (<i>Open</i>) is already preset in the <i>File</i> menu. The mask is executed using [DUE].</p>

```

File Edit Show View Options
-----
: 1 1. Open ... : : Welcome screen
: * . Close : -----
: * . Print documentation ... :
: * . Save as ... : e in Installation MONitor ***
: 5. Exit :
: .....:
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

Option 4 (*SOLIS2 delivery*) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 4 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... : -----
: 3. Installation definition file ... :
: 4. SOLIS2 delivery ... : ***
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel
: .....:
Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The following delivery information is then queried in the “SOLIS2 Delivery” dialog box that is opened: package name, user code, distribution medium (MTC 2). The set work file ID is displayed again in the lower section of the mask. This dialog box is executed using [DUE].

```

File Edit Show View Options
:-----:-----:
:                               Open                               :
:-----:-----:
:                               SOLIS2 delivery                    :
:-----:-----:
: Package name.....: <package-name>                               :
: User code.....: sol2p                                           :
: Distribution medium: 2 1. SOLIS2 volume                          2. Library      :
:                               3. Registered                          4. Local volume :
:                               Volume.....:                               :
:                               Device type:                       :
:-----:-----:
: Documentation library: 1 1. Standard                            2. Other        :
:                               Name:                               :
: Work file location: :I29A:$SYSSAG.                               :
: Reference File ... :I29A:$TSOS.SYS.IMON.SCI.REF                 :
:-----:-----:
: Supply unit selection 1 1.Manually (*BY-DIALOG)                :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                             :
:-----:-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

IMON then has all of the information needed to open the delivery on the data medium and it starts the processing operation, during which some messages are output to the screen:

IMON instructs ARCHIVE to read the specified MTC and extract files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries). The "IMON" jobs started for this purpose create or expand the documentation library and start a print job for the acknowledge form. You can switch back to IMON menu mode using [DUE].

```

% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142557', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0183' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142610', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0184' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0185
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0186
%PLEASE ACKNOWLEDGE

```

Confirm supply units

Five supply units were found on the specified MTC. IMON offers these for further processing (they are already highlighted). To select all of these supply units, execute the selection using [DUE].

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name          SU selection      Units      1 through  5 of      5
                   Version          Corr state  More:
X EDT              17.0          D30
X OPENFT          12.1          C10
X OPENFT-CR       12.1          B10
X PERCON          02.9          C00
X SORT            08.0          A30
                   *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select install

You must then switch to the *Edit* menu to install the units. Option 4 (*Install*) is already preset and needs only to be confirmed with [DUE].

```
File Edit Show View Options
-----
IM : 4 *. Search ... : code: SOL2P
----- : *. Add ... :
: *. Remove ... : gh 5 of 5
: 4. Installed ... : More:
: *. Park ... : te
X EDT : *. Generate installation definition file ... :
X OPE : *. Deinstall ... :
X OPE : *. Undo ... :
X PER : *. Activate ... :
X SOR : *. Check ... :
: *. Request correction delivery ... :
: *. Customer-Approved Install ... :
: .....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select parameter file

The following default installation parameters options are preset by IMON in the “IMON Parameter file” dialog box.

```
File Edit Show View Options
:-----:
:                               IMON Parameter file :
:-----:
: Parameter file ...: 2 1. None          2. Standard    3. Other      : 5
: File name:       :I29A:$TSOS.SYSPAR.IMON.LAST
:
: Save parameters...: 1 1. No          2. Standard    3. Other      :
:   File name:
:
: F1=Help F12=Cancel
:-----:
:                               *** End of SU selection ***
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

Specify target system

The following default installation settings are preset by IMON in the “Global installation parameters” dialog box.

```
File Edit Show View Options
:-----:
:                               Global installation parameters
:-----:
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 1 1. Default          2. Other
: Save old files.....: 1 1. No              2. With LMS
:                               3. With ARCHIVE          4. With MAREN
:
:                               VSN.....:
:                               Device type:      MAREN location: *STD
: Undo preparation.....: 2 1. Yes              2. No
: Placement mode.....: 1 1. Standard        2. Other
: Activation preparation mode: 1 1. Standard        2. Other
: Password file.....: 2 1. Yes              2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes              2. No
: Start.....: 1 1. Immediately    2. By user
: Configuration checks.....: 1 1. Yes              2. No
: Work file deleting.....: 1 1. Yes              2. No
:
: F1=Help F12=Cancel
:-----:
```

MM.N - current version

Input for this example

Target system:

- The supply units are to be installed for BS2000 current version (preset).
- The supply units are to be installed on the default pubset, i.e. the home pubset (preset)

Save old files:

- A "2" or "3" must be entered here to back up existing files that would be overwritten by the installation operation. Backup to an LMS library (option "2") was selected for this example.

The library will be created with the following name under the work file ID:

IMON.SAVE.LIB.<package name>.<customer ID>

i.e. in this example :I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name> .SOL2P

Undo preparation:

- A "1" must be specified here to create a backup that will allow you to restore the original state before installation (see [section "Undo - undoing an installation"](#)) The preset value "2" (i.e. no backup) was left unchanged in the example.

Placement mode:

- The default placement option is retained

Activation preparation mode:

- The default activation option is retained

Password file:

- This specification need not be changed

Print log files:

- This specification need not be changed: The logs are not to be printed out automatically once the installation is complete.

Start:

- This specification need not be changed: The installation procedure generated by IMON is to be started immediately and automatically.

Configuration check:

- This specification need not be changed: a configuration check should be executed.

Work file deleting:

- The work files generated under the work file ID by IMON during the installation are not to be deleted once the installation is complete. A "2" is also entered, which is different than the preset value.

The dialog box is executed with these entries and [DUE].

```

File Edit Show View Options
:-----:
: Global installation parameters :
:-----:
: Target system :
: BS2000 version.....: MM.N :
: Standard subset.....: 1 1. Default           2. Other :
: Save old files.....: 2 1. No                2. With LMS :
:                   : 3. With ARCHIVE      4. With MAREN :
: VSN.....: :
: Device type: MAREN location: *STD :
: Undo preparation.....: 2 1. Yes             2. No :
: Placement mode.....: 1 1. Standard       2. Other :
: Activation preparation mode: 1 1. Standard     2. Other :
: Password file.....: 2 1. Yes            2. No :
: File name....: :
: Read password: C' :
: Print log files.....: 2 1. Yes             2. No :
: Start.....: 1 1. Immediately 2. By user :
: Configuration checks.....: 1 1. Yes            2. No :
: Work file deleting.....: 2 1. Yes            2. No :
: :
: F1=Help F12=Cancel :
:-----:

```

Additional specifications are requested to select Installation items that belong to a certain hardware variant or a certain target system version with the “Target system parameters” dialog box.

The preset is 4 (All), i.e. the selection is made regardless of the hardware variant. The defaults displayed in the example are confirmed without change using [DUE]:

```

File Edit Show View Options
:-----:
: Global installation parameters :
:-----:
: Target system parameters :
:-----:
: Processor-type: 4 1. CISC-390 :
: * --- : 2. Other :
: 3. X86 : 2. With LMS :
: 4. All : 4. With MAREN :
: AREN location: *STD :
: Item-selection: 2 1. All :
: 2. Target-version-only : 2. No :
: : 2. Other :
: : 2. Other :
: : 2. No :
: F12=Cancel :
: : 2. No :
: Print log files.....: 2 1. Yes             2. No :
: Start.....: 1 1. Immediately 2. By user :
: Configuration checks.....: 1 1. Yes            2. No :
: Work file deleting.....: 2 1. Yes            2. No :
: :
: F1=Help F12=Cancel :
:-----:

```

**IMON
output**

IMON has then queried all of the information needed to install the delivery and starts the processing operation during which some messages are output on the screen:

- The :I29A:\$SYSSAG.<package-name>.APR91450162021.IE installation procedure will be generated:
 - Five supply units were selected for installation.
 - Various processing steps must be generated.
 - Activation of the listed supply units is generated.
- The installation procedure is then fully generated. It is started immediately and automatically.

```
Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' is being generated. '5'  
supply units have been selected  
Generation of :I29A:$SYSSAG..APR91450162021.IE startet:  
Generation of FSTAT-RENAME commands...  
Generation of SAVE-FILES commands...  
Generation of RESERVE-FILE commands...  
Generation of IMPORT-FILE commands...  
Generation of UPDATE-CATALOG-ENTRY commands...  
Generation of library processing commands...  
Generation of ADD-INSTALLATION-UNITS statements...  
Generation of reference file...  
SYSSII files processing started  
Generation of ACTIVATE-UNIT commands for supply units:  
EDT          17.0   D30  
OPENFT       12.1   C10  
OPENFT-CR    12.1   B10  
PERCON       02.9   C00  
SORT         08.0   A30  
Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' has been generated and  
entered  
PLEASE ACKNOWLEDGE
```

Checking the installation process

IMON switches back to menu mode once [DUE] is entered.

The SHOW-JOB-STATUS command can be used in the interactive task to check process whether the started installation task (job name "INSTALL") is already finished. The procedure sequence can be checked in the installation log.

As the installation procedure is still running asynchronously, the supply units to be installed are in an undefined state and are not displayed in this mask.

As the installation procedure is still running asynchronously, the supply units to be installed are in an undefined state and are not displayed in this mask.

If a subset of the supply units in the delivery were installed and not the complete set, this mask would display those supply units that were not selected.

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      0 through      0 of      0
                                SU selection
Unit name          Version      Corr state
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Close SOLIS2 delivery

Option 2 (*Close ...*) is preset in the *File* menu. The SOLIS2 delivery is closed with [DUE].

```
File Edit Show View Options
-----
: 2 *. Open ...           : name: <package-name> User code: SOL2P
: 2. Close ...           : -----
: 3. Print documentation ... :           Units      0 through      0 of      0
: 4. Save as ...         : selection
: 5. Exit ...            :           Version      Corr state
:.....: SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Checking the SCI

After successful installation, 168 supply units are displayed in the SCI. The newly entered supply units are sorted in alphabetical order. They can be checked by scrolling forwards using "+" in the command line and [DUE] or [F8].

In the following screen, the contents were scrolled forward to the new EDT entry. The EDT installation unit is entered in the SCI with its version and correction state.

```
File Edit Show View Options
-----
IMON: SCI: :PH21:$TSOS.SYS.IMON.SCI
-----
Unit name      Installation units selection 37 through 49 of 168
                  Version          Corr state      More: + -
DPRINTCL      01.2             A00
DPRINTCM      01.2             A00
DPRINTSV      01.2             A00
DRIVE         03.1             A10
DRIVE-COMP    03.1             A10
DRIVE-COMP-DOC 03.1             A10
DRIVE-COMP-LZS 03.1             A10
DRV           03.2             A02
DSSM          21.0             A00
DWS           11.0             A01
EDT           17.0             D30
ELFE         21.0             A00
ELSA         01.7             A10
-----
Command ==> +
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Naturally all other installation units in the delivery were entered also (cf. also the output after reopening the delivery, "[Example in menu mode](#)"). It was not possible to show a complete reproduction of the SCIs with the relevant masks here due to the space restrictions.

Terminate IMON

Installation of the SOLIS2 delivery is then complete. IMON itself can be terminated if no other IMON functions are needed (e.g. displaying information or carrying out other installations). IMON is terminated by entering 5 (*Exit*) in the *File* menu and [DUE] or [F3]).

```
File Edit Show View Options
-----
: 5 1. Open ... : IMON.SCI
: 2. Close : -----
: * . Print documentation ... : Units 37 through 49 of 168
: 4. Save as ... : tion units selection More: + -
: 5. Exit : Version Corr state
:.....:
DPRINTCM 01.2 A00
DPRINTSV 01.2 A00
DRIVE 03.1 A10
DRIVE-COMP 03.1 A10
DRIVE-COMP-DOC 03.1 A10
DRIVE-COMP-LZS 03.1 A10
DRV 03.2 A02
DSSM 21.0 A00
DWS 11.0 A01
EDT 17.0 D30
ELFE 21.0 A00
ELSA 01.7 A10
-----
Command ==> +
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

<p>Checking the work file ID</p>	<p>Files remain under the SYSSAG work file ID once the installation is concluded. As stipulated in the "Global installation parameters" dialog box (see "Example in menu mode"), they were not deleted automatically.</p> <pre> /show-file-attributes \$syssag. 48 :I29A:\$SYSSAG.A.SOLIS.B.<package-name>.SOL2P 15744 :I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P 9 :I29A:\$SYSSAG.RMS.OPT.ADM 54 :I29A:\$SYSSAG.RMS.OPT.DOC 327 :I29A:\$SYSSAG.SOLIS.LMSCONV 24 :I29A:\$SYSSAG.SOLLIB.IMON.PBD 24 :I29A:\$SYSSAG.SOLLIB.IMON.SYSSII 420 :I29A:\$SYSSAG.SOLLIB.IMON.020 3939 :I29A:\$SYSSAG.SOLLIB.SOLIS2.050 3939 :I29A:\$SYSSAG.SOLLIB.SOLIS2.050.RMS.071 495 :I29A:\$SYSSAG.SOLPAR.IMON.034.GEN 1596 :I29A:\$SYSSAG.SOLPAR.SOLIS2.050.AUSGEN 72 :I29A:\$SYSSAG.SOLPAR.SOLIS2.050.FMSG 306 :I29A:\$SYSSAG.SOLPRG.SOLIS2.050.GEN 3 :I29A:\$SYSSAG.SYSREP.EDT.170 12 :I29A:\$SYSSAG.SYSREP.OPENFT.121 3 :I29A:\$SYSSAG.SYSREP.PERCON.029 12 :I29A:\$SYSSAG.SYSREP.SORT.080 6 :I29A:\$SYSSAG.<package-name>.DP 303 :I29A:\$SYSSAG.<package-name>.IC 9 :I29A:\$SYSSAG.<package-name>.APR91450142010.DP 6 :I29A:\$SYSSAG.<package-name>.APR91450162010.IA 99 :I29A:\$SYSSAG.<package-name>.APR91450162010.IE 15 :I29A:\$SYSSAG.<package-name>.APR91450162010.II 210 :I29A:\$SYSSAG.<package-name>.APR91450162010.IL 9 :I29A:\$SYSSAG.<package-name>.APR91450162010.IP 30 :I29A:\$SYSSAG.<package-name>.APR91450162010.IR 3 :I29A:\$SYSSAG.<package-name>.APR91450162010.I0 456 :I29A:\$SYSSAG.<package-name>.APR91450162010.SCI 144 :I29A:\$SYSSAG.<package-name>.APR91450162010.SCI.GPN 12 :I29A:\$SYSSAG.<package-name>.RETURN.LETTER 96 :I29A:\$SYSSAG.<package-name>.SOL2P.DOC :I29A: PUBLIC: 32 FILES RES= 28735 FRE= 2403 REL== 2346 PAGES </pre>
<p>Post processing</p>	<p>For this example no manual post-processing is necessary (e.g. starting a product-specific procedure). Where manual post-processing is necessary, IMON displays messages to this effect on the console (these are also found in the installation protocol <package-name>.APR91450162021.IL or under the important console messages in the file <package-name>.IC on the work file ID).</p> <p>The delivery information can tell you whether further postprocessing is required. It should be referred to once more in this regard.</p>

Checking the installation

If the delivery was then opened once more (starting IMON, *Open: SOLIS2 delivery* menu), it should, as would be expected, exist in the SCI as a delivery that was already registered. Therefore by entering "3" for the distribution medium in the following dialog box, it should be opened as a delivery that has already been registered (the package name and user code must also be specified).

```

File Edit Show View Options
:-----:
:                               Open                               :
:-----:
:                               SOLIS2 delivery                    :
:-----:
: Package name.....: <package-name>                             :
: User code.....: sol2p                                           :
: Distribution medium: 3 1. SOLIS2 volume           2. Library       :
:                               3. Registered           4. Local volume   :
:                               Volume.....:                   :
:                               Device type:                   :
:-----:
: Documentation library: 1 1. Standard           2. Other         :
:                               Name:                               :
: Work file location: :I29A:$SYSSAG.                   :
: Reference File .... :I29A:$TSOS.SYS.IMON.SCI.REF      :
:-----:
: Supply unit selection 1 1. Manually (*BY-DIALOG)         :
:                               2. Automatically (*BY-CONFIGURATION) :
: F1=Help  F12=Cancel                                     :
:-----:
Command ==>
F1=Help  F3=Exit  F10=Menu  F12=Cancel

```

The delivery is known in the SCI. All five supply units were already installed.

```

File Edit Show View Options
:-----:
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
:-----:
Unit name  Vers  Corr  Package name  User code  Inst. Status  CAP Act
-----
          Units      1 through      5 of      5
          SU selection
EDT        17.0 D30  <package-name> SOL2P    Installed    N    Y
OPENFT     12.1 C10  <package-name> SOL2P    Installed    N    Y
OPENFT-CR  12.1 B10  <package-name> SOL2P    Installed    N    Y
PERCON     02.9 C00  <package-name> SOL2P    Installed    N    Y
SORT       08.0 A30  <package-name> SOL2P    Installed    N    Y
          *** End of SU selection ***
:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

3.1.1.2 Example with SDF statements

In this example, the default installation on the home pubset illustrated in “[Example in menu mode](#)” (“[Installation on the home pubset](#)”) will be completed using SDF statements. The procedure and objective remain the same.

All menus and user input offered by IMON that have to do with

- opening the SCI
- defining the work file ID
- opening the SOLIS2 delivery
- confirming the supply units selected by IMON for installation
- and specifying the target system can be realized here using two statements: MODIFY-IMON-OPTIONS and INSTALL-UNITS (see Points 3 through 8).

```
/modify-msg-attributes task-language=e _____ (1)
/ start-imon input-interface=*sdf _____ (2)
% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) IMON is started with the SDF interface by explicitly specifying INPUT-INTERFACE=*SDF.

Alternatively, IMON can also be started in the menu mode (it is sufficient to enter the START-IMON command because INPUT-INTERFACE=*STD is the preset and switches to menu mode).

After the user has explicitly opened the standard SCI with the *Open* option in the *File* menu, he or she can then select option 1 (*Switch to SDF Interface*) in the *Options* menu.

```
//modify-imon-options
  sci=*std, work-file-location=$syssag., reference-file=*std _____ (3)
// install-units _____ (4)
  unit-name=*from-solis2-delivery( _____ (5)
    package-name= <package-name> , user-code=sol2p,
    distribution-medium=*library,
    supply-units=*all), _____ (6)
  old-file-saving=*with-lms,
_____ (7)
  work-file-deleting=*no _____ (8)
```

-
- (3) The following statement is optional as the specified values correspond to the default setting:

The MODIFY-IMON-OPTIONS statement is used to explicitly specify the standard SCI (operand SOFTWARE-INVENTORY=*STD), the \$SYSSAG path on the home pubset (operand WORK-FILE-LOCATION=\$SYSSAG) as the storage location for work files and the default reference file for the configuration check (Operand REFERENCE-FILE=*STD).

- (4) The default installation is then carried out using the INSTALL-UNITS statement. The following operands are evaluated for this process:
- (5) The UNIT-NAME operand specifies the SOLIS2 delivery to be installed:

Package name: <package-name>

Customer ID: SOL2P

Delivery medium: Library

The delivery is opened implicitly with REGISTRATION=*EXTEND. This means that IMON evaluates any entries existing in the SCI for this delivery and changes the entries of the supply units in the SUPPLY-UNITS operand. Entries are already present if the delivery has already been opened at least once (e.g. after printing the delivery documentation with PRINT-DOCUMENTATION).

If all entries for this delivery in the SCI must be created first, the operand REGISTRATION=*REPLACE must be specified (see also "[INSTALL-UNITS Install and register software](#)").

- (6) The operand SUPPLY-UNITS=*ALL is the default: It selects all supply units for installation.

You can check and confirm the selected supply units in a dialog box in menu mode by specifying SUPPLY-UNITS=*BY-DIALOG (see [Example in menu mode](#)).

No further operands need to be specified for a default installation under the current operating system. The following non default specifications need to be set (cf. [Example in menu mode](#)):

- (7) Files on the system that will be overwritten by new files of the same name should first be backed up to an LMS library (cf. "[Example in menu mode](#)").

This library is called :I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P

- (8) The work files created by IMON are not to be deleted.

The statement is then executed using [DUE]:

```

% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.152557',
VERSION '12.0C00' _____ (9)
% ARC0033 ARCHIVE SUBTASK TSN '0190' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.152610',
VERSION '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0191' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 15:27, TSN = 0192

% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 15:27, TSN = 0193 — (10)
Job ':I29A:$SYSSAG. <package-name>.APR91527162021 .IE' is being generated. '5'
supply units have been selected _____ (11)
Generation of :I29A:$SYSSAG.<package-name>.APR91527162021.IE startet:
Generation of FSTAT-RENAME commands...
Generation of SAVE-FILES commands...
Generation of RESERVE-FILE commands...
Generation of IMPORT-FILE commands...
Generation of UPDATE-CATALOG-ENTRY commands...
Generation of library processing commands...
Generation of ADD-INSTALLATION-UNITS statements...
Generation of reference file...
SYSSII files processing started
Generation of ACTIVATE-UNIT commands for supply units:
  EDT          17.0   D30
  OPENFT       12.1   C10
  OPENFT-CR    12.1   B10
  PERCON       02.9   C00
  SORT         08.0   A30
Job ':I29A:$SYSSAG. <package-name>.APR91527162021 .IE' has been generated and
entered _____ (12)
// end _____ (13)

(9)  IMON reads the MTC (via ARCHIVE), extracts files that are needed for the subsequent installation (e.g.
      product movement file, IMON and SOLIS2 program libraries), and registers the delivery in the SCI.

(10) IMON then transfers the delivery documentation to the standard documentation library, which is created if
      necessary, and starts a print job for the acknowledge form (using the "IMON" job name).

(11) The :I29A:$SYSSAG.<package-name>.APR91527162021.IE installation procedure is generated:
      • five supply units were selected for installation
      • various processing steps must be generated
      • activation of the listed supply units is initiated.

(12) The installation procedure is then fully generated and started automatically as a background procedure
      (task with the job name "INSTALL").

(13) IMON is terminated using the END statement.

/show-job-status *job(install) _____ (14)

```

TSN: 0194 TYPE: 2 BATCHF NOW: 2021-04-09.152755
JOBNAME: INSTALL PRI: 9 210 SPOOLIN: 2021-04-09.1527
USERID: TSOS JCLASS: JCBTSOS LOGON: 2021-04-09.1527
ACCNB: ADMINSTR CPU-MAX: NTL CPU-USED:000003.2317
REPEAT: NO RERUN: NO FLUSH: NO
MRSCAT: HOLD: NO START: SOON
TID: 00020066 UNP/Q#: 17/004
CMD: EXECUTE SIZE: 70
PROG: :I29A:\$TSOS.SYSLNK.ARCHIVE.120(ARCHIVE,120,L)
ORIGFILE::I29A:\$SYSSAG. <package-name>.APR91527162021 .IE
CMD-FILE::I29A:\$TSOS.S.E.0194.2021-04-09.15.27.17
MONJV: *NONE

/show-job-status *tsn(0194) _____ (15)

% EXC0755 INFORMATION ON TASK WITH TSN '0194' CANNOT BE GIVEN

/show-file-attributes \$syssag. _____ (16)

48 :I29A:\$SYSSAG.A.SOLIS.B.<package-name>.SOL2P
15744 :I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P
9 :I29A:\$SYSSAG.RMS.OPT.ADM
54 :I29A:\$SYSSAG.RMS.OPT.DOC
327 :I29A:\$SYSSAG.SOLIS.LMSCONV
24 :I29A:\$SYSSAG.SOLLIB.IMON.PBD
24 :I29A:\$SYSSAG.SOLLIB.IMON.SYSSII
420 :I29A:\$SYSSAG.SOLLIB.IMON.020
3939 :I29A:\$SYSSAG.SOLLIB.SOLIS2.050
3939 :I29A:\$SYSSAG.SOLLIB.SOLIS2.050.RMS.071
495 :I29A:\$SYSSAG.SOLPAR.IMON.034.GEN
1596 :I29A:\$SYSSAG.SOLPAR.SOLIS2.050.AUSGEN
72 :I29A:\$SYSSAG.SOLPAR.SOLIS2.050.FMSG
306 :I29A:\$SYSSAG.SOLPRG.SOLIS2.050.GEN
3 :I29A:\$SYSSAG.SYSREP.EDT.170
12 :I29A:\$SYSSAG.SYSREP.OPENFT.121
3 :I29A:\$SYSSAG.SYSREP.PERCON.029
12 :I29A:\$SYSSAG.SYSREP.SORT.080
6 :I29A:\$SYSSAG.<package-name>.DP
303 :I29A:\$SYSSAG.<package-name>.IC
9 :I29A:\$SYSSAG.<package-name>.APR91450142010.DP
6 :I29A:\$SYSSAG.<package-name>.APR91450162010.IA
99 :I29A:\$SYSSAG.<package-name>.APR91450162010.IE
15 :I29A:\$SYSSAG.<package-name>.APR91450162010.II
210 :I29A:\$SYSSAG.<package-name>.APR91450162010.IL
9 :I29A:\$SYSSAG.<package-name>.APR91450162010.IP
30 :I29A:\$SYSSAG.<package-name>.APR91450162010.IR
3 :I29A:\$SYSSAG.<package-name>.APR91450162010.I0
456 :I29A:\$SYSSAG.<package-name>.APR91450162010.SCI
144 :I29A:\$SYSSAG.<package-name>.APR91450162010.SCI.GPN
12 :I29A:\$SYSSAG.<package-name>.RETURN.LETTER
96 :I29A:\$SYSSAG.<package-name>.SOL2P.DOC

:I29A: PUBLIC: 32 FILES RES= 28735 FRE= 2403 REL= 2346 PAGES

- (14) The SHOW-JOB-STATUS command is used to check whether the installation task is still running.
- (15) Once the installation task has run, the SHOW-JOB-STATUS command query is quit with message EXC0755, i.e. the task is terminated.
- (16) The SHOW-FILE-ATTRIBUTES command is used to display all files that were created under the work file ID.

`/show-file $sysmag.<package-name>.ic` _____ (17)

```

21-04-10 15:27:19 IMO2019 INSTALLATION FUNCTION EXECUTED FOR PACKAGE '<packag
21-04-10 15:27:22 IMO2001 FSTAT-RENAME : analyse of target system
21-04-10 15:27:43 IMO2002 SAVE-OLD-FILES procedure generation
21-04-10 15:27:44 IMO2003 SAVE-OLD-FILES procedure call
21-04-10 15:28:03 IMO2004 RESERVE-FILE : preparation of target system
21-04-10 15:28:07 IMO2005 IMPORT-FILE : import-procedure generation
21-04-10 15:28:08 IMO2024 IMPORT-FILE : import-procedure call
21-04-10 15:29:32 IMO2006 UPDATE-CATALOG-ENTRY : set file attributes
21-04-10 15:29:45 IMO2007 Library processing
21-04-10 15:29:48 IMO2008 ADD-INSTALLATION-UNITS : register installation unit
21-04-10 15:30:01 IMO2015 Reference file generation
21-04-10 15:30:12 IMO2011 RMS-PROCESSING: Revision packet put into RMS Depot
21-04-10 15:30:21 IMO2016 Processing of delivered SYSSII files
21-04-10 15:30:27 IMO4009 SUPPLY UNIT 'EDT' '17.0' 'D30' PROCESSED
21-04-10 15:30:27 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'EDT
21-04-10 15:30:27 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:29 IMO4010 ACTIVATE-UNIT 'EDT' '17.0' 'D30'
21-04-10 15:30:32 IMO4009 SUPPLY UNIT 'OPENFT' '12.1' 'C10' PROCESSED
21-04-10 15:30:32 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'OPE
21-04-10 15:30:32 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:34 IMO4010 ACTIVATE-UNIT 'OPENFT' '12.1' 'C10'
21-04-10 15:30:40 IMO4009 SUPPLY UNIT 'OPENFT-CR' '12.1' 'B10' PROCESSED
21-04-10 15:30:40 IMO4009 SUPPLY UNIT 'PERCON' '02.9' 'C00' PROCESSED
+
                                     S*SOF+          1(      1)

```

```

21-04-10 15:30:40 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'PER
21-04-10 15:30:40 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:43 IMO4010 ACTIVATE-UNIT 'PERCON' '02.9' 'C00'
21-04-10 15:30:46 IMO4009 SUPPLY UNIT 'SORT' '08.0' 'A30' PROCESSED
21-04-10 15:30:46 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'SOR
21-04-10 15:30:46 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:50 IMO4010 ACTIVATE-UNIT 'SORT' '08.0' 'A30'
21-04-10 15:30:53 IMO2010 RMS-PROCESSING: loader generation
21-04-10 15:31:05 IMO2027 SSCM global processing
21-04-10 15:31:10 IMO2021 INSTALLATION NORMALLY TERMINATED FOR PACKAGE '<pack

```

```

% SHO0301 WARNING: END OF FILE REACHED
e
                                     S*SOF+          24(      1)

```

(17) The SHOW-FILE command outputs the contents of the log file together with the most important console messages (file name: :I29A:\$SYSSAG.<package-name>.IC).

After scrolling to the file end, the last message logged (IMO2021) shows that the installation was terminated normally.

If manual post-processing is required, a console message to this effect will be displayed. The delivery information can tell you whether further postprocessing is required. It should be referred to once more in this context.

You can also check the installation sequence on the console (or in the CONSLOG file) and in the installation log. (The name of the installation log in the example is:

:I29A:\$SYSSAG.<package-name>.APR91527162021.IL).

After successful completion of the installation procedure, all supply units are registered in the SCI as installation units and the product files are installed in the stipulated storage location on the system.

```
/start-imon input-interface=*sdf _____ (18)
% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
//show-packages package-name= <package-name> ,user-code=sol2p,
inf=*par(supply-units=*yes,report-level=*minimum) _____ (19)
SCI Name = :I29A:$TSOS.SYS.IMON.SCI
=====
Package Name          User Code
Supply Unit Name      Version          Status
-----
<package-name>      SOL2P
EDT                   17.0D30         Installed
OPENFT                12.1C10         Installed
OPENFT-CR            12.1B10         Installed
PERCON                02.9C00         Installed
SORT                  08.0A30         Installed
(5 units)
//end _____ (20)
```

(18) IMON is called once more.

(19) All supply units of the previously installed delivery (package name <package-name>, customer ID SOL2P) that are now registered in the SCI are displayed with the SHOW-PACKAGES statement. Specification of the INFORMATION operand is optional in the example because the preset was used. The specification can be changed accordingly to request more detailed information.

(20) IMON is terminated using the END statement.

3.1.2 Installation on the imported pubset

Passages that differ from the default installation on the home pubset are displayed in a grid.

The schematic representation of the sequence can be found on "[Default installation on an imported pubset](#)".

3.1.2.1 Example in menu mode

Call IMON

IMON is started once English is set as the task-wide language for messages and masks. IMON switches to menu mode once the program is loaded and [DUE] entered.

```
/modify-msg-attributes task-language=e
/start-imon
% IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from fil
e ':I29A:$TSOS.SYSLNK.IMON-BAS.034.TU'
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved
PLEASE ACKNOWLEDGE
```

The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI are used.

```
File Edit Show View Options
-----
IMON: Welcome screen
-----

*** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

Open SCI

Now the standard SCI on the pubset 6A0B should be opened: The 1 (*Open ...*) option is already preset in the *File* menu. The mask is executed using [DUE].

```
File Edit Show View Options
-----
: 1 1. Open ... : : Welcome screen
: * . Close : : -----
: * . Print documentation ... :
: * . Save as ... : e in Installation MONitor ***
: 5. Exit :
: ..... :
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

Option 2 (*Foreign software configuration inventory*) is selected in the *Open* menu and confirmed with [DUE].

```

File Edit Show View Options
.....:-----
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :-----
: 3. Installation definition file ... :
: 4. SOLIS2 delivery ... : ***
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel
:
:.....:
:
Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The path name of the SCI on imported pubset 6A0B is entered in the "SCI name" field: : 6A0B:\$TSOS.SYS.IMON.SCI and confirmed with [DUE].

```

File Edit Show View Options
.....:-----
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :-----
: :
:.....:

```

The contents of the SCI are displayed with 163 entries, i.e. 163 installation units. The contents of the SCI are the same as those of the standard SCI in this example.

```

File Edit Show View Options
-----
IMON: SCI: :6A0B:$TSOS.SYS.IMON.SCI
-----
          Units      1 through      13 of 163
          Installation units selection      More: +
          Version      Corr state
ACS          21.0          A00
AID          03.5          A00
AIDSYS       21.0          A00
AIDSYSA      21.0          A00
ANITA        21.0          A00
APACHE       02.4          A00
ARCHIVE      12.0          C00
ASE          21.0          A00
ASSEMBH      01.4          A10
ASSEMBH-GEN  01.4          A10
ASSTRAN      05.4          A00
ASTI         21.0          A00
ATS          05.3          A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Define work file ID

To define the work file ID, select 2 (*IMON options*) in the *Options* menu bar. The mask is executed using [DUE].

```

File Edit Show View Options
-----
IMON: SCI: :6A0B: : 2 1. Switch to SDF interface :
-----
: 2. IMON options ... :
:.....: ough 13 of 163
: Installation units selection : More: +
Unit name Version Corr state
ACS 21.0 A00
AID 03.5 A00
AIDSYS 21.0 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The preset work file ID is displayed as a partial path name with a catalog ID (in this case with : I29A:\$SYSSAG., cf. "Example in menu mode "). As the work file ID should also be on he imported pubset, the catalog ID of this pubset is inserted (catalog ID 6A0B) and confirmed by pressing [DUE].

```

File Edit Show View Options
-----
: ..... IMON options :
: .....:
: Work file location: :6A0B:$SYSSAG. : 13 of 163
: Reference file ...: : More: +
: : I29A:$TSOS.SYS.IMON.SCI.REF :
: : :
: F1=Help F12=Cancel :
: .....:
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Open SOLIS2 delivery

The delivery should then be opened: Option 1 (*Open ...*) is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : IMON.SCI
: 2. Close :
: *. Print documentation ... : Units 1 through 13 of 163
: 4. Save as ... : tion units selection More: +
: 5. Exit : Version Corr state
: .....:
AID 21.0 A00
AIDSYS 03.5 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Option 4 (*SOLIS2 delivery ...*) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: .....:
: Open :
: .....:
: 4 *. Standard software configuration inventory : gh 13 of 163
: *. Foreign software configuration inventory ... : More: +
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel :
: .....:
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following specifications for the delivery are queried in the now open “SOLIS2 delivery” dialog box: package name, customer ID, delivery medium (MTC 2). In the lower part of the screen you can see the current work file ID setting. The mask is executed using [DUE].

```

File Edit Show View Options
:-----:
: Open :
:-----:
: SOLIS2 delivery :
:-----:
: Package name.....: <package-name> :
: User code.....: sol2p :
: Distribution medium: 2 1. SOLIS2 volume      2. Library :
:                   3. Registered           4. Local volume :
:                   Volume.....: :
:                   Device type: :
: : :
: Documentation library: 1 1. Standard      2. Other :
:                   Name: :
: Work file location: :6A0B:$SYSSAG. :
: Reference File ... : I29A:$TSOS.SYS.IMON.SCI.REF :
: : :
: Supply unit selection 1 1.Manually (*BY-DIALOG) :
:                   2.Automatically (*BY-CONFIGURATION) :
: : :
: F1=Help F12=Cancel :
:-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

IMON then has all of the information needed to open the delivery on the data medium and it starts the processing operation, during which some messages are output to the screen:

IMON instructs ARCHIVE to read the specified MTC and extract files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries). The "IMON" jobs started for this purpose create or expand the documentation library and start a print job for the acknowledge form. You can switch back to IMON menu mode using [DUE].

```

% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142557', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0183' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142610', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0184' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0185
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0186
%PLEASE ACKNOWLEDGE

```

Confirm supply units

Five supply units were found on the specified tape. IMON offers these for further processing (they are already highlighted). To select all of these supply units, execute the selection using [DUE].

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      1 through      5 of      5
                                SU selection  Version          Corr state
Unit name
X EDT                          17.0             D30
X OPENFT                       12.1             C10
X OPENFT-CR                   12.1             B10
X PERCON                       02.9             C00
X SORT                         08.0             A30
                                *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

You must then switch to the *Edit* menu to install the units. Option 4 (*Install ...*) is already preset and needs only to be confirmed with [DUE].

```

File Edit Show View Options
-----
IM : 4 *. Search ... : code: SOL2P
-----
: *. Add ... :
: *. Remove ... : gh 5 of 5
: 4. Installed ... : More:
: *. Park ... : te
X EDT : *. Generate installation definition file ... :
X OPE : *. Deinstall ... :
X OPE : *. Undo ... :
X PER : *. Activate ... :
X SOR : *. Check ... :
: *. Request correction delivery ... :
: *. Customer-Approved Install ... :
:.....:
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Select parameter file

The following default installation parameters options are preset by IMON in the “IMON parameter file” dialog box.

```

File Edit Show View Options
:-----
:                               IMON Parameter file :
:-----
: Parameter file ...: 2 1. None          2. Standard      3. Other        : 5
: File name:       :I29A:$TSOS.SYSPAR.IMON.LAST
:
: Save parameters...: 1 1. No          2. Standard      3. Other        :
:   File name:
:
: F1=Help F12=Cancel
:-----
:                               *** End of SU selection ***

```

```

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

The following default installation settings are preset by IMON in the “Global installation parameters” dialog box.

Specify target system

```

File Edit Show View Options
:-----
:                               Global installation parameters
:-----
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 2 1. Default          2. Other
:   Save old files.....: 2 1. No              2. With LMS
:                               3. With ARCHIVE          4. With MAREN
:
:                               VSN.....:
:                               Device type:      MAREN location: *STD
: Undo preparation.....: 2 1. Yes              2. No
: Placement mode.....: 1 1. Standard        2. Other
: Activation preparation mode: 1 1. Standard      2. Other
: Password file.....: 2 1. Yes              2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes              2. No
: Start.....: 1 1. Immediately  2. By user
: Configuration checks.....: 1 1. Yes              2. No
: Work file deleting.....: 2 1. Yes              2. No
:
: F1=Help F12=Cancel
:-----

```

MM.N – current version

Input for this example

Target system:

- The supply units are to be installed for BS2000 current version (default).
- The supply units are to be installed in the imported pubset 6A0B, a "2" must be entered here.

Save old files:

- A "2" or "3" must be entered here to back up existing files that would be overwritten by the installation operation. Backup to an LMS library (option "2") was selected for this example. The library will be created with the following name under the work file ID: `IMON.SAVE.LIB.<package name> <customer ID>`
i.e. in this example `:6A0B:$SYSSAG.IMON.SAVE.LIB.10MAI10617.SOL2P`

Undo process:

- A "1" must be specified here to create a backup that will allow you to restore the original state before installation (see [section "Undo - undoing an installation"](#)). The preset value "2" was left unchanged in the example.

Placement mode:

- The default placement option is retained

Activation preparation mode:

- The default activation option is retained

Password file:

- This specification need not be changed

Print log files:

- This specification need not be changed: The logs are not to be printed out automatically once the installation is complete.

Start:

- This specification need not be changed: The installation procedure generated by IMON is to be started immediately and automatically.

Configuration check:

- This specification need not be changed: a configuration check should be executed.

Work file deleting:

- The work files generated under the work file ID by IMON during the installation are not to be deleted once the installation is complete. Therefore the default setting must be changed to "2".

The dialog box is executed with these entries and [DUE]. The catalog ID 6A0B and the default system ID TSOS (which is already preset) are entered for the imported pubset in the “Pubset parameters” dialog box:

```

File Edit Show View Options
:-----:
: Global installation parameters :
:-----:
: Pubset parameters :
:-----:
: ..: MM.N :
: : 2 1. Default : 2. Other :
: Catid.: 6a0b : 2 1. No : 2. With LMS :
: : 3. With ARCHIVE : 4. With MAREN :
: DEFLUID: TSOS : SN.....:
: : evic type: MAREN location: *STD :
: F1=Help F12=Cancel : 2 1. Yes : 2. No :
: : 1 1. Standard : 2. Other :
: Activation preparation mode: 1 1. Standard : 2. Other :
: Password file.....: 2 1. Yes : 2. No :
: File name....:
: Read password: C' :
: Print log files.....: 2 1. Yes : 2. No :
: Start.....: 1 1. Immediately : 2. By user :
: Configuration checks.....: 1 1. Yes : 2. No :
: Work file deleting.....: 2 2. Yes : 2. No :
:
: F1=Help F12=Cancel :
:-----:

```

Additional specifications are requested to select Installation items that belong to a certain hardware variant or a certain target system version with the “Target system parameters” dialog box.

The preset is 4 (All), i.e. the selection is made regardless of the hardware variant. The defaults displayed in the example are confirmed without change using [DUE].

```

File Edit Show View Options
:-----:
: Global installation parameters :
:-----:
: Target system parameters :
:-----:
: Processor-type: 4 1. CISC-390 : 2. Other :
: : * --- : 2. With LMS :
: : 3. X86 : 4. With MAREN :
: : 4. All :
: : AREN location: *STD :
: Item-selection: 2 1. All : 2. No :
: : 2. Target-version-only : 2. Other :
: : : 2. Other :
: : : 2. No :
: F12=Cancel :
: : 2. No :
: Print log files.....: 2 1. Yes : 2. No :
: Start.....: 1 1. Immediately : 2. By user :
: Configuration checks.....: 1 1. Yes : 2. No :
: Work file deleting.....: 2 1. Yes : 2. No :
:
: F1=Help F12=Cancel :
:-----:

```

IMON has then queried all of the information needed to install the delivery and starts the processing operation during which some messages are output on the screen:

- The :6A0B:\$SYSSAG.10MAI10617.MAY121226442010.IE installation procedure will be generated:
 - five supply units were selected for installation
 - various processing steps must be generated
 - activation of the listed supply units is generated.
- The installation procedure is then fully generated. It is started immediately and automatically.

**IMON
output**

```

Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' is being generated. '5'
supply units have been selected
Generation of :I29A:$SYSSAG..APR91450162021.IE startet:
Generation of FSTAT-RENAME commands...
Generation of SAVE-FILES commands...
Generation of RESERVE-FILE commands...
Generation of IMPORT-FILE commands...
Generation of UPDATE-CATALOG-ENTRY commands...
Generation of library processing commands...
Generation of ADD-INSTALLATION-UNITS statements...
Generation of reference file...
SYSSII files processing started
Generation of ACTIVATE-UNIT commands for supply units:
  EDT           17.0   D30
  OPENFT        12.1   C10
  OPENFT-CR     12.1   B10
  PERCON        02.9   C00
  SORT          08.0   A30
Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' has been generated and
entered
PLEASE ACKNOWLEDGE

```

<p>Checking the installation process</p>	<p>IMON switches back to menu mode once [DUE] is entered.</p> <p>The installation procedure runs asynchronously and outputs to the console messages relating to the processing steps executed and any errors that arose. The procedure sequence can be checked in the installation log.</p> <p>The SHOW-JOB-STATUS command can be used in the interactive task to check whether the started installation task (job name "INSTALL") is already finished. The procedure sequence can be checked in the installation log.</p> <p>As the installation procedure is still running asynchronously, the supply units to be installed are in an undefined state and are not displayed in this mask. If a subset of the supply units in the delivery were installed and not the complete set, this mask would display those supply units that were not selected.</p> <div data-bbox="324 611 1419 1192" style="border: 1px solid gray; padding: 10px; margin: 10px 0;"> <pre> File Edit Show View Options ----- IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P ----- Units 0 through 0 of 0 SU selection Unit name Version Corr state *** End of SU selection *** ----- Command ==> F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ... </pre> </div>
<p>Terminate IMON</p>	<p>Installation of the SOLIS2 delivery is then complete. IMON itself can be terminated if no other IMON functions are needed (e.g. displaying information or carrying out other installations). IMON is terminated by entering 5 (Exit) in the File menu and [DUE] (it can also be terminated via the [F3] function key). This also closes the SOLIS2 delivery implicitly.</p> <p>Files remain under the SYSSAG work file ID once the installation is concluded. As stipulated in the "Global installation parameters" dialog box, they were not deleted automatically.</p>

<p>Checking the work file ID</p>	<pre> /show-file-attributes :6A0B:\$syssag. 48 :6A0B:\$SYSSAG.A.SOLIS.B.<package-name>.SOL2P 15744 :6A0B:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P 9 :A6A0B:\$SYSSAG.RMS.OPT.ADM 54 :6A0B:\$SYSSAG.RMS.OPT.DOC 327 :6A0B:\$SYSSAG.SOLIS.LMSCONV 24 :6A0B:\$SYSSAG.SOLLIB.IMON.PBD 24 :6A0B:\$SYSSAG.SOLLIB.IMON.SYSSII 420 :6A0B:\$SYSSAG.SOLLIB.IMON.020 3939 :6A0B:\$SYSSAG.SOLLIB.SOLIS2.050 3939 :6A0B:\$SYSSAG.SOLLIB.SOLIS2.050.RMS.071 495 :6A0B:\$SYSSAG.SOLPAR.IMON.034.GEN 1596 :6A0B:\$SYSSAG.SOLPAR.SOLIS2.050.AUSGEN 72 :6A0B:\$SYSSAG.SOLPAR.SOLIS2.050.FMSG 306 :6A0B:\$SYSSAG.SOLPRG.SOLIS2.050.GEN 3 :6A0B:\$SYSSAG.SYSREP.EDT.170 12 :6A0B:\$SYSSAG.SYSREP.OPENFT.121 3 :6A0B:\$SYSSAG.SYSREP.PERCON.029 12 :6A0B:\$SYSSAG.SYSREP.SORT.080 6 :6A0B:\$SYSSAG.<package-name>.DP 303 :6A0B:\$SYSSAG.<package-name>.IC 9 :6A0B:\$SYSSAG.<package-name>.APR91450142010.DP 6 :6A0B:\$SYSSAG.<package-name>.APR91450162010.IA 99 :6A0B:\$SYSSAG.<package-name>.APR91450162010.IE 15 :6A0B:\$SYSSAG.<package-name>.APR91450162010.II 210 :6A0B:\$SYSSAG.<package-name>.APR91450162010.IL 9 :6A0B:\$SYSSAG.<package-name>.APR91450162010.IP 30 :6A0B:\$SYSSAG.<package-name>.APR91450162010.IR 3 :6A0B:\$SYSSAG.<package-name>.APR91450162010.I0 456 :6A0B:\$SYSSAG.<package-name>.APR91450162010.SCI 144 :6A0B:\$SYSSAG.<package-name>.APR91450162010.SCI.GPN 12 :6A0B:\$SYSSAG.<package-name>.RETURN.LETTER 96 :I29A:\$SYSSAG.<package-name>.SOL2P.DOC :I29A: PUBLIC: 32 FILES RES= 28735 FRE= 2403 REL== 2346 PAGES F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ... </pre>
<p>Post-processing</p>	<p>For this example no manual post-processing is necessary (e.g. starting a product specific procedure). Where manual post-processing is required, IMON will display messages to this effect on the console (these are also found in the installation protocol <package-name>.APR121226442021.IL or under the important console messages in the file <package-name>.IC work file ID).</p> <p>168 supply units will be displayed if you call IMON once again and open the foreign SCI on pubset 6A0B (as in the default installation on the home pubset).</p>

3.1.2.2 Example with SDF statements

```
/modify-msg-attributes task-language=e _____ (1)
/start-imon input-interface=*sdf _____ (2)
% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) The INTERFACE=*SDF command is used to start IMON with the SDF interface.

Alternatively, you can also start IMON in menu mode (in which case it is sufficient to enter the START-IMON command, as INTERFACE=*STD is the default). After the user has explicitly opened an SCI with the *Open* option in the *File* menu, he or she can then select the option *1 (Switch to SDF interface)* in the *Options* menu.

```
//modify-imon-options
  sci=:6a0b:$tsos.sys.imon.sci,
  work-file-location=:6a0b:$syssag., reference-file=*std _____ (3)
//install-units _____ (4)
  unit-name=*from-solis2-delivery(
_____ (5)
  package-name= <package-name> ,user-code=sol2p,
  distribution-medium=*library,
  supply-units=*all-remaining),
_____ (6)
  target-system=*par( version=*current, _____ (7)
  pubset=*par(catid=6a0b,defluid=tsos)), _____ (8)
  old-file-saving=*with-lms, _____ (9)
  work-file-deleting=*no _____ (10)
```

- (3) With the MODIFY-IMON-OPTIONS statement the foreign SCI (operand SCI=:6A0B:\$TSOS.SYS.IMON.SCI) is set, the path \$SYSSAG. is set for work files on the imported pubset (operand WORK-FILE-LOCATION=:6A0B:\$SYSSAG.) and the default reference file for the configuration check (operand REFERENCE-FILE=*STD) is set.
- (4) The default installation is then carried out using the INSTALL-UNITS statement.

(5) The UNIT-NAME operand is used to specify the SOLIS2 delivery that is to be installed:

Package name: <package-
name>

User code: SOL2P

Distribution medium: Library

(6) The operand SUPPLY-UNITS=*ALL-REMAINING is the default: It selects all not yet installed supply units for installation.

You can check and confirm the selected supply units in a dialog box in menu mode by specifying SUPPLY-UNITS=*BY-DIALOG (see [Example in menu mode](#)).

(7) The installation is carried out for the current operating system version.

(8) The catalog ID 6A0B and the current default system ID are specified for the imported pubset.

No further operands need to be specified for a default installation under the current operating system. The following non default specifications need to be set (as in the example in menu mode):

(9) Files on the system that will be overwritten by new files of the same name should first be backed up to an LMS library. This library is called :6A0B:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P (cf. "[Example in menu mode](#)").

(10) The work files created by IMON are not to be deleted.

The statement is then executed using [DUE]:

```
% ARC0002 ANWEISUNG ANGENOMMEN. ARCHIVE-FOLGENUMMER 'A.100512.131703',  
VERSION '12.0C00' _____ (11)  
% ARC0033 ARCHIVE-SUBTASK MIT TSN '0APN' ERZEUGT  
% ARC0002 ANWEISUNG ANGENOMMEN. ARCHIVE-FOLGENUMMER 'A.100512.131719',  
VERSION '12.0C00'  
% ARC0033 ARCHIVE-SUBTASK MIT TSN '0APP' ERZEUGT  
% JMS0066 AUFTRAG 'IMON' ANGENOMMEN AM 21-04-12 UM 13:17, TSN = 0APQ (12)  
% JMS0066 AUFTRAG 'IMON' ANGENOMMEN AM 21-04-12 UM 13:17, TSN = 0APR
```

(11) IMON reads the MTC (via ARCHIVE), extracts files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries), and registers the delivery in the SCI.

(12) IMON then transfers the delivery documentation to the standard documentation library, which is created if necessary, and starts a print job for the acknowledge form (using the "IMON" job name).

Job ':6A0B:\$SYSSAG.<package-name>.APR121318292021.IE' is being generated. '5'

supply units have been selected _____ (13)

Generation of :6A0B:\$SYSSAG.<package-name>.APR121318292021.IE startet:

Generation of FSTAT-RENAME commands...

Generation of SAVE-FILES commands...

Generation of RESERVE-FILE commands...

Generation of IMPORT-FILE commands...

Generation of UPDATE-CATALOG-ENTRY commands...

Generation of library processing commands...

Generation of ADD-INSTALLATION-UNITS statements...

Generation of reference file...

SYSSII files processing started

Generation of ACTIVATE-UNIT commands for supply units:

EDT	17.0	D30
OPENFT	12.1	C10
OPENFT-CR	12.1	B10
PERCON	02.9	C00
SORT	08.0	A30

Job ':6A0B:\$SYSSAG.<package-name>.APR121318292021.IE' has been generated and

entered _____ (14)

//**end** _____ (15)

(13) The :6A0B:\$SYSSAG.<package-name>.APR121318292021.IE installation procedure will be generated:

- Five supply units were selected for installation.
- Various processing steps must be generated.
- Activation of the listed supply units is generated.

(14) The installation procedure is then fully generated and started automatically as a background procedure (task with the job name "INSTALL").

(15) IMON is terminated using the END statement.

The following SHOW commands are identical with those in the default installation on the home pubset. For space reasons, a detailed representation of the output was foregone as it contents differ from the home pubset only in the :6A0B: header of the imported pubset, if at all.

```
/show-job-status *job(install) _____ (16)
```

```
/show-job-status *tsn(0aps) _____ (17)
```

```
% EXC0755 INFORMATION ON TASK WITH TSN '0APS' CANNOT BE GIVEN
```

(16) The SHOW-JOB-STATUS command is used to check whether the installation task is still running (see also ["Example with SDF statements "](#)).

(17) Once the installation task has run, the SHOW-JOB-STATUS command query is quit with the message EXC0755, i.e. the task is terminated (see also ["Example with SDF statements "](#)).

```
/show-file-attributes :6a0b: $sysag. _____ (18)
```

```
/show-file :6a0b: $sysag. <package-name> .ic _____ (19)
```

(18) The SHOW-FILE-ATTRIBUTES command is used to display all files that were created under the work file ID (see also ["Example with SDF statements "](#)).

(19) The SHOW-FILE command is used to output the contents of the log file together with the most important console messages (see also ["Example with SDF statements "](#)).

```
File name: :6A0B:$SYSSAG.<package-name>.IC
```

After scrolling to the file end, the last message logged (IMO2021) shows that the installation was terminated normally.

Note

If manual post-processing is necessary, console messages to this effect will be displayed. The delivery information can tell you whether further postprocessing is required. It should be referred to once more in this context.

```
/start-imon input-interface=*sdf _____ (20)
```

```
//modify-imon-options software-inventory=:6a0b:$tsos.sys.imon.sci — — (21)
```

```
//show-supply-units
```

```
    unit-name=*from-solis2-del(
```

```
    package= <package-name>,user-code=sol2p),
```

```
    inf=*par(report=*summary) _____ (22)
```

```
/end _____ (23)
```

Once the installation procedure has run successfully, all supply units are registered in the SCI as installation units and the files of the products are stored on the system in the predefined depot location.

(20) IMON is called once more.

(21) The MODIFY-IMON-OPTIONS statement is used to define the foreign SCI.

(22) The SHOW-SUPPLY-UNITS statements is used to display all supply units of the previously installed delivery (package name, customer code SOL2P) that are now installed in the SCI (see also ["Example with SDF statements "](#)).

(23) IMON is terminated using the END statement.

3.1.3 Placement and activation in the default installation

Placement and activation are defined as follows in the default installation:

	In menu mode	Using the INSTALL-UNITS SDF statement
Placement		
Define default Placement	Via the menu: <i>Edit</i> , option 4 (<i>Install</i>): “Global installation parameters”: Placement mode: 1 (default; preset)	PLACEMENT-MODE= <u>*STD</u> (default; preset)
Overwrite existing files	Implicitly valid	REPLACE-OLD-FILES = <u>*YES</u> (default; preset)
Do not force user ID	Implicitly valid	FORCE-LOCATION = <u>*NO</u> (default; preset)
Activation		
Define default Activation	Via the menu: <i>Edit</i> , option 4 (<i>Install</i>): “Global installation parameters”: Activation preparation mode: 1 (default; preset)	ACTIVATION-MODE = <u>*STD</u> (default; preset)

Placement – specifying the depot location of the files to be installed

The file attributes of an installation item that is to be created are taken from the delivery information.

IMON reads the supply components (release items) of the supply units to be installed from the distribution medium and stores them as installation items on the target system. The file attributes of an installation item are taken from the product movement file of the SOLIS2 delivery. The depot location, i.e. the path name of the installation item is affected by the definitions in SYSSII file and other user input.

The path name is formed as follows in the default installation:

:<catid>:\$<userid>.<item-name>

where refers to the catalog ID of the pubset on which the supply units will be installed.

<catid> The catalog ID is taken from the target system specification
(TARGET-SYSTEM=*PARAMETERS(...,PUBSET=<catid>).

<userid> refers to the installation ID that is specified in the following order:

1. User ID that was defined explicitly for the delivery component in the customer-specific IMON parameter file (see [section "IMON parameter files" "IMON parameter files "](#)).
2. User ID that was defined explicitly in the product movement file for the supply component.
3. User ID that was defined explicitly in the product movement file for the supply group.
4. The default user ID of the current system or the default user ID specified for a foreign system.

<item-name> refers to the installation item.

If no such file exists, it is created with the file attributes defined for the supply component.

Existing files of the same name are overwritten without further user input (default is REPLACE-OLD-FILES=*YES; for more information on the other setting options, see "Customer-specific installation", "[Placement and activation in the customer-specific installation](#) "). The old file attributes are retained.

The following applies if a file cannot be overwritten because of a file lock, for example:

- If the file is a syntax or message file (installation items of the type SDF and MES), the installation procedure tries to cancel the file lock. If this attempt is unsuccessful, the procedure is canceled and a message that requires an answer is output on the operating console. The operator's response governs the subsequent processing:
 - "Repeat" The installation procedure attempts once again to cancel the file lock.
 - "Ignore" The installation is continued despite the file lock, i.e. the affected installation item is not installed.
 - "Abort" The installation is aborted with an error.
- The installation is canceled immediately if another installation item cannot be overwritten.

Handling password-protected files

The files remain password-protected. This is also the case for locked syntax and message files that are further processed automatically by IMON.

Handling libraries

Libraries (installation items of the type MOD, MAC, SRC, PL*, PLM, PLR and PLS) are handled as files in terms of the depot location and the installation item name.

An exception to this are libraries that are a subset of another library. These do not have to be installed under the defined path name, instead the elements they contain must be merged into the defined library. During the installation, these libraries are stored under the defined path in an intermediate step. The elements are then copied to the other library.

An alternative library that does not yet exist is configured in the target system with its elements and flagged as "mixed". If this library already exists, versions of elements for which a new element exists in the supplied library are deleted. The new elements of the supplied library are then added once more to the existing library. The library updated in this manner is then flagged as "mixed".

The installation is aborted if the library cannot be accessed (e.g. file lock).

Handling procedure files

Procedure files (installation items of the type DO or ENT) as handled in the same way as the other item types in the default installation.

Handling installation items of the type NST or %xx

Supply components that are not BS2000 files (e.g. publications, data volumes) are ignored by IMON.

Handling dummy items

Dummy items (installation items of the type *DF with full path names or of the type *DP with partial path names) are part of the delivery information.

They are generated during the installation or exist already under the installation ID. While dummy items are entered in the SCI with their logical names, they are not assigned a path name automatically.

In the following cases, a path name is entered during the installation process, where the correction version of the dummy item must always be <= the correction version of the installation item:

1. The delivery is a first-time delivery:

- The following applies if the dummy item is a SYSREP file: If the relevant release unit is assigned an RMS file and a default path name is defined in this RMS file, the first path name found in the RMS file is entered in the SCI. The path name of the generated loader is entered once RMS processing is concluded.
- If a default path name is defined in the product movement file, a new path name is formed from the catalog ID of the target system and the default path name. The default system ID is used if the default path name does not contain a user ID.

With this type of processing, path names that were transferred from RMS files are also overwritten by dummy items of SYSREP files. The path name of the generated loader is entered however once RMS processing is concluded.

2. The delivery is a correction delivery:

The SCI does not as yet contain a path name for the dummy item:

- The following applies if the dummy item is a SYSREP file: If the relevant release unit is assigned an RMS file and a default path name is defined in this RMS file, the first path name found in the RMS file is entered in the SCI.

The path name of the generated loader is entered once RMS processing is concluded.

- The path names are not updated if a default path name is defined in the product movement file.

The SCI already contains a path name for the dummy item:

- The path name in the SCI is retained regardless of whether or not the dummy item is contained in the correction delivery.

Handling SYSSII files

The SYSSII files are first transferred to the target system, merged with the PLAM library SOLLIB.IMON.SYSSII under the work file ID and then deleted again.

You can suppress transfer of the SYSSII files to the target system if you use an IMON parameter file (see [section "IMON parameter files"](#)).

Central repositories for certain files

When you use an IMON parameter file you can specify central storage locations for documentation files and subsystem declarations (see [section "IMON parameter files" "IMON parameter files"](#)).

Activation – preparing for activation of the software to be installed

Syntax files

- IMON enters syntax files in the SDF parameter file of the target pubset without the need for further user input, IMON processing the SDF parameter file with the default names \$TSOS.SYSPAR.SDF.

If another SDF parameter file is activated in the target system, IMON first saves a copy under the default name and then processes the parameter file with the default name.

The next system start is entered as the activation time (corresponds to MODIFY-SDF-PARAMETERS with SCOPE=*NEXT-SESSION). IMON obtains the name of the subsystem and the syntax file type from the product movement file.

If the DSSM catalog contains a definition for the syntax file of the subsystem, IMON does not perform activation but the syntax file is automatically activated by DSSM.

Message files

Message files (installation items of the type MES) can be activated at different times. The activation time is queried by IMON.

Message files of subsystems that were loaded with or before DSSM are activated via the Class 2 system parameter without the need for further user input:

- The BS2CP message file SYSMES.BS2CP.170 (in BS2000 V8.0) corresponds to the default value of the MSGFIL01 system parameter. It is simply installed. If it is already activated, it is first deactivated, installed, and then reactivated.
- The message files of the release units BCAM, CALENDAR, DSSM, FITC, MIP, and SRPMNUC are merged in the global message file that is defined in the MSGFIL02 system parameter (default is \$TSOS.SYMES.EKP.01). If the global message file is already activated, it is first deactivated, extended, and then reactivated.

All other message files are entered by IMON in the MIP parameter file of the target pubset.

The next system start is entered as the activation time (corresponding to the command MODIFY-MIP-PARAMETERS with SCOPE=*NEXT-SESSION).

Note on reinstallation of syntax and message files which have already been activated

In the course of reinstallation IMON deactivates the file concerned, renames the file <file>.nnn (suffix nnn = 001 through 999), and reactivates the renamed file. Subsequently the new component is installed under its standard name. When the next startup takes place, the syntax and message files with a suffix are automatically deleted.

Subsystem

IMON modifies the static subsystem catalog in preparation for activation of a subsystem (installation item SSC /SSD). To do this, IMON ascertains the path name of the static subsystem catalog (default is \$TSOS.SYS.SSD.CAT.X on the target pubset) and requests its confirmation. In the case of subsystems for which a remove option exists, you can specify whether an earlier version of the subsystem should be removed prior to the installation (default: earlier version is removed).

A backup copy is created under the name <subsystem-catalog>.<time-stamp> before the subsystem catalog is modified.

IMON also generates a subsystem catalog source file under the name <subsystem-catalog>.SRC. It can be used as a basis from which to regenerate the static subsystem catalog in its default format at any time (e.g. if problems occur after the subsystem settings are changed).

Once the subsystem catalog is generated, IMON searches for a procedure with the name <subsystem-catalog>.USER-MODIF. If this procedure exists, it is started using the CALL-PROCEDURE command. You can use this procedure to define repeated installation activities.

RMS files

In the case of an installation where no user input is required and the delivery was not first parked, system corrections are transferred to the RMS depot. IMON generates the rep loader. IMON enters the logical names and the path names generated by RMS in the SCI.

POSIX satellites

The POSIX items identify the commands that IMON must provide to the /START-POSIX-INSTALLATION program. This program is called at POSIX subsystem start.

For each POSIX item, IMON writes an “add product” command in the file “\$SYSROOT.IMON.ACTIONS.ADD”. Any old version of the same product must be removed from the POSIX system. Therefore, before adding a new one, IMON retrieves from the POSIX configuration file (\$SYSROOT.POSIX-CONFIGURATION) the old version(s) of this product that must be removed. The associated “remove” commands are stored in “\$SYSROOT.IMON.ACTIONS.REM”.

! CAUTION!

The following must be borne in mind before restarting the POSIX subsystem:

- The POSIX installation program can access the installation units registered in the SCI only if these are in the LOCKED=NO status, i.e. after the installation units have been activated.

- If activation has not yet taken place, access is not possible.

In this case the \$SYSROOT.IMON.ACTIONS.ADD and \$SYSROOT.IMON.ACTIONS.REM files must be saved under another name before the POSIX subsystem is restarted.

Procedures

Procedures (type DO and ENT) that are to run after first querying the installation parameters, are started automatically by SOLIS without the need for further user input in the case of installation on the home pubset. Batch procedures (type ENT) are also deleted automatically once they run.

In the case of installation on an imported pubset, these procedures must be started manually as part of postprocessing. Procedures of the type DO are started using the ENTER-PROCEDURE or CALL-PROCEDURE command, procedures of the type ENT are started using the ENTER-JOB command.

In both cases the names of the procedures to be started manually are output to SYSOUT when the Installation procedure is generated or to the console when the installation procedure is running (see also the console log).

3.2 Customer-specific installation under BS2000

In a customer-specific installation, the entire delivery or only a part thereof is stored on the home pubset or on another, imported pubset. In contrast to a default installation, in a customer-specific installation the depot location and activation mode can differ from the default.

Parking and single or multiple installation from the park ID is also considered to be a type of customer-specific installation, whereby installation from the park ID itself can be carried out as a default installation.

System environment and requirements for the customer-specific sample installations

The following settings and requirements are valid for the customer-specific installations described on pages [123 \(Installation on the home pubset with customizing \)](#) through [191 \(Example with SDF statements \)](#). The first table lists the general system environment and the features of the SOLIS2 delivery.

Current system	MM.N
Target system of the installation	MM.N
Package name of the SOLIS2 delivery	<package-name>
Customer identifier	SOL2P
Data volume identifier	QE6171
Device type	TAPE-C4
Supply units contained	EDT, OPENFT, OPENFT-CR, PERCON, SORT

MM.N - current version

The next two tables detail the differences between the IDs and file names for an installation on the home pubset, an installation where the supply units are first parked, and a multiple installation.

For an installation on the home pubset

Catalog ID of the home pubset	I29A
Standard SCI	:I29A:\$TSOS.SYS.IMON.SCI
Default system ID	TSOS
Work file ID	SYSSAG on the home pubset
Path name for work files	:I29A:\$SYSSAG.
Library for the supply documentation	:I29A:\$SYSSAG.<package-name>.SOL2P.DOC
Backup library	:I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P
Installation procedure	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IE

Log for important message files	:I29A:\$SYSSAG.<package-name>.IC
Installation log	:I29A:\$SYSSAG.<package-name>.<time-stamp>.IL
Modified installation ID for the EDT and PERCON supply units	OSID1 (instead of the default system ID TSOS)
Modified location of the RMS depot	:I29A:\$OSD6
Handling of earlier subsystem versions	Entries from earlier versions remain in the DSSM catalog

For an installation where the supply units are first parked

Catalog ID of the home pubset	I29A
Standard SCI	:I29A:\$TSOS.SYS.IMON.SCI
Park SCI (foreign SCI for parking)	:I29A:\$PARKSW1.SYS.IMON-PARK.SCI
Default system ID	TSOS
Work file ID	PARKSW1 on the home pubset, with WORK prefix
Park ID	PARKSW1 on the home pubset
Path name for work files	:I29A:\$PARKSW1.WORK.
Library for the supply documentation	:I29A:\$PARKSW1.<package-name>.SOL2P.DOC
Backup library	:I29A:\$PARKSW1.WORK.IMON.SAVE.LIB.<package-name>.SOL2P
Park procedure	:I29A:\$PARKSW1..WORK.<package-name>.<time-stamp>.PE
Log for important console messages	:I29A:\$PARKSW1.WORK.<package-name>.IC
Park log	:I29A:\$PARKSW1.WORK.<package-name>.<time-stamp>.PL

For multiple installations (of parked software)

Catalog ID of the home pubset	I29A
Standard SCI	:I29A:\$TSOS.SYS.IMON.SCI
Park SCI (foreign SCI for parking)	:I29A:\$PARKSW1.SYS.IMON-PARK.SCI
Default system ID	TSOS

Catalog ID of the imported pubset	6A0B
Default system ID for the installation on the imported pubset	TSOS
Work file ID	PARKSW1 on the home pubset, with WORK prefix
Park ID	PARKSW1 on the home pubset
Path name for work files	:I29A:\$PARKSW1.WORK.
Backup library	:I29A:\$PARKSW1.WORK.IMON.SAVE.LIB.<package-name>.SOL2P
Installation procedure	:I29A:\$PARKSW1.WORK.<package-name>.<time-stamp>.IE
Log for important console messages	:I29A:\$PARKSW1.WORK.<package-name>.IC
Installation log	:I29A:\$PARKSW1.WORK.<package-name>.<time-stamp>.IL

The <time-stamp> part of the name is the timestamp of the installation in the format <month><day><time><year>, where

<month> is the first three letters of the month

<day> is a one or two-digit day of the month

<time> is the time at which the installation started in the form hhmmss

<year> is a four-digit figure

General steps in a customer-specific installation (parking also)

Call IMON

Depending on the call, IMON branches to menu mode or waits for SDF statements.

Open SCI

The standard SCI \$TSOS.SYS.IMON.SCI on the home pubset is opened implicitly (in the SDF mode when started, in menu mode when the delivery is opened). If a foreign SCI is to be created/written, it must be opened explicitly. This procedure is presented in [section "Installation on the imported pubset"](#)

Define work file ID

IMON creates work files on the home pubset under the SYSSAG user ID by default.

Open SOLIS2 delivery

ARCHIVE reads the data volume. Information on the package name, user code, VSN, and volume type from the accompanying information is needed during this process. A delivery that is already open can be opened as a registered delivery.

**Confirm
supply
units**

All supply units are offered for further processing by IMON, confirm this selection with [DUE]

To install or park parts of the delivery only, modify the selection accordingly and confirm it with [DUE].

When a registered delivery is opened, all supply units not yet parked or installed are offered for further processing.

**Select
install
or park**

The option 4 (*Install*) for installing the selected supply units is preset in the *Edit* menu once you select the supply units in menu mode.

**Select
parameter
file**

The default parameter file SYSPAR.IMON.LAST of the current environment is preset. If this file exists, IMON uses the specific installation parameters stored there as defaults in the subsequent dialog boxes

**Specify
target
system**

The operating system version and the home pubset of the active system are preset.

**Customize
depot
location**

The depot location (installation or park ID) is adapted to the customer's requirements. (Placement: part of the customizing process)

**Customize
activation
parameters**

The activation parameters are adapted to the customer's requirements.

(Activation: part of the customizing process)

This point is omitted if the supply units are parked.

IMON generates and starts the installation or park procedure. It either loads all of the files in the supplied products to your specified location and registers them in the SCI as installation units or the products are registered in the SCI as supply units with the status "parked".

**Close
SOLIS2
delivery**

The SOLIS2 delivery is closed explicitly if IMON is to be used again (e.g. to check the SCI). Otherwise IMON can be terminated directly.

**Terminate
IMON**

If the SOLIS2 delivery was not already closed, it is closed implicitly when IMON is terminated.

**Any
postprocessing**

Any necessary postprocessing and the order in which it should be carried out can be found in the delivery information.

This point is omitted if the supply units are parked.

The subsequent installation from the park ID can be carried out as a default installation or a customer-specific installation.

The schematic representation of the sequence can be found on "[Customer-specific installation on the home pubset with customizing](#)" (for the home pubset with customizing), "[Customer-specific installation with prior parking](#)" (for installation with parking) or "[Sequence of a customer-specific multiple installation of a previously installed delivery](#)" (for a multiple installation).

3.2.1 Installation on the home pubset with customizing

The schematic representation of the sequence can be found on "[Customer-specific installation on the home pubset with customizing](#)".

3.2.1.1 Example in menu mode

Call IMON

IMON is started once English is set as the task-wide language for messages and masks. IMON switches to menu mode once the program is loaded and [DUE] entered.

```
/modify-msg-attributes task-language=e
/start-imon
% IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from file
':I29A:$TSOS.SYSLNK.IMON-BAS.034.TU'
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved
PLEASE ACKNOWLEDGE
```

The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI is used.

```
File Edit Show View Options
-----
IMON: Welcome screen
-----

*** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----

Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

Open SCI

The standard SCI on the home pubset is to be opened: The 1 (*Open ...*) option is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : : Welcome screen
: * Close : -----
: * Print documentation ... :
: * Save as ... : e in Installation MONitor ***
: 5. Exit :
: .....:
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

In the *Open* menu, select 1 (*Standard SCI*) and confirm with [DUE].

```

File Edit Show View Options
-----
: .....:
: 1 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... : -----
: 3. Installation definition file ... :
: 4. SOLIS2 delivery ... : ***
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel
:
: .....:
File: Open: Standard SCI
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The screen displays the contents of the open standard SCI. It contains 163 entries, i.e. 163 installation units that can be viewed by scrolling backwards and forwards using +/- in the command line. Before you can select a menu function, you may need to delete from the command line any presetting caused by the scrolling operation.

File Edit Show View Options

IMON: SCI: :I29A:\$TSOS.SYS.IMON.SCI

Unit name	Installation	Units Version	units selection 1 through Corr state	13 of 168 More: +
ACS		21.0	A00	
AID		03.5	A00	
AIDSYS		21.0	A00	
AIDSYSA		21.0	A00	
ANITA		21.0	A00	
APACHE		02.4	A00	
ARCHIVE		12.0	C00	
ASE		21.0	A00	
ASSEMBH		01.4	A10	
ASSEMBH-GEN		01.4	A10	
ASSTRAN		05.4	A00	
ASTI		21.0	A00	
ATS		05.3	A20	

Command ==>

F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

Define work file ID

The following steps can be skipped when the default value for the work file ID is kept, as is the case in this example (continue then with "Open SOLIS2 Delivery", "[Example in menu mode](#)"). The work file ID can be respecified by entering 2 (IMON Options) in the Options menu. Execute the mask using [DUE].

```

File Edit Show View Options
-----
IMON: SCI: :I29A: : 2 1. Switch to SDF interface :
----- : 2. IMON options ... : -----
: ..... ough 13 of 168
: ..... Installation units selection More: +
Unit name Version Corr state
ACS 21.0 A00
AID 03.5 A00
AIDSYS 21.0 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The preset work file ID is displayed as a partial path name with a catalog ID (in this case with :I29A:\$SYSSAG.). This default work file ID is accepted by entering [DUE].

```

File Edit Show View Options
-----
: ..... IMON options : -----
: .....
: Work file location: :I29A:$SYSSAG. : 13 of 168
: Reference file ...: : More: +
: :I29A:$TSOS.SYS.IMON.SCI.REF :
: .....
: F1=Help F12=Cancel :
: .....
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Open SOLIS2 delivery

Now the delivery will be opened: The 1 (*Open ...*) option is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : IMON.SCI
: 2. Close :
: *. Print documentation ... :
: 4. Save as ... : tion units selection 1 through 13 of 163
: 5. Exit : Version Corr state More: +
:.....:
AID 21.0 A00
AIDSYS 03.5 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Option 4 (SOLIS2 delivery) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: .....:
: Open :
: .....:
: 4 *. Standard software configuration inventory : gh 13 of 163
: *. Foreign software configuration inventory ... : More: +
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel :
: .....:
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following delivery information is then queried in the “SOLIS2 delivery” dialog box that is opened: package name, user code, distribution medium (MTC is preset), the VSN, and the MTC device type. The set work file ID is displayed again in the lower section of the mask. This dialog box is executed using [DUE].

```

File Edit Show View Options
:-----:
:                               Open                               :
:-----:
:                               SOLIS2 delivery                    :
:-----:
: Package name.....: <package-name>                             :
: User code.....: sol2p                                           :
: Distribution medium: 1 1. SOLIS2 volume                          2. Library           :
:                               3. Registered                        4. Local volume      :
:                               Volume.....:                          :
qe6171                               :
:                               Device type: tape-                 :
c4                                     :
: Documentation library: 1 1. Standard                            2. Other             :
:                               Name:                               :
: Work file location: :I29A:$SYSSAG.                               :
: Reference File ...: :I29A:$TSOS.SYS.IMON.SCI.REF                :
:                               :                                   :
: Supply unit selection 1 1.Manually (*BY-DIALOG)                 :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                             :
:-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

IMON then has all of the information needed to open the delivery on the data medium and it starts the processing operation, during which some messages are output to the screen:

IMON instructs ARCHIVE to read the specified MTC and extract files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries). The "IMON" jobs started for this purpose create or expand the documentation library and start a print job for the acknowledge form. You can switch back to IMON menu mode using [DUE].

```

% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142557', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0183' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142610', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0184' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0185
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0186
%PLEASE ACKNOWLEDGE

```

Confirm supply units

Five supply units were found on the specified MTC. IMON offers these for further processing (they are already highlighted). To select all of these supply units, execute the selection using [DUE].

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name          SU selection  Units  1 through  5 of  5
                   Version      Corr state
X EDT              17.0        D30
X OPENFT           12.1        C10
X OPENFT-CR        12.1        B10
X PERCON           02.9        C00
X SORT             08.0        A30
*** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select install

You must then switch to the *Edit* menu to install the units. Option 4 (*Install*) is already preset and needs only to be confirmed with [DUE].

```
File Edit Show View Options
-----
IM : 4 *. Search ... : code: SOL2P
---- : *. Add ... :
: *. Remove ... : gh 5 of 5
: 4. Installed ... : More:
: 5. Park ... : te
X EDT : *. Generate installation definition file ... :
X OPE : *. Deinstall ... :
X OPE : *. Undo ... :
X PER : *. Activate ... :
X SOR : *. Check ... :
: *. Request correction delivery ... :
: *. Customer-Approved Install ... :
:.....:
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select parameter file

The following default installation parameters options are preset by IMON in the “IMON parameter file” dialog box.

```

File Edit Show View Options
:-----
:                               IMON Parameter file :
:-----
: Parameter file ...: 2 1. None          2. Standard      3. Other      : 5
: File name: :I29A:$TSOS.SYSPAR.IMON.LAST
:
: Save parameters...: 1 1. No            2. Standard      3. Other      :
:   File name:
:
: F1=Help F12=Cancel
:-----
:                               *** End of SU selection ***
:-----

Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

The following default installation settings are preset by IMON in the “Global installation parameters” dialog box.

Specify target system

```

File Edit Show View Options
:-----
:                               Global installation parameters
:-----
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 1 1. Default          2. Other
:   Save old files.....: 1 1. No              2. With LMS
:                               3. With ARCHIVE      4. With MAREN
:
:                               VSN.....:
:                               Device type:      MAREN location: *STD
: Undo preparation.....: 2 1. Yes                2. No
: Placement mode.....: 1 1. Standard          2. Other
: Activation preparation mode: 1 1. Standard        2. Other
: Password file.....: 2 1. Yes                2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes                2. No
: Start.....: 1 1. Immediately          2. By user
: Configuration checks.....: 1 1. Yes                2. No
: Work file deleting.....: 1 1. Yes                2. No
:
: F1=Help F12=Cancel
:-----

```

MM.N – current version

Input for this example

Target system:

- The supply units are to be installed for BS2000 MM.N (preset).
- The supply units are to be installed on the default pubset, i.e. the home pubset (preset)

Save old files:

- A "2" or "3" must be entered here to back up existing files that would be overwritten by the installation operation. Backup to an LMS library (option "2") was selected for this example. The library will be created with the following name under the work file ID:

`IMON.SAVE.LIB.<package name>.<customer ID>`

i.e. in this example `:I29A:$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P`

Undo preparation:

- A "1" must be specified here to create a backup that will allow you to restore the original state before installation (see [section "Undo - undoing an installation"](#)). The preset value "2" (i.e. no backup) was left unchanged in the example.

Placement mode:

- The default placement mode should be changed to the customer's requirements, therefore enter "2" here.

Activation mode:

- The default activation mode should be changed to the customer's requirements, therefore enter "2" here.

Password file:

- This specification need not be changed

Print log files:

- This specification need not be changed: The logs are not to be printed out automatically once the installation is complete.

Start:

- This specification need not be changed: The installation procedure generated by IMON is to be started immediately and automatically.

Configuration check:

- This specification need not be changed: A configuration check should be executed.

Work file deleting:

- The work files generated under the work file ID by IMON during the installation are not to be deleted once the installation is complete. Therefore the default setting must be changed to "2".

The dialog box is executed with these entries and [DUE].

```

File Edit Show View Options
:-----:
:                               Global installation parameters                               :
:-----:
: Target system                                                            :
:   BS2000 version.....: MM.N                                           :
:   Standard subset.....: 1 1. Default                   2. Other                :
: Save old files.....: 1 1. No                         2. With LMS             :
:                               3. With ARCHIVE                          4. With MAREN           :
:                               VSN.....:                                 :
:                               Device type:           MAREN location: *STD :
: Undo preparation.....: 2 1. Yes                         2. No                   :
: Placement mode.....: 2 1. Standard                   2. Other                :
: Activation preparation mode: 2 1. Standard                   2. Other                :
: Password file.....: 2 1. Yes                         2. No                   :
:   File name.....:                               :
:   Read password: C'                               :
: Print log files.....: 2 1. Yes                         2. No                   :
: Start.....: 1 1. Immediately                   2. By user              :
: Configuration checks.....: 1 1. Yes                         2. No                   :
: Work file deleting.....: 1 1. Yes                         2. No                   :
:                               :
: F1=Help  F12=Cancel
:-----:

```

Additional specifications are requested to select Installation items that belong to a certain hardware variant or a certain target system version with the “Target system parameters” dialog box.

The preset is 4 (All), i.e. the selection is made regardless of the hardware variant. The defaults displayed in the example are confirmed without change using [DUE].

```

File Edit Show View Options
:-----:
:                               Global installation parameters                               :
:-----:
:                               Target system parameters                               :
:-----:
: Processor-type: 4 1. CISC-390                                           2. Other                :
:                               *  ---                                     2. With LMS             :
:                               3. X86                                     4. With MAREN           :
:                               4. All                                     :
:                               : MAREN location: *STD                   :
: Item-selection: 2 1. All                                               2. No                   :
:                               2. Target-version-only                   2. Other                :
:                               :                                         2. Other                :
:                               :                                         2. No                   :
:                               F12=Cancel                               :
:                               :                                         :
:                               :                                         2. No                   :
: Print log files.....: 2 1. Yes                         2. No                   :
: Start.....: 1 1. Immediately                   2. By user              :
: Configuration checks.....: 1 1. Yes                         2. No                   :
: Work file deleting.....: 2 1. Yes                         2. No                   :
:                               :
: F1=Help  F12=Cancel
:-----:

```

Customize depot location

The following screen shows the placement parameters of the 5 supply units to be installed:

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Supply units placement parameters
Units 1 through 5 of 5
More:
Replace Force
Unit name Vers Corr Catid Userid Prefix files Loc.
EDT 17.0 D30 *DEF TSOS *NONE Y N
OPENFT 12.1 C10 *DEF *STD *NONE Y N
OPENFT-CR 12.1 B10 *DEF *STD *NONE Y N
PERCON 02.9 C00 *DEF TSOS *NONE Y N
SORT 08.0 A30 *DEF *STD *NONE Y N
*** End of Supply units placement parameters
***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The installation ID "TSOS" is changed for the supply units EDT and PERCON to the user ID "OSID1". All other presets are left unchanged:

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Supply units placement parameters
Units 1 through 5 of 5
More:
Replace Force
Unit name Vers Corr Catid Userid Prefix files Loc.
EDT 17.0 D30 *DEF osidl *NONE Y N
OPENFT 12.1 C10 *DEF *STD *NONE Y N
OPENFT-CR 12.1 B10 *DEF *STD *NONE Y N
PERCON 02.9 C00 *DEF osidl *NONE Y N
SORT 08.0 A30 *DEF *STD *NONE Y N
*** End of Supply units placement parameters
***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

There is a library for each of the EDT and SORT supply units that is a subsets of an alternative library. The libraries are to be installed and merged with the alternative library. To accomplish this, confirm the preset "Y" in the "Library" column without changing it:

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                Units      1 through      2 of      2
                Supply units placement parameters
                More:
Unit name      Vers      Corr      Librairies
EDT            17.0     D30      Y
SORT          08.0     A30      Y
                *** End of Supply units placement parameters ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Customize activation parameters

The screen below shows the activation parameters of the delivery to be installed:

```

                Global activation preparation parameters
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Syntax file processing.: 1 1. Yes                2. No
SDF param file.: :I29A:$TSOS.SYSPAR.SDF
Message file processing: 1 1. Yes                2. No
MIP param file.: :I29A:$TSOS.SYSPAR.MIP.210
MES file.....: :I29A:$TSOS.SYSMES.EKP.01
DSSM processing.....: 1 1. Yes                2. No
Catalog name.: :I29A:$TSOS.SYS.SSD.CAT.X
Keep old version: 2 1. Yes                2. No
RMS processing.....: 1 1. Depot+loaders  2. Depot only  3. No
Depot location.: 1 1. Standard          2. Enforced
Location: :I29A:$TSOS
POSIX processing.....: 1 1. Cfg          2. All          3. No
F1=Help F12=Cancel

```

This option is changed from the default, i.e. the earlier DSSM versions are to be retained for the entire delivery and the user ID is changed from TSOS to USR for the RMS depot location:

```

Global activation preparation parameters
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Syntax file processing: 1 1. Yes                2. No
SDF param file.: :I29A:$TSOS.SYSPAR.SDF

Message file processing: 1 1. Yes                2. No
MIP param file.: :I29A:$TSOS.SYSPAR.MIP.210
MES file.: :I29A:$TSOS.SYSMES.EKP.01

DSSM processing.: 1 1. Yes                2. No
Catalog name.: :I29A:$TSOS.SYS.SSD.CAT.X
Keep old version: 1 1. Yes                2. No

RMS processing.: 1 1. Depot+loaders  2. Depot only  3. No
Depot location.: 1 1. Standard      2. Enforced
Location: :I29A:$usr.

POSIX processing.: 1 1. Cfg          2. All          3. No

F1=Help F12=Cancel

```

The following screen then once more displays the activation parameters for the 5 supply units to be installed. The values can be changed separately for each supply unit. In this case however, they are confirmed unchanged:

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                Units      1 through      5 of      5
                Supply units activation preparation parameters
                More:
Unit name      Vers      Corr      SDF      MSG      SUB      PSX
EDT             17.0     D30      Y        Y        Y        -
OPENFT         12.1     C10      Y        Y        Y        -
OPENFT-CR      12.1     B10      -        -        -        -
PERCON         02.9     C00      Y        Y        Y        -
SORT           08.0     A30      Y        Y        Y        -
*** End of Supply units activation preparation parameters
***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following screen then shows once more the supply units for which earlier subsystem versions will be retained. In terms of this activation parameter, in this case all of the displayed supply units are marked (here with "x") and the selection is confirmed with [DUE].

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
              Units      1 through      4 of      4
SU selection for keeping old subsystem versions      More:
Unit name      Version      Corr state
X EDT          17.0          D30
X OPENFT      12.1          C10
X PERCON      02.9          C00
X SORT        08.0          A30
*** End of selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

**IMON
output**

IMON has then queried all of the information needed to install the delivery and starts the processing operation during which some messages are output on the screen:

- The :I29A:\$SYSSAG.<package-name>.APR101149532021.IE installation procedure will be generated:
 - five supply units were selected for installation
 - various processing steps must be generated
 - activation of the listed supply units is generated.
- The installation procedure is then fully generated. It is started immediately and automatically.

```

Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' is being generated. '5'
supply units have been selected
Generation of :I29A:$SYSSAG.<packaga-name>.APR91450162021.IE startet:
Generation of FSTAT-RENAME commands...
Generation of SAVE-FILES commands...
Generation of RESERVE-FILE commands...
Generation of IMPORT-FILE commands...
Generation of UPDATE-CATALOG-ENTRY commands...
Generation of library processing commands...
Generation of ADD-INSTALLATION-UNITS statements...
Generation of reference file...
SYSSII files processing started
Generation of ACTIVATE-UNIT commands for supply units:
EDT          17.0      D30
OPENFT      12.1      C10
OPENFT-CR   12.1      B10
PERCON      02.9      C00
SORT        08.0      A30
Job ':I29A:$SYSSAG.<package-name>.APR91450162021.IE' has been generated and
entered
PLEASE ACKNOWLEDGE

```

Checking the installation process

IMON switches back to menu mode once [DUE] is entered.

The installation procedure runs asynchronously and outputs to the console messages relating to the processing steps executed and any errors that arose. The procedure sequence can be checked in the installation log.

The SHOW-JOB-STATUS command can be used in the interactive task to check whether the started installation task (job name "INSTALL") is already finished. The procedure sequence can be checked in the installation log.

As the installation procedure is still running asynchronously, the supply units to be installed are in an undefined state and are not displayed in this mask. If a subset of the supply units in the delivery were installed and not the complete set, this mask would display those supply units that were not selected.

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units    0 through    0 of    0
                                More:
Unit name      SU selection  Version      Corr state
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Close SOLIS2 delivery

The SOLIS2 delivery is closed by entering 2 (*Close ...*) in the *File* menu and [DUE].

```
File Edit Show View Options
-----
: 2 *. Open ... : name: <package-name> User code: SOL2P
: 2. Close ... : -----
: 3. Print documentation ... : Units    0 through    0 of    0
: 4. Save as ... : selection
: 5. Exit ... : Version      Corr state
:.....: SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Checking the SCI

The SOLIS2 delivery is closed by entering 2 (*Close ...*) in the *File* menu and [DUE].

168 supply units are then displayed in the SCI. The newly entered supply units are sorted in alphabetical order. They can be checked by scrolling forwards using “+” in the command line and [DUE] or [F8].

In the following screen, the contents were scrolled forward to the new EDT entry. The EDT installation unit is entered in the SCI with its version and correction state.

```
File Edit Show View Options
-----
IMON: SCI: :PH21:$TSOS.SYS.IMON.SCI
-----
Unit name          Installation units selection  Units 37 through 49 of 168
                    Version          Corr state  More: + -
DPRINTCL           01.2          A00
DPRINTCM           01.2          A00
DPRINTSV           01.2          A00
DRIVE              03.1          A10
DRIVE-COMP         03.1          A10
DRIVE-COMP-DOC     03.1          A10
DRIVE-COMP-LZS     03.1          A10
DRV                03.2          A02
DSSM               21.0          A00
DWS               11.0          A01
EDT               17.0          D30
ELFE              21.0          A00
ELSA              01.7          A10
-----
Command ==> +
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Naturally all other installation units in the delivery were entered also (cf. also the output after reopening the delivery, "[Example in menu mode](#)"). It was not possible to show a complete reproduction of the SCIs with the relevant masks here due to the space restrictions.

Terminate IMON

Installation of the SOLIS2 delivery is then complete. IMON itself can be terminated if no other IMON functions are needed (e.g. displaying information or carrying out other installations). IMON is terminated by entering 5 (Exit) in the *File* menu and [DUE] (it can also be terminated via the [F3] function key):

```
File Edit Show View Options
-----
: 5 1. Open ... : IMON.SCI
: 2. Close :
: *. Print documentation ... :
: 4. Save as ... : tion units selection  Units 37 through 49 of 168
: 5. Exit : Version          Corr state  More: + -
: .....:
DPRINTCM           01.2          A00
DPRINTSV           01.2          A00
DRIVE              03.1          A10
DRIVE-COMP         03.1          A10
DRIVE-COMP-DOC     03.1          A10
DRIVE-COMP-LZS     03.1          A10
DRV                03.2          A02
DSSM               21.0          A00
DWS               11.0          A01
EDT               17.0          D30
ELFE              21.0          A00
ELSA              01.7          A10
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

**Checking
the work
file ID**

Files remain under the SYSSAG work file ID once the installation is concluded. As stipulated in the “Global installation parameters” dialog box, they were not deleted automatically.

```
/show-file-attributes $syssag.  
48 :I29A:$SYSSAG.A.SOLIS.B.<package-name>.SOL2P  
15744 :I29A:$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P  
9 :I29A:$SYSSAG.RMS.OPT.ADM  
54 :I29A:$SYSSAG.RMS.OPT.DOC  
327 :I29A:$SYSSAG.SOLIS.LMSCONV  
24 :I29A:$SYSSAG.SOLLIB.IMON.PBD  
24 :I29A:$SYSSAG.SOLLIB.IMON.SYSSII  
420 :I29A:$SYSSAG.SOLLIB.IMON.020  
3939 :I29A:$SYSSAG.SOLLIB.SOLIS2.050  
3939 :I29A:$SYSSAG.SOLLIB.SOLIS2.050.RMS.071  
495 :I29A:$SYSSAG.SOLPAR.IMON.034.GEN  
1596 :I29A:$SYSSAG.SOLPAR.SOLIS2.050.AUSGEN  
72 :I29A:$SYSSAG.SOLPAR.SOLIS2.050.FMSG  
306 :I29A:$SYSSAG.SOLPRG.SOLIS2.050.GEN  
3 :I29A:$SYSSAG.SYSREP.EDT.170  
12 :I29A:$SYSSAG.SYSREP.OPENFT.121  
3 :I29A:$SYSSAG.SYSREP.PERCON.029  
12 :I29A:$SYSSAG.SYSREP.SORT.080  
6 :I29A:$SYSSAG.&lt;package-name>.DP  
303 :I29A:$SYSSAG.<package-name>.IC  
9 :I29A:$SYSSAG.<package-name>.APR91450142010.DP  
6 :I29A:$SYSSAG.<package-name>.APR91450162010.IA  
99 :I29A:$SYSSAG.<package-name>.APR91450162010.IE  
15 :I29A:$SYSSAG.<package-name>.APR91450162010.II  
210 :I29A:$SYSSAG.<package-name>.APR91450162010.IL  
9 :I29A:$SYSSAG.<package-name>.APR91450162010.IP  
30 :I29A:$SYSSAG.<package-name>.APR91450162010.IR  
3 :I29A:$SYSSAG.<package-name>.APR91450162010.IO  
456 :I29A:$SYSSAG.<package-name>.APR91450162010.SCI  
144 :I29A:$SYSSAG.<package-name>.APR91450162010.SCI.GPN  
12 :I29A:$SYSSAG.<package-name>.RETURN.LETTER  
96 :I29A:$SYSSAG.<package-name>.SOL2P.DOC  
:I29A: PUBLIC: 32 FILES RES= 28735 FRE= 2403 REL== 2346 PAGES
```

Postprocessing

For this example no manual post-processing is necessary (e.g. starting a product-specific procedure). Where manual post-processing is required, IMON will display messages to this effect on the console (these are also found in the installation protocol <package-name>.APR101149532021.IL or under the important console messages in the file <package-name>.IC work file ID).

The delivery information can tell you whether further postprocessing is required. It should be referred to once more in this regard.

If the delivery was then opened once more (starting IMON, Open: SOLIS2 delivery menu), it should, as would be expected, exist in the SCI as a delivery that was already registered.

Therefore by entering “3” in the following dialog box, it should be opened as a delivery that has already been registered.

**C hecking
the
installation**

```

File Edit Show View Options
:-----:
:                               Open                               :
:                               SOLIS2 delivery                    :
:-----:
: Package name.....: <package-name>                               :
: User code.....: sol2p                                           :
: Distribution medium: 3 1. SOLIS2 volume                          2. Library           :
:                               3. Registered                          4. Local volume     :
:                               Volume.....:                               :
:                               Device type:                          :
:                               :                               :
: Documentation library: 1 1. Standard                            2. Other             :
:                               Name:                               :
: Work file location: :I29A:$SYSSAG.                               :
: Reference File ... :I29A:$TSOS.SYS.IMON.SCI.REF                 :
:                               :                               :
: Supply unit selection 1 1.Manually (*BY-DIALOG)                 :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                             :
:-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The delivery is known in the SCI. All five supply units were already installed.

```

File Edit Show View Options
:-----:
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
:-----:
Units      1 through      5 of      5
           More:
Unit name  Vers  Corr  Park userid  Install status
EDT        17.0 D30   *NONE        Installed
OPENFT     12.1 C10   *NONE        Installed
OPENFT-CR  12.1 B10   *NONE        Installed
PERCON     02.9 C00   *NONE        Installed
SORT       08.0 A30   *NONE        Installed
           *** End of SU selection ***
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

3.2.1.2 Example with SDF statements

```
/modify-msg-attributes task-language=e _____ (1)
```

```
/start-imon input-interface=*sdf _____ (2)
```

```
% IMOLOAD Program 'IMON-TU', version '03.4A00' of '2021-02-22' loaded from  
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
```

```
% IMOCOPY Copyright (C) 2021 Fujitsu Technology Solutions All Rights  
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) The INTERFACE=*SDF command is used to start IMON with the SDF interface.

Alternatively, you can also start IMON in menu mode (in which case it is sufficient to enter the START-IMON command, as INTERFACE=*STD is the default).

After the user has explicitly opened an SCI with the *Open* option in the *File* menu, he or she can then select the option *1 (Switch to SDF interface)* in the *Options* menu.

```
//modify-imon-options  
    sci=*std, _____ (3)
```

```
    work-file-location=$syssag., reference-file=*std _____ (4)
```

```
//install-units _____ (4)
```

```
    unit-name=*from-solis2-delivery( _____ (5)
```

```
        package-name=<package-name>, user-code=sol2p,
```

```
        distribution=*solis2-volume(volume=qe6171, device-type=tape-c4),
```

```
        supply-units=*by-dialog), _____ (6)
```

```
    old-file-saving=*with-lms, _____ (7)
```

```
    placement-mode=*by-dialog, _____ (8)
```

```
    activation-mode=*par( _____ (9)
```

```
        dssm-processing=*by-dialog, rep-processing=*by-dialog),
```

```
    work-file-deleting=*no _____ (10)
```

- (3) The following statement is optional as the specified values correspond to the default setting:

The MODIFY-IMON-OPTIONS statement is used to specify the standard SCI (SOFTWARE-INVENTORY=*STD operand), the \$SYSSAG. path on the home pubset (WORK-FILE-LOCATION=\$SYSSAG. operand) for work files and the default reference file for the optional configuration check (operand REFERENCE-FILE=*STD).

- (4) The customer-specific installation is then carried out using the INSTALL-UNITS statement.

- (5) The UNIT-NAME operand is used to specify the SOLIS2 delivery that is to be installed:

Package name: <package-name>

User code: SOL2P

Distribution medium: MTC of the type TAPE-C4 with the VSN QE6171

The delivery is opened implicitly with REGISTRATION=*EXTEND. This means that IMON evaluates any entries existing in the SCI for this delivery and changes the entries of the supply units in the SUPPLY-UNITS operand. Entries are already present if the delivery has already been opened at least once (e.g. after printing the delivery documentation with PRINT-DOCUMENTATION).

If all entries for this delivery in the SCI must be created first, then the operand REGISTRATION=*REPLACE must be specified (see also "[INSTALL-UNITS Install and register software](#)").

- (6) The operand SUPPLY-UNITS=*BY-DIALOG can be used to check and control the selected supply units in a mask:

IMON initially offers all of the supply units in the delivery for installation (corresponds to the default SUPPLY-UNITS=*ALL). In this example, the selection is confirmed unchanged using [DUE].

- (7) Files on the system that will be overwritten by new files of the same name should first be backed up to an LMS library.

This library is called :I29A:\$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P (see also "[Example in menu mode](#)").

- (8) The operand PLACEMENT-MODE=*BY-DIALOG can be used to request a control dialog box in which you can adapt the default placement parameters to your specific requirements.

- (9) The operand ACTIVATION-MODE=*PARAMETERS(...) can be used to request a control dialog box in which you can adapt the default activation parameters for subsystem administration (DSSM-PROCESSING=*BY-DIALOG) and for processing system corrections (REP-PROCESSING=*BY-DIALOG) to your specific requirements.

- (10) The work files created by IMON are not to be deleted.

The statement is then executed using [DUE]:

```
% IMO0951 Delivery with package name '<package-name>' and user code 'SOL2P' up
to now not registered in SCI
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100510.123258',
VERSION '12.0C00' _____ (11)
% ARC0033 ARCHIVE SUBTASK TSN '0206' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100510.123347',
VERSION '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0207' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-18 AT 15:04, TSN = 0208
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-18 AT 15:04, TSN = 0209 _____ (12)
PLEASE ACKNOWLEDGE
```

- (11) IMON reads the MTC (via ARCHIVE), extracts files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries), and registers the delivery in the SCI.
- (12) IMON then transfers the delivery documentation to the standard documentation library, which is created if necessary, and starts a print job for the acknowledge form (using the "IMON" job name).

```

File  Edit  Show  View  Options
-----
          Units      1 through      5 of      5
          SU selection
Unit name      Version      Corr state
X EDT          17.0          D30
X OPENFT       12.1          C10
X OPENFT-CR    12.1          B10
X PERCON       02.9          C00
X SORT         08.0          A30
          *** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

- (13) IMON selects all supply units and displays the selection so you can check it in a mask (see also point 6). The selection is confirmed with [DUE] without changing it.

```

IMON: SOLIS2 delivery: Package name: <package-name>  User code: SOL2P
-----
          Units      1 through      5 of      5
          Supply units placement parameters
Unit name      Vers Corr Catid  Userid  Prefix  Replace Force
EDT            17.0 D30  *DEF  osidl  *NONE   Y       N
OPENFT         12.1 C10  *DEF  *STD   *NONE   Y       N
OPENFT-CR      12.1 B10  *DEF  *STD   *NONE   Y       N
PERCON         02.9 C00  *DEF  osidl  *NONE   Y       N
SORT           08.0 A30  *DEF  *STD   *NONE   Y       N
          *** End of Supply units placement parameters ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

- (14) The placement parameters for the selected supply units are displayed. The EDT and PERCON supply units should not be installed under the TSOS user ID, therefore the new user ID OSID1 is entered in the Userid column and the mask is executed with [DUE].

```

IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                Units      1 through      2 of      2
          Supply units placement parameters      More:
Unit name      Vers      Corr      Librairies
EDT            17.0      D30            Y
SORT          08.0      A30            Y
          *** End of Supply units placement parameters ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

- (15) There is a library for each of the EDT and SORT supply units that is a subset of an alternative library. The libraries are to be installed and merged with the alternative library. To accomplish this, confirm the preset “Y” in the “Library” column without changing it.

```

                Global activation preparation parameters
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Syntax file processing.: 1 1. Yes                2. No
  SDF param file... :I29A:$TSOS.SYSPAR.SDF
Message file processing: 1 1. Yes                2. No
  MIP param file... :I29A:$TSOS.SYSPAR.MIP.210
  MES file..... :I29A:$TSOS.SYSMES.EKP.01
DSSM processing.....: 1 1. Yes                2. No
  Catalog name.... :I29A:$TSOS.SYS.SSD.CAT.X
  Keep old version: 2 1. Yes                2. No
RMS processing.....: 1 1. Depot+loaders 2. Depot only 3. No
  Depot location..: 1 1. Standard          2. Enforced
  Location: :I29A:$TSOS
POSIX processing.....: 1 1. Cfg            2. All            3. No
F1=Help F12=Cancel

```

- (16) The parameters for preparing the activation are displayed in a dialog box with the default values (as in a default installation). The parameters initially apply globally for all selected supply units.

```

Global activation preparation parameters
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Syntax file processing.: 1 1. Yes                2. No
  SDF param file.: :I29A:$TSOS.SYSPAR.SDF
Message file processing: 1 1. Yes                2. No
  MIP param file.: :I29A:$TSOS.SYSPAR.MIP.210
  MES file.: :I29A:$TSOS.SYSMES.EKP.01
DSSM processing.: 1 1. Yes                      2. No
  Catalog name.: :I29A:$TSOS.SYS.SSD.CAT.X
  Keep old version: 1 1. Yes                    2. No
RMS processing.: 1 1. Depot+loaders  2. Depot only  3. No
  Depot location.: 1 1. Standard      2. Enforced
  Location: :I29A:$usr.
POSIX processing.: 1 1. Cfg           2. All           3. No
F1=Help F12=Cancel

```

(17) The earlier versions are to be retained in DSSM processing, therefore a “1” (Yes) is entered in the “Keep old version” field.

For RMS processing, the RMS depot should be used under the USR user ID (also on the home pubset), therefore the user ID TSOS is overwritten with the USR user ID in the “Depot location” field.

```

IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Supply units activation preparation parameters
Unit name      Units 1 through 5 of More:
                Vers  Corr SDF  MSG  SUB  PSX
EDT             17.0  D30  Y   Y   Y   -
OPENFT         12.1  C10  Y   Y   Y   -
OPENFT-CR      12.1  B10  -   -   -   -
PERCON         02.9  C00  Y   Y   Y   -
SORT           08.0  A30  Y   Y   Y   -
*** End of Supply units activation preparation parameters
***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

```

IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
              Units      1 through      4 of      4
SU selection for keeping old subsystem versions      More:
              Unit name      Version      Corr state
X EDT              17.0              D30
X OPENFT           12.1              C10
X PERCON           02.9              C00
X SORT              08.0              A30
*** End of selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

- (18) The most important current activation parameters (syntax file processing, message file processing, and DSSM processing) for the supply units to be installed are displayed once more. The mask is confirmed unchanged using [DUE].
- (19) The following screen shows the supply units for which earlier subsystem versions will be retained. In terms of this activation parameter, in this case all of the displayed supply units are marked (here with “x”) and the selection is confirmed with [DUE] (the “+” must first be deleted).

```

Job ':I29A:$SYSSAG.<package-name>.APR101242452021.IE' is being generated. '5'
supply units have been selected _____ (20)
Generation of :I29A:$SYSSAG.<package-name>.APR101242452021.IE startet:
Generation of FSTAT-RENAME commands...
Generation of SAVE-FILES commands...
Generation of RESERVE-FILE commands...
Generation of IMPORT-FILE commands...
Generation of UPDATE-CATALOG-ENTRY commands...
Generation of library processing commands...
Generation of ADD-INSTALLATION-UNITS statements...
Generation of reference file...
SYSSII files processing started
Generation of ACTIVATE-UNIT commands for supply units:
  EDT              17.0      D30
  OPENFT           12.1      C10
  OPENFT-CR        12.1      B10
  PERCON           02.9      C00
  SORT              08.0      A30

Job ':I29A:$SYSSAG.<package-name>.APR101242452021.IE' has been generated and
entered _____ (21)
//end _____ (22)

```

(20) The :I29A:\$SYSSAG.<package-name>.APR101242452021.IE installation procedure is generated:

- five supply units were selected for installation
- various processing steps must be generated
- activation of the listed supply units is generated

(21) The installation procedure is then fully generated and started automatically as a background procedure (task with job name "INSTALL").

(22) IMON is terminated using the END statement.

/show-job-status *job(install) _____ (23)

```
TSN:      0210      TYPE:    2 BATCH      NOW:      2021-04-10.124316
JOBNAME:  INSTALL  PRI:      9 210      SPOOLIN: 2021-04-10.1242
USERID:   TSOS     JCLASS:   TSOSBAT   LOGON:    2021-04-10.1242
ACCNB:    ADMINSTR CPU-MAX:  NTL        CPU-USED: 000002.4347
REPEAT:   NO       RERUN:    NO        FLUSH:    NO
MRSCAT:                   HOLD:     NO        START:    SOON
TID:      0007003B UNP/Q#:   03/004
CMD:      EXECUTE                   SIZE:     205
PROG:     :I29A:$TSOS.LMSCONV
ORIGFILE: :I29A:$SYSSAG.<package-name>.APR101242452021.IE
CMD-FILE: :I29A:$TSOS.S.E.0205.2021-04-10.12.42.54
MONJV:    *NONE
```

(23) A check is performed as to whether the installation task is still running.

/show-job-status *tsn(0210) _____ (24)

% EXC0755 INFORMATION ON TASK WITH TSN '0210' CANNOT BE GIVEN

/show-file-attributes \$syssag. _____ (25)

```
48 :I29A:$SYSSAG.A.SOLIS.B.<package-name>.SOL2P
15744 :I29A:$SYSSAG.IMON.SAVE.LIB.<package-name>.SOL2P
9 :I29A:$SYSSAG.RMS.OPT.ADM
54 :I29A:$SYSSAG.RMS.OPT.DOC
327 :I29A:$SYSSAG.SOLIS.LMSCONV
24 :I29A:$SYSSAG.SOLLIB.IMON.PBD
24 :I29A:$SYSSAG.SOLLIB.IMON.SYSSII
420 :I29A:$SYSSAG.SOLLIB.IMON.020
3939 :I29A:$SYSSAG.SOLLIB.SOLIS2.050
3939 :I29A:$SYSSAG.SOLLIB.SOLIS2.050.RMS.071
495 :I29A:$SYSSAG.SOLPAR.IMON.034.GEN
1596 :I29A:$SYSSAG.SOLPAR.SOLIS2.050.AUSGEN
72 :I29A:$SYSSAG.SOLPAR.SOLIS2.050.FMSG
306 :I29A:$SYSSAG.SOLPRG.SOLIS2.050.GEN
3 :I29A:$SYSSAG.SYSREP.EDT.170
12 :I29A:$SYSSAG.SYSREP.OPENFT.121
3 :I29A:$SYSSAG.SYSREP.PERCON.029
12 :I29A:$SYSSAG.SYSREP.SORT.080
6 :I29A:$SYSSAG.<package-name>.DP
303 :I29A:$SYSSAG.<package-name>.IC
9 :I29A:$SYSSAG.<package-name>.APR91450142021.DP
6 :I29A:$SYSSAG.<package-name>.APR91450162021.IA
99 :I29A:$SYSSAG.<package-name>.APR91450162021.IE
15 :I29A:$SYSSAG.<package-name>.APR91450162021.II
210 :I29A:$SYSSAG.<package-name>.APR91450162021.IL
9 :I29A:$SYSSAG.<package-name>.APR91450162021.IP
30 :I29A:$SYSSAG.<package-name>.APR91450162021.IR
3 :I29A:$SYSSAG.<package-name>.APR91450162021.I0
456 :I29A:$SYSSAG.<package-name>.APR91450162021.SCI
144 :I29A:$SYSSAG.<package-name>.APR91450162021.SCI.GPN
12 :I29A:$SYSSAG.<package-name>.RETURN.LETTER
96 :I29A:$SYSSAG.<package-name>.SOL2P.DOC
:I29A: PUBLIC: 32 FILES RES= 28735 FRE= 2403 REL== 2346 PAGES
```

(24) Once the installation task has run, the SHOW-JOB-STATUS command query is quit with the message EXC0755, i.e. the task is terminated.

(25) All files that were created under the work file ID are displayed.

/show-file \$syssag. <package-name> .IC _____ (26)

```

21-04-10 15:27:19 IMO2019 INSTALLATION FUNCTION EXECUTED FOR PACKAGE '<packag
21-04-10 15:27:22 IMO2001 FSTAT-RENAME : analyse of target system
21-04-10 15:27:43 IMO2002 SAVE-OLD-FILES procedure generation
21-04-10 15:27:44 IMO2003 SAVE-OLD-FILES procedure call
21-04-10 15:28:03 IMO2004 RESERVE-FILE : preparation of target system
21-04-10 15:28:07 IMO2005 IMPORT-FILE : import-procedure generation
21-04-10 15:28:08 IMO2024 IMPORT-FILE : import-procedure call
21-04-10 15:29:32 IMO2006 UPDATE-CATALOG-ENTRY : set file attributes
21-04-10 15:29:45 IMO2007 Library processing
21-04-10 15:29:48 IMO2008 ADD-INSTALLATION-UNITS : register installation unit
21-04-10 15:30:01 IMO2015 Reference file generation
21-04-10 15:30:12 IMO2011 RMS-PROCESSING: Revision packet put into RMS Depot
21-04-10 15:30:21 IMO2016 Processing of delivered SYSSII files
21-04-10 15:30:27 IMO4009 SUPPLY UNIT 'EDT' '17.0' 'D30' PROCESSED
21-04-10 15:30:27 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'EDT
21-04-10 15:30:27 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:29 IMO4010 ACTIVATE-UNIT 'EDT' '17.0' 'D30'
21-04-10 15:30:32 IMO4009 SUPPLY UNIT 'OPENFT' '12.1' 'C10' PROCESSED
21-04-10 15:30:32 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'OPE
21-04-10 15:30:32 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:34 IMO4010 ACTIVATE-UNIT 'OPENFT' '12.1' 'C10'
21-04-10 15:30:40 IMO4009 SUPPLY UNIT 'OPENFT-CR' '12.1' 'B10' PROCESSED
21-04-10 15:30:40 IMO4009 SUPPLY UNIT 'PERCON' '02.9' 'C00' PROCESSED
+
S*SOF+ 1( 1)

```

```

21-04-10 15:30:40 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'PER
21-04-10 15:30:40 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:43 IMO4010 ACTIVATE-UNIT 'PERCON' '02.9' 'C00'
21-04-10 15:30:46 IMO4009 SUPPLY UNIT 'SORT' '08.0' 'A30' PROCESSED
21-04-10 15:30:46 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'SOR
21-04-10 15:30:46 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:50 IMO4010 ACTIVATE-UNIT 'SORT' '08.0' 'A30'
21-04-10 15:30:53 IMO2010 RMS-PROCESSING: loader generation
21-04-10 15:31:05 IMO2027 SSCM global processing
21-04-10 15:31:10 IMO2021 INSTALLATION NORMALLY TERMINATED FOR PACKAGE '<pack

```

```

% SHO0301 WARNING: END OF FILE REACHED
e
S*SOF+ 24( 1)

```

- (26) The contents of the log file (:!29A:\$SYSSAG.<package-name>.IC) together with the most important console messages are output.

After scrolling to the file end, the last message logged (IMO2021) shows that the installation was terminated normally.

Note

If manual post-processing is required, console messages to this effect will be displayed. The delivery information can tell you whether further postprocessing is required. It should be referred to once more in this context.

After successful completion of the installation procedure, all supply units are registered in the SCI as installation units and the product files are installed in the stipulated storage location on the system.

```
/start-imon input-interface=*sdf _____ (27)
```

```
% IMOLOAD Program 'IMON-TU', version '03.4A00' of '2021-02-22' loaded from  
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
```

```
% IMOCOPY Copyright (C) 2021 Fujitsu Technology Solutions All Rights  
Reserved
```

```
//show-packages package-name= <package-name>,user-code=sol2p,  
inf=*par(supply-units=*yes,report-level=*minimum) _____ (28)
```

```
=====
```

Package Name	User Code	
Supply Unit Name	Version	Status
<package-name>	SOL2P	
EDT	17.0D30	Installed
OPENFT	12.1C10	Installed
OPENFT-CR	12.1B10	Installed
PERCON	02.9C00	Installed
SORT	08.0A30	Installed

```
(5 units)
```

```
/end _____ (29)
```

(27) IMON is called once more.

You can also check the installation sequence on the console (or in the CONSLOG file) and in the installation log. (The name of the installation log in the example is:

:I29A:\$SYSSAG.<package-name>.APR101242452021.IL).

(28) All supply units of the previously installed delivery (package name <package-name>, customer ID SOL2P) that are now registered in the SCI are displayed with the SHOW-PACKAGES statement. The specification of the INFORMATION operand is optional in the example because the preset was used. The specification can be changed accordingly to request more detailed information.

(29) IMON is terminated using the END statement.

3.2.2 Installation with prior parking of the software

The schematic representation of the sequence can be found on "[Customer-specific installation with prior parking](#)".

Parking of the supplied software is carried out in much the same way as installation. Instead of the installation ID, however, a park ID is specified.

IMON generates the park procedure and starts it automatically. The result of this is that all of the files in the supplied products are transferred to the park ID. The products are registered in the SCI as supply units with the status "parked".

Installation from the park ID

The installation from the park ID can be a default installation or a customer-specific installation. The following must be noted for the installation procedure:

- The SOLIS2 delivery must be opened as a registered delivery. To do this, the same SCI and the same work file ID must be used as when parking.
- The supply units to be installed must be selected.

3.2.2.1 Example in menu mode

Call IMON

Part 1: Parking

IMON is started once English is set as the task-wide language for messages and masks. IMON switches to menu mode once the program is loaded and [DUE] entered.

```
/modify-msg-attributes task-language=e
/start-imon
% IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from fil
e ':I29A:$TSOS.SYSLNK.IMON-BAS.034.TU'
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved
PLEASE ACKNOWLEDGE
```

The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI is used.

```
File Edit Show View Options
-----
IMON: Welcome screen
-----

*** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

**Open
SCI**

In the Open submenu of the File menu you select 2 (Foreign SCI...) and confirm with[DUE].

```
File Edit Show View Options
-----
: 2 1. Standard software configuration inventory      :
: 2. Foreign software configuration inventory ...   : -----
: 3. Installation definition file ...               :
: 4. SOLIS2 delivery ...                           : ***
: 5. Structure and installation information file ... :
:
: F1=Help F12=Cancel
:
:.....:

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

The path name for the park SCI is entered in the "Foreign SCI" dialog box: \$PARKSW1.SYS.IMON-PARK.SCI. The default catalog ID for the park ID PARKSW1 is the catalog ID of the home pubset in this example.

```
File Edit Show View Options
-----
: 2 1. Standard software configuration inventory      :
: 2. Foreign software configuration inventory ...   : -----
:
:.....:
```

As the park SCI does not yet exist, it was created the first time it was opened. It does not as yet contain any entries:

```
File Edit Show View Options
-----
IMON: SCI: :I29A:$PARKSW1.SYS.IMON-PARK.SCI
-----
Unit name          SU selection      Units      0 through  0 of 0
                   Version          Corr state  More:
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Define work file ID

To define the work file ID, select 2 (IMON options) in the Options menu bar and execute the mask using [DUE].

```
File Edit Show View Options
-----
IMON: SCI: :I29A: : 2 1. Switch to SDF interface :
-----
:          2. IMON options ... :
:.....: ough 0 of 0
:          SU selection          More:
Unit name          Version      Corr state
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

The path name of the default work file ID is displayed in the dialog box with :I29A:\$SYSSAG. For parking, the work files are also to be created under the park ID with the WORK prefix in the example. The path name is therefore edited accordingly and is then :I29A:\$PARKSW1.WORK.

```
File Edit Show View Options
-----
:          IMON options          :
:-----:
: Work file location: :I29A:$parksw1.work. : 0 of 0
: Reference file ...: : More:
: :I29A:$TSOS.SYS.IMON.SCI.REF :
: : :
: F1=Help F12=Cancel :
:.....:
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Open SOLIS2 delivery

The delivery should then be opened: Option 1 (*Open*) is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : IMON.SCI
: 2. Close :
: *. Print documentation ... : Units 0 through 0 of 0
: 4. Save as ... : tion units selection More:
: 5. Exit : Version Corr state
: .....: lation units selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Option 4 (*SOLIS2 delivery*) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: ..... Open :
: .....:
: 4 *. Standard software configuration inventory : gh 0 of 0
: *. Foreign software configuration inventory ... : More:
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel :
: .....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following delivery information is then queried in the “SOLIS2 delivery” dialog box that is opened: package name, user code, distribution medium (magnetic tape is preset), the VSN, and the MTC device type. The set work file ID is displayed again in the lower section of the mask. This dialog box is executed using [DUE].

```

File Edit Show View Options
:-----:
:                               Open                               :
:-----:
:                               SOLIS2 delivery                     :
:-----:
: Package name.....: <package-name>                             :
: User code.....: sol2p                                           :
: Distribution medium: 1 1. SOLIS2 volume           2. Library       :
:                               3. Registered           4. Local volume   :
:                               Volume.....: ge6171           :
:                               Device type: tape-c4           :
:-----:
: Documentation library: 1 1. Standard           2. Other           :
:                               Name:                               :
: Work file location: :I29A:$SYSSAG.               :
: Reference File ... :I29A:$TSOS.SYS.IMON.SCI.REF :
:-----:
: Supply unit selection 1 1.Manually (*BY-DIALOG) :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                           :
:-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

IMON then has all of the information needed to open the delivery on the data medium and it starts the processing operation, during which some messages are output to the screen:

IMON instructs ARCHIVE to read the specified MTC and extract files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS2 program libraries). The "IMON" jobs started for this purpose create or expand the documentation library and start a print job for the acknowledge form. You can switch back to IMON menu mode using [DUE].

```

% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142557', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0183' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100509.142610', VERSIO
N '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0184' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0185
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-09 AT 14:27, TSN = 0186
%PLEASE ACKNOWLEDGE

```

Confirm supply units

Five supply units were found on the specified MTC. IMON offers these for further processing (they are already highlighted). To select all of these supply units, execute the selection using [DUE].

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
          Units      1 through      5 of      5
          SU selection  Version      Corr state  More:
X EDT      Unit name      17.0        D30
X OPENFT   12.1        C10
X OPENFT-CR 12.1        B10
X PERCON   02.9        C00
X SORT     08.0        A30
          *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select park

You must then switch to the Edit menu to install the units. Option 4 (Install) is already preset. You select the Park option by entering a "5" and confirming it with [DUE].

```
File Edit Show View Options
-----
IM : 5 * . Search ... : er code: SOL2P
-----: * . Add ... : -----
: * . Remove ... : ough 5 of 5
: 4. Install ... : More:
: 5. Park ... : tatus
X EDT : * . Generate installation definition file ... :
X OPENF: * . Deinstall ... :
X OPENF: * . Undo ... :
X PERCO: * . Activate ... :
X SORT : * . Check ... :
: * . Request correction delivery ... :
: * . Customer-Approved Install ... :
:.....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Specify target system


```

File Edit Show View Options
:-----:-----:-----:
:                               Park parameters                               :
:-----:-----:-----:
: Target system                                                           : 5
:   BS2000 version               : MM.N
:   Standard pubset.....: 1 1. Default          2. Other
:                               Catid:
:
: Save old files.....: 2 1. No          2. With LMS
:                               3. With ARCHIVE  4. With MAREN
:                               VSN.....:
:                               Device type....:
:                               MAREN location : *STD
: Update RMS depot.....: 1 1. Yes          2. No
:   Depot Location: *STD
:
: File location.....: :I29A:$PARKSW1
:
: F1=Help  F12=Cancel
:
:-----:-----:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

IMON has then queried all of the information needed to park the delivery and starts the processing operation during which some messages are output on the screen:

- The :I29A:\$PARKSW1.WORK.<package-name>.APR101333272021.PE park procedure will be generated:
 - five supply units were selected for parking
 - various processing steps must be generated.
- The park procedure is then fully generated. It is started immediately and automatically.

<p>IMON output</p>	<pre> Job ':I29A:\$PARKSW1.WORK.<package-name>.APR91450162021.PE' is being generated. '5' supply units have been selected Generation of :I29A:\$ PARKSW1.WORK..APR91450162021.PE startet: Generation of FSTAT-RENAME commands... Generation of SAVE-FILES commands... Generation of RESERVE-FILE commands... Generation of IMPORT-FILE commands... Generation of RMS processing commands... Generation of ACTIVATE-UNIT commands for supply units: EDT 17.0 D30 OPENFT 12.1 C10 OPENFT-CR 12.1 B10 PERCON 02.9 C00 SORT 08.0 A30 Job ':I29A:\$ PARKSW1.WORK.<package-name>.APR91450162021.PE' has been generated and entered PLEASE ACKNOWLEDGE </pre> <p>The installation procedure runs asynchronously and outputs to the console messages relating to the processing steps executed and any errors that arose. The procedure sequence can be checked in the installation log.</p> <p>The SHOW-JOB-STATUS command can be used in the interactive task to check whether the started installation task (TSN 0216) is already finished. The procedure sequence can be checked in the installation log.</p> <p>IMON switches to menu mode once [DUE] is entered.</p> <p>As the installation procedure is still running asynchronously, the supply units to be parked are in an undefined state and are not displayed in this mask.</p> <p>If a subset of the supply units in the delivery were parked and not the complete set, this mask would display those supply units that were not selected.</p>
<p>Close SOLIS2 delivery</p>	<p>The SOLIS2 delivery is closed: Option 2 (Close) is entered in the File menu and confirmed with [DUE].</p> <p>This concludes the parking operation. The parked supply components can be installed at any time and as often as necessary. The rest of the example shows how to install the supply units from the park ID.</p>
<p>Open SOLIS2 delivery</p>	<p><i>Part 2: Installing</i></p> <p>The delivery should be installed from the park ID once it is parked: Option 1 (Open) is already preset in the File menu. The mask is executed using [DUE].</p>

```

File Edit Show View Options
-----
: 1 *. Open ... : name: <package-name> User code: SOL2P
: 2. Close ... :
: 3. Print documentation ... : Units 0 through 0 of 0
: 4. Save as ... : selection
: 5. Exit ... : Version Corr state
:.....: SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Option 4 (*SOLIS2 delivery*) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: .....:
: Open :
: .....:
: 4 *. Standard software configuration inventory : gh 0 of 0
: *. Foreign software configuration inventory ... : More:
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
: .....:
: F1=Help F12=Cancel :
: .....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following delivery information is then queried in the “SOLIS2 delivery” dialog box that is opened: package name, user code. “3” for registered delivery is entered in the distribution medium field. The set work file ID is displayed again in the lower section of the mask. This dialog box is executed using [DUE].

```

File Edit Show View Options
-----
:                               Open                               :
:                               SOLIS2 delivery                   :
:-----
: Package name.....: <package-name>                             :
: User code.....: sol2p                                         :
: Distribution medium: 3 1. SOLIS2 volume           2. Library       :
:                               3. Registered           4. Local volume   :
:                               Volume.....:                 :
:                               Device type:                 :
:-----
: Documentation library: 1 1. Standard           2. Other         :
:                               Name:                 :
: Work file location: :I29A:$SYSSAG.                 :
: Reference File ... :I29A:$TSOS.SYS.IMON.SCI.REF     :
:-----
: Supply unit selection 1 1.Manually (*BY-DIALOG)         :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                     :
:-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The 5 parked supply units are displayed with the status "Parked":

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Units      1 through      5 of      5
More:
Unit name  Vers  Corr  SU selection  Park userid  Install status
EDT        17.0 D30  :I29A:$PARKSW1  Parked
OPENFT     12.1 C10  :I29A:$PARKSW1  Parked
OPENFT-CR  12.1 B10  :I29A:$PARKSW1  Parked
PERCON     02.9 A10  :I29A:$PARKSW1  Parked
SORT       08.0 A30  :I29A:$PARKSW1  Parked
*** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Confirm supply units

For the installation, the 5 supply units are selected by highlighting them. The selection is then confirmed with [DUE].

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      1 through      5 of      5
                                SU selection
Unit name      Vers  Corr  Park userid  Install status
X EDT          17.0 D30  :I29A:$PARKSW1  Parked
X OPENFT       12.1 C10  :I29A:$PARKSW1  Parked
X OPENFT-CR    12.1 B10  :I29A:$PARKSW1  Parked
X PERCON       02.9 A10  :I29A:$PARKSW1  Parked
X SORT         08.0 A30  :I29A:$PARKSW1  Parked
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select install

You must then switch to the *Edit* menu to install the units. Option 4 (*Install*) is already preset and needs only to be confirmed with [DUE].

```
File Edit Show View Options
-----
IM : 5 * . Search ... : er code: SOL2P
-----: * . Add ... : :
: * . Remove ... : ough 5 of 5
: 4. Install ... : More:
: 5. Park ... : tatus
X EDT : * . Generate installation definition file ... :
X OPENF: * . Deinstall ... :
X OPENF: * . Undo ... :
X PERCO: * . Activate ... :
X SORT : * . Check ... :
: * . Request correction delivery ... :
: * . Customer-Approved Install ... :
:.....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

**Select
parameter
file**

The following default installation parameters options are preset by IMON in the "IMON parameter file" dialog box.

```
File Edit Show View Options
: .....
: IMON Parameter file :
: .....
: Parameter file ...: 2 1. None      2. Standard    3. Other      : 5
: File name: :I29A:$TSOS.SYSPAR.IMON.LAST
: .....
: Save parameters...: 1 1. No          2. Standard    3. Other      :
:   File name:
: .....
: F1=Help F12=Cancel
: .....
: *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

Specify target system

In the “Global installation parameters” dialog box, the version and the standard pubset are already correctly preset under Target system. Option “2” in the “Work file deleting” field retains the supply units to be installed under the park ID. This entry is the minimum required if a further installation is intended.

```
File Edit Show View Options
:-----Global installation parameters-----:
: Target system                               :
:   BS2000 version.....: MM.N          :
:   Standard pubset.....: 1 1. Default      2. Other      :
:   Save old files.....: 1 1. No          2. With LMS   :
:                               3. With ARCHIVE  4. With MAREN :
:                               VSN.....:          :
:                               Device type: MAREN location: *STD :
: Undo preparation.....: 2 1. Yes         2. No         :
: Placement mode.....: 1 1. Standard   2. Other      :
: Activation preparation mode: 1 1. Standard  2. Other      :
: Password file.....: 2 1. Yes         2. No         :
:   File name.....:          :
:   Read password: C'      '          :
: Print log files.....: 2 1. Yes         2. No         :
: Start.....: 1 1. Immediately  2. By user    :
: Configuration checks.....: 1 1. Yes         2. No         :
: Work file deleting.....: 2 1. Yes         2. No         :
:                               :
: F1=Help F12=Cancel
:-----:

```

The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customer-specific).

As the open park SCI is not the standard SCI for the target system, an additional query is output when generating the installation procedure (see ["Example with SDF statements"](#)): The installation can be registered in the (foreign) SCI currently open or in the standard SCI of the home pubset. The installation is registered in the standard SCI (as in a direct installation on the home pubset) if you respond with “N”.

The subsequent procedure up to creating and starting the installation procedure can be found in the relevant examples.

Terminate IMON

IMON is terminated by entering 5 (Exit) in the File menu and [DUE] (it can also be terminated via the [F3] function key).

The [section “Multiple installation of parked software”](#) shows a further installation on an imported pubset from the park ID.

3.2.2.2 Example with SDF statements

Part 1: Parking

```
/modify-msg-attributes task-language=e _____ (1)
//start-imon input-interface=*sdf _____ (2)

% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from file :
I29A:$TSOS.SYSLNK.IMON-BAS.034.TU

% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) The INTERFACE=*SDF command is used to start IMON with the SDF interface.

Alternatively, you can also start IMON in menu mode (in which case it is sufficient to enter the START-IMON command, as INTERFACE=*STD is the default). After the user has explicitly opened an SCI with the *Open* option in the *File* menu, he or she can then select option 1 (*Switch to SDF interface*) in the *Options* menu.

```
//modify-imon-options
    sci=$parksw1.sys.imon-park.sci,
    work-file-location=$parksw1.work., _____ (3)
    reference-file=*std
//park-units _____ (4)
unit-name=*from-solis2-delivery( _____ (5)
    package-name= <package-name> , user-code=sol2p,
    distribution-medium=*solis2-volume(volume=qe6171,dev-type=tape-c4),
    supply-units=*by-dialog), _____ (6)
old-file-saving=*with-lms, _____ (7)
file-location=$parksw1. _____ (8)
```

- (3) The MODIFY-IMON-OPTIONS statement is used to explicitly specify the park SCI (SOFTWARE-INVENTORY=\$PARKSW1.SYS.IMON-PARK.SCI operand) and the PARKSW1 park ID, the WORK file name prefix (WORK-FILE-LOCATION= \$PARKSW1.WORK. operand) for work files and the standard reference file for the configuration check (operand REFERENCE-FILE=*STD).
- (4) The park operation is then carried out using the PARK-UNITS statement.

- (5) The UNIT-NAME operand is used to specify the SOLIS2 delivery that is to be installed:

Package name: <package-name>

User code: SOL2P

Distribution medium: MTC of the type TAPE-C4 with the VSN QE6171

The delivery is opened implicitly with REGISTRATION=*EXTEND. This means that IMON evaluates any entries existing in the SCI for this delivery and changes the entries of the supply units in the SUPPLY-UNITS operand. Entries are already present if the delivery has already been opened at least once (e.g. after printing the delivery documentation with PRINT-DOCUMENTATION).

If all entries for this delivery in the SCI must be created first, the operand REGISTRATION=*REPLACE must be specified (see also "[INSTALL-UNITS Install and register software](#)").

- (6) You can check and confirm the selected supply units in a dialog box by specifying SUPPLY-UNITS=*BY-DIALOG:

IMON initially offers all of the supply units in the delivery for installation (corresponds to the default SUPPLY-UNITS=*ALL). In this example, the selection is confirmed unchanged with [DUE].

- (7) Files on the system that will be overwritten by files of the same name should first be backed up to an LMS library. This library is called:

:I29A:\$PARKSW1.WORK.IMON.SAVE.LIB.<package-name>.SOL2P (cf. "[Example in menu mode](#)").

- (8) The FILE-LOCATION=\$PARKSW1. operand is used to specify that the files to be parked are to be stored on the home pubset under the user ID PARKSW1.

The statement is then executed using [DUE]:

```
% IMO0951 Delivery with package name '<package-name>' and user code 'SOL2P' up
to now not registered in SCI
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100510.142243',
VERSION '12.0C00' _____ (9)
% ARC0033 ARCHIVE SUBTASK TSN '0218' GENERATED
% ARC0002 STATEMENT ACCEPTED. ARCHIVE SEQUENCE NUMBER 'A.100510.142313',
VERSION '12.0C00'
% ARC0033 ARCHIVE SUBTASK TSN '0219' GENERATED
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-10 AT 14:24, TSN = 0220
% JMS0066 JOB 'IMON' ACCEPTED ON 21-04-10 AT 14:24, TSN = 0221 _____ (10)
PLEASE ACKNOWLEDGE
```

- (9) IMON reads the MTC (via ARCHIVE), extracts files that are needed for the subsequent installation (e.g. product movement file, IMON and SOLIS program libraries), and registers the delivery in the SCI.
- (10) IMON then transfers the delivery documentation to the standard documentation library, which is created if necessary, and starts a print job for the acknowledge form.

```
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name          SU selection  Units      1 through  5 of  5
                   Version      Corr state  More:
X EDT              17.0       D30
X OPENFT          12.1       C10
X OPENFT-CR       12.1       B10
X PERCON          02.9       C00
X SORT            08.0       A30
                   *** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

(11)

- (11) IMON selects all of the supply units and displays the selection in a mask so that it can be checked (cf. Point 6). The selection is confirmed unchanged with [DUE].

Job ':I29A:\$PARKSW1.WORK.<package-name>.APR101426212021.PE' is being generated.

'5' supply units have been selected _____ (12)

Generation of :I29A:\$PARKSW1.WORK.<package-name>.APR101426212021.PE started:

Generation of FSTAT-RENAME commands...

Generation of SAVE-FILES commands...

Generation of RESERVE-FILE commands...

Generation of IMPORT-FILE commands...

Generation of RMS processing commands...

Park generation processed for supply units:

EDT	17.0	D30
OPENFT	12.1	C10
OPENFT-CR	12.1	B10
PERCON	02.9	C00
SORT	08.0	A30

% JMS0066 JOB 'PARK' ACCEPTED ON 21-04-10 AT 14:26, TSN = 0222

Job ':I29A:\$PARKSW1.WORK.<package-name>.APE101426212021.PE' has been generated

and entered _____ (13)

//end _____ (14)

(12) The park procedure is generated. File name:

:I29A:\$PARKSW1.WORK.<package-name>.APR101426212021.PE

- five supply units were selected for parking
- various processing steps must be generated
- activation of the listed supply units is initiated.

(13) The park procedure is then fully generated and started automatically as a background procedure (task with the TSN 0222).

(14) IMON is terminated using the END statement.

% JMS0066 JOB 'PARK' ACCEPTED ON 21-04-10 AT 14:26, TSN = 0222

/show-job-sta *tsn(0at6) _____ (15)

TSN: 0222 TYPE: 2 BATCHF NOW: 2021-04-10.142629
JOBNAME: PARK PRI: 9 210 SPOOLIN: 2021-04-10.1524
USERID: TSOS JCLASS: JCBTSOS LOGON: 2021-04-10.1524
ACCNB: ADMINSTR CPU-MAX: NTL CPU-USED:000001.1239
REPEAT: NO RERUN: NO FLUSH: NO
MRSCAT: HOLD: NO START: SOON
TID: 0002007B UNP/Q#: 03/004
CMD: EXECUTE SIZE: 581
PROG: :I29A:\$TSOS.LMSCONV
ORIGFILE::I29A:\$PARKSW1.WORK.<package-name>.APR101426212021.PE
CMD-FILE::I29A:\$TSOS.S.E.0222.2021-04-10.14.26.21
MONJV: *NONE

/show-job-sta *tsn(0222) _____ (16)

% EXC0755 INFORMATION ON TASK WITH TSN '0222' CANNOT BE GIVEN

/show-file-attr \$parksw1. _____ (17)

:

- (15) The SHOW-JOB-STATUS command is used to check whether the park task is still running.
- (16) Once the park task has run, the SHOW-JOB-STATUS command query is quit with the message EXC0755, i.e. the task is terminated.
- (17) The SHOW-FILE-ATTRIBUTES command is used to display all files that were created under the work file ID.

Part 2: Installing from the park ID

```

/start-imon input-interface=*sdf _____ (18)
% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
//modify-imon-options
    software-inventory=$parkswl.sys.imon-park.sci,
    work-file-location=$parkswl.work. _____ (19)

```

- (18) IMON is restarted with the SDF interface to install the supply units from the park ID (see also Point 2).
- (19) The MODIFY-IMON-OPTIONS statement is used, as before in parking, to define the park SCI and the same work file ID explicitly (see Point 3).

```

//install-units _____ (20)
    unit-name=*from-solis2-delivery( _____ (21)
        package-name= <package-name> ,user-code=sol2p,
        distribution=*registered-medium,
        supply-units=*by-dialog), _____ (22)
    work-file-deleting=*no _____ (23)

```

- (20) The installation is then carried out using the INSTALL-UNITS statement:
- (21) The parked SOLIS2 delivery that is to be installed is specified using the UNIT-NAME operand:

```

Package name:    <package-name>
User code:       SOL2P
Distribution medium: *REGISTERED-MEDIUM (default)

```

- (22) The SUPPLY-UNITS=*BY-DIALOG operand can be used to display the supply units to be installed in a dialog box.
The supply units can also be specified directly in the SUPPLY-UNITS operand.
- (23) The WORK-FILE-DELETION=*NO operand can be used to retain the supply units to be installed under the park ID (for further installations).

```

IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name      Vers  Corr  SU selection  Units  1 through  5 of  5
                Park userid  Install status
EDT            17.0 D30   :I29A:$PARKSW1  Parked
OPENFT        12.1 C10   :I29A:$PARKSW1  Parked
OPENFT-CR     12.1 B10   :I29A:$PARKSW1  Parked
PERCON        02.9 A10   :I29A:$PARKSW1  Parked
SORT          08.0 A30   :I29A:$PARKSW1  Parked
                *** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

(24)

- (24) After executing the statement, all five supply units in the requested dialog box are marked and the dialog box is executed with [DUE].

The specifications in the INSTALL-UNITS statement are the minimum required for installation on the home pubset. The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customerspecific). As the open park SCI is not the default SCI for the target system, an additional query is output when generating the installation procedure:

```

Job ':I29A:$PARKSW1.WORK.<package-name>.APR101437462021.IE' is being
generated.

'5' supply units have been selected

Generation of  :I29A:$PARKSW1.WORK.<package-name>.APR101437462021.
IE startet:

*****

* Local SCI  :I29A:$PARKSW1.SYS.IMON-PARK.SCI
* is not a standard one.  Should the installation results
* be registered in this one or in the standard SCI of the
* target system (:I29A:$TSOS.SYS.IMON.SCI) ?
* Reply Y=local SCI/ N=std SCI

*****

*n _____ (25)

```

-
- (25) It is possible that the installation will be registered in the currently open foreign SCI or standard SCI of the home pubset. Responding with “N” will cause the registration to be performed in the standard SCI (just like for a direct installation on the home pubset). Further processing up to the creation and start of the installation procedure can be taken from the appropriate examples. In the following it is assumed that the installation procedure has already successfully executed:

```
      :  
      //modify-imon-options software-inventory=*std _____ (26)  
      //show-supply-units unit-name=*from-solis2-delivery(  
          package-name= <package-name> ,user-code=sol2p),  
          select=*by-attributes(installation-status=*installed) _____ (27)  
      :  
      //end _____ (28)
```

- (26) The standard SCI of the home pubset is opened.
- (27) All of the supply units in the previously installed delivery (package name <package-name> , user code SOL2P) that are registered in the standard SCI are displayed.
- (28) IMON is terminated using the END statement.

3.2.3 Multiple installation of parked software

The schematic representation of the sequence can be found on "[Sequence of a customer-specific multiple installation of a parked delivery](#)".

3.2.3.1 Example in menu mode

Call IMON

IMON is started once English is set as the task-wide language for messages and masks. IMON switches to menu mode once the program is loaded and [DUE] entered.

```
/modify-msg-attributes task-language=e
/start-imon
% IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from fil
e ':I29A:$TSOS.SYSLNK.IMON-BAS.034.TU'
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved
PLEASE ACKNOWLEDGE
```

The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI is used.

```
File Edit Show View Options
-----
IMON: Welcome screen
-----

*** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----

Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

**Open
SCI**

In the *Open* submenu of the *File* menu you select 2 (*Foreign SCI...*) and confirm with [DUE].

```
File Edit Show View Options
:-----:
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :-----:
: 3. Installation definition file ... :
: 4. SOLIS2 delivery ... : ***
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel
:
:.....:

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----:
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

The path name for the park SCI is entered in the “Foreign SCI” dialog box: \$PARKSW1.SYS.IMON-PARK.SCI. The default catalog ID for the park ID PARKSW1 is the catalog ID of the home pubset in this example.

```
File Edit Show View Options
:-----:
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :-----:
: :
:.....:
```

For the first installation the installation units were entered in the standard SCI of the home pubset. Therefore there are no entries (for installation units) present.

```
File Edit Show View Options
:-----:
IMON: SCI: :I29A:$PARKSW1.SYS.IMON-PARK.SCI
:-----:
Units 0 through 0 of 0
SU selection More:
Unit name Version Corr state
*** End of SU selection ***

-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Define work file ID

To define the work file ID, select 2 (*IMON options*) in the *Options* menu bar and execute the mask using [DUE].

```
File Edit Show View Options
-----
IMON: SCI: :I29A: : 2 1. Switch to SDF interface :
-----
: 2. IMON options ... :
:.....: ough 0 of 0
: SU selection : More:
Unit name Version Corr state
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

The path name of the default work file ID is displayed in the dialog box with :I29A:\$SYSSAG. The path name should be set as during the park operation. It is therefore edited accordingly and is then :I29A:\$PARKSW1.WORK.

```
File Edit Show View Options
: .....:
: IMON options :
: .....:
: Work file location: :I29A:$parksw1.work. : 0 of 0
: Reference file ...: : More:
: :I29A:$TSOS.SYS.IMON.SCI.REF :
: :
: F1=Help F12=Cancel :
: .....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Open SOLIS2 delivery

The delivery should then be opened: Option 1 (*Open*) is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : IMON.SCI
: 2. Close :
: * . Print documentation ... : Units 0 through 0 of 0
: 4. Save as ... : tion units selection More:
: 5. Exit : Version Corr state
: .....: lation units selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Option 4 (SOLIS2 delivery) for opening the SOLIS2 delivery is preset. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: .....:
: Open :
: .....:
: 4 * . Standard software configuration inventory : gh 0 of 0
: * . Foreign software configuration inventory ... : More:
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel :
: .....:
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The following delivery information is then queried in the “SOLIS2 delivery” dialog box that is opened: package name, user code. “3” for registered delivery is entered in the distribution medium field. The set work file ID is displayed again in the lower section of the mask. This dialog box is executed using [DUE].

```

File Edit Show View Options
-----
:                               Open                               :
:                               SOLIS2 delivery                   :
:-----
: Package name.....: <package-name>                             :
: User code.....: sol2p                                           :
: Distribution medium: 3 1. SOLIS2 volume                2. Library   :
:                               3. Registered                4. Local volume :
:                               Volume.....:                   :
:                               Device type:                   :
:-----
: Documentation library: 1 1. Standard                2. Other      :
:                               Name:                   :
: Work file location: :I29A:$SYSSAG.                   :
: Reference File ...: :I29A:$TSOS.SYS.IMON.SCI.REF         :
:-----
: Supply unit selection 1 1.Manually (*BY-DIALOG)         :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help F12=Cancel                                       :
:-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The 5 parked supply units are displayed with the status "Installed":

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Units      1 through      5 of      5
          More:
Unit name  Vers  Corr  SU selection  Park userid  Install status
EDT        17.0 D30  :I29A:$PARKSW1  Installed
OPENFT     12.1 C10  :I29A:$PARKSW1  Installed
OPENFT-CR  12.1 B10  :I29A:$PARKSW1  Installed
PERCON     02.9 A10  :I29A:$PARKSW1  Installed
SORT       08.0 A30  :I29A:$PARKSW1  Installed
          *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Confirm supply units

For another installation the PERCON supply unit is selected by marking it and the selection is confirmed with [DUE].

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      1 through      5 of      5
                                SU selection
Unit name      Vers  Corr   Park userid  Install status
EDT            17.0 D30   :I29A:$PARKSW1  Installed
OPENFT        12.1 C10   :I29A:$PARKSW1  Installed
OPENFT-CR     12.1 B10   :I29A:$PARKSW1  Installed
X PERCON      02.9 A10   :I29A:$PARKSW1  Installed
SORT          08.0 A30   :I29A:$PARKSW1  Installed
                *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select install

You must then switch to the *Edit* menu to install the units. Option 4 (*Install*) is already preset and needs only to be confirmed with [DUE].

```
File Edit Show View Options
-----
IM : 4 *. Search ... : er code: SOL2P
-----: *. Add ... : -----
: *. Remove ... : ough 5 of 5
: 4. Install ... : More:
: 5. Park ... : tatus
EDT : *. Generate installation definition file ... :
OPENF: *. Deinstall ... :
OPENF: *. Undo ... :
X PERCO: *. Activate ... :
SORT : *. Check ... :
: *. Request correction delivery ... :
: *. Customer-Approved Install ... :
:.....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select parameter file

The following default installation parameters options are preset by IMON in the “IMON parameter file” dialog box.

```
File Edit Show View Options
:-----
:                               IMON Parameter file :
:-----
: Parameter file ...: 2 1. None          2. Standard      3. Other      : 5
: File name: :I29A:$TSOS.SYSPAR.IMON.LAST
:
: Save parameters...: 1 1. No          2. Standard      3. Other      :
:   File name:
:
: F1=Help F12=Cancel
:-----
:                               *** End of SU selection ***
```

```
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

Specify target system

A different pubset is specified in the “Global installation parameters” dialog box under Target System in the “Standard Pubset” field with a “2”, the value 8.0 for the preset for the version is already correct. Entering “2” in the “Delete work files” field will make the supply units to be installed remain available on the park ID. These entries are the minimum requirement to add another installation on an imported pubset.

```
File Edit Show View Options
:-----
:                               Global installation parameters
:-----
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 2 1. Default          2. Other
:   Save old files.....: 1 1. No              2. With LMS
:                               3. With ARCHIVE      4. With MAREN
:                               VSN.....:
:                               Device type:      MAREN location: *STD
: Undo preparation.....: 2 1. Yes              2. No
: Placement mode.....: 1 1. Standard        2. Other
: Activation preparation mode: 1 1. Standard      2. Other
: Password file.....: 2 1. Yes              2. No
:   File name...:
:   Read password: C'
: Print log files.....: 2 1. Yes              2. No
: Start.....: 1 1. Immediately  2. By user
: Configuration checks.....: 1 1. Yes              2. No
: Work file deleting.....: 2 1. Yes              2. No
:
: F1=Help F12=Cancel
:-----
```

The catalog ID 6A0B and the system ID TSOS (preset) are set in the “Pubset parameters” dialog box for the imported pubset and are confirmed with [DUE].

```

File Edit Show View Options
:-----:
:                               Global installation parameters                               :
:-----:
: Pubset parameters              : ..: MM,N                                         :
: :                               : ..: 2 1. Default                               2. Other   :
: Catid..: 6a0b                  : ..: 1 1. No                                   2. With LMS :
: :                               : ..: 3. With ARCHIVE                          4. With MAREN :
: DEFLUID: TSOS                  : SN.....:                                         :
: :                               : evice type:                                MAREN location: *STD :
: F1=Help F12=Cancel            : ..: 2 1. Yes                                   2. No     :
: :                               : ..: 1 1. Standard                             2. Other  :
: Activation preparation mode: 1 1. Standard   2. Other  :
: Password file.....: 2 1. Yes   2. No     :
:   File name.....:                                         :
:   Read password: C' '                                         :
: Print log files.....: 2 1. Yes   2. No     :
: Start.....: 1 1. Immediately   2. By user  :
: Configuration checks.....: 1 1. Yes   2. No     :
: Work file deleting.....: 2 2. Yes   2. No     :
: :                               :                                         :
: F1=Help F12=Cancel          :                                         :
:-----:

```

The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customer-specific).

As the open park SCI is not the standard SCI for the target system, an additional query is output when generating the installation procedure (see "[Example with SDF statements](#)"): The installation can be registered in the (foreign) SCI currently open or in the standard SCI of the home pubset. The installation is registered in the standard SCI (as in a direct installation on the home pubset) if you respond with “N”.

The subsequent procedure up to creating and starting the installation procedure can be found in the relevant examples.

Check the installation process

The PERCON supply unit is not displayed until the installation is ended and therefore can no longer be selected.

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      1 through      4 of      4
                                SU selection
Unit name      Vers  Corr  Park userid  Install status
EDT            17.0 D30   :I29A:$PARKSW1  Installed
OPENFT        12.1 C10   :I29A:$PARKSW1  Installed
OPENFT-CR     12.1 B10   :I29A:$PARKSW1  Installed
SORT          08.0 A30   :I29A:$PARKSW1  Installed
                *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Close SOLIS2 delivery

The SOLIS2 delivery is closed by entering 2 (*Close ...*) in the *File* menu.

```
File Edit Show View Options
-----
: 2 1. Open ...           : name: <package-name> User code: SOL2P
: 2. Close               : -----
: * . Print documentation ... : Units      1 through      4 of      4
: 4. Save as ...         : selection
: 5. Exit                : rk userid      Install status
: .....                : A:$PARKSW1     Installed
OPENFT              12.1 C10   :I29A:$PARKSW1  Installed
OPENFT-CR           12.1 B10   :I29A:$PARKSW1  Installed
SORT                08.0 A30   :I29A:$PARKSW1  Installed
                    *** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Check the SCI

After closing the park SCI (Option 2 (*Close*) in the *File* menu), the standard SCI of the imported subset should then be opened: Option 1 (*Open*) is already preset in the *File* menu and needs simply to be confirmed with [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : : Nothing opened
: 2. Close : :
: * . Print documentation ... : :
: 4. Save as ... : :
: 5. Exit : :
: .....: :

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

In the *Open* menu 2 (*Foreign SCI ...*) is selected and confirmed with [DUE].

```

File Edit Show View Options
-----
: .....: :
: Open : :
: .....: :
: 2 1. Standard software configuration inventory : :
: 2. Foreign software configuration inventory ... : :
: 3. Installation definition file ... : :
: 4. SOLIS2 delivery ... : :
: 5. Structure and installation information file ... : :
: : :
: F1=Help F12=Cancel : :
: .....: :

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The path name of the subset 6A0B standard SCI is entered in the “Foreign SCI” dialog box: :
6A0B:\$TSOS.SYS.IMON.SCI.

```

File Edit Show View Options
-----
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :
:
:           Foreign software configuration inventory :
:           SCI name: :6A0B:$tsos.sys.imon.sci :
: F :
:....: F1=Help F12=Cancel :
:
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The SCI was previously empty. The PERCON installation unit is now displayed. The RMS supply unit was automatically installed by IMON since it was not already present.

```

File Edit Show View Options
-----
IMON: SCI: :6A0B:$TSOS.SYS.IMON.SCI
-----
Units          1 through      2 of      2
SU selection   Version      Corr state  More:
PERCON        Unit name
RMS           02.9         C00
              07.1         G00
              *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

**Terminate
IMON**

IMON is terminated by entering 5 (*Exit*) in the *File* menu and [DUE] (it can also be terminated via the [F3] function key).

3.2.3.2 Example with SDF statements

```
/modify-msg-attributes task-language=e _____ (1)
//start-imon input-interface=*sdf _____ (2)

% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU

% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) The INTERFACE=*SDF command is used to start IMON with the SDF interface. Alternatively, you can also start IMON in menu mode (in which case it is sufficient to enter the START-IMON command, as INTERFACE=*STD is the default). After the user has explicitly opened an SCI with the *Open* option in the *File* menu, he or she can then select option 1 (*Switch to SDF interface*) in the *Options* menu.

```
//modify-imon-options
  sci=$parksw1.sys.imon-park.sci,
  work-file-location=$parksw1.work., _____ (3)
  reference-file=*std
//install-units _____ (4)
  unit-name=*from-solis2-delivery( _____ (5)
    package-name= <package-name> ,user-code=sol2p,
    distribution=*registered-medium
    supply-units=*by-dialog), _____ (6)
  target-system=*par(
    version=*current, _____ (7)
    pubset=*par(catid=6a0b,defluid=tsos)), _____ (8)
  work-file-deleting=*no _____ (9)
```

- (3) The MODIFY-IMON-OPTIONS statement is used to specify the park SCI (SCI=\$PARKSW1.SYS.IMON-PARK.SCI operand) and the PARKSW1 park ID, the WORK file name prefix (WORK-FILE-LOCATION=\$PARKSW1.WORK. operand) for work files and the standard reference file for the configuration check (operand REFERENCE-FILE=*STD).

The park SCI and work file ID must be set in the same way as they were for the park operation (cf. also ["Example with SDF statements"](#), Point 3).

- (4) The installation is then carried out using the INSTALL-UNITS statement.
- (5) The UNIT-NAME operand is used to specify the parked SOLIS2 delivery that is to be installed:

```
Package name:    <package-name>
User code:      SOL2P
Distribution medium: *REGISTERED-MEDIUM (default)
```

-
- (6) You can check and confirm the selected supply units in a dialog box by specifying SUPPLY-UNITS=*BY-DIALOG.

The supply units can also be specified directly in the SUPPLY-UNITS operand (in this case using SUPPLY-UNITS=PERCON, cf. Point 10).

- (7) The installation is carried out for the current operating system version.
- (8) The catalog ID 6A0B and the TSOS default system ID are entered for the imported pubset.
- (9) The WORK-FILE-DELETION=*NO operand can be used to retain the supply units to be installed under the park ID (for further installations).

```
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name      Vers  Corr  SU selection  Units  1 through  5 of  5
                More:
EDT            17.0  D30   :I29A:$PARKSW1  Installed
OPENFT        12.1  C10   :I29A:$PARKSW1  Installed
OPENFT-CR    12.1  B10   :I29A:$PARKSW1  Installed
X PERCON      02.9  A10   :I29A:$PARKSW1  Installed
SORT         08.0  A30   :I29A:$PARKSW1  Installed
                *** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

- (10)

- (10) After executing the statement, the PERCON supply unit is marked in the dialog box requested, and the dialog box is executed using [DUE].

The specifications in the INSTALL-UNITS statement are the minimum required for installation on the imported pubset. The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customerspecific). As the open park SCI is not the standard SCI for the target system, an additional query is output when generating the installation procedure:

Job ':I29A:\$PARKSW1.WORK.<package-name>.APR101333272021.IE' is being generated.

'1' supply unit has been selected

Generation of :I29A:\$PARKSW1.WORK.<package-name>.APR101333272021.IE started:

* Local SCI :I29A:\$PARKSW1.SYS.IMON-PARK.SCI

* is not a standard one. Should the installation results

* be registered in this one or in the standard SCI of the

* target system (:6A0B:\$TSOS.SYS.IMON.SCI) ?

* Reply Y=local SCI/ N=std SCI

* n _____ (11)

- (11) The installation can be registered in the (foreign) SCI currently open or in the standard SCI of the imported pubset. The installation is registered in the standard SCI of the imported pubset (as in a direct installation on the imported pubset) if you respond with "N".

The subsequent procedure up to creating and starting the installation procedure can be found in the relevant examples. In the following examples we will assume that the installation procedure was already successfully started:

```
//modify-imon-options
```

```
sci=: 6a0b:$tsos.sys.imon.sci _____ (12)
```

```
//show-supply-units unit-name=*from-solis2-delivery(
```

```
package-name= <package-name> ,user-code=sol2p),  
select=*by-attributes(installation-status=*installed) _____ (13)
```

```
:
```

```
//end _____ (14)
```

- (12) The standard SCI of the imported pubset (:6A0B:\$TSOS.SYS.IMON.SCI) is opened using the MODIFY-IMON-OPTIONS statement.
- (13) All of the supply units in the previously installed delivery (package name <package-name>, user code SOL2P) that are registered in the standard SCI are displayed (in this case only the PERCON and RMS supply units, cf. also the contents of the SCI on ["Example in menu mode "](#)).
- (14) IMON is terminated using the END statement.

3.2.4 Multiple installations of software, which is already installed

A block diagram illustrating the sequence is given on "[Sequence of a customer-specific multiple installation of a previously installed delivery](#)".

A multiple installation can also take place without parking. A prerequisite is that the software has already been installed once and the work files exist. The installation can be a repeat of the standard installation or a customer-specific installation. For the sequence of the installation the following points must be observed:

- The SCI in which the software, which is already installed, is registered, must be opened. Also, the same work file ID must be used as for the installation which has already been executed.
- The supply units, which are to be reinstalled, must be selected. For the choice of the supply units in menu mode, the display of the installed supply units or deliveries is requested with *View: Supply units* or with *View: Packages (deliveries)*.

3.2.4.1 Example in menu mode

Call IMON

IMON is started once English is set as the task-wide language for messages and masks. IMON switches to menu mode once the program is loaded and [DUE] entered.

```
/modify-msg-attributes task-language=e
/start-imon
% IMOLOAD Program 'IMON-TU', version 'V03.4' of 2021-02-22' loaded from fil
e ':I29A:$TSOS.SYSLNK.IMON-BAS.034.TU'
% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights Reserved
PLEASE ACKNOWLEDGE
```

The welcome screen describes the menu items for the first two steps of the installation when the standard SCI or foreign SCI is used.

```
File Edit Show View Options
-----
IMON: Welcome screen
-----

*** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI
-----

Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel
```

Open SCI

The standard SCI, from which an already-installed supply unit is to be installed, is to be opened. In the following example, the supply unit is imported from the default SCI to the home pubset. The 1 (*Open ...*) option is already preset in the *File* menu. The mask is executed using [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : : Welcome screen
: * . Close : -----
: * . Print documentation ... :
: * . Save as ... : e in Installation MONitor ***
: 5. Exit :
: .....:
: File: Open: Standard SCI
: to display standard SCI contents.
:
: Use menu
: File: Open: SOLIS2 delivery
: to install a delivery in standard SCI.
:
: Use menu
: File: Open: Foreign SCI
: and then menu
: File: Open: SOLIS2 delivery
: to install a delivery in a foreign SCI
:
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

In the *Open* menu, select 1 (*Standard SCI*) and confirm with [DUE].

```

File Edit Show View Options
-----
: 1 1. Standard software configuration inventory : -----
: 2. Foreign software configuration inventory ... : -----
: 3. Installation definition file ... :
: 4. SOLIS2 delivery ... : ***
: 5. Structure and installation information file ... :
: :
: F1=Help F12=Cancel
:
: .....:
:
: Use menu
: File: Open: SOLIS2 delivery
: to install a delivery in standard SCI.
:
: Use menu
: File: Open: Foreign SCI
: and then menu
: File: Open: SOLIS2 delivery
: to install a delivery in a foreign SCI
:
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The screen displays the contents of the open standard SCI. It contains 168 entries, i.e. 168 installation units that can be viewed by scrolling backwards and forwards using +/- in the command line. Before you can select a menu function, you may need to delete from the command line any presetting caused by the scrolling operation.

File Edit Show View Options

IMON: SCI: :I29A:\$TSOS.SYS.IMON.SCI

Unit name	Installation	Units Version	1 through units selection Corr state	13 of 168 More: +
ACS		21.0	A00	
AID		03.5	A00	
AIDSYS		21.0	A00	
AIDSYSA		21.0	A00	
ANITA		21.0	A00	
APACHE		02.4	A00	
ARCHIVE		12.0	C00	
ASE		21.0	A00	
ASSEMBH		01.4	A10	
ASSEMBH-GEN		01.4	A10	
ASSTRAN		05.4	A00	
ASTI		21.0	A00	
ATS		05.3	A20	

Command ==>

F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

Define work file ID

The following steps can be skipped when the default value for the work file ID is kept, as is the case in this example (continue then with View:Supply units), "Example in menu mode "). The work file ID can be respecified, as they were set at the time of the installation of the desired supply units, by entering 2 (IMON Options) in the Options menu. Execute the mask using [DUE].

```

File Edit Show View Options
-----
IMON: SCI: :I29A: : 2 1. Switch to SDF interface :
----- : 2. IMON options ... : -----
: ..... ough 13 of 168
: ..... Installation units selection More: +
Unit name Version Corr state
ACS 21.0 A00
AID 03.5 A00
AIDSYSA 21.0 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The preset work file ID is displayed as a partial path name with a catalog ID (in this case with : I29A:\$SYSSAG.). This default work file ID is accepted by entering [DUE].

```

File Edit Show View Options
-----
: ..... IMON options : -----
: ..... : 13 of 168
: Work file location: :I29A:$SYSSAG. : More: +
: Reference file ...:
: :I29A:$TSOS.SYS.IMON.SCI.REF
:
: F1=Help F12=Cancel
: .....
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Confirm supply units

In the View menu, the supply units registered in the SCI are displayed if 2 (Supply units) is selected. Alternatively, if 4 (Packages (deliveries)) is selected, the registered deliveries are displayed.

```

File Edit Show View Options
-----
: Filter :
:-----
: 2 1. Installation units : nits 1 through 13 of 168
: 2. Supply units ... : selection More: +
: 3. Supply units for Request Delivery : on Corr state
: 4. Packages (deliveries) : 0 A00
: : 5 A00
: F12=Cancel : 0 A00
:-----
: ANITA 21.0 A00
: APACHE 02.4 A00
: ARCHIVE 12.0 C00
: ASE 21.0 A00
: ASSEMBH 01.4 A10
: ASSEMBH-GEN 01.4 A10
: ASSTRAN 05.4 A00
: ASTI 21.0 A00
: ATS 05.3 A20
:-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The view is limited to supply units from this delivery by Package name=2 (Specific) and Name=<package-name>. So that the selection only contains supply units that are already installed, in the Installation Status the marker “/” is kept before the Installed condition (the markers of the other conditions must be erased):

```

File Edit Show View Options
-----
: Filter : IMON.SCI
:-----
: Filter supply units : 168
:-----
: Package name: 2 1. All 2. Other :
: Name: <package-name> :
: User code...: 1 1. All 2. Other :
: Name: :
: Customer approved...: 1 1. All 2. Yes 3. No :
: Last installation...: 1 1. All 2. Last :
: Activable.....: 1 1. All 2. Yes 3. No :
: Installation Status: / Installed / Being installed :
: / Parked / In Library :
: / On SOLIS2 support / On Local support :
: / Being deinstalled / Partially installed :
: / Others :
- : F1=Help F12=Cancel :-----
C :
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

The 5 parked supply units are displayed with the status “Installed”:

```

File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Unit name      Vers Corr Package name  User code  Inst. Status  More:  CAP Act
-----
EDT            17.0 D30 <package-name> SOL2P      Installed    N         N
Y
OPENFT        12.1 C10 <package-name> SOL2P      Installed    N         Y
OPENFT-CR     12.1 B10 <package-name> SOL2P      Installed    N         Y
PERCON        02.9 C00 <package-name> SOL2P      Installed    N         Y
SORT          08.0 A30 <package-name> SOL2P      Installed    N         Y
*** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

For another installation the EDT supply unit is selected by marking it and the selection is confirmed with [DUE].

```

File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Unit name      Vers Corr Package name  User code  Inst. Status  More:  CAP Act
-----
X EDT          17.0 D30 <package-name> SOL2P      Installed    N         N
Y
OPENFT        12.1 C10 <package-name> SOL2P      Installed    N         Y
OPENFT-CR     12.1 B10 <package-name> SOL2P      Installed    N         Y
PERCON        02.9 C00 <package-name> SOL2P      Installed    N         Y
SORT          08.0 A30 <package-name> SOL2P      Installed    N         Y
*** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Select install

You must then switch to the *Edit* menu to install the units. Option 4 (*Install*) is already preset and needs only to be confirmed with [DUE].

```
File Edit Show View Options
-----
IM : 4 *. Search ... :
---- : *. Add ... :
      : 3. Remove : gh 5 of 5
      : 4. Installed : More:
      U : *. Park ... : Status CAP Act
X EDT : 6. Generate installation definition file ... : d N
Y
X OPEN : 7. Deinstall ... : d N Y
X OPEN : 8. Undo ... : d N Y
X PERC : 9. Activate ... : d N Y
X SORT : 10. Check ... : d N Y
      : *. Request correction delivery :
      : 12. Customer-Approved Install ... :
:.....:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Select parameter file

The following default installation parameters options are preset by IMON in the "IMON parameter file" dialog box.

```
File Edit Show View Options
-----
: IMON Parameter file :
: Parameter file ...: 2 1. None 2. Standard 3. Other : 5
: File name: :I29A:$TSOS.SYSPAR.IMON.LAST :
: Save parameters...: 1 1. No 2. Standard 3. Other :
: File name: :
: F1=Help F12=Cancel :
: *** End of SU selection ***
:

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Parameter file:

- IMON will use the default parameter file SYSPAR.IMON.LAST from the current location, when present, to preset specific installation parameters in the following dialog boxes (preset).

Save parameters:

- No saving of the installation parameters will be performed (preset).

Specify target system

A different pubset is specified in the “Global installation parameters” dialog box under Target System in the “Standard Pubset” field with a “2”, the value 8.0 for the preset for the version is already correct (the target system version must match the version of the previous installation). Entering “2” in the “Delete work files” field will make the supply units to be installed remain available on the park ID. These entries are the minimum requirement to add another installation on an imported pubset.

```

File Edit Show View Options
: .....
:                               Global installation parameters
: .....
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 2 1. Default                2. Other
: Save old files.....: 1 1. No                    2. With LMS
:                               3. With ARCHIVE          4. With MAREN
:                               VSN.....:
:                               Device type:          MAREN location: *STD
: Undo preparation.....: 2 1. Yes                2. No
: Placement mode.....: 1 1. Standard          2. Other
: Activation preparation mode: 1 1. Standard          2. Other
: Password file.....: 2 1. Yes                2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes                2. No
: Start.....: 1 1. Immediately          2. By user
: Configuration checks.....: 1 1. Yes                2. No
: Work file deleting.....: 2 1. Yes                2. No
:
: F1=Help F12=Cancel
: .....

```

MM.N – current version

The catalog ID 6A0B and the system ID TSOS (preset) are set in the “Pubset parameters” dialog box for the imported pubset and are confirmed with [DUE].

```

File Edit Show View Options
: .....
:                               Global installation parameters
: .....
: Pubset parameters
: .....: ..: MM.N
:                               : ..: 2 1. Default                2. Other
:                               : ..: 1 1. No                    2. With LMS
: Catid.: 6a0b                : ..: 3. With ARCHIVE          4. With MAREN
:                               : ..:
: DEFLUID: TSOS                : SN.....:
:                               : evic type:          MAREN location: *STD
: F1=Help F12=Cancel          : ..: 2 1. Yes                2. No
:                               : ..: 1 1. Standard          2. Other
: Activation preparation mode: 1 1. Standard          2. Other
: Password file.....: 2 1. Yes                2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes                2. No
: Start.....: 1 1. Immediately          2. By user
: Configuration checks.....: 1 1. Yes                2. No
: Work file deleting.....: 2 2. Yes                2. No
:
: F1=Help F12=Cancel
: .....

```

The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customer-specific).

The subsequent procedure up to creating and starting the installation procedure can be found in the relevant examples.

Check the installation process

The installation process is checked below:

```

File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
                                Units      1 through      5 of      5
                                More:
Unit name  Vers Corr Package name  User code  Inst. Status  CAP Act
-----
Y EDT      17.0 D30 <package-name> SOL2P      Installed      N      Y
OPENFT    12.1 C10 <package-name> SOL2P      Installed      N      Y
OPENFT-CR 12.1 B10 <package-name> SOL2P      Installed      N      Y
PERCON    02.9 C00 <package-name> SOL2P      Installed      N      Y
SORT      08.0 A30 <package-name> SOL2P      Installed      N      Y
                                *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Close SOLIS2 delivery

The SOLIS2 delivery is closed by entering 2 (*Close ...*) in the *File* menu.

```

File Edit Show View Options
-----
: 2 *. Open ... : N.SCI
: 2. Close ... : -----
: *. Print documentation ... : Units      1 through      5 of      5
: 4. Save as ... : selection More:
: 5. Exit ... : name User code Inst. Status CAP Act
: .....: e-name> SOL2P Installed N
Y
OPENFT    12.1 C10 <package-name> SOL2P      Installed      N      Y
OPENFT-CR 12.1 B10 <package-name> SOL2P      Installed      N      Y
PERCON    02.9 C00 <package-name> SOL2P      Installed      N      Y
SORT      08.0 A30 <package-name> SOL2P      Installed      N      Y
                                *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Check the SCI

The standard SCI of the imported pubset should then be opened: Option 1 (*Open ...*) is already preset in the *File* menu and needs simply to be confirmed with [DUE].

```

File Edit Show View Options
-----
: 1 1. Open ... : : Nothing opened
: 2. Close : :
: * . Print documentation ... : :
: 4. Save as ... : :
: 5. Exit : :
: .....:

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

In the *Open* menu 2 (*Foreign SCI ...*) is selected and confirmed with [DUE].

```

File Edit Show View Options
-----
: .....:
: Open : :
: .....:
: 2 1. Standard software configuration inventory : :
: 2. Foreign software configuration inventory ... : :
: 3. Installation definition file ... : :
: 4. SOLIS2 delivery ... : :
: 5. Structure and installation information file ... : :
: : :
: F1=Help F12=Cancel : :
: .....:

-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

The path name of the subset 6A0B standard SCI is entered in the “Foreign SCI” dialog box:
:6A0B:\$TSOS.SYS.IMON.SCI.

```

File Edit Show View Options
-----
: 2 1. Standard software configuration inventory :
: 2. Foreign software configuration inventory ... :
:
:           Foreign software configuration inventory :
:           SCI name: :6A0B:$tsos.sys.imon.sci :
: F :
:....: F1=Help F12=Cancel :
:
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

Finally, the default SCI of the pubset 6A0B contains the installation units PERCON and RMS (i.e. after the end of the example installation in [section "Multiple installation of parked software"](#)). The installation unit EDT is now also displayed.

```

File Edit Show View Options
-----
IMON: SCI: :6A0B:$TSOS.SYS.IMON.SCI
-----
          Units      1 through      3 of      3
          SU selection  Version      Corr state
Unit name
EDT          17.0          D30
PERCON       02.9          C00
RMS          07.1          G00
          *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

**Terminate
IMON**

IMON is terminated by entering 5 (*Exit*) in the *File* menu and [DUE] (it can also be terminated via the [F3] function key).

3.2.4.2 Example with SDF statements

```
/modify-msg-attributes task-language=e _____ (1)
/start-imon input-interface=*sdf _____ (2)

% IMOLOAD Program 'IMON-TU', version 'V03.4A00' of '2021-02-22' loaded from
file :I29A:$TSOS.SYSLNK.IMON-BAS.034.TU

% IMOCOPY Copyright (C) Fujitsu Technology Solutions 2021 All Rights
Reserved
```

- (1) The MODIFY-MSG-ATTRIBUTES command is used to set English for message output and language-dependent menus, as in the menu mode example.
- (2) The INTERFACE=*SDF command is used to start IMON with the SDF interface. Alternatively, you can also start IMON in menu mode (in which case it is sufficient to enter the START-IMON command, as INTERFACE=*STD is the default). After the user has explicitly opened an SCI with the *Open* option in the *File* menu, he or she can then select option 1 (*Switch to SDF interface*) in the *Options* menu.

```
/modify-imon-options
  sci=*std,
  work-file-location=$syssag., _____ (3)
  reference-file=*std
//install-units _____ (4)
  unit-name=*from-sci( _____ (5)
    package-name= <package-name> ,user-code=sol2p,
    supply-units=*by-dialog), _____ (6)
  target-system=*par(
    version=*current, _____ (7)
    pubset=*par(catid=6a0b,defluid=tsos)), _____ (8)
  work-file-deleting=*no _____ (9)
```

- (3) The MODIFY-IMON-OPTIONS statement is used to specify the park SCI (SOFTWARE-INVENTORY=\$PARKSW1.SYS.IMON-PARK.SCI operand) and the PARKSW1 park ID, the WORK file name prefix (WORK-FILE-LOCATION=\$PARKSW1.WORK. operand) for work files and the default reference file for the configuration check (operand REFERENCE-FILE=*STD). The park SCI and work file ID must be set in the same way as they were for the park operation (also compare with point 3 in example [section "Example with SDF statements"](#)).
- (4) The installation is then carried out using the INSTALL-UNITS statement.
- (5) The operand UNIT-NAME=*FROM-SCI(...) specifies a SOLIS2 delivery, which is already installed and registered, from which is to be installed again:

Package name: <package-name>

User code: SOL2P

- (6) You can check and confirm the selected supply units in a dialog box by specifying SUPPLY-UNITS=*BY-DIALOG.

The supply units can also be specified directly in the SUPPLY-UNITS operand (in this case using SUPPLY-UNITS=EDT, cf. Point 10 (Example with SDF statements)).

- (7) The installation is carried out for the current operating system version (preset). The target system version for this method of installation must match the target system version of the SOLIS2 delivery, which is already installed.
- (8) The catalog ID 6A0B and the TSOS default system ID are entered for the imported pubset.
- (9) The WORK-FILE-DELETION=*NO operand can be used to retain the supply units to be installed under the park ID (for further installations).

```

File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
          Units      1 through      5 of      5
          SU selection
Unit name  Vers Corr Package name  User code  Inst. Status  CAP Act
X EDT      17.0 D30 <package-name> SOL2P    Installed
Y
OPENFT     12.1 C10 <package-name> SOL2P    Installed      N   Y
OPENFT-CR  12.1 B10 <package-name> SOL2P    Installed      N   Y
PERCON     02.9 C00 <package-name> SOL2P    Installed      N   Y
SORT       08.0 A30 <package-name> SOL2P    Installed      N   Y
          *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

- (10)
- (10) After executing the statement, the EDT supply unit is marked in the dialog box requested, and the dialog box is executed using [DUE].

The specifications in the INSTALL-UNITS statement are the minimum required for installation on the imported pubset. The remaining specifications in this dialog box and also the subsequent procedure depend on the installation type required (default or customerspecific).

The further sequence up to creation and the start of the installation procedure can be seen from the respective examples. In the following it is assumed that the installation procedure has already been executed successfully:

```

//modify-imon-options
  sci=: 6a0b :$tsos.sys.imon.sci _____ (11)
//show-supply-units unit-name=*from-solis2-delivery(
  package-name= <package-name> ,user-code=sol2p),
  select=*by-attributes(installation-status=*installed) _____ (12)
:
//end _____ (13)

```

-
- (11) The standard SCI of the imported pubset (:6A0B:\$TSOS.SYS.IMON.SCI) is opened using the MODIFY-IMON-OPTIONS statement.
 - (12) With the statement SHOW-SUPPLY-UNITS, all supply units are displayed from the delivery with the package name <package-name> and the customer ID SOL2P, which are registered with the status "Installed" in the default SCI of the imported pubset (in this case only the supply units EDT, PERCON, and RMS, also compare with contents of SCIs on ["Example in menu mode "](#)).
 - (13) IMON is terminated using the END statement.

3.2.5 Placement and activation in the customer-specific installation

ePlacement and activation can be defined as follows in the customer-specific installation:

	In menu mode	Using the INSTALL-UNITS statement
Placement		
Customer-specific placement	Via the menu: <i>Edit</i> , option 4 (<i>Install</i>): “Global installation parameters”: Placement mode: 2 (Other) The context switches to the “Supply units placement parameters” dialog box. The following values can be changed there:	PLACEMENT-MODE= *BY-DIALOG
Modify path names (catalog ID, user ID, prefix)	Modify the values in the “Catid”, “Userid”, “Prefix” columns	Modify the values in the “Catid”, “Userid”, “Prefix” columns
Overwrite existing files <ul style="list-style-type: none"> • Yes • Only in certain cases • No 	Modify the value in the “Replace files” column <ul style="list-style-type: none"> • Y (preset) • M • N 	REPLACE-OLD-FILES= <ul style="list-style-type: none"> • *YES (preset) • *MINIMUM • *NO
Force location <ul style="list-style-type: none"> • No • Only in certain cases • Yes 	Modify the value in the “Force Loc.” column <ul style="list-style-type: none"> • N (preset) • M • Y 	FORCE-LOCATION= <ul style="list-style-type: none"> • *NO (preset) • *MINIMUM • *YES
Activation		
Customer-specific activation	Via the menu: <i>Edit</i> , option 4 (<i>Install</i>): “Global installation parameters”: Activation preparation mode: 2 (Other) The “Global activation preparation parameters” dialog box is displayed. The following values can be modified there:	ACTIVATION-MODE= <u>*PARAMETERS(...)</u>
Syntax file processing	<ul style="list-style-type: none"> • 1 (Yes) <ul style="list-style-type: none"> ◦ 2 (No) SDF param file	SYNTAX-FILE-PROCESS= *BY-DIALOG / *NO / *YES

Message file processing	<ul style="list-style-type: none"> • 1 (Yes) • 2 (No) MIP param file System message file name	MESSAGE-FILE-PROCESS= *BY-DIALOG / *NO / *YES
DSSM processing	<ul style="list-style-type: none"> • 1 (Yes) • 2 (No) DSSM Catalog name Keep old version: <ul style="list-style-type: none"> • 1 (Yes) • 2 (No) 	DSSM-FILE-PROCESS= *BY-DIALOG / *NO / *YES
RMS processing	<ul style="list-style-type: none"> • 1 (Depot and Loader) • 2 (Depot only) • 3 (No) Depot location: <ul style="list-style-type: none"> • 1 (Standard) • 2 (Enforced) Location	REP-FILE-PROCESS= *BY-DIALOG / *NO / *YES
POSIX processing	<ul style="list-style-type: none"> • 1 (cfg) • 2 (all) • 3 (no) 	POSIX-PROCESSING= *BY-DIALOG / *BY- CONFIGURATION-FILE / *ALL / *NO

Placement – specifying the depot location of the files to be installed

IMON reads the supply components (release items) of the supply units to be installed from the distribution medium and stores them as installation items on the target system. The file attributes of an installation item are taken from the delivery information. The depot location, i.e. the path name of the installation item is affected by the definitions in the delivery information and other user input.

The path name is formed as follows in the case of further user input:

```
:<catid>:$<userid>.[<prefix>]<item-name>
```

where:

- <catid> Refers to the catalog ID of the pubset on which the supply units will be installed. The catalog ID is taken from the target system specification (TARGET-SYSTEM=*PARAMETERS(...,PUBSET = <catid>).

<userid> Refers to the installation ID. If you specify a user ID that differs from that for the default installation, the user ID to be used is formed in accordance with the FORCE-LOCATION setting and the stipulations of the delivery information (see under: "Specifying the installation ID" "[Placement and activation in the customer-specific installation](#) ").

<prefix> refers to a prefix that is used if the path name for the installation item can be selected freely. The prefix is specified in the following sequence:

1. Prefix of the associated supply group
2. Prefix of the associated supply unit
3. Otherwise no prefix

<item-name> refers to the installation item.

The storage location of an installation item specified in the delivery information can be changed when evaluating a customer-specific IMON parameter file (see [section "IMON parameter files" \(IMON parameter files\)](#)) using the options set there. The IMON parameter file can contain the following options relating to the storage location:

- One installation ID can be specified for each supply unit specified.
- The catalog ID and user ID as well as the prefix can be specified for centrally storing documentation files and subsystem declarations.

Specifying the installation ID

- FORCE-LOCATION=*NO

Modification of the installation ID is not to be forced.

It is specified in the following sequence:

1. Location that was defined explicitly for the associated supply group.
2. Location that was specified explicitly for the associated supply unit.

If a location for the associated supply group or supply unit is prescribed explicitly (mandatory) in the delivery information, the installation is aborted if a different location is specified.

- FORCE-LOCATION=*MINIMUM

Modification of the location (installation ID and prefix) is not forced in the case of permanently defined user IDs.

It is specified in the following sequence:

1. Location that is defined permanently in the product movement file for the associated supply group.
2. Location that is defined permanently in the product movement file for the associated supply unit.
3. Location that was specified explicitly for the associated supply group.
4. Location that was specified explicitly for the associated supply unit.

If a location for the associated supply group or supply unit is prescribed explicitly (mandatory) in the delivery information, the installation is continued using the mandatory location even if a different location is specified.

-
- FORCE-LOCATION=*YES

Modification of the installation ID is to be forced.

It is specified in the following sequence:

1. Location that was defined explicitly for the associated supply group.
2. Location that was specified explicitly for the associated supply unit.

If a location for the associated supply group or supply unit is prescribed explicitly (mandatory) in the delivery information, the installation is continued even if a different location is specified.

Files for which a non default location (installation ID and prefix) was forced are listed in a specific log file as this type of installation can stop the product behaving correctly.

Specifying the installation item name

If no such file exists, it is created with the file attributes defined for the supply component.

Existing files of the same name are overwritten in accordance with the user input (REPLACE-OLD-FILES=*MINIMUM/*YES). Whether or not the files are overwritten also depends on whether the delivery information rules the original file names to be mandatory.

If the file name is not mandatory, existing files are not overwritten by the user input REPLACE-OLD-FILES=*MINIMUM/*NO. The installation item is stored in a new file under the substitute path name generated by IMON. The substitute path name is formed as follows:

```
:<catid>.$<userid>.[<prefix>]<item-name>.nnn
```

The suffix nnn refers to a consecutive number from 001 through 999. If the appended suffix renders the file name too long, the installation items is shortened accordingly from the end.

The file attributes of the new file are taken from the delivery information. If the file name is mandatory, existing files are overwritten by the user input REPLACE-OLD-FILES=*MINIMUM (as with *YES also). The previous file attributes are retained.

The following applies if a file cannot be overwritten because of a file lock, for example:

- If the file is a syntax or message file (installation items of the type SDF and MES), the installation procedure tries to cancel the file lock. If this attempt is unsuccessful, the procedure is canceled and a message that requires an answer is output on the operating console. The operator's response governs the subsequent processing:
 - “Repeat” The installation procedure attempts once again to cancel the file lock.
 - “Ignore” The installation is continued despite the file lock, i.e. the affected installation item is not installed.
 - “Abort” The installation is aborted with an error.
- The installation is canceled immediately if another installation item cannot be overwritten.

If the file name is mandatory, existing files are not overwritten by the user input REPLACE-OLD-FILES=*NO and the installation is canceled.

Activation – preparing for activation of the installed software

Activation of the installed software is started using the INSTALL-UNITS statement, operand ACTIVATION-MODE=*PARAMETERS(...) or, in menu mode, in the “Global Activation Preparation Parameter” screen.

Syntax files

ACTIVATION-MODE=*PARAMETERS(SYNTAX-FILE-PROCESS=*BY-DIALOG/*NO/*YES)

Unlike in the default installation, in the customer-specific installation you can

- prevent activation of the syntax files
- modify the name of the SDF parameter file for the next system start (corresponds to MODIFY-SDF-PARAMETERS with SCOPE=*NEXT-SESSION)

Message files

ACTIVATION-MODE=*PARAMETERS(MESSAGE-FILE-PROCESS=*BY-DIALOG/*NO/*YES)

Unlike in the default installation, in the customer-specific installation you can

- prevent activation of the message files
- modify the name of the MIP parameter file
- modify the value of the MSGFIL02 system parameter and thereby the name of the global message file

Note on reinstallation of syntax and message files which have already been activated

In the course of reinstallation IMON deactivates the file concerned, renames the file <file>.nnn (suffix nnn = 001 through 999), and reactivates the renamed file. Subsequently the new component is installed under its standard name. When the next startup takes place, the syntax and message files with a suffix are automatically deleted.

Subsystem

ACTIVATION-MODE=*PARAMETERS(DSSM-PROCESSING=*BY-DIALOG/*NO/*YES)

Unlike in the default installation, in the customer-specific installation you can

- prevent processing of the subsystem catalog
- modify the name of the subsystem catalog
- retain or remove the earlier version of a removable subsystem

The subsystem catalog name specified in the selection window may be displayed there when using a customer-specific IMON parameter file (see [section "IMON parameter files"](#)).

RMS files

ACTIVATION-MODE=*PARAMETERS(REP-PROCESSING=*BY-DIALOG/*NO)

When you park a supply unit, the system corrections (REP installation item) are transferred to the RMS depot (if the preset of the operand UPDATE-RMS-DEPOT from the statement PARK-UNITS or the option *RMS Depot actualise=1* (Yes) on the "Park parameter" screen is kept).

You can specify that the Rep loader is to be generated during the parking procedure when using a customer-specific IMON parameter file (see [section "IMON parameter files"](#)).

If the supply unit has not yet been parked, you can specify transfer to the RMS depot or the name of the RMS depot. You can also specify whether the rep loader is to be generated during the installation.

If the rep loader is generated during the installation, IMON enters the logical names in the SCI and extends the current path names with the default values from the RMS file.

If the rep loader is generated after the installation, the current path names are not contained in the SCI. The current path names must then be entered in the SCI in accordance with the SET-INSTALLATION-PATH command.

POSIX satellites

ACTIVATION-MODE=*PARAMETERS(POSIX-PROCESSING=*BY-DIALOG / *BY-CONFIGURATION-FILE / *NO / *ALL)

Unlike in the default installation, in the customer-specific installation you can prevent activation of the POSIX files.

Procedures

Procedures (of types DO and ENT) that are to be executed during installation are displayed when the installation parameters are queried. Automatic start is the default when installing on the home pubset. If the automatic start is explicitly suppressed, then the procedures must be started manually after installation is complete (just like for an installation on an imported pubset). In this case the names of the procedures to be started manually are output to SYSOUT when generating the installation procedure or to the console while the installation procedure is executing (see also the console log). Procedures of the type DO are started using the ENTER-PROCEDURE or CALL-PROCEDURE command, procedures of the type ENT are started using the ENTER-JOB command.

4 IMON functions and interfaces (IMON-BAS)

This chapter is an introduction to IMON-BAS. It tells you how to start IMON and describes the functions that IMON-BAS supports using menu options, SDF statements and macro calls.

Overview of the IMON-BAS functions

The following overview shows the IMON-BAS menu options, along with the corresponding SDF statements and macro calls. In the menu options at least one work SCI must have been opened (*File: Open Standard SCI or Foreign SCI*).

Function	Menu option	Statement	Macro
Add installation unit	<i>File: Open ...</i> <i>IDF file ... or SYSSII file ...</i> <i>Edit: Add ...</i>	ADD-INSTALLATION- UNITS	-
Activate supply unit	<i>View: Filter...</i> <i>Supply units ...</i> <i>Edit: Activate ...</i>	ACTIVATE-UNITS	-
Check installation unit	<i>Edit: Check ...</i>	CHECK-UNITS	-
Check supply unit	<i>View: Filter...</i> <i>Supply units ...</i> <i>Edit: Check ...</i>	CHECK-UNITS	-
Deinstall supply unit	<i>View: Filter...</i> <i>Supply units ...</i> <i>Edit: Add ...</i>	DEINSTALL-SUPPLY-UNIT	-
Export installation unit	<i>Edit: Generate IDF ...</i>	GENERATE-IDF	-
Install supply unit	<i>File: Open ... SOLIS2 Delivery ...</i> <i>Edit: Install ...</i>	INSTALL-UNITS	-
Install "approved" supply unit	<i>View: Filter...</i> <i>Supply units ...</i> (and <i>Customer approved=Yes</i>) <i>Edit:Customer-Approved Install...</i>	INSTALL-UNITS UNIT= *CUSTOMER-APPROVED	-

Park supply unit	<i>File: Open ... SOLIS2 Delivery ...</i> <i>Edit: Park ...</i>	PARK-UNITS	-
Print documentation	<i>File: Open ... SOLIS2 Delivery ...</i> <i>File: Print documentation ...</i>	PRINT-DOCUMENTATION	-
Remove installation unit	<i>Edit: Remove</i>	REMOVE-INSTALLATION-UNITS	-
Delete supply unit	<i>View: Filter...</i> <i>Supply units ...</i> <i>Edit: Remove</i>	REMOVE-SUPPLY-UNITS	-
Delete registered SOLIS2 delivery	<i>View: Filter ...</i> <i>Packages (deliveries)</i> <i>Edit: Remove</i>	REMOVE-PACKAGES	-
Close / open SCI	<i>File: Close</i> <i>File: Open ... SCI</i>	MODIFY-IMON-OPTIONS	-
Set work file ID	<i>Options: IMON options ...</i>	MODIFY-IMON-OPTIONS	
Set reference file	<i>Options: IMON options ...</i>	MODIFY-IMON-OPTIONS	
Save SCI	<i>File: Save as ...</i>	SAVE-SOFTWARE-INVENTORY	-
Show installation units	<i>Show: Installation units ...</i> <i>Show: From formatted file ...</i>	SHOW-INSTALLATION-UNITS	IMOSHIU
Show installation items	<i>Show: Installation item ...</i> <i>Show: From formatted file ...</i>	SHOW-INSTALLATION-ITEMS	IMOSHII
Show formatted file	<i>Show: Formatted file</i>	SHOW-FORMATTED-FILE	IMOSHFF
Output registered SOLIS2 delivery	<i>View: Filter ...</i> <i>Packages (deliveries)</i> <i>Show: Packages</i>	SHOW-PACKAGES	-

Show supply units	<i>View: Filter ...</i> <i>Supply units ...</i> <i>Show: Supply units ...</i> <i>Show: From formatted file ...</i>	SHOW-SUPPLY-UNITS	IMOSHSU
Switch between menu and statement mode	<i>Options: Switch to SDF interface</i>	SWITCH-TO-FHS	-
Undo last installation of the supply unit	<i>View: Filter ...</i> <i>Supply units ...</i> <i>Edit: Undo ...</i>	UNDO-SUPPLY-UNIT	-
Request correction delivery	<i>View: Filter ...</i> <i>Supply units for Request Delivery ...</i> <i>Edit: Request correction delivery ...</i>	REQUEST-CORRECTION-DELIVERY	
Set approval ID for supply unit	-	SET-CUSTOMER-APPROVAL	-
Reset approval ID of a supply unit	-	RESET-CUSTOMER-APPROVAL	-
Terminate Imon	<i>File: Exit or [DUE]</i>	END	-

The descriptions of the IMON-BAS menu functions start on "[Description of the IMON-BAS menu options](#) ", the IMON-BAS statements start on "[The IMON-BAS statements](#) ", and the IMON-BAS macro calls on "[The IMON-BAS macros](#) ".

The scope of the output or the availability of a function depend on the processing status and the user's status, i.e. nonprivileged or privileged user (SUBSYSTEM-MANAGEMENT privilege).

4.1 Starting and terminating IMON-BAS

IMON-BAS is called using the command START-IMON. If the TPR subsystem IMON is not yet active, this command starts it automatically.

Two interfaces are available for operating IMON-BAS:

- Menu interface

IMON-BAS runs in menu mode. The menu interface is only available in interactive dialog.

General information on the IMON-BAS menu interface can be found in the [section "Working with the menu interface of IMON-BAS"](#). The menu functions are described in the [section "Description of the IMON-BAS menu options"](#).

- SDF interface

IMON-BAS runs in statement mode, i.e. using SDF statements. The statements are described in the [section "The IMON-BAS statements"](#).

The INPUT-INTERFACE operand determines which interface is used to start IMON-BAS in the interactive dialog. The interface can also be switched here while the program is running.

A welcome screen is output when IMON-BAS is started in the menu mode that informs the user of calls to some important functions..

START-IMON	Alias: IMON
VERSION = *STD / <product-version>(..)	
,MONJV = *NONE / <filename 1..54 without-gen-vers>	
,CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>	
,INPUT-INTERFACE = *STD / *SDF	

VERSION =

The IMON version to be used.

VERSION = *STD

Use the version set with the SELECT-PRODUCT-VERSION command. If no version was selected, the highest available version of IMON is loaded by default.

VERSION = <product-version >

Defines the product version in the format mm.n[a[so]], e.g. 2.9 or 2.9A or 2.9A00 (see also description of SDF data types in section "[SDF syntax representation](#)").

MONJV =

A job variable for monitoring the IMON session.

MONJV = *NONE

No job variable.

MONJV = <filename 1..54 without-gen-vers>

The name of the job variable that is to monitor the IMON session.

CPU-LIMIT =

Maximum CPU time in seconds allowed for the program at runtime.

CPU-LIMIT = *JOB-REST

Use the remaining CPU time for the task.

CPU-LIMIT = <integer 1..32767 seconds>

The maximum time to be used.

INPUT-INTERFACE =

The interface to be used to start IMON-BAS.

INPUT-INTERFACE = *STD

Start IMON-BAS with the menu interface in interactive mode.

Start IMON-BAS with the SDF interface in procedure or batch mode.

INPUT-INTERFACE = *SDF

Start IMON-BAS with the SDF interface.

Notes

1. The TPR subsystem IMON-GPN must be started.
2. The TPR subsystem IMON is started automatically using the START-IMON command provided IMON-BAS and IMON-GPN were installed correctly and IMON-GPN is already started.
3. When you start IMON at the console, you must specify the operand INPUT-INTERFACE=*SDF.
4. When the program ends, the monitoring job variable can assume the following values:

Status display (byte 0 - 2)	Return code (byte 3 - 7)	Meaning
\$T	0000	No error
\$T	1010	Statement or function rejected, program is resumed.
\$A	2010	Statement or function rejected, program is aborted. Spin-off takes place.
\$A	2015	Unexpected EOF on SYSDTA.
\$A	3020	Internal inconsistency.

Terminating IMON-BAS

The way in which you terminate IMON-BAS depends on the current operating interface:

- You can terminate IMON-BAS from the menu interface either by pressing [F3] or by selecting the *Exit* option in the *File* menu.
- If the SDF interface is set, you can terminate IMON-BAS using the END statement.

4.2 Working with the menu interface of IMON-BAS

When you invoke IMON in interactive mode using the START-IMON command, the basic IMON mask appears when the program starts. This mask offers you a choice of menu options and the option of entering keystroke commands or control statements in the instruction area (see [section "Description of the IMON-BAS menu options"](#)). See the manual "FHS" [7] for details.

The welcome screen informs the user which menu options are required to call some important IMON functions.

If IMON is started in a procedure or as a batch job, the SDF interface of IMON is called.

```
File Edit Show View Options
(1)
-----
(2)
                                     IMON: Welcome screen
-----

                                     *** Welcome in Installation MONitor ***

Use menu
File: Open: Standard SCI
to display standard SCI contents.

Use menu
File: Open: SOLIS2 delivery
(3)
to install a delivery in standard SCI.

Use menu
File: Open: Foreign SCI
and then menu
File: Open: SOLIS2 delivery
to install a delivery in a foreign SCI

-----
Command ==>
(4)
F1=Help F3=Exit F10=Menu F12=Cancel
(5)
```

Figure 9: Basic IMON mask

Menu bar (1)

The menu bar shows the available menu titles. Each menu title represents a group of menu options that are displayed in a pull-down menu beneath the menu title. You use these menus to select the functions of IMON.

The IMON menu bar contains the menus *File*, *Edit*, *Show*, *View* and *Options*.

Status bar (2)

The status bar provides information on the contents of the mask and the currently opened object. No objects are opened after IMON is called, and the welcome screen is output first.

Body (3)

The body contains information about the currently opened object: All the elements of the object, which the user can select for further actions, are displayed. If all the elements cannot be displayed in the body, the user can move the body as required by scrolling. An element is selected by entering any character in the input box in front of the element name. Help information is displayed on the welcome screen immediately after IMON is called.

The body also displays dialog boxes, in which the user is prompted to enter the parameters that are necessary for an action, or in which messages are displayed.

Instruction area (4)

The instruction area enables you to control the dialog using control statements (see the [section "Description of the IMON-BAS menu options"](#)).

Key area (5)

The key area contains an overview of the current function key template.

Working with the basic IMON mask

In the basic IMON mask, you can select menu options, enter abbreviated commands (keystroke commands) as well as enter control statements in the instruction area.

The various functions are displayed in the form of menus in the menu bar. These menus are used to select the IMON-BAS functions.

Press [F10] to activate the menu bar. Use the tab key to move from one menu title to the next. The allocated menu options can then be displayed by pressing the [DUE] key to confirm the selected menu title.

The options are numbered consecutively within the menu (starting with 1). A specific option in the menu can be selected by entering the corresponding number in the input field that appears in front of the first menu option. The input field for certain options may already have a preset option number, depending on the menu and the processing status. You must confirm the entry by pressing [DUE].

If IMON needs additional information in order to perform an action, a dialog box appears. Enter the parameters that IMON needs to perform the action in this dialog box. In this manual, a detailed description of the parameters follows the description of each function.

Messages

IMON messages are displayed in a dialog box. Close this dialog box by pressing either [F3] or [F12].

Abbreviated commands and control statements

Abbreviated commands (keystroke commands) facilitate the input of standard actions. An abbreviated command is executed by pressing the corresponding function key. The assignment of function keys for a number of standard cases is shown in the key area. The instruction area accepts the input of control statements. These statements are described in detail in the manual "FHS" [7].

The following table compares IMON abbreviated commands (keystroke commands using function keys) with control statements. Control statements can be entered in the instruction area as an alternative to the abbreviated commands.

Key	Control statement	Meaning
[F1]	HELP	Call context-sensitive help of FHS
[F3]	EXIT	Exit IMON Close dialog box
[F5]	PREVIOUS	Display the previous installation unit or supply unit (see "1. Edit: Search ..." (Edit menu))

[F6]	NEXT	Display the next installation unit or supply unit (see "1. Edit: Search ..." (Edit menu))
[F7]	BACKWARD; -	Page up one page
	--	Go to first page
[F8]	FORWARD; +	Page down one page
	++	Go to last page
[F10]	MENU	Select menu functions
[F12]	CANCEL	Abort function, undo selection, close dialog box

If there are no function keys available, the SETP control statement can be used to allocate corresponding function keys to the P keys. The allocation made using SETP only affects the current IMON run: It is lost if you switch to the SDF interface from menu mode and then switch back again to the FHS menu interface using the SWITCH-TO-FHS statement.

The SETP control statement has the following syntax:

$$\text{SETP } \left\{ \begin{array}{l} P_n \\ (P_n, \dots, P_m) \\ P_n-P_m \end{array} \right\} \left\{ \begin{array}{l} \text{ON} \\ \text{OFF} \end{array} \right\} [, \dots]$$

|

Meaning of the operands

- P_n is the P key with the number n.
- (P_n, \dots, P_m) refers to all enumerated P keys: P_n, \dots, P_m .
- (P_n-P_m) refers to the P keys from P_n to P_m .
- ON allocates the corresponding F keys to the specified keys.
- OFF removes the allocation of the specified keys.

The SETP operands can be specified more than once. In this case, they must be separated by a comma; see example below. Each comma can be surrounded by any number of blanks.

Example

```
SETP P1 ON, P2 OFF, (P3,P7,P8) ON, (P4-P6) OFF
```

This SETP control statement makes the following allocations:

P1 - F1, P3 - F3, P7 - F7, P8 - F8

The allocations of P2 as well as P4, P5 and P6 were removed.

Switching between menu and statement mode

The *Switch to SDF interface* option in the *Options* menu allows you to switch from the menu mode to the statement mode of IMON. The SCI you opened prior to the switch remains available.

Alternatively, the SWITCH-TO-FHS statement can be used to switch to the IMON menu mode.

4.3 Description of the IMON-BAS menu options

Menu options for operating IMON-BAS using the menu interface are displayed in the *File*, *Edit*, *Show*, *View* and *Options* menus. These menu options are used to select the object on which the user wants to work, and the desired action is then selected and activated.

Selecting an object is a two-step procedure:

1. Select and open the SCI, IDF file, SOLIS2 delivery or SYSSII file (*File* menu, *Open* option).
2. Select and edit the associated installation units and supply units (*Edit* and *Show* menu).

Overview of all menu options

File	Show
<ul style="list-style-type: none">1. Open ...2. Close3. Print documentation ... ¹⁾4. Save ...5. Exit IMON	<ul style="list-style-type: none">1. Installation units ...2. Installation item ...3. From formatted file ...4. Formatted file5. Supply units ...6. Packages
Edit	View
<ul style="list-style-type: none">1. Find ...2. Add ...3. Remove4. Install ... ¹⁾5. Park ... ¹⁾6. Generate installation definition file ...7. Deinstall ... ¹⁾8. Undo ... ¹⁾9. Activate ... ¹⁾10. Check ... ¹⁾11. Request correction delivery ...12. Customer-Approved Install ...	<ul style="list-style-type: none">1. Filter ...
	Options
	<ul style="list-style-type: none">1. Switch to SDF interface2. IMON options ...

The options marked ¹⁾ must be selected under the TSOS user ID because privileged functions are invoked.

Notes

- The menu options available depend on the processing state and currently also on the caller's privileges. The number of the option is replaced by the "*" character for options that are temporarily not available.
- No object is open initially after IMON is started. If no SCI is open, after a IDF file, a SOLIS2 delivery or a SYSSII file are opened, the following editing steps are executed implicitly for the standard SCI. If the foreign SCI is to be edited, then it must be opened explicitly via *File: Open... Foreign SCI*.

The options in the *File* menu are used to open or close an SCI, an IDF file, a SOLIS2 delivery or a SYSSII file. In addition, the currently opened SCI is saved, delivery documentation is printed, or IMON is terminated in this menu.

The options in the *Edit* menu are used to search through the installation units and/or supply units for a certain string, add or delete installation units in the SCI, start the installation process, park selected software (supply units), generate an IDF file, start a deinstallation process, undo the last installation for a supply unit (Undo), activate an installation unit or supply unit or to check the correctness and up-to-dateness of registered installation or supply units.

The options in the *Show* menu are used to display information from an SCI, a SOLIS2 delivery, and a formatted file. This information is displayed online or is written to a formatted file.

The options in the *Select* menu are used to specify if other selected installation units, supply units or deliveries are to be displayed.

The options in the *Options* menu are used to switch to the statement mode, to change the work file ID or the reference file.

4.3.1 File menu

Options in the *File* menu:

1. *Open ...*

Opens an SCI, an IDF file, a SOLIS2 delivery or a SYSSII file.

2. *Close*

Closes an SCI, an IDF file, a SOLIS2 delivery or a SYSSII file.

3. *Print documentation ...*

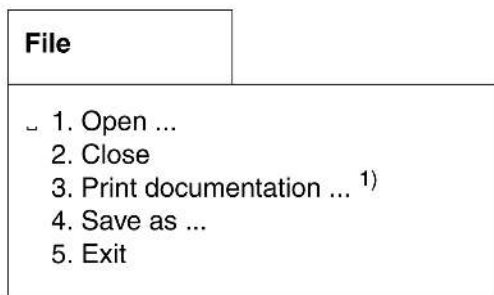
Prints the delivery documentation.

4. *Save as ...*

Creates a backup of the currently open SCI. The backup is performed in pairs for the IMON-SCI and the GPN-SCI.

5. *Exit*

Terminates IMON.



The options marked ¹⁾ must be selected under the TSOS user ID, because privileged functions are invoked.

1. File: Open ...

Option 1 (*Open ...*) in the *File* menu opens an object. The following objects can be selected using the option numbers in the dialog box that is displayed:

1. Standard SCI
2. Foreign SCI
3. IDF file
4. SOLIS2 delivery
5. SYSSII file

The welcome screen is output and no objects are open initially when IMON is started. The IDF file, SOLIS2 delivery and SYSSII file objects can only be viewed with the SUBSYSTEM-MANAGEMENT privilege.

```

File Edit Show View Options
-----
: 1 1. Standard software configuration inventory      :
: 2. Foreign software configuration inventory ...   :
: 3. Installation definition file ...              :
: 4. SOLIS2 delivery ...                           : ***
: 5. Structure and installation information file ... :
:
: F1=Help F12=Cancel
:
:.....:
:
:           Use menu
:           File: Open: SOLIS2 delivery
:           to install a delivery in standard SCI.
:
:           Use menu
:           File: Open: Foreign SCI
:           and then menu
:           File: Open: SOLIS2 delivery
:           to install a delivery in a foreign SCI
:
-----
Command ==>
F1=Help F3=Exit F10=Menu F12=Cancel

```

Figure 10: Open dialog box

The selection options depend on the objects that are already open:

- If no object is open, any object can be opened.
- If no SCI is open, IMON implicitly opens the standard SCI when an IDF file, a SOLIS2 delivery or a SYSSII file is opened, i.e. the following editing steps are carried out in the standard SCI.
- If an SCI is open, an IDF file, a SOLIS2 delivery or a SYSSII file can be opened, but not another SCI. This function can only be executed if you have the SUBSYSTEM- MANAGEMENT privilege.
- The *Open* option in the *File* menu is not available if:
 - an SCI and an IDF file are open
 - an SCI and a SOLIS2 delivery are open.
 - an SCI and a SYSSII file are open

The *Close* option in the *File* menu closes the last object you opened.

Open: Standard SCI

When you select option number “1”, the standard SCI is opened on the home pubset. For privileged users the SCI is opened with write access, otherwise it is opened for read access only.

All installation units from the standard SCI are displayed in the selection window (see figure 12). You can select installation units for further processing (see "File menu ").

Open: Foreign SCI

When you select option number “2”, a foreign SCI is opened.

You are prompted to enter the file name of the foreign SCI in the following dialog box.


```

File Edit Show View Options
:-----:
:                               Open                               :
:-----:
:                               SOLIS2 delivery                    :
:-----:
: Package name.....: <package-name>                               :
: User code.....: sol2p                                           :
: Distribution medium: 2 1. SOLIS2 volume                2. Library      :
:                               3. Registered                4. Local volume  :
:                               Volume.....:                :
:                               Device type:                :
:-----:
: Documentation library: 1 1. Standard                2. Other      :
:                               Name:                :
: Work file location: :I29A:$SYSSAG.                    :
: Reference File ... : :I29A:$TSOS.SYS.IMON.SCI.REF          :
:-----:
: Supply unit selection 1 1.Manually (*BY-DIALOG)        :
:                               2.Automatically (*BY-CONFIGURATION) :
: F1=Help  F12=Cancel                                     :
:-----:
Command ==>
F1=Help  F3=Exit  F10=Menu  F12=Cancel

```

Figure 14: Open a SOLIS2 delivery

Package name

Package name as stated in the footer of the delivery contents.

User code

User code as stated in the footer of the delivery contents.

Distribution medium

Details of the distribution medium. If 1 (SOLIS2 support) or 2 (Library) is specified here, the supply units that can be installed are determined by the delivery information for the delivery; if 3 (Registered) or 1 (Local support) is specified here, the SCI content determines the supply units that can be installed.

1 (SOLIS2 support)

The delivery is located on a data volume shipped with SOLIS2 (tape cartridge, CD or DVD). The volume serial number (VSN) and the device type can be taken from the delivery information. All supply units of the specified delivery are available for selection. The delivery information is read from MTC nd registered in the SCI.

Note

If a delivery is already registered in the SCI, a warning is issued and all informations about the delivery and all included supply units are then overwritten. The following effects can be observed:

- Even files that are already installed are read again from the data volume during installation or parking.
- Supply units of this delivery, which were alerady installed, can only be used again for an installation with UNIT-NAME=*FROM-SCI, after they have been installed again from the data volume.

Volume

VSN of the data volume which contains the delivery.

Device type

Device type of the data volume.

2 (Library)

The delivery is located in a PLAM library under the name SOLFTR.<package-name>.<user-code> under the user ID SYSSAG, where <package-name> and <user-code> are the values of "Package name" and "User code".

The delivery information is read from the library and registered in the SCI.

3 (Registered)

Read the delivery information, the VSN and the device type of the dddddddd from the open SCI. Files that have already been installed are not read again from the data volume.

It is now possible to install the same supply unit for the same system multiple times using the *REGISTERED-MEDIUM functionality (e.g. after deleting a file of the supply unit by mistake). Moreover, the "Installed" status of a supply unit will now be set only in the SCI of the system where the supply unit is effectively installed and no longer in the SCI from where the installation is triggered (where it remains unchanged).

Specifying this option results in the installation of:

- parked software
- partially installed software
- software for which you printed the delivery information beforehand.

4 (Local support)

The delivery is located on a data volume created by the customer (tape cartridge or CD-ROM or virtual volume in a CentricStor). The data volume information of the product movement file is ignored. All supply units of the specified delivery are available for selection. The delivery information is read from MTC and registered in the SCI.

Note

If a delivery is already registered in the SCI, a warning is issued and all informations about the delivery and all included supply units are then overwritten. The effects are the same as for 1 (SOLIS2 support).

Documentation library

Name of the library in which the delivery documentation will be stored. If you specify 1 (Standard) here, the standard name is used; if you specify 2 (Other), the explicitly entered name is used. If the library already exists, the new delivery documentation is added to it.

1 (Standard)

Use the standard name <work-file-location>.<package-name>.DOC, where <package-name> is the value of "Package name" and <work-file-location> is the value of "Work file location".

2 (Other)

Explicit specification of the library name in the “Name” field.

Name

Name of the library.

Work file location

Specifies the location at which either work files or software are to be buffered (“parked”).

The default entry in this field is the path name of the current work file ID setting.

If a different work file code is specified, this declaration remains effective until the delivery is closed. Temporary files cannot be specified here.

Reference file

Specifies the location of the reference file, which is to be used as the basis for a configuration check (see [section "IMON reference files"](#)). The field is preset with the location of the currently set reference file.

Supply unit selection Specifies method of supply unit selection

1 Select manually supply unit
(Manually)

2 (Automatically)

Install all supply units based on configuration.

All supply units from delivery will be installed if they are in higher or equal version than already installed in system.

If supply unit was not in system, then it's installation will be skipped.

Selecting supply units (from the SOLIS2 delivery)

When you open a SOLIS2 delivery, the status bar displays the package name and the user code of the delivery. All supply units of the SOLIS2 delivery are displayed in the selection window. The supply units can be selected for further processing.

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                                Units      1 through      5 of      5
                                SU selection
                                Version      Corr state
X EDT                          17.0      D30
X OPENFT                       12.1      C10
X OPENFT-CR                    12.1      B10
X PERCON                       02.9      C00
X SORT                         08.0      A30
                                *** End of SU selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 15: Select supply units of the SOLIS2 delivery

You can select the supply units by entering any character in front of the desired supply unit. Scrolling is allowed while making your selection. Confirm your selection by pressing the [DUE] key. Any control statements (e.g. "+") in the instruction area must first be deleted.

When you open a "non-registered" delivery (on MTC or in the PLAM library), all supply units are initially marked "X".

When you open a registered delivery, all supply units that are not yet parked or installed are initially marked "X".

As soon as you select at least one supply unit, you can install or park the selected supply units using the *Install* or *Park* options in the *Edit* menu. The delivery documentation can be printed even if you have not selected a supply unit.

Open: SYSSII file

Select option "5" to open a SYSSII file. Enter the file name of the SYSSII file in the following dialog box.

```

File Edit Show View Options
:-----:
: Open :
:-----:
: 5 *. Standard software configuration inventory : gh 13 of 163
: *. Foreign software configuration inventory ... : More: +
: 3. Installation definition file ... : te
: 4. SOLIS2 delivery ... :
: 5. Structure and installation information file ... :
:-----:
: F : Structure and installation information file :
:-----:
AP : SII name: :
AR : :
AS : F1=Help F12=Cancel :
AS :-----:
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 16: Open a SYSSII file

SII name Name of the SYSSII file.

When you open the SYSSII file, the items of this SYSSII file are displayed in the body.

As soon as you open the SYSSII file, the option *Add* is available in the *Edit* menu. Press [DUE] to confirm the execution of a standard registration (without further user data). If your registration requires further user data, you must select the desired items in the selection window.

2. File: Close

Option 2 (*Close*) in the *File* menu closes the last object you opened. In addition to an SCI (standard or foreign SCI), a SOLIS2 delivery, an IDF file or a SYSSII file can be open at the same time. As soon as you close an object, another object can be opened and processed (e.g. a SYSSII file can be opened after closing a SOLIS2 delivery).

All open objects are closed when you terminate the IMON run (*Exit* option in the *File* menu, [F3] or END statement).

3. File: Print documentation

Option 3 (*Print documentation ...*) in the *File* menu prints the delivery documentation for an open SOLIS2 delivery. You can select the documents to be printed in the following dialog box. The dialog box is also output when exactly one delivery (package) was selected for the display of deliveries (set using *Select: Filter... Packages (deliveries)*).

You need to have the SUBSYSTEM-MANAGEMENT privilege to use this function.

```

File Edit Show View Options
-----
: Print documentation :e-name> User code: SOL2P
:-----
: Return letter.....: 1 1. Yes 2. No : 1 through 5 of 5
: Delivery contents.....: 1 1. Yes 2. No : More:
: Release notices.....: 1 1. Yes 2. No : Corr state
: : D30
: F1=Help F12=Cancel : C10
:-----
: : B10
X PERCON 02.9 C00
X SORT 08.0 A30
*** End of SU selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 17: Define scope of printing for delivery documentation

- Return letter Print/do not print return letter for systems support.
 Answer 1 (Yes) or 2 (No); default is 1.
- Delivery contents Print/do not print delivery information.
 Answer 1 (Yes) or 2 (No); default is 1.
- Release notices Print/do not print release notices for supply units.
 Answer 1 (Yes) or 2 (No); default is 1.

4. File: Save as ...

Option 4 (Save as ...) in the *File* menu allows you to copy the current SCI (IMON and IMON-GPN files) during an IMON run and save it under the file name specified in the dialog box. The corresponding IMON-GPN-SCI is saved with the same file name and the suffix “.GPN”.

```

File Edit Show View Options
:-----:
: Save software configuration inventory as :
:-----:
: Destination filename: :I29A:$TSOS.SYS.IMON.SCI.20100509135406 :
: :
: F1=Help F12=Cancel :
:-----:
ADAM                15.0          A00
AID                 03.0          C00
AIDSYS             15.0          A00
AIDSYSA            15.0          A00
ANITA              15.0          A00
APACHE             01.3          A11
ARCHIVE            07.0          B05
ASSEMBH-GEN        01.2          C00
BCAM               17.0          A08
BCAM-DIAG          01.0          A06
BCAM-GEN           01.0          A00
BINDER             02.3          A00
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

```

File Edit Show View Options
:-----:
: Save software configuration inventory as :
:-----:
: Destination filename: :I29A:$TSOS.SYS.IMON.SCI.20210409135406 :
: :
: F1=Help F12=Cancel :
:-----:
AID                 03.5          A00
AIDSYS             21.0          A00
AIDSYSA            21.0          A00
ANITA              21.0          A00
APACHE             02.4          A00
ARCHIVE            12.0          C00
ASE                21.0          A00
ASSEMBH            01.4          A10
ASSEMBH-GEN        01.4          A10
ASSTRAN            05.4          A00
ASTI               21.0          A00
ATS                05.3          A20
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 18: Save SCI file

Destination filename File name of the backup copy. The default file name is the name of the SCI with the date and time as the suffix (.yyyymmddhhmmss).

5. File: Exit

Terminate IMON by selecting option 5 (*Exit*) in the *File* menu and pressing [DUE]. You can also terminate IMON by pressing [F3]

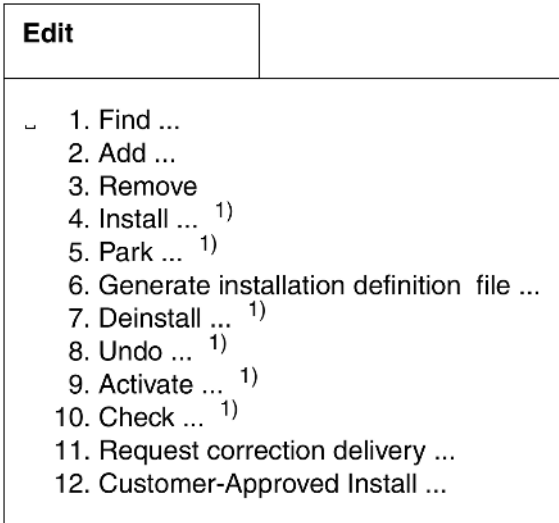
Objects that are still open are closed automatically when you terminate.

4.3.2 Edit menu

The *Edit* menu contains the following options:

1. *Search ...*
Searches for a specific string in the installation units.
2. *Add ...*
Adds installation units to the open SCI.
3. *Remove*
Removes installation units or supply units from the SCI.
4. *Install ...*
Starts the installation process.
5. *Park ...*
Stores software delivered on a data volume in the system.
6. *Generate installation definition file ...*
Adds supply units to an SCI using IDF.
7. *Deinstall ...*
Starts the deinstallation process.
8. *Undo ...*
Undoes the last installation of a supply unit.
9. *Activate ...*
Starts the dynamic activation of an installation unit or supply unit.
10. *Check ...*
Starts the check of the selected installation or supply units.
11. *Request correction delivery ...*
Request information on corrections and/or on a correction delivery for supply units.
12. *Customer-Approved Install ...*
Starts the installation process for supply units that are already installed on a pubset and that have a customer approval ID for release for other installations.

At least one object must be open before you can select the options in the *Edit* menu. In addition to an SCI (standard or foreign SCI), a SOLIS2 delivery, an IDF file or a SYSSII file can be open at the same time. The last object you opened is displayed in the status bar of the basic mask.



The functions marked ¹⁾ must be called under the TSOS user ID, because privileged functions are invoked.

1. Edit: Search ...

Use option 1 (*Search*) in the *Edit* menu to search for installation units or supply units whose names contain a specific string. The search function allows you to quickly display certain entries in the body without having to search and scroll manually.

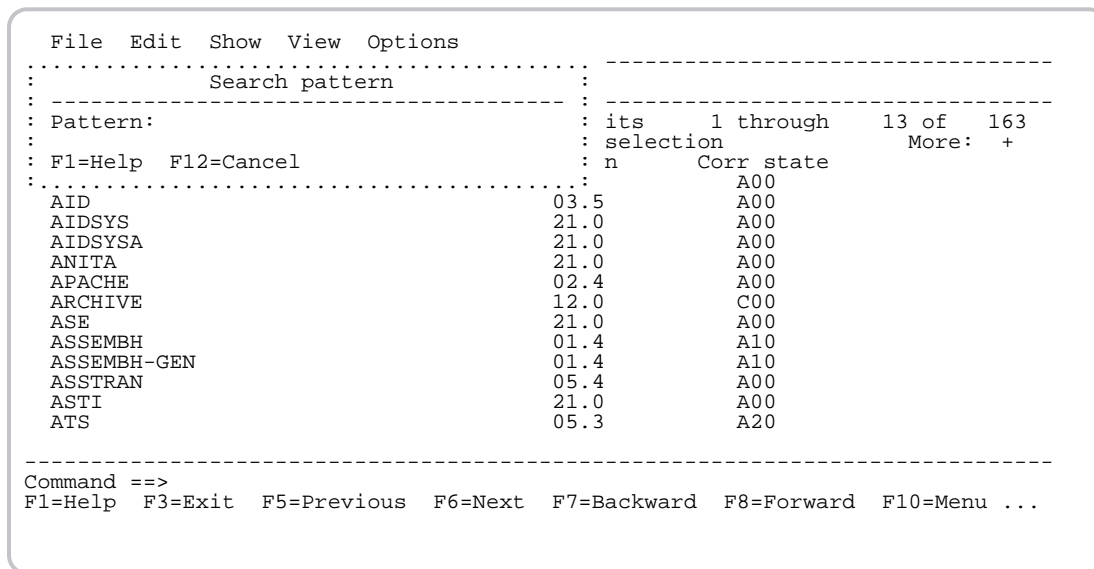


Figure 19: Search for installation units or supply units using a specific string

Enter a search pattern in the “Search pattern” dialog box. You can use the “*” and “/” characters as wildcards in a simple search pattern. An asterisk “*” stands for any string, including blanks. A slash “/” stands for any one character. Complex search patterns can be structured using BS2000 wildcards (see also the SDF metasyntax, ["SDF syntax representation"](#)).

When the search pattern is found, the cursor is positioned in the selection column in which the entry was found, and this can be selected as required for further processing. An error message is displayed if no matching strings are found.

Press [F6] or enter NEXT in the instruction area to view the next matching string. Press [F5] or enter PREVIOUS in the instruction area to return to the preceding matching string.

2. Edit: Add

With option 2 (*Add ...*) in the *Edit* menu you can enter private software and software components that are already installed in the system to the open SCI as installation units. This function requires the SUBSYSTEM-MANAGEMENT privilege.

Before registering, you must open an IDF or a SYSSII file containing information about the installation units and the installation items of the private software. You are prompted to enter the output target for the result log in the “Add installation units” dialog box.

```

File Edit Show View Options
-----
: Add installation units
-----
: Pubset: 1 1. Default          2. Other          : ersion : V02.1A00
:                               Catid:                :
:                               :                   : through 10 of 10
: Userid: 1 1. Standard        2. Other          :                   : More:
:                               Userid:                :
:                               :                   :
: Output: 1 1. Sysout          2. Syslst         :
:                               Syslst number: STD      :
:                               :                   :
: F1=Help  F12=Cancel         :                   :
:                               :                   :
:                               :                   :
SYSSDF.ACS.021
SYSREP.ACS.021
SYSSSC.ACS.021
*** End of RI selection ***
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 20: Add installation units to an SCI

Pubset

Pubset on which the installation items are to be installed.

1 (Default)

Default: Install the installation items on the home pubset of the current system under the default catalog ID of the corresponding user ID.

2 (Other)

Install the installation items on the catalog ID specified by “Catid”.

Catid

Catalog ID name.

Userid

User ID under which the installation items are to be installed

1 (Standard)

Default: Use the standard user ID from the IDF or SYSSII file.

2 (Other)

Use the user ID specified by "Userid".

Userid

User ID name.

Output

The output target for the result log.

1 (Sysout)

Default: The result log is output to SYSOUT

2 (Syslst)

The result log is output to SYSLST.

Syslst number

An explicitly specified SYSLST file number.

Possible entries: STD/<integer 1..99>

If STD is the default setting, the log is output to SYSLST.

Notes on registration

Each file with a path named in the IDF or SYSSII file is checked. The Add option sets the placement status. If the item is found, the attributes are taken from the system.

A result log is output as soon as the statement is executed. The log contains a list of the newly added installation units, plus their associated installation items or error messages.

3. Edit: Remove

Use option 3 (*Remove*) in the *Edit* menu to remove administrative information on superfluous installation units, supply units or packages (deliveries) from an open SCI. This function requires the SUBSYSTEM-MANAGEMENT privilege.

The option is available under the following conditions:

- At least one installation unit was selected from the installation units of the SCI displayed in the work area (see [figure 12 \(File menu\)](#)).
- At least one supply unit was selected from the supply units of the SCI displayed in the work area (set with *View: Filter... Supply units*). Supply units in the “Parked” or “Installed” installation state are not removed when *Remove* is selected.
- At least one of the deliveries displayed in the work area of the SCI was viewed (set with *View: Filter... Packages (Deliveries)*). Supply units, which are in the condition “Parked” or “Installed” are not deleted.

When a delivery is deleted, the related supply units are also deleted, provided they are in the installation status “Parked” or “Installed”. The following dialog box displays a prompt asking whether the related work files are to be deleted:

```

File Edit Show View Options
:-----:
: Remove package option :
:-----:
: Work Files Deletion : 2 1. Yes 2. No : age 1 through 1 of 1
: : : n More:
: Output: 1 1. Sysout :
: 2. Syslst :
: Syslst number: STD : ction ***
: :
: F1=Help F12=Cancel :
:-----:

Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 21: Optional deletion of work files when deleting a SOLIS2 delivery

Work Files Deletion	Specifies whether work files from this delivery on the currently selected work file ID will be deleted.
1 (Yes)	Work files from this delivery will be deleted.
2 (No)	Default: Work files from this delivery will remain.
Output	The output target for the result log.
1 (Sysout)	Default: Output the result log to SYSOUT.
2 (Syslst)	Output the result log to SYSLST.
Syslst number	An explicitly specified SYSLST file number.
	Possible entries: STD/<integer 1..99>
	The default setting STD outputs the result log to SYSLST.

4. Edit: Install

Use option 4 (*Install ...*) in the *Edit* menu to start the installation process. Refer to the section "[Installation](#)" for details of the installation process. This function requires the SUBSYSTEM-MANAGEMENT and USER-ADMINISTRATION privileges.

If the generated installation procedure is started manually (see parameter "Start = 2"), the installation, including the creation of the installation procedure, can be executed under any user ID to which both privileges are assigned. The current user ID must therefore be set as the Work file location (see *IMON options* in the *Options* menu). As a result, the generated installation procedure must be started under the TSOS user ID.

Exception

If you are installing from MTC, or if a user ID other than the current user ID is used for the installation, the entire installation must be performed as before under the TSOS ID.

It is essential that you have an open SCI and that you have opened a SOLIS2 delivery by selecting *Open* in the *File* menu. If you selected supply units when you opened the SOLIS2 delivery, only these are installed. If you did not explicitly specify certain supply units, the entire delivery is installed.

The installed software is locked to the user of the system until the next system start (see command LOCK-PRODUCT-VERSION in the „Commands“ [4] manual).

Installation parameter file to use or to generate are entered in the "IMON Parameter file" dialog box:

```
File Edit Show View Options
: .....
: IMON Parameter file :
: .....
: Parameter file ...: 2 1. None      2. Standard  3. Other
: File name:
:
: Save parameters...: 1 1. No      2. Standard  3. Other
: File name:
:
: F1=Help F12=Cancel
: .....
:
: .....
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Figure 22: IMON Parameter file

Parameter file

Specifies if a parameter file with predefined installation values has to be used. This input field only refers to the IMON parameter file for default installation parameters (default name is \$TSOS.SYSPAR.IMON.LAST).

Irrespective of this, the customer-specific IMON parameter file \$SYSPAR.IMON.<kkz> which optionally exists or the file \$TSOS.SYSPAR.IMON is always evaluated (see [section "IMON parameter file for optional installation functions"](#)).

1 (None)

The \$TSOS.SYSPAR.IMON.LAST is ignored. The \$TSOS.SYSPAR.IMON.<user-code> or \$TSOS.SYSPAR.IMON parameter file will be taken into consideration. In the case of entries which are contained in both files, the value from the parameter file \$TSOS.SYSPAR.IMON.LAST is used.

2 (Standard)

Both parameter files \$TSOS.SYSPAR.IMON.LAST and \$TSOS.SYSPAR.IMON[.<user-code>] will be used.

3 (Other)

A user-specific parameter file is to be used. In addition, the parameter file \$TSOS.SYSPAR.IMON[.<kkz>] is used. In the case of entries which are contained in both files, the value from the user-specific parameter file is used.

Save parameters

Specifies if the installation parameters used during the current installation process have to be saved.

1 (No)

Default: do not save the installation parameters.

2 (Standard)

The installation parameters will be saved either in the file \$TSOS.SYSPAR.IMON.LAST (when no input parameter file specified) or in the input parameter file specified by the user.

3 (Other)

Saving will be done in a user specified file.

File name

Name of the user parameter file where the installation parameters will be saved.

Parameters that apply to all supply units to be installed are entered in the “Global installation parameters” dialog box:

```

File Edit Show View Options
:-----
:                               Global installation parameters
:-----
: Target system
:   BS2000 version.....: MM.N
:   Standard pubset.....: 1 1. Default          2. Other
: Save old files.....: 1 1. No              2. With LMS
:                               3. With ARCHIVE      4. With MAREN
:
:                               VSN.....:
:                               Device type:      MAREN location: *STD
: Undo preparation.....: 2 1. Yes                2. No
: Placement mode.....: 1 1. Standard          2. Other
: Activation preparation mode: 1 1. Standard      2. Other
: Password file.....: 2 1. Yes                2. No
:   File name....:
:   Read password: C'
: Print log files.....: 2 1. Yes                2. No
: Start.....: 1 1. Immediately          2. By user
: Configuration checks.....: 1 1. Yes          2. No
: Work file deleting.....: 2 1. Yes          2. No
:
: F1=Help F12=Cancel
:-----

```

Figure 23: Install installation units using global installation parameters

Target system

System for which installation takes place.

BS2000

BS2000 version of the target system for which the installation is performed.

Permitted entries: 19.0 / 20.0 / 21.0

Standard pubset

Target pubset for installation. Install to the home pubset of the current system.

Default

Install to an imported pubset. You must enter the catalog ID and the

Other

system default ID in the “Pubset parameters” dialog box (see [figure 24](#)).

Save old files

Save/do not save system files to be overwritten when the new files are installed.

1 (No)

Default: Do not save the files.

2 (With LMS)

Save the files with LMS/LMSSCONV.

The save library is called <work-file-location>.IMON.SAVE.LIB.<package-name>, where <package-name> is the value of “Package name” and <work-file-location> is the value of “Work file location”.

3 (With ARCHIVE)

Save the files with ARCHIVE.

VSN

VSN of the data volume on which the files are to be saved.

Device type

Device type of the data volume.

4 (With MAREN)

This option is only shown if the MAREN subsystem is loaded.

The files are backed up with ARCHIVE, in the process of which the MAREN subsystem automatically specifies the VSN of a free volume (provided that a storage location is stated in the *MAREN Location* field).

MAREN Location

Name of the storage location from which the free volume is to be selected.

*STD is preset, i.e. the ARCHIVE backup is carried out without MAREN support, with the default settings of ARCHIVE.

Undo processing

This determines whether a backup should be created for the purpose of restoring the original status prior to installation (undo function). In this case, all the files updated or deleted during installation are first saved with ARCHIVE. In addition, all metadata required to restore the original status is saved in separate files (undo files).

1 (Yes)

A backup is performed for the call of the undo function. The ARCHIVE directory and the VSN and the device type of the volume are queried in a dialog box (see [figure 24](#)).

2 (No)

Default value: no backup is performed.

Placement mode

The location at which the installation items are placed.

1 (Standard)

Default: Preset entries from the SYSSII files are used for the placement location, for overwriting existing files and for dealing with user ID conflicts.

2 (Other)

The current placement parameters are displayed in the body and can be changed ([figure 27](#)).

Activation preparation mode

Defines which installation items are to be prepared for subsequent activation during the installation procedure.

1 (Standard)

Default: The standard measures are performed.

2 (Other)

The current activation parameters are displayed in the body and can be changed ([figure 29](#)).

Password file

Determines whether the ADD-PASSWORD commands for password-protected files were stored in a file.

1 (Yes)

The passwords were stored in a file.

File name

Name of the file containing the ADD-PASSWORD commands for the password-protected files. These commands are integrated into the procedure file. The syntax of the password commands must be correct.

Read password

The read password of the password file. Only alphanumeric characters can be used in the read password.

2 (No)

Default: The passwords were not stored in a file.

Print log files

Print/do not print the installation logs.

1 (Yes)

Print the installation logs.

2 (No)

Do not print the installation logs.

Start

Determines whether the installation is started immediately and automatically, or by the user.

1 (Immediately)

Default: Start the installation immediately.

2 (By user)

Initiate the installation by starting the installation procedure using ENTER-PROCEDURE. The name of the installation procedure is reported when the file is created.

Configuration checks

Specifies whether a configuration check should be carried out. The current set reference file is the basis of the configuration check.

1 (Yes)

Default: the configuration check is executed.

2 (No)

The installation is executed without a configuration check.

Work file deleting

Delete/do not delete the work files that were established for “Work file location” using *IMON options* in the *Options* menu.

1 (Yes)

Default: Delete the work files.

2 (No)

Do not delete the work files.

Pubset parameters of an imported pubset

You must enter the parameters for the imported pubset, on which the supply units are to be installed, in this dialog box.

```

File Edit Show View Options
:-----:
:                               Global installation parameters                               :
:-----:
: Pubset parameters              :
:                               :
:                               : ..: MM.N
:                               : ..: 2 1. Default                2. Other
: Catid...: 6a0b                 : ..: 1 1. No                2. With LMS
:                               : ..: 3. With ARCHIVE        4. With MAREN
: DEFLUID: TSOS                 : SN.....:
:                               : evice type:                MAREN location: *STD
: F1=Help F12=Cancel           : ..: 2 1. Yes                2. No
:                               : ..: 1 1. Standard          2. Other
: Activation preparation mode: 1 1. Standard          2. Other
: Password file.....: 2 1. Yes                2. No
:   File name.....:
:   Read password: C'          '
: Print log files.....: 2 1. Yes                2. No
: Start.....: 1 1. Immediately          2. By user
: Configuration checks.....: 1 1. Yes                2. No
: Work file deleting.....: 2 2. Yes                2. No
:                               :
: F1=Help F12=Cancel
:-----:

```

Figure 24: Pubset parameters of an imported pubset

Catid Catalog ID of the imported pubset. The default Catid is the catalog ID of the home pubset.
This field is not filled anymore and must be filled with the HOME catalog ID of the target system.

DEFLUID System default user ID to be used for the installation on the imported pubset.

This field is filled by default either by the current DEFLUID of your system or by *STD (when a installation parameter file is used). In this last case, *STD will be converted to:

- the current DEFLUID of your system when no PVS-INFO record is found in the parameter file for the specified target pubset.

to the value found in the parameter file in the PVS-INFO record associated to the specified target pubset (see [section "IMON reference files"](#)).

Undo parameters

The parameters for saving the data, needed for restoration to the original state (Undo function) are queried in this dialog box.

```
File Edit Show View Options
: .....
:                               Global installation parameters
: .....
:                               UNDO Parameters
: -----
: Retention period: 030
: .....
: Save medium.....: 1 1. ARCHIVE directory 2. Tape
:                               3. Library
: File name....:
: VSN.....: Device type:
: .....
: F1=Help F12=Cancel
: .....
: File name....:
: Read password: C'
: Print log files.....: 2 1. Yes 2. No
: Start.....: 1 1. Immediately 2. By user
: Configuration checks.....: 1 1. Yes 2. No
: Work file deleting.....: 2 2. Yes 2. No
: .....
: F1=Help F12=Cancel
: .....
: .....
```

Figure 25: Undo parameters

Retention period	Specifies after how many days the Undo files generated may be changed or deleted. The default setting is 30 days.
Save medium	Specification of the data medium for saving the data.
1 (ARCHIVE directory)	Default: The data medium is determined from the pool in the ARCHIVE-directory file.
File name	Name of the ARCHIVE directory file.
2 (Tape)	The data medium is specified explicitly.
VSN	VSN of the data medium to be used for saving the data.

Device type	Device type of the data medium (must be specified in both cases).
3 (Library)	Data are saved in a PLAM library. Library name: <work-file-location>.IMON.UNDO.LIB.<package>

Target system parameters

This dialog box queries additional parameters for release items that are installed as a function of the target system.

```

File Edit Show View Options
:-----:
: Global installation parameters :
:-----:
: Target system parameters :
:-----:
: Processor-type: 4 1. CISC-390 : 2. Other :
: * .--- : 2. With LMS :
: 3. X86 : 4. With MAREN :
: 4. All : :
: AREN location: *STD :
: Item-selection: 2 1. All : 2. No :
: 2. Target-version-only : 2. Other :
: : 2. Other :
: : 2. No :
: F12=Cancel : :
:-----: : 2. No :
: Print log files.....: 2 1. Yes : 2. No :
: Start.....: 1 1. Immediately : 2. By user :
: Configuration checks.....: 1 1. Yes : 2. No :
: Work file deleting.....: 2 1. Yes : 2. No :
: : :
: F1=Help F12=Cancel : :
:-----:

```

Figure 26: Installing target system-dependent installation units

Processor-type Specifies the hardware type of the target system for the installation of hardware-dependent release items. The default and user-defined values are dependent on the target system version:

Target system version The presetting of Processor-type

BS2000 V19.0 – V21.0 4 (All); all values can be selected except 2 (*.--)

- 1 The target system runs on a /390 machine. Only those release items are installed that have the (CISC-390) TARGET=A (hardwareindependent) or the TARGET=S (/390 variant) code.
- 2 (*.--) Unsupported
- 3 (X86) The target system runs on a X86 machine.
Only those release items are installed that have the TARGET=A (hardware-independent) or the TARGET=K (X86 variant) code.

4 (All) Those release items are installed whose hardware variant is supported in the target system.

Target system version Supported hardware type

BS2000 V19.0 – V21.0 4 (All); all values can be selected except 2 (*.---)

Item Specifies if only those release items that are intended for the specified target system version are to be installed. The default value is 2; in other words, release items are installed only for the specified target system version.

1 All release items are installed regardless of the target system version.
(Both)

2 Only release items that are specified for the relevant target system are installed. Release items that are specified for other target system versions are not installed (this refers in particular to the item types SSC, SSD, MSV, SDF, *DA and REP).
(Target-version-only)

Placement parameters for supply units

The selected supply units are displayed together with the previously assigned global placement parameters in the following mask. The global placement parameters can be modified or new parameters can be added for the selected supply units.

```
File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                Units      1 through      5 of      5
                Supply units placement parameters
                More:
Unit name      Vers Corr Catid  Userid  Prefix  Replace Force
files         Loc.
EDT            17.0 D30  *DEF  TSOS   *NONE   Y       N
OPENFT        12.1 C10  *DEF  *STD   *NONE   Y       N
OPENFT-CR    12.1 B10  *DEF  *STD   *NONE   Y       N
PERCON        02.9 C00  *DEF  TSOS   *NONE   Y       N
SORT          08.0 A30  *DEF  *STD   *NONE   Y       N
*** End of Supply units placement parameters ***
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 27: Display selected supply units

- Unit name Name of the supply unit.
- Vers Version of the supply unit.
- Corr Correction state of the supply unit.

Catid	Catalog ID under which the supply unit will be installed. *DEF stands for the default catalog ID (DEFCAT) of the specified user ID.
Userid	<p>User ID under which the supply unit will be installed. *STD means that various release units/items of the supply unit will be installed under different user IDs.</p> <p>If the string *RU is specified for a supply unit, a follow-on mask appears in which you can enter specific details for the release units of the supply unit. Details of the release unit have priority over those of the supply unit.</p>
Prefix	Prefix for the installation file name (default: no prefix). If you specify a prefix, your specification must be fully qualified and a value must be specified for "Userid".
Replace files	<p>Overwrite/do not overwrite an existing file during installation.</p> <p>Y Default: Overwrite existing files.</p> <p>M Do not overwrite existing files, if possible.</p> <p>N Do not overwrite existing files.</p>
Force loc.	<p>Determines how IMON behaves when placing release items if the desired location (determined by catalog ID, user ID and prefix) is not the same as the stipulated user ID or prefix of the release unit or of the release item.</p> <p>Y Use the desired location for the release items. Any stipulated user IDs or prefixes are ignored.</p> <p>M Release units and items with a stipulated user ID or prefix are stored under the stipulated path name. Otherwise, the desired user ID is used.</p> <p>N The placement of the release items is aborted, if the location or the user ID is specifically stated for one of the release units or release items. Otherwise, the desired user ID is used.</p>

The following mask displays only the selected supply units for which a library, which is to be merged into an alternative library, exists. The way in which the library is handled can be modified for these supply units.

```

File Edit Show View Options
-----
IMON: SCI: :6A0B:$TSOS.SYS.IMON.SCI
-----
Units      1 through      2 of      2
Unit name  SU selection  Version  Corr state  More:
EDT        02.9         D30
SORT       08.0         A30
*** End of SU selection ***
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 28: Display selected supply units with library

Unit name Name of the supply unit.

Version Version of the supply unit.

Corr. state Correction state of the supply unit.

Library Determines how IMON handles the libraries of the supply units.

Y Catalog the libraries of the supply units under their original names on the target system. In addition, IMON merges their elements into alternative libraries.

N Catalog the libraries of the supply units only under their original names in the target system.

- The supply unit does not contain a library file.

Activation parameters of supply units

The selected supply units are displayed together with the previously assigned global activation parameters in a dialog box. The global activation parameters can be modified for the selected supply units.

```

Global activation preparation parameters
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Syntax file processing.: 1 1. Yes                2. No
  SDF param file.: :I29A:$TSOS.SYSPAR.SDF
Message file processing: 1 1. Yes                2. No
  MIP param file.: :I29A:$TSOS.SYSPAR.MIP.210
  MES file.: :I29A:$TSOS.SYSMES.EKP.01
DSSM processing.: 1 1. Yes                      2. No
  Catalog name.: :I29A:$TSOS.SYS.SSD.CAT.X
  Keep old version: 2 1. Yes                    2. No
RMS processing.: 1 1. Depot+loaders  2. Depot only  3. No
  Depot location.: 1 1. Standard      2. Enforced
  Location: :I29A:$TSOS
POSIX processing.: 1 1. Cfg            2. All          3. No
F1=Help F12=Cancel

```

Figure 29: Global parameters for preparing for activation

Syntax file processing	Process/do not process syntax files during the installation procedure.
1 (Yes)	Process syntax files.
SDF param file	Name of the SDF parameter file.
2 (No)	Do not process syntax files.
Message file processing	Process/do not process message files during the installation procedure.
1 (Yes)	Process message files.
MIP param file	Name of the MIP parameter file.
MES file	Name of the message file into which the message files of the subsystems are to be merged using CREATION-TIME=BEFORE-DSSM-LOAD and AT-DSSM-LOAD.
2 (No)	Do not process message files.
DSSM processing	Process/do not process the static DSSM subsystem catalog during the installation procedure.
1 (Yes)	Process the static DSSM subsystem catalog.
Catalog name	Defines which static DSSM subsystem catalog is processed. If you do not enter a name here, the default name of the static catalog is used. IMON saves an existing catalog as <old-catalog-name>.<timestamp> before replacing it with the new catalog. If the name is too long, <old-catalog-name> is abbreviated.
Keep old version	Keep/do not keep old versions of the processed subsystems in the DSSM subsystem catalog for subsystems that allow the removal of old versions.
1 (Yes)	Retain old subsystem versions. This option displays a dialog box for the explicit selection of supply units whose subsystem versions are to be retained.

2 (No)	Do not retain old subsystem versions.
2 (No)	Do not process the static DSSM subsystem catalog.
RMS processing	Process/do not process the RMS depot during the installation procedure.
1 (Depot + loaders)	Transfer the data into the RMS depot and generate the associated REP loaders.
2 (Depot only)	Transfer the data into the RMS depot.
Depot location	The name of the RMS depot for transferring data and generating the REP loaders.
1 (Standard)	The RMS depot that was specified for parking is used for parked software. Otherwise, the RMS depot specified for "Location" is used.
2 (Enforced)	The RMS depot specified for "Location" is always used.
Location	Explicit specification of the RMS depot.
3 (No)	The RMS depot is not updated, no REP loaders are generated. Important: If there are no REP loaders, the system may not be able to achieve a "System Ready".
POSIX processing	Process / do not process satellites during the installation procedure (installation calls in POSIX).
1 (Cfg)	There will be a POSIX installation only for those POSIX products inside the delivered package, which are already registered to \$SYSROOT.POSIX.CONFIGURATION.
2 (All)	There will be a POSIX installation for all POSIX products inside the delivered package, no matter whether they are already registered to \$SYSROOT.POSIX.CONFIGURATION or not.
3 (No)	No POSIX installation will be performed

Even when no POSIX configuration file (\$SYSROOT.POSIX.CONFIGURATION) is found on the target system, the field is prefilled with "1". I.e. preset in this case is that no POSIX installation is performed. The installation of the POSIX items contained in the delivery can be forced by the specification of "2" (All).

The current activation parameters for the supply units are displayed in the following mask and these can be modified or new parameters added for each supply unit.

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
                Units      1 through      5 of      5
Supply units activation preparation parameters
More:
Unit name      Vers      Corr      SDF      MSG      SUB      PSX
EDT             17.0      D30      Y      Y      Y      -
OPENFT         12.1      C10      Y      Y      Y      C
OPENFT-CR      12.1      B10      -      -      -      -
PERCON         02.9      C00      Y      Y      Y      -
SORT           08.0      A30      Y      Y      Y      -
*** End of Supply units activation preparation parameters ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 30: Display selected supply units

- Unit name Name of the supply unit.
- Version Version of the supply unit.
- Corr state Correction state of the supply unit.
- SDF Process/do not process the syntax files of the supply units.
 - Y Process the syntax files.
 - N Do not process the syntax files.
 - The supply unit does not have a syntax file or no processing was requested.
- MSG Process/do not process the message files of the supply units.
 - Y Process the message files.
 - N Do not process the message files.
 - The supply unit does not have a message file or no processing was requested.
- SUB Process/do not process the subsystem declarations.
 - Y Process the subsystem declarations.
 - N Do not process the subsystem declarations.
 - The supply unit does not have a subsystem declaration or no processing was requested.
- PSX Process/do not process POSIX satellites during the installation procedure for the corresponding supply unit.
 - The supply unit does not have POSIX satellite file or no POSIX processing was requested.

- N The POSIX satellites of the selected supply unit products will not be processed.
- A There will be a POSIX installation, no matter whether the supply unit is already registered in the file \$SYSROOT.POSIX.CONFIGURATION or not.
- C There will be a POSIX installation only if the supply unit is already registered in \$SYSROOT.POSIX.CONFIGURATION.

If “Keep old version=1(Yes)” was selected in the global activation parameters for DSSM processing, the following mask displays the supply units with the subsystem versions to be retained. Only those subsystem versions that the user selects explicitly are retained.

```

File Edit Show View Options
-----
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
              Units      1 through      4 of      4
SU selection for keeping old subsystem versions      More:
Unit name      Version      Corr state
X EDT          17.0          D30
X OPENFT      12.1          C10
X PERCON      02.9          C00
X SORT        08.0          A30
*** End of selection ***

-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 31: Selecting subsystem versions to be retained

5. Edit: Park

Use option 5 (*Park ...*) in the *Edit* menu to read software from MTC and park it on a user ID. This menu option allows you to prepare for installation, while at the same time registering the delivery in the SCI.

This function requires the SUBSYSTEM-MANAGEMENT privilege.

This menu option is available when an SCI and a SOLIS2 delivery, which was supplied on tape cartridge, are open. Supply units of this delivery are parked on the user ID and catalog ID specified as the Work file location in the *Options: IMON options* function.

You can install parked software using the *Install* option, but the SOLIS2 delivery must first be opened as a registered SOLIS2 delivery (see *Open* option in the *File* menu).

You must enter the specifications required for parking in the “Park parameters” dialog box.

```

File Edit Show View Options
:-----
:                               Park parameters                               :-----
:-----
: Target system                               :                               5
: BS2000 version          : MM.N
: Standard pubset.....: 1 1. Default                2. Other
:                               :                               Catid:
:                               :                               :
: Save old files.....: 1 1. No                    2. With LMS
:                               :                               3. With ARCHIVE  4. With MAREN
:                               :                               VSN.....:
:                               :                               Device type...:
:                               :                               MAREN location : *STD
: Update RMS depot.....: 1 1. Yes                    2. No
: Depot Location: *STD
:
: File location.....: :I29A:$PARKSW1
:
: F1=Help  F12=Cancel
:
:-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 32: Park parameters

Target system

System for which installation is to take place later.

BS2000 version

BS2000 version of the target system for which the installation is performed.

Permitted entries: 19.0 / 20.0 / 21.0

Standard pubset

Home pubset of the target system for installation. This specification can only be evaluated if the RMS depot is to be updated (see "Update RMS depot" on the next page).

1 (Home)

Default: Install under the standard catalog ID of the relevant user ID on the home pubset of the current system.

2 (Other)

Install on an imported pubset. You must enter the catalog ID of the pubset in the next field "Catid".

Catid

Catalog ID of the imported pubset.

Save old files

Save/do not save the files to be overwritten when the new files are parked.

1 (No)

Default: Do not save the files.

2 (With LMS)

Save the files with LMS/LMSCONV.

The name of the save library is <work file ID>.IMON.SAVE.LIB.<package name>.<cid>., where <package name> is the value of the "Package name" field, <cid> is the value of the "Customer ID" field and <work file ID> is the current work file ID setting or the <file-location> of the park ID specified in the "File location" field.

3 (With ARCHIVE)

Save the files with ARCHIVE.

VSN

VSN of the data volume to which the files are to be saved.

Device type

Device type of the data volume.

4 (With MAREN)

This option is only displayed if the MAREN subsystem is loaded.

The files are backed up with ARCHIVE. During this process, the MAREN subsystem automatically specifies the VSN of a free volume (provided that a storage location is stated in the *MAREN Location* field).

MAREN Location

Name of the storage location, from which the free volume is to be selected.

*STD is preset, i.e. the ARCHIVE backup is carried out without MAREN support, with the default settings of ARCHIVE.

Update RMS depot

Transfer/do not transfer the RMS delivery set into the RMS depot.

1 (Yes)

Default: Transfer the RMS delivery set into the RMS depot.

Depot Location

*STD / [:<catid>:]\$[<userid>].

Defines the catalog and/or user IDs under which the RMS depot is created. The default name is "RMS. DEPOT". When the default setting *STD is used, <catid> is the catalog ID of the pubset from the target system.

<userid> is taken from the delivery information.

2 (No)

Do not transfer the data into the RMS depot.

File location

The location (specified in the form [:<cat-id>:]\$<userid>.) at which the release items are to be parked. This specification replaces the previous location definition (see *Work file location*, *Options* menu, *IMON options* option).

6. Edit: Generate IDF file

Installation units or supply units can be copied from open SCI via an IDF file to another SCI with the 6 (*Generate IDF file ...*) option in the *Edit* menu (Import/ Export function, see also the GENERATE-IDF statement, "[GENERATE-IDF Export SCI entries using IDF](#)").

This function requires the SUBSYSTEM-MANAGEMENT privilege.

This menu option is available when an SCI is open and at least one supply unit or one installation unit is selected.

```
File Edit Show View Options
.....
:                                     Generation parameters
:-----
: Renaming.....: 1 1. No          2. Complete  3. Composed
: Copy procedure generation: 1 1. No          2. Yes+save  3. Yes
: Saving location:
:
: SCI.....: 1 1. Std          2. Other
: SCI filename...:
:
: Generated file prefix:
:
: Output.....: 1 1. Sysout        2. Syslst
: Syslst number:
:
: F1=Help  F12=Cancel
:
:-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 33: Parameters for exporting installation units or supply units

Renaming

Defines if and how the saving locations (and thus the path names) of the associated installation items are to be renamed.

1 (No)

Do not rename the saving locations.

2 (Complete)

Completely rename the saving locations. New names can be allocated to old saving locations in the "Complete renaming parameters" dialog box (see "[Edit menu](#)"). Old and new path names are specified with a catalog ID, a user ID and if necessary, a prefix.

3 (Composed)

Rename individual parts of the saving locations: catalog ID and/or user ID and/or prefix. Allocations can be specified for each part in the "Composed renaming parameters" dialog box (see "[Edit menu](#)").

Copy procedure generation

Generate/do not generate a copy procedure. The copy procedure contains COPY-FILE commands for all the selected installation items. The “Renaming” specifications define the source and target location of the copy action.

1 (No)

Default: Do not generate a copy procedure.

2 (Yes + Save)

Generate a copy procedure. The copy procedure also contains COPY-FILE commands for all the selected installation items. These commands are used to create a backup of the installation items that already exist in the target location.

Saving location

Name of the saving location for the backups.

3 (Yes)

Generate the copy procedure without backups.

SCI

The SCI to which the selected entries are to be added.

1 (Std)

Add the selected entries to the standard SCI.

2 (Other)

Add the selected entries to the SCI specified below.

SCI filename

File name of the SCI to which the selected entries are to be added.

Generated file prefix

Defines the names of the generated procedures. The import procedure has the suffix .GEN, while the copy procedure has the suffix .COPY.

Output

The output target for the result log.

1 (Sysout)

Default: Output the result log to SYSOUT.

2 (Syslst)

Output the result log to SYSLST.

7. Edit: Deinstall

Unneeded supply units are removed from a BS2000 system (deinstalled) with the 7 (*Deinstall ...*) option in the *Edit* menu. Details on the deinstallation procedure are given in [section "Deinstallation"](#). SUBSYSTEM-MANAGEMENT privileges are required to use the function.

One requirement is that there is an open SCI and at least one supply unit has been selected for deinstallation in the work area. The supply units registered in the SCI are displayed using the *Select* menu via the option *Filter: Supply units*.

When a supply unit is deinstalled, all actions that were performed during its installation are undone:

- activated files are deactivated, if necessary
- installed files are deleted, if necessary
- entries are removed from the SCI

The test mode is set via the *Execution=2* parameter. In this case only the preliminary analysis is performed to check to see if the deinstallation executed successfully.

Parameters that will apply to all supply units to be deinstalled are entered in the "Global deinstallation parameters" dialog box:

```
File Edit Show View Options
:-----Deinstallation parameters-----:
: Target system                          :
: BS2000/OSD Version.....: 1 1. Current                2. Other          :
:                               version:  :
: DSSM catalog name: :I29A:$TSOS.SYS.SSD.CAT.X :
: Save files .....: 1 1. No                2. With LMS       :
:                               3. With ARCHIVE  4. With prefix   :
:                               VSN.....:      P:          :
:                               Device type: :
: Execution.....: 1 1. Yes                2. No             :
: Output.....: 1 1. Sysout                2. Syslst        :
:                               Syslst number: STD :
: F1=Help F12=Cancel
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Figure 34: Global deinstallation parameters

Target system

Specifications on the target system for which the deinstallation is being performed. The target pubset is already specified by the catalog ID of the open SCI.

BS2000 Version

BS2000 version of the target system under which the deinstallation will be performed.

1 (Current version)

Default: Version of the currently running system.

2 (Other)

Explicit specification of the version

Permitted entries: 19.0 / 20.0 / 21.0

DSSM catalog name

Name of the static DSSM catalog from which the subsystem to be deinstalled is to be removed. The preset is the standard DSSM catalog \$TSOS.SYS.SSD.CAT.X on the target pubset.

Save files

Specifies if the files that will be deleted during deinstallation are to be saved beforehand.

1 (No)

Default: The files are not saved.

2 (With wsLMS)

The files are saved with LMS/LMSCONV in a library. Name of the save library:

\$<work file ID>.IMONDEI.<time-stamp>.SAVE.LIB, where <work file ID> is the current work file ID setting (see the statement MODIFY-IMON-OPTIONS "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

3 (With ARCHIVE)

The files are saved using ARCHIVE.

VSN

VSN of the data medium on which the data will be saved.

Device type

Device type of the data medium.

4 (With prefix)

Backup copies are made of the files. The file name of each backup copy is formed from the prefix specified in the P field and the original path name, where the catalog ID and the dollar symbol are omitted from the left side of the user ID.

P

File name prefix of the backup copy in the form <partial-filename 1..16>. The file name prefix may not be longer than 16 characters after adding the catalog ID and user ID of the caller.

Example

During deinstallation of the EDT (caller TSOS, prefix "D1.") the file

:I29A:\$TSOS.SYSLNK.EDT.170 is deleted, among others. The backup copy is saved under the name :I29A:\$TSOS.D1.TSOS.SYSLNK.EDT.170.

Execution

Specifies if the actual deinstallation is to be executed after performing the preliminary analysis. During the preliminary analysis a check is performed to see if the most important requirements to perform the deinstallation are fulfilled (see "[Deinstallation](#)").

1 (Yes)

The deinstallation is executed after a successful preliminary analysis.

2 (No)

The call is made in the test mode, i.e. only the preliminary analysis is performed for the deinstallation and any errors arising during it are recorded.

Output

Specifies where the result log is output.

1 (Sysout)

Default: The result log is output to SYSOUT.

2 (Syslst)

The result log is output to SYSLST.

Syslst number

Explicit specification of the number of the SYSLST file.

Possible entries: STD/<integer 1..99>

Output is sent to SYSLST using the preset value STD.

8. Edit: Undo

The last installation of a supply unit in a BS2000 system is undone with the option *8 (Undo ...)* in the *Edit* menu . The supply unit is deinstalled the original state before the supply unit was installed is restored based on the Undo files saved during installation. Details on the Undo function are described in [section "Undo - undoing an installation"](#).

SUBSYSTEM-MANAGEMENT privileges are required for the function.

The menu option is available when an SCI is open and at least one supply unit for the Undo function was selected in the work area. The supply units registered in the SCI are displayed via the *Filter: Supply units* option in the *Select* menu.

Requirements

- All changed or deleted files were saved during installation and the metadata of the installation was recorded in Undo files (see dialog box "Undo parameters" in the menu option *Edit: Install...*, or in the UNDO-
PREPARATION=*PARAMETERS(...) option on "INSTALL-UNITS Install and register software" for the INSTALL-
UNITS statement).
- Only the standard SCI (\$TSOS.SYS.IMON.SCI) on the home pubset or an imported pubset is open. The Undo function is rejected for another SCI. The test mode is set using the parameter *Execution=2*. In this case only the preliminary analysis used to monitor the successful execution of the Undo function is performed.

The necessary parameters are entered in the "Undo parameters" dialog box:

```

File Edit Show View Options
.....
:                                UNDO parameters                                :
:-----:
: DSSM catalog name: :I29A:$TSOS.SYS.SSD.CAT.X                               :
:
: Execution.....: 1 1. Yes                                           2. No           :
:
: Output.....: 1 1. Sysout                                           2. Syslst      :
:                                     Syslst number: STD              :
:
: F1=Help F12=Cancel
:
:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 35: Undo parameters

DSSM catalog name

Name of the static DSSM catalog to be restored. The default setting is the standard DSSM catalog :
<catid>:\$TSOS.SYS.SSD.CAT.X on the target pubset.

Execution

Specifies if the actual restoration is to be executed after performing the preliminary analysis. During the preliminary analysis a check is performed to see if the most important requirements to perform the restoration are fulfilled (see "Preparing the Undo function" "Undo - undoing an installation").

1 (Yes)

The restoration is executed after a successful preliminary analysis.

2 (No)

Only the preliminary analysis is performed for the restoration. Any errors arising during it are recorded.

Output

Specifies where the result log is output.

1 (Sysout)

Default: The result log is output to SYSOUT.

2 (Syslst)

The result log is output to SYSLST.

Syslst number

Explicit specification of the SYSLST file number.

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

9. Edit: Activate

Use option 9 (*Activate ...*) in the *Edit* menu, to activate previously installed supply units in the current system (the activation preparations during the installation only relate the next system start). Details of the dynamic activation are found in [section "Dynamic activation"](#) . The SUBSYSTEM-MANAGEMENT privilege is needed for this function.

If the generated activation procedure is manually started (see parameter "Start=2"), the activation and the creation of the activation procedure can be carried out under any desired user ID, which has both privileges. The current user ID must be set as the work file location (see the field of this name in *IMON Options* In the *Options* menu). The generated activation procedure must then be started under the user ID TSOS.

The prerequisites for this are: a SCI must be open; at least one supply unit must have been selected in the work area. The supply units registered in the SCI are displayed in the *View* menu under the *Filter: Supply units* option.

Parameters for dynamic activation of the selected supply units are entered in the "ACTIVATE parameters" dialog box:

```

File Edit Show View Options
.....
:                               ACTIVATE parameters                               :
:-----:-----:-----:-----:-----:-----:-----:-----:-----:-----:
: File prefix.....: 1 1. Std                               2. Other                               :
:   File prefix partial name.:                               :
: :
: Start .....: 1 1. Immediately                               2. By user                               :
: :
: Output .....: 1 1. Sysout                               2. Syslst                               :
:                               Syslst number:                               :
: :
: View level .....: 1 1. Supply units                               2. Installation units                   :
:                               3. None                               :
: :
: Keep old version .....: 2 1. Yes                               2. No                                   :
: :
: F1=Help F12=Cancel                               :
:-----:-----:-----:-----:-----:-----:-----:-----:-----:
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 36: Global Activation parameters

File prefix

Specifies the prefix of the activation procedure, which is saved at the following location as default:

`$<userid>.<prefix>.<time-stamp>.DA`

On the format of the time stamp `<time-stamp>` see "[The troubleshooting process](#)".

1 (Std)

Default: The prefix is the character string IMONACU.

2 (Other)

A different prefix is to be used.

File prefix partial name

A character string, which is to be used as a prefix.

Start

Specifies when the generated activation procedure is started.

1 (Immediately)

Default: The activation procedure is started immediately.

2 (By user)

The activation is initiated by manual starting of the activation procedure with the command ENTER-PROCEDURE. The name of the activation procedure is shown at the creation of the file.

Output

Specifies where the result log is output.

1 (Sysout)

Default: The result log is output to SYSOUT.

2 (Syslst)

The result log is output to SYSLST.

Syslst number

Explicit specification of the SYSLST file number.

Possible entries: `STD/<integer 1..99>`

The default setting STD outputs the information to SYSLST.

View level

Specifies whether an additional window is to be opened for the supply units, which are to be activated.

1 (Supply units)

Default: All previously selected, activable supply units are shown in an additional window.

2 (Installation units)

In an additional window, all installation units are shown, which belong to the previously selected supply units and are activable.

3 (None)

No additional windows are opened.

Keep old version

Specifies (on subsystems that permit version removal) whether old versions of the processed subsystems are to remain in the DSSM subsystem catalog or are to be removed.

1 (Yes)

Old subsystem versions are to be kept. Users are shown a dialog mask for explicit selection of the supply units whose subsystem versions can be retained.

2 (No)

Old subsystem versions are not kept.

10. Edit: Check

Use option *10 (Check ...)* in the *Edit* menu, to check the completeness, the correctness and the up-to-dateness of previously installed supply or installation units. This function requires the SUBSYSTEM-MANAGEMENT privilege.

The function is available under the following conditions:

- Only the standard SCI (\$TSOS.SYS.IMON.SCI) on the home pubset or on any imported pubset is open. The “Check” function is not available for another SCI.
- At least one installation unit was selected from the installation units of the SCI displayed in the work area (see [figure 12 \(File menu\)](#)).
- At least one supply unit was selected from the supply units of the SCI displayed in the work area (set with *View: Filter... Supply units*). Only supply units in the “Installed” installation state may be checked.

```

File Edit Show View Options
-----
: Check units : SCI
:-----
: Output: 1 1. Sysout : Units 1 through 13 of 168
: : 2. Syslst : units selection More: +
: Syslst number: : Version Corr state
: : 21.0 A00
: F12=Cancel : 03.5 A00
:-----
X AIDSYS 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 37: Check Units

- Output Specifies where the check result log is output.
 - 1 (Sysout) Default: The result log is output to SYSOUT.
 - 2 (Syslst) The result log is output to SYSLST.
- Syslst number Explicit specification of the SYSLST file number.
 - Possible entries: STD/<integer 1..99>
 - The default setting STD outputs the information to SYSLST.

11. Edit: Request correction delivery

Option 11 (*Request correction delivery ...*) in the *Edit* menu is used to request information on the available corrections of a supply unit or on a correction delivery for the supply unit. The option is available if after *View: Filter* the output of *Supply units for Request Delivery* is set and at least one supply unit was selected. This function requires the SUBSYSTEM-MANAGEMENT privilege.

IMON creates a correction request (with the file name \$TSOS.IMON.DELREQ.<timestamp>) from the user input and sends it by mail to the software distribution center of Fujitsu.

The parameters for requesting the corrections for the selected supply units are entered in the "Request parameters" dialog box.

```

File Edit Show View Options
.....
Request parameters
-----
: Delivery scope.....: 2 1. Modified items          2. All items          :
:                   3. Information only          :
:
: Distribution medium: 1 1. FT BS2000          2. FTP via WWW       :
:                   3. OpenFT via WWW          4. Other             :
:                   Device type:              :
:
: Delivery date.....: 1 1. Standard          2. Other             :
:                   Date:                    :
:
: Send request email.: 1 1. By user              :
:                   2. Immediately using local SMTP-SERVER :
:                   3. Immediately using      :
:                   SMTP-Server:            :
:
: F1=Help  F12=Cancel
:
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 38: Request parameters

Delivery scope

Defines the scope of the correction request.

1 (Modified items)

delivery that contains only the modified release items of the supply unit.

2 (All items)

Requests a correction delivery for the entire supply unit.

3 (Information only)

Requests only information on corrections available for the supply unit.

Distribution medium

Specifies the distribution medium for the requested corrections.

1 (FT BS2000)

Only for customers with a direct BS2000 network connection:

The correction delivery is to be transfered directly to the customer system using openFT.

2 (FTP via WWW)

The correction delivery is to be made available to the customer on the WWW for collection using FTP.

3 (OpenFT via WWW)

The correction delivery is to be made available to the customer on the WWW for collection using openFT.

4 (Other device type)

The correction delivery is to be distributed on a data volume.

The device type of the data volume should be entered in the following field.

Delivery date

Specifies the desired delivery date.

1 (Standard)

The correction delivery should be distributed or made available within two working days. WWW deliveries are generally made available one or two hours after request receipt.

2 (Other date)

The correction delivery should be distributed or made available on the specified date. The desired delivery date should be entered in the following field.

Send request email

Specifies how the mail with the correction request is to be sent to the software distribution center of Fujitsu.

1 (By user)

The mail is sent manually by the user.

2 (Immediately using local SMTP-SERVER)

The mail is sent immediately using the local mail server.

3 (Immediately using SMTP-SERVER:)

The mail is sent immediately using the mail server specified in the following field.

12. Edit: Customer-Approved Install

Option 12 (*Customer-Approved Install* in the *Edit* menu starts the installation process for supply units that are already installed on a pubset and that have a customer approval ID for release on further installations. A delta installation is then performed for the selected supply units; in other words, only those installation items that have an older installation timestamp in the target system are installed. Details on the installation procedure are described in [section "Install on the basis of customer approval IDs"](#).

If the generated installation procedure is started manually (see "Start = 2" parameter), installation, including creation of the installation procedure, can be executed under any user ID to which both privileges are assigned. The current user ID must therefore be set as the current work file location (see field of the same name in the *IMON options* in the *Options* menu). The generated installation procedure must then be started under the TSOS user ID.

Output of further menu screens and dialog boxes to prompt for installation parameters is the same as described for option 4 (*Install ...*) in the [section "Edit menu"](#).

SUBSYSTEM-MANAGEMENT and USER-ADMINISTRATION privileges are required to use this function.

4.3.3 Show menu

The *Show* menu contains the following options:

1. *Installation units ...*

Outputs information about installation units of the open SCI.

2. *Installation item ...*

Outputs information about installation items of the open SCI.

3. *From formatted file ...*

Outputs information about installation units, installation items or supply units from a formatted file.

4. *Formatted File*

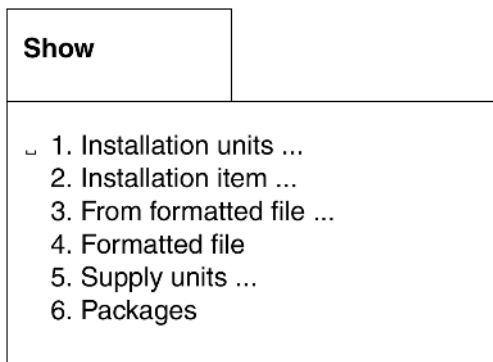
Outputs the contents of a formatted file in legible form to SYSLST.

5. *Supply units ...*

Outputs information about supply units of the open SCI.

6. *Packages*

Outputs information about registered deliveries of the open SCI.



1. Show: Installation units ...

Use option 1 (*Installation units...*) in the *Show* menu to output the information contained in an SCI about previously selected installation units (see "[Installation unit \(IU\)](#) "). You must enter the necessary parameters in the "Show installation units" dialog box.

If you are a nonprivileged user, this function will return only the path names for which DMS assess is possible.

The *Installation units...* option is available when an SCI is open and installation units have been selected in the body.

```

File Edit Show View Options
:-----:
: Show installation units :
:-----:
: Information: Installation items: 1 1. Yes          2. No          :
: Report level.....: 1 1. Minimum          2. All attributes :
:-----:
: Output.....: 1 1. Sysout          2. Syslst          3. Formatted file :
:-----:
: Syslst number: STD          :
: File name.....:          :
: Write mode...: 2 1. Extend          2. Replace          :
:-----:
: F1=Help  F12=Cancel          :
:-----:
ASSEMBH-GEN          01.4          A10
ASSTRAN              05.4          A00
ASTI                 21.0          A00
ATS                  05.3          A20
:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 39: Show installation units of an SCI

Information

The scope of the information to be output.

Installation items

Show/do not show the installation items in an installation unit.

Report level

The reporting level for the installation units.

1 (Minimum)

Default: Show the names of the installation units.

2 (All attributes)

Show the names and attributes of the installation units.

Output

The entry 1, 2 or 3 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Default: Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for “Output = 2”.

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for “Output = 3”.

Write mode

The write mode for the formatted file. This entry is only evaluated for “Output = 3”.

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

Notes

- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user. However, the contents of the formatted file can be output in legible form to SYSOUT and SYSLST using the options 3 (*From formatted File ...*) and 4 (*Formatted file*).
- An asterisk (*) is substituted for each path name that the user is not authorized to read.
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

2. Show: Installation item ...

Use option 2 (*Installation item...*) in the *Show* menu to output information about the installation items entered in the open SCI (see "[Installation item \(II\)](#)").

If you are a nonprivileged user, this function will return only the path names to which DMS access is possible.

The *Installation item* option is available if a SCI is opened and no installation units or supply units are marked in the body.

You can define how the desired installation items are to be specified in more detail in the “Show installation item” dialog box:

```

File Edit Show View Options
-----
: ..... Show installation item : N.SCI
: .....
: 1 1. By item name ... : Units 1 through 13 of 168
: 2. By installation path ... : ection More: +
: : Version Corr state
: F 1=Help F12=Cancel : 21.0 A00
: .....
AIDSYS 21.0 A00
AIDSYSA 21.0 A00
ANITA 21.0 A00
APACHE 02.4 A00
ARCHIVE 12.0 C00
ASE 21.0 A00
ASSEMBH 01.4 A10
ASSEMBH-GEN 01.4 A10
ASSTRAN 05.4 A00
ASTI 21.0 A00
ATS 05.3 A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 40: Show installation items of an SCI

Dialog box options:

1. By item name

If you know the name of the item, use “By item name” to search for the installation item. The dialog box for entering the selection criteria is described on ["Show menu "](#).

2. By installation path

If you know the installation path of the item, use “By installation path” to search for the installation item. The dialog box for entering the selection criteria is described on ["Show menu "](#).

Notes

- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.
- An asterisk (*) is substituted for each path name that the user is not authorized to read.

Show: Installation item (By item name)

If you know the name of an item, you can search for the associated installation item by entering the required parameters in the following dialog box.

```

File Edit Show View Options
.....
: Show installation item: by item name
:-----
: Item name.....:
: Version.....: 1 1. All          2. Highest          3. Other
:                                     Version:
: Unit name.....: 1 1. All          2. Other
: Name.....:
: Version.....: 1 1. All          2. Highest          3. Other
:                                     Version:
: Corr state...: 1 1. All          2. Highest          3. Lowest
:                                     4. Other
:                                     Corr State:
: Report level...: 1 1. Minimum      2. All attributes
: Output.....: 1 1. Sysout          2. Syslst          3. Formatted file
: Syslst number: STD
: File name....:
: Write mode...: 2 1. Extend          2. Replace
:
: F1=Help F12=Cancel
:-----
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 41: Show installation item by item name

Item name

Name of the installation item, see "Installation item (II)".

Version

Version of the installation item.

1 (All)

Select all versions.

2 (Highest)

Select the highest version.

3 (Other)

Specify the version explicitly in the "Version" field.

Version

Version.

Unit name

Name of the installation unit.

1 (All)

Search all installation units entered for the installation item.

2 (Other)

Specify the name, the version and the correction state of the installation unit.

Name

Name of the installation unit to be searched for the installation item, see "[Installation unit \(IU\)](#)".

Version

Version of the installation unit

1 (All)

Search all versions of the installation unit.

2 (Highest)

Search the highest version of the installation unit.

3 (Other)

Specify the version explicitly in the "Version" field.

Version

Version.

Corr state

Correction state of the installation unit.

1 (All)

Search all correction states of the installation unit.

2 (Highest)

Search the highest correction state of the installation unit.

3 (Lowest)

Search the lowest correction state of the installation unit.

4 (Other)

Specify the correction state explicitly in the "Corr State" field.

Corr State

Correction state in the format: <aso>.

Report Level

The reporting level for the installation item attributes.

1 (Minimum)

Show the names of the installation items.

2 (All attributes)

Show the names and attributes of the installation items.

Output

The entry 1, 2 or 3 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Default: Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted File)

Output the information to a formatted file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for "Output = 3".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

Show: Installation item (By installation path)

If you know the path name of an item, you can search for the associated item by entering the required parameters in the following dialog box.

```

File Edit Show View Options
-----
: ..... Show installation item ..... : N.SCI
: -----
: 1 1. By item name ... : Units 1 through 13 of 168
: 2. By installation path ... : ection More: +
: .....
: Show installation item: by installation path :
: -----
: File name.....: :
: :
: Report level...: 1 1. Minimum 2. All attributes :
: :
: Output.....: 1 1. Sysout 2. Syslst 3. Formatted file :
: Syslst number: STD :
: File name....: :
: Write mode...: 2 1. Extend 2. Replace :
: :
: F1=Help F12=Cancel :
: -----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 42: Show installation item by path name

File name

Path name of the installation item.

Report level

The reporting level for the installation item attributes.

1 (Minimum)

Show the names of the installation items.

2 (All attributes)

Show the names and attributes of the installation items.

Output

The entry 1, 2 or 3 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Default: Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for "Output = 3".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

3. Show: From formatted file ...

Use option 3 (*From formatted file...*) in the *Show* menu to obtain information on installation units, installation items or supply units from a formatted file.

The *From formatted file* option is available if no installation units or supply units are marked in the body.

You can enter the file name and specify which objects are to be displayed in the "Show from formatted file" dialog box:

```

File Edit Show View Options
-----
:                                     Show from formatted file :
:-----
: Formatted file:
:
: 1. Installation units ...
: 2. Installation items ...
: 3. Supply units ...
:
: F1=Help  F12=Cancel
:-----
APACHE           02.4           A00
ARCHIVE          12.0           C00
ASE              21.0           A00
ASSEMBH          01.4           A10
ASSEMBH-GEN      01.4           A10
ASSTRAN          05.4           A00
ASTI             21.0           A00
ATS              05.3           A20
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 43: Show installation units, installation items or supply units from a formatted file

Formatted file Name of the formatted file.

Options for selecting objects:

1. Installation units ...

Show all installation units. The dialog box for specifying the scope of the information and the output medium is described on "[Show menu](#)".

2. Installation items ...

Show all installation items. The dialog box for specifying the scope of the information and the output medium is described on "[Show menu](#)".

3. Supply units ...

Show all supply units. The dialog box for specifying the scope of the information and the output medium is described on "[Show menu](#)".

Notes

- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST. An asterisk (*) is substituted for each path name that the user is not authorized to read.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

Show: From formatted file: Installation units

This option returns information about the installation units in a formatted file (see "[Installation unit \(IU\)](#)").

Enter the scope of the information and the output medium in the following dialog box:

```
File Edit Show View Options
:-----:
:                               Show from formatted file                               :
:-----:
:                               Show installation units from formatted file           :
:-----:
: Information: Installation items: 1 1. Yes                                     2. No   :
: Report level.....: 1 1. Minimum                                           2. All attributes :
:-----:
: Output.....: 1 1. Sysout                                                     2. Syslst :
:              3. Formatted file                                             4. Input formatted file :
:-----:
: Syslst number: STD                                                         :
: File name....:                                                            :
: Write mode...: 2 1. Extend                                                 2. Replace :
:-----:
: F1=Help  F12=Cancel                                                         :
:-----:
: ASTI                                           21.0           A00 :
: ATS                                           05.3           A20 :
:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 44: Show installation units from a formatted file

Information

The scope of the information to be output:

Installation items

Show/do not show the installation items. Possible entries: 1 (Yes; this is the default) or 2 (No)

Report level

The reporting level for the installation item attributes.

1 (Minimum)

Default: Show only the names of the installation items.

2 (All attributes)

Show the names and attributes of the installation items.

Output

The entry 1, 2, 3 or 4 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Default: Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

4 (Input formatted file)

Output the information to the formatted file used as the input file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for "Output = 3".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

Show: From formatted file: Installation items

This option returns information about the installation items in a formatted file (see "[Installation item \(II\)](#)").

Enter the scope of the information and the output medium in the following dialog box:

```
File Edit Show View Options
:-----:
:               Show from formatted file           :
:-----:
:               Show installation items from formatted file
:-----:
: Report level: 1 1. Minimum                       2. All attributes
: Output.....: 1 1. Sysout                         2. Syslst
:               3. Formatted file                 4. Input formatted file
:   Syslst number: STD
:   File name....:
:   Write mode...: 2 1. Extend                     2. Replace
:
: F1=Help  F12=Cancel
:-----:
ASSEMBH           01.4           A10
ASSEMBH-GEN       01.4           A10
ASSTRAN           05.4           A00
ASTI              21.0           A00
ATS               05.3           A20
:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 45: Show installation items from a formatted file

Report level

The reporting level for the installation item attributes.

1 (Minimum)

Default: Show only the names of the installation items.

2 (All attributes)

Show the names and attributes of the installation items.

Output

The entry 1, 2, 3 or 4 defines the target for output. The next fields

Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

4 (Input formatted file)

Output the information to the formatted file used as the input file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST

File name

Name of the formatted file. This entry is only evaluated for "Output = 3 or 4".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3 or 4".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

Show: From formatted file: Supply units

This function returns information about the supply units in a formatted file (see "Supply unit ").

You can enter the required parameters in the dialog box that appears.

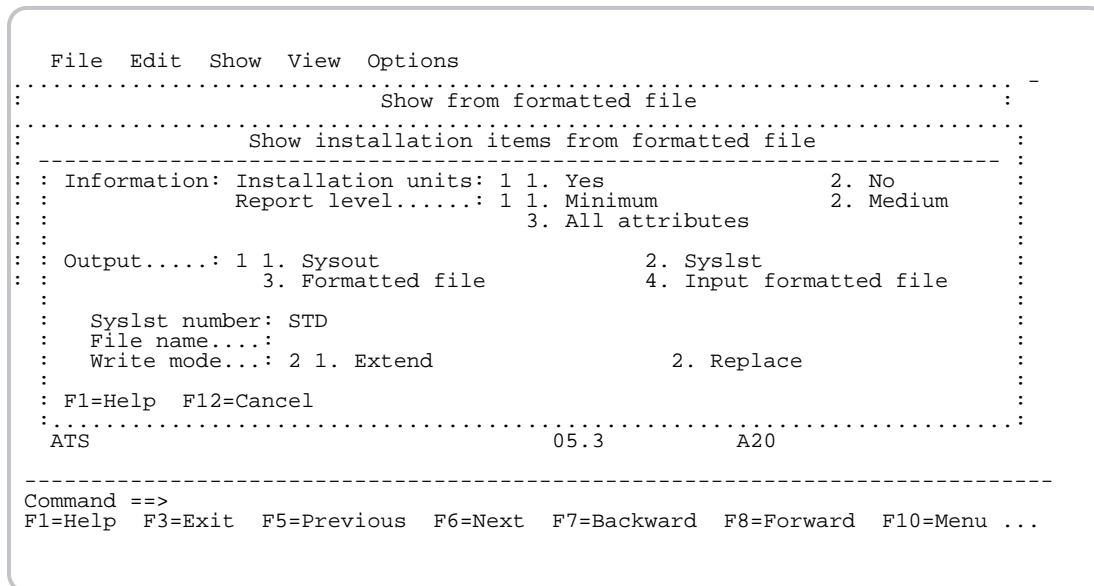


Figure 46: Show supply units from a formatted file

Information

The scope of the information to be output:

Installation units

Show/do not show the installation units. Possible entries: 1 (Yes; this is the default) or 2 (No)

Report level

The reporting level for the installation unit attributes.

1 (Minimum)

Default:

Do not show the attributes of the supply units.

2 (Medium)

No attributes of the supply units, except creation time, are shown. The output is sorted by customer ID.

3 (All attributes)

Show the attributes of the supply units for each correction state.

Output

The entry 1, 2, 3 or 4 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

4 (Input formatted file)

Output the information to the formatted file used as the input file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for "Output = 3 or 4".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3 or 4".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

4. Show: Formatted file

Use option 4 (*Formatted file ...*) in the *Show* menu to output the contents of a formatted file to SYSLST (default) in the form of information that has been edited so as to be legible. Alternatively, the output can be directed to a SYSLST file from the set SYSLST01 through SYSLST99.

```
File Edit Show View Options
:-----:-----:-----:-----:-----:-----:-----:-----:-----:-----:
:                               Show formatted file                               :
:-----:-----:-----:-----:-----:-----:-----:-----:-----:-----:
: Formatted file:                                                            : 68
: Output.....: Syslst number: STD                                          :
:                                                     : nd
: F1=Help   F12=Cancel
:-----:-----:-----:-----:-----:-----:-----:-----:-----:-----:
AIDSYS                21.0          A00
AIDSYSA                21.0          A00
ANITA                 21.0          A00
APACHE                02.4          A00
ARCHIVE               12.0          C00
ASE                   21.0          A00
ASSEMBH               01.4          A10
ASSEMBH-GEN           01.4          A10
ASSTRAN               05.4          A00
ASTI                  21.0          A00
ATS                   05.3          A20
:-----:-----:-----:-----:-----:-----:-----:-----:-----:-----:
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 47: Show formatted file

Formatted file

Name of the formatted file.

Output

The target for output.

Syslst number

An explicitly specified SYSLST file number.

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

Note

If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

5. Show: Supply units ...

Option 5 (*Supply units ...*) in the *Show* menu allows you to output information about the supply units (see "[Supply unit](#) ") of the open SCI, which you selected beforehand using the *Filter* option in the *View* menu.

The SUBSYSTEM-MANAGEMENT privilege is required in order to show information about supply units from the SCI.

You must enter further specifications about the scope of the information and the output medium in the following dialog box:

```
File Edit Show View Options
:-----:
:                Show supply units                :
:-----:
: Information: Installation units: 1 1. Yes          2. No          :
:           Report level.....: 1 1. Minimum        2. Medium         :
:                               3. All attributes      : e
:
: Output.....: 1 1. Sysout                2. Syslst          3. Formatted file :
:
: Syslst number: STD
: File name....:
: Write mode...: 2 1. Extend                2. Replace
:
: F1=Help  F12=Cancel
:-----:

Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 48: Show supply units from a formatted file

Information

The scope of the information to be shown:

Installation units

Show/do not show installation units.

Report level

The reporting level for installation unit attributes.

1 (Minimum)

Default:

Do not show the attributes of the supply units.

2 (Medium)

No attributes of the supply units, except creation time, are shown. The output is sorted by customer ID.

3 (All attributes)

Show the attributes of the supply units for each correction state.

Output

The entry 1, 2 or 3 defines the target for output. The next fields Syslst number, File name and Write mode contain supplementary specifications for the output target.

1 (Sysout)

Default: Output the information to SYSOUT.

2 (Syslst)

Output the information to SYSLST.

3 (Formatted file)

Output the information to a formatted file.

Syslst number

An explicitly specified SYSLST file number. This number is only evaluated for "Output = 2".

Possible entries: STD/<integer 1..99>

The default setting STD outputs the information to SYSLST.

File name

Name of the formatted file. This entry is only evaluated for "Output = 3".

Write mode

The write mode for the formatted file. This entry is only evaluated for "Output = 3".

1 (Extend)

The formatted file is created. If it already exists, then the new information is appended to it.

2 (Replace)

Default: The formatted file is created. If it already exists, then it is overwritten.

6. Show: Packages

Information about registered deliveries of the open SCI are output with the 6 (*Packages*) option in the *Show* menu (see "[Supply unit](#)").

The option is available when the output of deliveries was specified beforehand for display in the work area (*Filter* option in the *Select* menu) and at least one delivery (package) was selected.

SUBSYSTEM-MANAGEMENT privileges are required to obtain information about registered deliveries from the SCI.

Additional specifications on the scope of information to be output and the output medium are requested in the following dialog box:

```

File Edit Show View Options
-----
:                               Show packages                               :
-----
: Information: Supply units.....: 1 1. Yes                               2. No :
:               Report level.....: 1 1. Minimum                       2. All attributes :
: Output.....: 1 1. Sysout                                           2. Syslst :
: Syslst number: STD :
: F1=Help F12=Cancel :
:-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 49: Output deliveries

Information

Specifies the scope of the information to be output:

Supply units

Specifies if deliveries are to be output.

Report level

Controls the output of the attributes of the deliveries.

1 (Minimum)

Default:

The attributes of the deliveries are not output.

2 (All attributes)

The attributes of the deliveries are output.

Output

Output destination. The Syslst number field below it contains additional specifications on the output destination.

1 (Sysout)

Default: The information is output to SYSOUT.

2 (Syslst)

The information is output to SYSLST.

Syslst number

Explicit specification of the SYSLST file number. This specification is only evaluated for “Output = 2”.

Possible entries: `STD/<integer 1..99>`

The default setting STD outputs the information to SYSLST.

4.3.4 View menu

The *View* menu contains the following option:

1. *Filter ...*

Controls whether installation units, supply units or packages are displayed in the body.



1. View: Filter

You decide if installation units, supply units, correction deliveries or registered deliveries (packages) of the currently open SCI are to be displayed in the work area with the option *1 (Filter)* of the *Select* menu.

Enter the desired setting using the “Filter” dialog box:

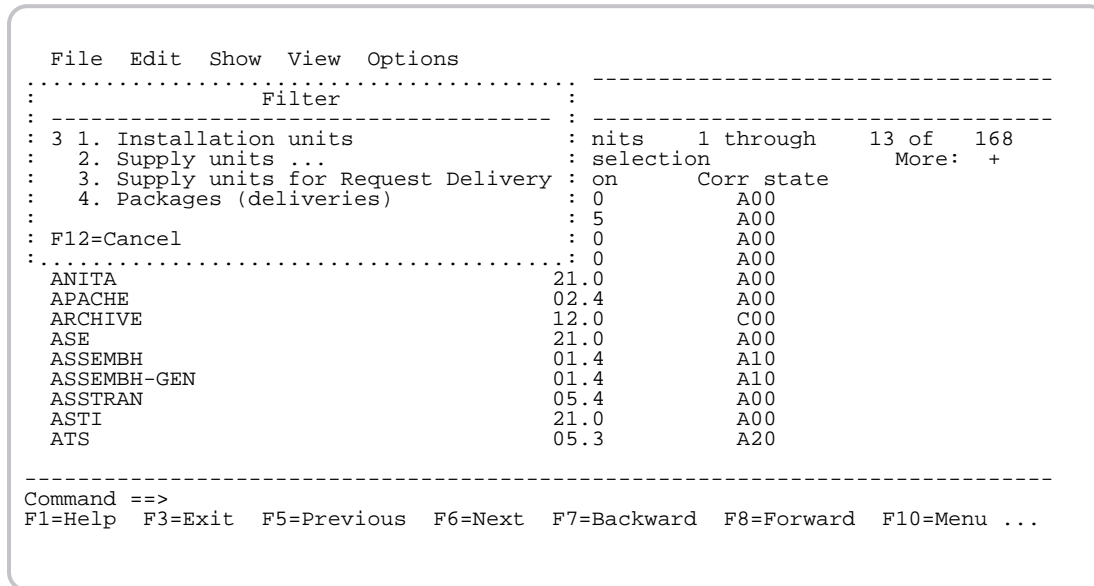


Figure 50: Initial selection of installation units and supply units

Options in the dialog box:

1. Installation units
2. Supply units
3. Supply units for Request Delivery
4. Packages (deliveries)

Selecting the installation units

Default: installation units are to be displayed in the work area. The following screen is output:

```

File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Unit name      Installation units selection  Units 1 through 13 of 168
                  Version          Corr state      More: +
ACS            21.0            A00
AID            03.5            A00
AIDSYS        21.0            A00
AIDSYSA       21.0            A00
ANITA         21.0            A00
APACHE        02.4            A00
ARCHIVE       12.0            C00
ASE           21.0            A00
ASSEMBH       01.4            A10
ASSEMBH-GEN   01.4            A10
ASSTRAN       05.4            A00
ASTI          21.0            A00
ATS           05.3            A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 51: Selection of installation units

Note

Installation units that could not be allocated (no SYSSII file) are indicated by a hash character (#).

Dialog box for selecting the supply units

View supply units in the body. Another dialog box appears in which you can specify supply units with specific package names and user codes and specific characteristics, such as “last installation”, “activable” or a particular installation status that you want to view.

```

File Edit Show View Options
-----
: Filter :
:-----:
: Filter supply units : 68
:-----:
: Package name: 1 1. All 2. Other :
: Name: :
: User code...: 1 1. All 2. Other :
: Name: :
: Customer approved...: 1 1. All 2. Yes 3. No :
: Last installation...: 1 1. All 2. Last :
: Activable.....: 1 1. All 2. Yes 3. No :
: Installation Status: / Installed / Being installed :
: / Parked / In Library :
: / On SOLIS2 Volume / On Local Volume :
: / Being deinstalled / Partially installed :
: / Others :
- : F1=Help F12=Cancel :
C :-----:
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 52: Initial selection of supply units

Package name

Display all supply units or only the supply units of a certain delivery (selection criterion: Package name).

- 1 (All)

Default: Display all supply units.

2 (Other)

Display only the supply units of the delivery with the package name specified in the "Name" field.

Name

The package name.

User code

Display all supply units or only the supply units of a certain delivery (selection criterion: User code).

1 (All)

Default: Display all supply units.

2 (Other)

Display only the supply units from the delivery or the deliveries with the user code specified in the "Name" field.

Name

User code.

Customer-Approved

States whether only the supply units with a customer approval ID are to be displayed.

1 (All)

Default: all supply units are displayed.

2 (Yes)

Only the supply units with a customer approval ID are displayed.

3 (No)

Only the supply units without a customer approval ID are displayed.

Last installation

States whether the last installed supply unit is to be displayed.

1 (All)

Default: all supply units are displayed.

2 (Last)

Only the last installed versions of supply units are displayed.

Activable

States whether the supply units which are activable in the current session should be displayed.

1 (All)

Default: all supply units are displayed.

2 (Yes)

Only supply units that can still be activated are displayed.

3 (No)

Only supply units that cannot be activated are displayed.

Installation Status

Selects the supply units, which are in one of the indicated installation status (OR-gate). On the meaning of the installation status also see "[Installation state of supply units](#)" ([Fetching information from the SCI](#)).

Initially all conditions are marked with “/”, i.e. selection is independent of this selection criterion.

Select supply units from the SCI

When you open an SCI and set the output of supply units using the *Filter* option in the *View* menu, the supply units that are registered in the SCI are displayed in the selection window.

You can now select supply units for further processing from this list.

```
File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Units      1 through      13 of      168
SU selection  More: +
Unit name    Vers Corr Package name  User code  Inst. Status  CAP Act
ACS          21.0 A00 <package-name> SOL2P      Installed    N   Y
AID          03.5 A00 <package-name> SOL2P      Installed    N   Y
AIDSYS      21.0 A00 <package-name> SOL2P      Installed    N   Y
AIDSYSA     21.0 A00 <package-name> SOL2P      Installed    N   Y
ANITA       21.0 A00 <package-name> SOL2P      Installed    N   Y
APACHE      02.4 A00 <package-name> SOL2P      Installed    N   Y
ARCHIVE     12.0 C00 <package-name> SOL2P      Installed    N   Y
ASE         21.0 A00 <package-name> SOL2P      Installed    N   Y
ASSEMBH     01.4 A10 <package-name> SOL2P      Installed    N   Y
ASSEMBH-GEN 01.4 A10 <package-name> SOL2P      Installed    N   Y
ASSTRAN     05.4 A00 <package-name> SOL2P      Installed    N   Y
ASTI        21.0 A00 <package-name> SOL2P      Installed    N   Y
ATS         05.3 A20 <package-name> SOL2P      Installed    N   Y
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 53: Select supply units in the selection window

Select the supply units by entering any character in front of the desired supply unit and confirm your selection by pressing [DUE].

Select correction deliveries (supply units)

All supply units with their most recently installed main version are to be displayed in the work area.

The most recently installed main version is displayed in the work area of each supply unit registered in the SCI after a SCI has been opened and output of *Supply units for Request Delivery* has been set using the *Filter* option of the *View* menu (otherwise as in [figure 53](#)).

Supply units can now be selected in the work area for further processing. The *Request correction delivery* option of the *Edit* menu is now available for the selected supply units (see "Edit menu").

Selecting the deliveries

The deliveries registered in the SCI are to be displayed with their package names in the work area.

In a further dialog box, the display can be limited to deliveries which have been registered in the SCI within a specific time period, or which were produced within a specific time period, or which contain supply units in a specific installation status:

```

File Edit Show View Options
:-----
: Filter : IMON.SCI
:-----
: Filter packages
:-----
: Registration Date (in SCI):
: From: 1 1. Any 2. Date
: (jjjj-mm-tt):
: To: 1 1. Any 2. Date
: (jjjj-mm-tt):
: Generation Date (package name):
: From: 1 1. Any 2. Package
: (JJMMmnnnnn):
: To: 1 1. Any 2. Package
: (JJMMmnnnnn):
:
: Contained SU Status: 1 1. Any 2. All installed
: 3. Partially installed 4. Nothing installed
:
: F1=Help F12=Cancel
-
-- Command
==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 54: Limitation to registered deliveries with specific criteria

Registration Date (in SCI)

Selects deliveries which have been entered in the SCI within the stated time interval.

From

Selects deliveries whose creation date >= the stated date.

1 (Any)

Default: The time interval has no lower limit.

2 (Date)

The date stated in the following field sets a lower limit for the time interval.

To

Selects deliveries whose registration date <= the stated date.

1 (Any)

Default: The time interval has no upper limit.

2 (Date)

The date stated in the following field sets an upper limit for the time interval.

Generation Date (package name)

Selects deliveries which have been produced within the stated interval.

From

Selects deliveries whose generation date \geq the stated date.

1 (Any)

Default: The interval has no lower limit.

2 (Package)

The package name stated in the following field sets a lower limit for the interval.

To

Selects deliveries whose generation date \leq the stated date.

1 (Any)

Default: The interval has no upper limit.

2 (Package)

The package name stated in the following field sets an upper limit for the interval.

Contained SU Status

Selects deliveries depending on the installation status of the supply units they contain.

1 (Any)

Default: The installation status of the contained supply units is not a selection criterion.

2 (All installed)

Only selects deliveries in which all of the supply units are already installed.

3 (Partially installed)

Selects deliveries where only part of all the supply units have already been installed.

4 (Nothing installed)

Only selects deliveries in which none of the supply units have yet been installed.

Select deliveries from the SCI

When you open an SCI and set the output of supply units using the *Filter* option in the *View* menu, the deliveries that are registered in the SCI are displayed in the selection window.

You can now select deliveries for further processing from this list.

```
File Edit Show View Options
-----
IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Package name      User code      Package selection      1 through      3 of      3
Package name      User code      Package selection      More:
x <package-name-1> SOL2p
<package-name-2> SOL2p
<package-name-3> SOL2P
*** End of Package selection ***
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...
```

Figure 55: Selection of registered deliveries

4.3.5 Options menu

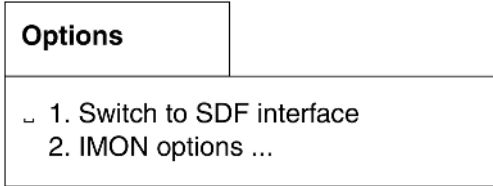
The *Options* menu contains the following options:

1. *Switch to SDF interface*

Switches from the menu mode to the statement mode.

2. *IMON options ...*

Defines the work file ID and the reference file.



1. Options: Switch to SDF interface

Use the option 1 (*Switch to SDF interface*) in the *Options* menu to switch to the statement mode. Once you switch interfaces, you can now enter IMON statements (see also [section "The IMON-BAS statements"](#)). You can use the SWITCH-TO-FHS statement to return to the menu mode.

This option is available when an SCI is open and no other selection has been made.

2. Options: IMON options

Use the *IMON options ...* option in the *Options* menu to define the location for storing the parked software and the work files (*Work file location*).

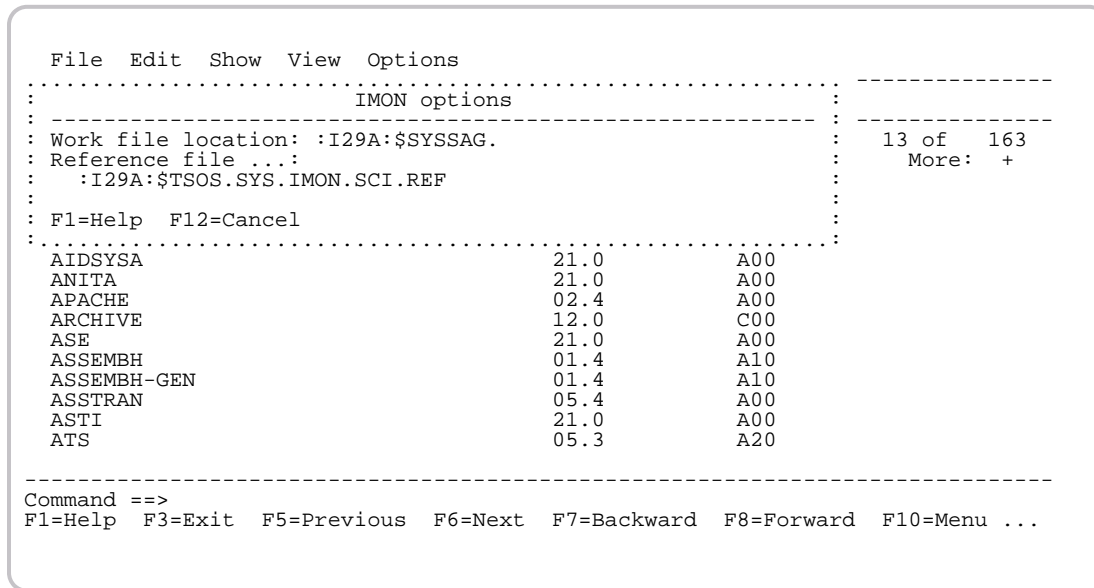


Figure 56: Change the work file location

Work file location

The location for parking either work files or software. The location is specified in the form <partial-filename 2..23> and can contain a catalog ID, a user ID and a file name prefix. To prevent errors when automatically generating file names, it is recommended not to exceed a total length of 18 characters (including catalog ID and user ID) when a prefix is used.

The path name of the current work file ID is displayed by default in this field. When you start IMON using START-IMON, the user ID SYSSAG is generally set on the home pubset. If this ID does not exist, the user ID TSOS and the file name prefix IMON are set on the home pubset. Temporary files cannot be specified here.

Reference file

Selects a reference file which is to be used as the basis for a configuration check during installations. The path name of the current reference file is displayed by default in this field.

When you start IMON using START-IMON, the the reference file \$TSOS.SYS.IMON.SCI.REF is generally set on the home pubset.

4.4 The IMON-BAS statements

In the procedure and batch modes the SDF interface is always started by IMON-BAS by default using the START-IMON command. IMON-BAS also expects statements in the SDF format as additional input.

The statement mode is also available in the interactive dialog.

The SDF interface can be started directly using the operand INPUT-INTERFACE=*SDF in the START-IMON command.

In menu mode, you can use the *Switch to SDF interface* option in the *Options* menu to switch

to the statement mode (see ["Switching between menu and statement mode"](#) (*Working with the menu interface of IMON-BAS*)).

The open SCI remains open.

Overview of the IMON-BAS statements and the standard SDF statements

Statement	Function
ACTIVATE-UNITS	Installed software is dynamically activated
ADD-INSTALLATION-UNITS	Registered installed software in the SCI
CHECK-UNITS	Validate correctness and up-to-dateness of registered software ¹⁾
DEINSTALL-SUPPLY-UNITS	Deinstall any software not needed anymore ¹⁾
END	Close file and end IMON-BAS
GENERATE-IDF	Export SCI entries using IDF
INSTALL-UNITS	Install software and register it in the SCI ¹⁾
MODIFY-IMON-OPTIONS	Select IMON options
PARK-UNITS	Park software ¹⁾
PRINT-DOCUMENTATION	Print delivery documentation ¹⁾
REMOVE-INSTALLATION-UNITS	Remove installation units from the SCI
REMOVE-SUPPLY-UNITS	Remove supply units from the SCI
REQUEST-CORRECTION-DELIVERY	Request correction delivery for registered supply units
RESET-CUSTOMER-APPROVAL	Reset approval timestamp for supply unit
SAVE-SOFTWARE-INVENTORY	Copy current SCI
SET-CUSTOMER-APPROVAL	Set approval timestamp for supply unit

SHOW-FORMATTED-FILE	Show formatted file
SHOW-INSTALLATION-ITEMS	Show installation items
SHOW-INSTALLATION-UNITS	Show installation units
SHOW-PACKAGES	Show deliveries
SHOW-SUPPLY-UNITS	Show supply units
UNDO-SUPPLY-UNIT	Undo the installation ¹⁾
SWITCH-TO-FHS	Switch to menu mode

The functions marked ¹⁾ must be called under the TSOS user ID, because privileged functions are invoked.

You can also use the following standard SDF statements.

Statement	Function
EXECUTE-SYSTEM-CMD	Execute command during program run
HOLD-PROGRAM	Switch to command mode
HELP-MSG-INFORMATION	Output system message text to SYSOUT
MODIFY-SDF-OPTIONS	Modify SDF parameter settings
REMARK	Output comment
RESET-INPUT-DEFAULTS	Reset task-specific default values
RESTORE-SDF-INPUT	Redisplay last input
SHOW-INPUT-DEFAULTS	Display task-specific default values
SHOW-INPUT-HISTORY	Dump input buffer to SYSOUT
SHOW-SDF-OPTIONS	Show SDF parameter settings
SHOW-STMT	Output the syntax description of a statement
STEP	Marks a section used for error handling (only in procedure or ENTER file)

WRITE-TEXT	Write text to SYSOUT/SYSLST
------------	-----------------------------

The standard SDF statements are not described in this manual. See the manual “Dialog Interface” [2 ([Related publications](#))] for details.

The IMON-BAS statements are described in alphabetical order; each description is structured as follows:

- statement name and function
- privileges
- description of function
- presentation of the statement format

A description of the SDF syntax can be found in the Appendix from "[SDF syntax representation](#)".

- description of operands
- notes

4.4.1 ACTIVATE-UNITS Dynamically activating installed software

Privilege: SUBSYSTEM-MANAGEMENT

Function description

With the ACTIVATE-UNITS statement, software can be dynamically activated, i.e. without interruption of system running, after the end of installation (which prepares the activation for the next system run). The selection of the software to be activated is performed either at the level of supply units or installation units.

For the stated supply units or installation units, IMON generates an activation procedure, which contains all the commands needed for activation. The procedure is started automatically. As an alternative, the start can be performed manually. The DSSM commands needed for activation are also listed in a report file and should be checked prior to calling the activation procedure.

Format

ACTIVATE-UNITS

UNIT-NAME = *SUPPLY-UNIT(...) / *INSTALL-UNIT(...)

***SUPPLY-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...**

| <text 1..30 without-sep>(...

| | **VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>**

| | **,CORRECTION-STATE = *HIGHEST-EXISTING / <alphanum-name 3..3>**

***INSTALL-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...**

| <text 1..30 without-sep>(...

| | **VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>**

| | **,CORRECTION-STATE = *HIGHEST-EXISTING / <alphanum-name 3..3>**

,SELECT = *LAST-INSTALLATION / *ALL / list-poss(15): *SOLIS2-DELIVERY(...)

***SOLIS2-DELIVERY(...)**

| **PACKAGE-NAME = <alphanum-name 1..12>**

| **,USER-CODE = <alphanum-name 1..8>**

,START = *IMMEDIATELY / *BY-USER

,FILE-PREFIX = *STD / <partial-filename 2..20 without-cat-user>

,OUTPUT = *SYSOUT / *SYSLST(...)

***SYSLST(...)**

| **SYSLST-NUMBER = *STD / <integer 1..99>**

,KEEP-OLD-VERSION = *NO / *YES

Operands

UNIT-NAME =

Selection of supply units or installation units, which are to be activated.

UNIT-NAME = *SUPPLY-UNIT(...)

Selection of supply units which are to be activated.

UNIT-NAME = *ALL

All supply units, which are entered in the SCI, are to be activated.

UNIT-NAME = *BY-DIALOG

The selection of the desired supply units is performed via dialog masks.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

The names of the desired supply units are explicitly stated. Up to 30 supply units can be stated in the list.

VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>

Specification of the version of the stated supply unit.

VERSION = *HIGHEST-EXISTING

Highest Version of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING / <alphanum-name 3..3>

Specification of the correction state of the stated supply units with the stated version.

CORRECTION-STATE = *HIGHEST-EXISTING

Highest correction state of the supply unit.

UNIT-NAME = *INSTALL-UNIT(…)

Selection of the installation units, which are to be activated.

UNIT-NAME = *ALL

All installation units, which are entered in the SCI, are to be activated.

UNIT-NAME = *BY-DIALOG

The selection of the desired installation units is performed via dialog masks.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

The names of the desired installation units is explicitly stated.

VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>

Specification of the version of the stated installation unit.

VERSION = *HIGHEST-EXISTING

Highest Version of the installation unit.

CORRECTION-STATE = *HIGHEST-EXISTING / <alphanum-name 3..3>

Specification of the correction state of the stated installation units with the stated version.

CORRECTION-STATE = *HIGHEST-EXISTING

Highest correction state of the installation unit.

SELECT =

States whether the number of supply units or installation units stated under UNIT-NAME should be further limited.

SELECT = *LAST-INSTALLATION

Only supply units or installation units from the latest installation are processed.

SELECT = *ALL

All displayed supply units or installation units are processed.

SELECT = list-poss(15): *SOLIS2-DELIVERY(...)

All supply units or installation units, which come from a specific SOLIS2 delivery, are processed. Up to 15 different SOLIS2 deliveries can be stated in a list.

PACKAGE-NAME = <alphanum-name 1..12>

States the package (as stated in the footer of the supply information).

USER-CODE = <alphanum-name 1..8>

States the customer ID (as stated in the footer of the supply information).

START =

Specifies whether the activation procedure should be automatically started or started by the user. The procedure name is shown after the generation of the file or at the automatic start.

The activation procedure is started with the job name ACTIVATE.

START = *IMMEDIATELY

The activation procedure is started immediately, automatically.

START = *BY-USER

The dynamic activation is initiated by the manual starting of the activation procedure

(ENTER-PROCEDURE command).

FILE-PREFIX = *STD / <partial-filename 2..20 without-cat-user>

Specifies the file name prefix of the generated activation procedure, which is saved under the current user ID with the name <prefix>.<time-stamp>.DA (on the format of the time stamp <time-stamp> see "[The troubleshooting process](#)").

The default is *STD, i.e. the character string "IMONACU" is used. If the prefix is explicitly stated, the final stop must be entered with it.

OUTPUT =

The output target for the result log.

OUTPUT = *SYSOUT

Output the result log to SYSOUT.

OUTPUT = *SYSLST(...)

Output the result log to SYSLST.

SYSLST-NUMBER =

Number of the SYSLST file.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

An explicitly specified SYSLST number.

KEEP-OLD-VERSION = *NO / *YES

Specifies (for subsystems that permit removal) whether old versions of the processed subsystems are retained in the DSSM subsystem catalog or are to be removed. The default is *NO. In other words, the old versions are removed from the subsystem catalog.

Note

The statement is not executed and is rejected with an error message in the following circumstances:

- The processed SCI is not a standard SCI, i.e. the location is not :<catid>.\$TSOS.SYS.IMON.SCI.
- One of the selected supply units or installation units is not registered in the currently open SCI.
- One of the selected supply units is in the installation status "Installed".

4.4.2 ADD-INSTALLATION-UNITS Register installed software

Privileges: SUBSYSTEM-MANAGEMENT

Function

Private software and software components already installed in the system can be entered as installation units in the open SCI with the ADD-INSTALLATION-UNITS statement.

The source of input for this function is either an IDF file (Installation Definition File, see "[Installation Definition File \(IDF file\)](#) ") that contains the information on the installation units and the corresponding installation items of the private software or a SYSSII file (structure and installation file, see "[Structure and Installation Information file \(SYSSII file\)](#) ").

If the installation units are registered from a SYSSII file, the entry you make in the SCI can be restricted to a certain target system version (TARGET-SYSTEM-VERSION operand).

This means that you only enter installation items that are specific to the specified target system version.

Format

ADD-INSTALLATION-UNITS

```
FROM-FILE = *SYSDTA / <filename 1..54 without-gen-vers> / *SII(...)  
*SII(...)  
  | FILE-NAME = <filename 1..54 without-gen-vers>  
  | ,INSTALLATION-NAME = *PARAMETERS(...)  
  | *PARAMETERS(...)  
  |   | PUBSET = *DEFAULT / <cat-id 1..4>  
  |   | ,USER-ID = *STD / <alphanum-name 1..8>  
  |   | ,INSTALLATION-PATH = *STD / *BY-DIALOG  
  | ,TARGET-SYSTEM-VERS = *CURRENT / *ANY / 19.0 / 20.0 / 21.0  
,OUTPUT = *SYSOUT / *SYSLST(...)  
*SYSLST(...)  
  | SYSLST-NUMBER = *STD / <integer 1..99>
```

Operands

FROM-FILE =

Name of the input file. The input file is either an IDF file or a SYSSII file.

FROM-FILE = *SYSDTA

This entry can only be made in the procedures generated by IMON.

The IDF file (in internal format; see "[Installation definition file \(IDF file, internal format\)](#)") is read by SYSDTA in an import procedure generated using GENERATE-IDF.

FROM-FILE = <filename 1..54 without-gen-vers>

Name of the IDF file.

FROM-FILE = *SII(...)

Register the software using the specifications of the associated SYSSII file.

FILE-NAME = <filename 1..54 without-gen-vers>

Name of the SYSSII file.

INSTALLATION-NAME = *PARAMETERS(...)

The path name of the installation items.

PUBSET =

Pubset on which all the installation items are to be installed.

PUBSET = *DEFAULT

Install the installation items under the default catalog ID of the corresponding user ID on the home pubset of the active system.

PUBSET = <cat-id 1..4>

Use the specified catalog ID.

USER-ID = *STD / <alphanum-name 1..8>

User ID under which the installation items are to be installed.

Default: *STD, i.e. use the default user ID from the SYSSII file.

INSTALLATION-PATH =

File names of the installation items.

INSTALLATION-PATH = *STD

Register the installation items using the file name from the SYSSII file. No file name is registered for dummy installation items (except for SYSREP).

INSTALLATION-PATH = *BY-DIALOG

The installation items are offered for selection in dialog masks (e.g. select according to PROCESSOR-TYPE). You can modify their file names.

TARGET-SYSTEM-VERS = *CURRENT / *ANY / 19.0 / 20.0 / 21.0

Specifies whether only the installation items related to the specified target system version are to be registered.

Default: *CURRENT, i.e. only items relevant for the current version of the system are registered. With *ANY, all installation items are registered.

OUTPUT =

The output target for the log.

OUTPUT = *SYSOUT

Output the log to SYSOUT.

OUTPUT = *SYSLST(...)

Output the log to SYSLST.

SYSLST-NUMBER = *STD / <integer 1..99>

Output the information to the system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99.

Default: *STD, i.e. Output the information to the system file SYSLST.

Notes

1. The statement is rejected in the following cases and an error message is displayed:
2. The SCI cannot be accessed for write operations.

3.

- The IDF file cannot be accessed.
- The SYSSII file cannot be accessed.
- An error was detected in the IDF file.
- An error was detected in the SYSSII file.
- An installation unit that has to be added already exists in the SCI.

4. In the following case, the statement is canceled when the first defective installation unit is detected and an error message is displayed:

An IDF file generated using the GENERATE-IDF statement is read by SYSDTA and an installation unit that has to be added already exists in the SCI.

5. Each file with a path named in the IDF is checked. The ADD-INSTALLATION-UNITS statement sets the placement status. If the item is found, the attributes are taken from the system.
6. The log is output once the statement has executed. The log is a list of installation items added, plus their associated installation items or the error messages, as applicable.
7. Installation items with a logical name are registered in the IMON-GPN-SCI.
8. Various target system version-dependent installation items of an installation unit have the same logical name, so that with the default value, TARGET-SYSTEM-VERS=*ANY, the last allocation by IMON for the logical name becomes effective. IMON processes the installation items of an installation unit in alphabetical order.
9. The items to be registered are selected automatically on the basis of the specified target system and the item's TARGET ID.

The items are assigned as follows:

- In target system versions V19.0 and higher, items with TARGET= A, S, and K are processed.

4.4.3 CHECK-UNITS Check installed software

Privileges: SUBSYSTEM-MANAGEMENT

Function

This statement allows to check the completeness, the correctness and the up-to-dateness of installed installation- or supply-unit(s) on the system.

This functionality validates the completeness of an installed unit:

- if all the files contained in this unit are still present on the system
- if all the files are still consistent
- if all the files were not modified since their last installation

It validates also the up-to-dateness of the installed unit, i.e. if some items contained in the unit were successfully activated (if the activation was requested during the installation process):

- the activation of message files and syntax files
- the update of DSSM catalog and RMS depot
- the validity of the REP file.

The motivation of the installation check functionality is to detect the problems encountered during installation or after installation:

- installation step(s) ignored by the user during installation process
- files partially or completely destroyed after a device or system crash
- erroneous manipulations of the user in the installation job procedure.

Format

CHECK-UNITS

UNIT-NAME = *SUPPLY-UNIT(...) / *INSTALL-UNIT(...)

***SUPPLY-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / *LAST-INSTALLED** / list-poss(30): <text 1..30 without-sep>(…)

| <text 1..30 without-sep>(…)

| | **VERSION = *HIGHEST-EXISTING / *ALL** / <product-version without-man-corr>

| | **,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL** / <alphanum-name 3..3>

***INSTALL-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / *LAST-INSTALLED** / list-poss(30): <text 1..30 without-sep>(…)

| <text 1..30 without-sep>(…)

| | **VERSION = *HIGHEST-EXISTING / *ALL** / <product-version without-man-corr>

| | **,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL** / <alphanum-name 3..3>

,OUTPUT = *SYSOUT / *SYSLST(...)

***SYSLST(...)**

| **SYSLST-NUMBER = *STD** / <integer 1..99>

Operands

UNIT-NAME =

Specifies the level at which the check is to be performed.

UNIT-NAME = *SUPPLY-UNIT(...)

The check is done at supply unit level.

UNIT-NAME = *ALL

All supply units, registered in the SCI and with “Installed” installation status are to be checked.

UNIT-NAME = *BY-DIALOG

All supply units, registered in the SCI and with “Installed” installation status are proposed for interactive selection.

UNIT-NAME = *LAST-INSTALLED

All supply units from the last installation process are to be checked.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Explicitly specified name of the desired supply units. Up to 30 supply units can be specified in a list.

VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>

Specification of the version of the stated supply unit.

VERSION = *HIGHEST-EXISTING

The highest version must be processed.

VERSION = *ALL

All versions of the selected supply units are to be checked.

CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / <alphanum-name 3..3>

Specification of the correction state of the stated supply units with the stated version.

CORRECTION-STATE = *HIGHEST-EXISTING

The highest correction state must be processed.

CORRECTION-STATE = *ALL

All correction states must be processed.

UNIT-NAME = *INSTALL-UNIT(…)

The check is done at installation unit level.

UNIT-NAME = *ALL

All installation units, registered in the SCI and with “Installed” installation status are to be checked.

UNIT-NAME = *BY-DIALOG

All installation units, registered in the SCI and with “Installed” installation status are proposed for interactive selection.

UNIT-NAME = *LAST-INSTALLED

All installation units from the last installation process are to be checked.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Explicitly specified name of the desired installation units. Up to 30 installation units can be specified in a list.

VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>

Specification of the version of the stated installation unit.

VERSION = *HIGHEST-EXISTING

The highest version must be processed.

VERSION = *ALL

All versions of the selected installation units are to be checked.

CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / <alphanum-name 3..3>

Specification of the correction state of the stated installation units with the stated version.

CORRECTION-STATE = *HIGHEST-EXISTING

The highest correction state must be processed.

CORRECTION-STATE = *ALL

All correction states must be processed.

OUTPUT =

The output target for the log.

OUTPUT = *SYSOUT

Output the log to SYSOUT.

OUTPUT = *SYSLST(...)

Output the log to SYSLST.

SYSLST-NUMBER = *STD / <integer 1..99>

Output the information to the system file SYSLST or to a SYSLST file from the set SYSLST01 through SYSLST99.

Default: *STD, i.e. Output the information to the system file SYSLST.

Note

The statement is rejected and an error message is displayed in following cases:

- The opened SCI does not correspond to a standard one of the current system (name differs from <home-catid>\$TSOS.SYS.IMON.SCI).
- The specified installation unit or supply units do not exist in the currently opened SCI.
- The specified supply units have an installation status different from "INSTALLED".
- The work userid SYSSAG does not exist or some necessary work files for SSCM processing are no more present there.

Example

```
//START-IMON
//CHECK-UNITS UNIT-NAME=*INSTALLATION-UNIT(UNIT-NAME=(FDDRL,SORT,PERCON))
***** IU 'FDDRL' '17.0A00'
File ':MP14:$TSOS.SPMLNK.FDDRL.170' does not exist
File ':MP14:$TSOS.SKMLNK.FDDRL.170' does not exist
File ':MP14:$TSOS.SYSFGM.FDDRL.170.D' does not exist
File ':MP14:$TSOS.SYSFGM.FDDRL.170.E' does not exist
Info : File ':MP14:$TSOS.SYSMES.FDDRL.170' not merged during installation
Info : File ':MP14:$TSOS.SYSSDF.FDDRL.170' not merged during installation
Info : File ':MP14:$TSOS.SYSSSC.FDDRL.170' not merged during installation
Info : File ':MP14:$TSOS.SYSRMS.FDDRL.170' not merged during installation
***** IU 'SORT' '07.9C00'
***** IU 'PERCON' '02.9A10'
Warning : Initial creation attributes '20100322 144227' differ from
current creation attributes '20100322 145709' for file
':P1:$TSOS.SYSSDF.PERCON.029'
Warning : File ":P1:$TSOS.SYSSDF.PERCON.029" modified since its last
installation
Warning : Initial creation attributes '20100322 144228' differ from
current creation attributes '20100322 145710' for file
':P1:$TSOS.SYSLNK.PERCON.029'
Warning : File ":P1:$TSOS.SYSLNK.PERCON.029" modified since its last
installation
//END
```

4.4.4 DEINSTALL-SUPPLY-UNITS Deinstall software

10MAI10617Privileges: SUBSYSTEM-MANAGEMENT

Function

Supply units not needed anymore are removed (deinstalled) with the DEINSTALL-SUPPLY-UNITS statement. The deinstallation is performed for every installation unit of the specified supply unit except for the installation units that are also currently assigned to other installation units and that will remain installed in the system.

During the deinstallation of a supply unit all actions performed during its installation are undone:

- activated files are deactivated, if necessary
- installed files are deleted, if necessary
- entries in the IMON-SCI are removed

The test mode can be set via the operand EXECUTION=*NO . In this case only the preliminary analysis used to check if the deinstallation executed without error is performed.

You will find more details on deinstallation in [section "Deinstallation"](#).

Format

DEINSTALL-SUPPLY-UNITS

UNIT-NAME = *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(…)

<text 1..30 without-sep>(…)

| **VERSION = *ALL / *EXCEPT-HIGHEST-EXISTING / *LOWEST-EXISTING /**

| list-poss(30): <product-version>(…)

| <product-version>(…)

| | **,CORRECTION-STATE = *ALL / *HIGHEST-EXISTING / *LOWEST-EXISTING /**

| | list-poss(15): <alphanum-name 3..3>

,FILE-SAVING = *NO / *WITH-LMS / *WITH-PREFIX(…) / *WITH-ARCHIVE(…)

***WITH-PREFIX(…)**

| **PREFIX = <partial-filename 2..16>**

***WITH-ARCHIVE(…)**

| **VOLUME = <vsn>**

| **,DEVICE-TYPE = <device>**

,TARGET-SYSTEM = *PARAMETERS(…)

***PARAMETERS(…)**

| **VERSION = *CURRENT / <product-version without-man-corr>**

| **,DSSM-CATALOG = *DEFAULT / <filename 1..54 without-gen-vers>**

,EXECUTION = *YES / *NO

,OUTPUT = *SYSOUT / *SYSLST(…)

***SYSLST(…)**

| **SYSLST-NUMBER = *STD / <integer 1..99>**

Operands

UNIT-NAME =

Selection of the supply units to be deinstalled.

UNIT-NAME = *BY-DIALOG

The selection criteria and the corresponding supply units are selected interactively (see "[Interactive selection of supply units](#)" ([INSTALL-UNITS Install and register software](#))).

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Name of the supply unit to be deinstalled. In the VERSION operand the deinstallation can be limited to certain versions of this supply unit.

Up to 30 supply units can be specified in a list.

VERSION =

Specifies which version of the specified supply unit will be deinstalled.

VERSION = *ALL

All versions are deinstalled.

VERSION = *EXCEPT-HIGHEST-EXISTING

All versions are deinstalled except for the highest version.

VERSION = *LOWEST-EXISTING

The lowest version present is deinstalled.

VERSION = list-poss(30): <product-version>(…)

The specified version is deinstalled. The deinstallation can be limited to certain correction states in the CORRECTION-STATE operand.

Up to 30 versions can be specified in a list.

CORRECTION-STATE =

Specifies which correction state of the specified version are deinstalled.

CORRECTION-STATE = *ALL

All correction states of the specified version are deinstalled.

CORRECTION-STATE = *EXCEPT-HIGHEST-EXISTING

All correction states of the specified version are deinstalled except for the highest correction state.

CORRECTION-STATE = *LOWEST-EXISTING

The lowest correction state present is deinstalled.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

The correction state of the supply unit to be deinstalled is specified explicitly.

Format: <aso>

FILE-SAVING = *NO / *WITH-LMS / *WITH-PREFIX(…) / *WITH-ARCHIVE(…)

Specifies if the files deleted during deinstallation are to be saved beforehand.

FILE-SAVING = *NO

The files are not saved.

FILE-SAVING = *WITH-LMS

The files are saved in a library with LMS/LMSCONV. Name of the save library:

\$<work>.IMONDEI.<time-stamp>.SAVE.LIB,

where <work> is the current work file ID setting (see the statement MODIFY-IMON-OPTIONS on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

FILE-SAVING = *WITH-PREFIX(...)

Backup copies of the files are created. The file name of each backup copy is formed from the prefix specified in the PREFIX operand and the original path, where the catalog ID and the dollar symbol are omitted from the left side of the user ID.

PREFIX = <partial-filename 2..16>

File name prefix for the backup copy. The file name prefix may not be longer than 16 characters after adding the catalog ID and user ID of the caller.

Example

During deinstallation of the EDT (caller TSOS, prefix "D1.") the file
:B503:\$OSD4.SYSLNK.EDT.170 is deleted, among others. The backup copy is
saved under the name :B503:\$TSOS.D1.TSOS.SYSLNK.EDT.170.

FILE-SAVING = *WITH-ARCHIVE(...)

The files are saved with ARCHIVE.

VOLUME = <vsn>

VSN of the data medium on which the files are to be saved using ARCHIVE.

DEVICE-TYPE = <device>

Device type of the data medium.

TARGET-SYSTEM = *PARAMETERS(...)

Specifications on the target system for which the deinstallation is to be performed.

VERSION =

BS2000 version of the target system.

VERSION = *CURRENT

The current BS2000 version is used.

VERSION = <product-version without-man-corr>

Explicit specification of the BS2000 version.

Permitted entries: V19.0, V20.0 and V21.0

DSSM-CATALOG = *DEFAULT / <filename 1..54 without-gen-vers>

Name of the static DSSM catalogs from which the subsystems to be deinstalled are to be removed.

EXECUTION =

Specifies if the actual deinstallation is to be executed after performing the preliminary analysis. During the preliminary analysis a check is performed to see if the most important requirements to perform the deinstallation are fulfilled:

-
1. Every file to be deleted must be accessible.
 2. Every SDF parameter file from which the entries for syntax files are to be removed must be accessible.
 3. Every MIP parameter file from which the entries for message files are to be removed must be accessible.
 4. Every static DSSM catalog and every SSCM source file from which the entries for subsystems are to be removed must be accessible.
 5. Every subsystem to be deleted must be stopped.

EXECUTION = *YES

The deinstallation is executed after a successful preliminary analysis. If the preliminary analysis determines that a requirement is not fulfilled, then an error is logged and error

handling is initiated:

- In the procedure or batch mode the processing mode is changed to EXECUTION=*NO, i.e. the preliminary analysis is continued and all errors detected are recorded. The actual deinstallation is not started, however.
- In the interactive dialog processing is interrupted with a message requiring a response. Depending on the response of the user, processing is simply continued (the error is ignored) processing is continued with EXECUTION=*NO or the current step of the check is repeated. When an error is ignored IMON assumes during processing after ignoring the error that no error was detected during this step of the check. Repeating this step of the check only makes sense when the cause of the error was eliminated in the meantime.

EXECUTION = *NO

The call is made in the test mode, i.e. only the preliminary analysis for the deinstallation is performed and any errors detected during analysis are recorded.

OUTPUT = *SYSLST(...)

Error messages are output to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Error messages are output to the standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specification of the SYSLST number.

Example

In the following example a check is made to see if the supply unit PERCON could be deinstalled without error.

```
//deinstall-supply-units unit-name=*by-dialog,execution=*no _____ (1)
```

```
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Units      1 through      5 of      5
More:
Unit name  Vers  Corr  SU selection  Park userid  Install status
EDT        17.0  D30   :I29A:$PARKSW1  Installed
OPENFT     12.1  C10   :I29A:$PARKSW1  Installed
OPENFT-CR  12.1  B10   :I29A:$PARKSW1  Installed
X PERCON   02.9  A10   :I29A:$PARKSW1  Installed
SORT       08.0  A30   :I29A:$PARKSW1  Installed
*** End of SU selection ***
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

```
===== Deinstallation executed with EXECUTION=*NO: ===== (3)
Subsystem "PERCON" was removed from the SDF parameter file
":I29A:$TSOS.SYSPAR.SDF"
Message file ':I29A:$OSD4.SYSMES.PERCON.029' removed from the message system
Message file ':I29A:$OSD4.SYSMES.PERCON.029' removed from the MIP parameter
file ':I29A:$TSOS.SYSPAR.MIP.170' removed if present
Subsystem "PERCON" "V02.9" removed from catalog ":I29A:$TSOS.SYS.SSD.CAT.X"
File ':I29A:$TSOS.SYSFGM.PERCON.029.D' deleted
File ':I29A:$TSOS.SYSFGM.PERCON.029.E' deleted
File ':I29A:$TSOS.SYSLNK.PERCON.029' deleted
File ':I29A:$TSOS.SYSMES.PERCON.029' deleted
File ':I29A:$TSOS.PERCON' deleted
File ':I29A:$TSOS.SYSRMS.PERCON.029' deleted
File ':I29A:$TSOS.SYSSDF.PERCON.029.TSOS' deleted
File ':I29A:$TSOS.SYSSDF.PERCON.029' deleted
File ':I29A:$TSOS.SYSSSC.PERCON.029' deleted
File ':I29A:$TSOS.SYSSSC.PERCON.029.LOW' deleted
LE 'PERCON' Version "02.9C00" (package-name) removed
IU 'PERCON' Version "02.9C00" removed
```

- (1) The statement DEINSTALL-SUPPLY-UNITS is called in the test mode (EXECUTION=*NO). The supply units are to be selected in the dialog (UNIT=*BY-DIALOG).
- (2) The supply units are displayed in a dialog box. In the example the supply unit PERCON was selected. The selection is then confirmed with [DUE].
- (3) The messages in the deinstallation steps to be performed are output to SYSOUT. In this example no errors that could have arisen while deinstalling PERCON were detected.

Notes

1. In the following cases the statement was rejected without taking any action and an error message was output:

-
- No standard SCI (\$TSOS.SYS.IMON.SCI) opened.
 - One of the selected supply units is not registered in the currently open SCI.
 - One of the selected supply units is registered in the currently open SCI but is not in the “Installed” or “Parked” state.
 - VERSION=*EXCEPT-HIGHEST-EXISTING or *LOWEST-EXISTING was specified and only one version is registered in the SCI.
 - The preliminary analysis was not executed without errors in the procedure or batch mode.
2. The package name is ignored when selecting the supply units. If a supply unit is in several SOLIS2 deliveries, then all supply units are deinstalled.
 3. Installation units are only deinstalled when they are not assigned to any of the supply units left in the system.
 4. Files are only deactivated and deleted when they are not assigned to any of the supply units left in the system.
 5. The access rights to files are checked by IMON using the file reservation with the SECURE-RESOURCE-ALLOCATION command.
 6. Updating the DSSM catalog:
If the dynamic DSSM catalog cannot be updated, then a warning is output.
The static DSSM catalog update is performed for all subsystems to be deleted in a single step. If the catalog cannot be saved after that, then a warning is output. If the user requests the save to be repeated, then this can be done using the parameter ENFORCE=YES .
 7. “Being Deinstalled” installation state:
If the actual deinstallation for a supply unit is started, IMON sets the installation state to “Being Deinstalled” until the supply unit is removed from the SCI at the end of deinstallation.
This installation state is shown in the output of the SHOW-SUPPLY-UNITS command. It prevents SCI entries of supply units that are now being deinstalled from being exported at the same time using the GENERATE-IDF statement.
 8. Restoring an aborted deinstallation:
If the deinstallation process is aborted by the user or by IMON itself, then a restart can be performed with continuation of the deinstallation simply by calling the deinstallation function again with the same selection of supply units.

4.4.5 GENERATE-IDF Export SCI entries using IDF

Privileges: SUBSYSTEM-MANAGEMENT

Function

The GENERATE-IDF statement allows you to copy (export) SCI entries from selected supply units or installation units to a different SCI. IMON generates an import procedure in which the desired target SCI is opened and a generated IDF file, which describes the structure of the selected supply components or installation units, (see "[Installation definition file \(IDF file, internal format\)](#)"), is read in by SYSDTA using the ADD-INSTALLATION-UNITS statement.

Note

When selecting installation units (UNIT-NAME=*INSTALL-UNIT(...)) the IDF file generated does not contain information about the assignments to supply units. Since this information may also be missing when registering later on in another SCI, the installation units exported in this manner cannot be deinstalled with the DEINSTALL-SUPPLY-UNITS statement.

In addition, the installation path (in its entirety or only the catalog ID, the user ID or the file name prefix) can be changed during the export operation. In this case, it is also possible to generate a copy procedure containing all the COPY-FILE commands for the relevant installation items and the import procedure call. When you call the copy procedure, the installation items are first transferred to the new location (placement) and are then registered in the specified SCI.

In the case of supply units whose path names belong to non-accessible files, you must enter the statement under TSOS. If you use different user IDs, the statement is not actually rejected, but the path names of the non-accessible files are replaced by an asterisk (*) in the generated IDF file and in the copy procedure. The ADD-INSTALLATION-UNITS statement is then rejected during the import procedure because of the defective IDF file. Likewise, the corresponding commands are also rejected as being defective during the copy procedure.

Format

GENERATE-IDF

UNIT-NAME = *SUPPLY-UNIT(...) / *INSTALLATION-UNIT(...)

***SUPPLY-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...**
| **<text 1..30 without-sep>(...**
| | **VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>**
| | **,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING /**
| | **list-poss(15): <alphanum-name 3..3>**

***INSTALLATION-UNIT(...)**

| **UNIT-NAME = *ALL / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...**
| **<text 1..30 without-sep>(...**
| | **VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>**
| | **,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING /**
| | **list-poss(15): <alphanum-name 3..3>**

,RENAMING = *NONE / *COMPLETE-LOCATION(...) / *COMPOSED-LOCATION(...)

***COMPLETE-LOCATION(...)**

| **NAME = list-poss(15): *PARAMETERS(...)**
| ***PARAMETERS(...)**
| | **OLD-NAME = <partial-filename 2..23>**
| | **,NEW-NAME = <partial-filename 2..23> / *PROMPT**

***COMPOSED-LOCATION(...)**

| **CATID = *UNCHANGED / list-poss(15): *PARAMETERS(...)**
| ***PARAMETERS(...)**
| | **OLD-NAME = <cat-id 1..4>**
| | **,NEW-NAME = <cat-id 1..4>**
| **,USERID = *UNCHANGED / list-poss(15): *PARAMETERS(...)**
| ***PARAMETERS(...)**
| | **OLD-NAME = <name 1..8>**

```

|      | ,NEW-NAME = <name 1..8>
| ,PREFIX = *UNCHANGED / list-poss(15): *PARAMETERS(...)
| *PARAMETERS(...)
|      | OLD-NAME = *NONE / <partial-filename 2..7 without-cat-user> / *ALL
|      | ,NEW-NAME = *NONE / <partial-filename 2..7 without-cat-user>
, COPY-PROC-GENERATION = *NO / *YES(...)
  *YES(...)
    | SAVE-FILE-LOCATION = *NONE / <partial-filename 2..23>
, SCI-NAME = *STD / <filename 1..50 without-gen-vers>
, GENERATE-FILE-PREFIX = <filename 1..49 without-gen-vers>
, IDF-FORMAT = *CURRENT / *V1 / *V2
, OUTPUT = *SYSOT / *SYSLST(...)
  *SYSLST(...)
    | SYSLST-NUMBER = *STD / <integer 1..99>

```

Operands

UNIT-NAME =

Selection of the installation units or supply units to be added to the import procedure.

UNIT-NAME = *SUPPLY-UNIT(...)

Select the installation units to be included in the import procedure.

UNIT-NAME = *ALL

Include all installation units from the SCI in the import procedure.

UNIT-NAME = *BY-DIALOG

Select the desired installation units using dialog boxes.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Explicitly specified name of the desired installation unit. Up to 30 installation units can be specified in a list.

VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>

Specification of the version of the specified installation unit.

VERSION = *HIGHEST-EXISTING

Highest version of the installation unit.

VERSION = *ALL

All versions of the installation unit.

**CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING / list-poss(15):
<alphanum-name 3..3>**

Specification of the correction state of the specified installation unit with the specified version.

CORRECTION-STATE = *HIGHEST-EXISTING

Highest correction state of the installation unit.

CORRECTION-STATE = *ALL

All correction states of the installation unit.

CORRECTION-STATE = *LOWEST-EXISTING

Lowest correction state of the installation unit.

UNIT-NAME = *INSTALLATION-UNIT(...)

Select the supply units to be included in the import procedure.

UNIT-NAME = *ALL

Include all the supply units from the SCI in the import procedure.

UNIT-NAME = *BY-DIALOG

Select the desired supply units using dialog boxes.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(...)

Explicitly specified name of the desired supply unit. Up to 30 supply units can be specified in a list.

VERSION =

Specification of the version of the specified supply unit.

VERSION = *HIGHEST-EXISTING

Highest version of the supply unit.

VERSION = *ALL

All versions of the supply unit.

**CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING / list-poss(15):
<alphanum-name 3..3>**

Specification of the correction state of the specified supply unit with the specified version.

CORRECTION-STATE = *HIGHEST-EXISTING

Highest correction state of the supply unit.

CORRECTION-STATE = *ALL

All correction states of the supply unit.

CORRECTION-STATE = *LOWEST-EXISTING

Lowest correction state of the supply unit.

RENAMING =

Rename/do not rename the saving locations (and thus the path names) of the associated installation items.

RENAMING = *NONE

Do not rename the saving locations.

RENAMING = *COMPLETE-LOCATION(...)

Completely rename the saving locations. New saving locations can be allocated to old saving locations in the NAME operand. The old and new saving locations are specified using a catalog ID, a user ID and a prefix, if necessary.

NAME = list-poss(15): *PARAMETERS(...)

The allocation of the old saving location to the new saving location. The complete path name of the saving location [:<cat-id>:]<userid>[prefix] is always specified.

Up to 15 allocations can be specified in a list.

OLD-NAME = <partial-filename 2..23 without-cat-user>

Old saving location.

NEW-NAME = <partial-filename 2..23 without-cat-user>

New saving location.

NEW-NAME = *PROMPT

You are prompted to enter the new saving location when the generated procedure is being executed.

RENAMING = *COMPOSED-LOCATION(...)

Rename individual parts of the saving locations: catalog ID and/or user ID and/or prefix.

The desired allocations can be specified in the CATID, USERID and PREFIX operands.

CATID =

Change/do not change the catalog ID.

CATID = *UNCHANGED

Do not change the catalog ID.

CATID = list-poss(15): *PARAMETERS(...)

The allocation of the old catalog ID to the new catalog ID. Up to 15 allocations can be specified in a list.

OLD-NAME = <cat-id 1..4>

Old catalog ID.

NEW-NAME = <cat-id 1..4>

New catalog ID.

USERID =

Change/do not change the user ID.

USERID = *UNCHANGED

Do not change the user ID.

USERID = list-poss(15): *PARAMETERS(...)

The allocation of the old user ID to the new user ID. Up to 15 allocations can be specified in a list.

OLD-NAME = <name 1..8>

Old user ID.

NEW-NAME = <name 1..8>

New user ID.

PREFIX =

Change/do not change the prefix.

PREFIX = *UNCHANGED

Do not change the prefix.

PREFIX = list-poss(15): *PARAMETERS(...)

The allocation of the old prefix to the new prefix. Up to 15 allocations can be specified in a list.

OLD-NAME = *NONE / <partial-filename 2..7 without-cat-user> / *ALL

Old prefix. Default is *NONE, i.e. all installation items without a prefix are given the prefix specified in the NEW-NAME operand.

Specify *ALL causes all installation items to be assigned the prefix specified in the NEW-NAME operand regardless of their prefix.

NEW-NAME = *NONE / <partial-filename 2..7 without-cat-user>

New prefix. Default is *NONE, i.e. the installation items do not have a prefix after renaming.

COPY-PROC-GENERATION =

Generate/do not generate a copy procedure. The copy procedure contains COPY-FILE commands for all the selected installation items. The RENAMING operand defines the source and target location of the copy action.

COPY-PROC-GENERATION = *NO

Do not generate a copy procedure.

COPY-PROC-GENERATION = *YES(...)

Generate a copy procedure.

SAVE-FILE-LOCATION = *NONE / <partial-filename 2..23>

Determines whether backups of the installation items that already exist at the target saving location are made before copying. The backups are created at the saving location specified here. Default: *NONE, i.e. do not make a backup.

SCI-NAME = *STD / <filename 1..50 without-gen-vers>

Specifies the SCI to which the selected supply units or installation units are to be added. Default is *STD, i.e. the entries are imported into the standard SCI.

GENERATE-FILE-PREFIX = <filename 1..49 without-gen-vers>

The names of the generated procedures. The import procedure is assigned the suffix .GEN, while the copy procedure is given the suffix .COPY.

IDF-FORMAT =

Specifies which format version of IDF has to be generated, i.e. for which version of IMON this IDF is intended.

IDF-FORMAT = *CURRENT

The generated IDF is intended to be used with IMON-BAS from the current version and higher.

IDF-FORMAT = *V1

The generated IDF is intended to be used with an IMON-BAS version lower than V3.0.

IDF-FORMAT = *V2

The generated IDF is intended to be used with an IMON-BAS version from V3.0 and higher.

OUTPUT =

The output target for the result log.

OUTPUT = *SYSOUT

Output the result log to SYSOUT.

OUTPUT = *SYSLST(...)

Output the result log to SYSLST.

SYSLST-NUMBER =

Number of the SYSLST file.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

An explicitly specified SYSLST number.

4.4.6 INSTALL-UNITS Install and register software

Privileges: SUBSYSTEM-MANAGEMENT,
USER-ADMINISTRATION

Function

The INSTALL-UNITS statement enables you to install the supply units of a SOLIS2 delivery and update the open SCI with the new information. When you open a new delivery (i.e. one not yet registered in the SCI), the delivery is registered in the SCI. For details of the installation procedure, see [section "Installation"](#).

The installation can be executed for START=*BY-USER, up to and including the generation of the installation procedure, under any user ID with both privileges. The current user ID must be specified as the WORK-FILE-LOCATION using MODIFY-IMON-OPTIONS. Following this, the installation procedure must then be started under TSOS.

Exception: If you are installing from a data volume, or if you are using a user ID other than the current user ID for the installation, the entire installation must be performed under TSOS.

If you are installing the delivery from data volume (original or copy created by the customer) or from a library (DISTRIBUTION-MEDIUM= *SOLIS2-VOLUME/ *LOCAL-VOLUME/ *LIBRARY), the delivery documentation is created and stored in a library.

The software, which has been installed, is locked for users of the system until the next system start (also see LOCK-PRODUCT-VERSION command in the „Commands“ [\[4\]](#) manual).

If, at installation, the installation item of an “approved” supply unit is changed, IMON automatically resets the customer approval ID of the supply unit (see also RESET-CUSTOMER-APPROVAL statement ([RESET-CUSTOMER-APPROVAL](#) [Reset customer approval ID of a supply unit](#))).

Format

INSTALL-UNITS

UNIT-NAME = *FROM-SOLIS2-DELIVERY(...) / *FROM-SCI(...) / *CUSTOMER-APPROVED(...)

***FROM-SOLIS2-DELIVERY(...)**

| **PACKAGE-NAME = <alphanum-name 1..12>**

| **,USER-CODE = <alphanum-name 1..8>**

| **,DISTRIBUTION-MEDIUM = *REGISTERED-MEDIUM / *LIBRARY(...) / *SOLIS2-VOLUME(...) / *LOCAL-VOLUME(...)**

| ***LIBRARY**(...)

| | **DOCUMENT-LIBRARY = *STD / <filename 1..54 without-gen-vers>**

| | **,REGISTRATION = *EXTEND / *REPLACE**

| ***SOLIS2-VOLUME**(...)

| | **VOLUME = <vsn>**

| | **,DEVICE-TYPE = <device>**

| | **,DOCUMENT-LIBRARY = *STD / <filename 1..54 without-gen-vers>**

| | **,REGISTRATION = *EXTEND / *REPLACE**

| ***LOCAL-VOLUME**(...)

| | **VOLUME = <vsn>**

| | **,DEVICE-TYPE = <device>**

| | **,DOCUMENT-LIBRARY = *STD / <filename 1..54 without-gen-vers>**

| | **,REGISTRATION = *EXTEND / *REPLACE**

| **,SUPPLY-UNITS = *ALL-REMAINING(...) / *COMPLETE-DELIVERY(...) / *BY-DIALOG / *BY-CONFIGURURATION(...)/ list-poss(30): <text 1..30 without-sep>(...)**

| ***ALL-REMAINING**(...)

| | **REPLACE-OLD-FILES = *YES / *MINIMUM / *NO**

| | **,FORCE-LOCATION = *NO / *MINIMUM / *YES**

| ***COMPLETE-DELIVERY**(...)

| | **REPLACE-OLD-FILES = *YES / *MINIMUM / *NO**

| | **,FORCE-LOCATION = *NO / *MINIMUM / *YES**

```

|
|
|      REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|
|      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
|
|      <text 1..30 without-sep>(…)
|
|      VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>
|
|      ,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING / list-
|      poss(15): <alphanum-name 3..3>
|
|      ,REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|
|      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
|
|*FROM-SCI(…)
|
| PACKAGE-NAME = <alphanum-name 1..12> / *BY-SUPPLY-UNITS
|
| ,USER-CODE = <alphanum-name 1..8> / *BY-SUPPLY-UNITS
|
| ,SUPPLY-UNITS = *ALL(…) / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(…)
|
|   *ALL(…)
|
|      REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|
|      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
|
|      <text 1..30 without-sep>(…)
|
|      VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>
|
|      ,CORRECTION-STATE = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING / list-
|      poss(15): <alphanum-name 3..3>
|
|      ,REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|
|      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
|
|*CUSTOMER-APPROVED(…)
|
| PACKAGE-NAME = <alphanum-name 1..12> / *BY-SUPPLY-UNITS
|
| ,USER-CODE = <alphanum-name 1..8> / *BY-SUPPLY-UNITS
|
| ,SUPPLY-UNITS = *ALL(…) / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(…)
|
|   *ALL(…)
|
|      REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|
|      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
|
|      <text 1..30 without-sep>(…)

```

```

|      |      VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>
|      |      ,REPLACE-OLD-FILES = *YES / *MINIMUM / *NO
|      |      ,FORCE-LOCATION = *NO / *MINIMUM / *YES
,TARGET-SYSTEM = *PARAMETERS(...)
  *PARAMETERS(...)
|      |      VERSION = *CURRENT / <product-version without-man-corr>
|      |      ,PUBSET = *DEFAULT / *PARAMETERS(...)
|      |      *PARAMETERS(...)
|      |      CATID = <cat-id 1..4>
|      |      ,DEFLUID = *STD / <name 1..8>
|      |      ,PROCESSOR-TYPE = *STD / *ANY / *CISC-390 / *X86
|      |      ,ITEM-SELECTION = *STD / *ALL / *TARGET-VERSION-ONLY
,OLD-FILE-SAVING = *STD / *NO / *WITH-LMS / *WITH-ARCHIVE(...)
  *WITH-ARCHIVE(...)
|      |      VOLUME = <vsn> / *FROM-OPERATOR(...)
|      |      *FROM-OPERATOR(...)
|      |      LOCATION = *STD / <alphanum-name 1..8>
|      |      ,DEVICE-TYPE = <device>
,PASSWORD-FILE = *NO / <filename 1..54 without-gen-vers>(...
  <filename 1..54 without-gen-vers>(...
|      |      READ-PASSWORD = <c-string 1..4>
,LOG-PRINTING = *STD / *NO / *YES
,PLACEMENT-MODE = *STD / *BY-DIALOG
,ACTIVATION-MODE = *STD / *PARAMETERS(...)
  *PARAMETERS(...)
|      |      SYNTAX-FILE-PROCESS = *BY-DIALOG / *NO / *YES
|      |      ,MESSAGE-FILE-PROCESS = *BY-DIALOG / *NO / *YES
|      |      ,DSSM-PROCESSING = *BY-DIALOG / *NO / *YES
|      |      ,REP-PROCESSING = *BY-DIALOG / *NO / *YES
|      |      ,POSIX-PROCESSING = *BY-DIALOG / *BY-CONFIGURATION-FILE / *NO / *ALL

```

```

,START = *IMMEDIATELY / *BY-USER
,WORK-FILE-DELETING = *STD / *YES / *NO
,UNDO-PREPARATION = *STD / *NO / *PARAMETERS(...)
  *PARAMETERS(...)
  | RETENTION-PERIOD = 30 / <integer 0..999 days>
  | ,SAVE-MEDIUM = *ARCHIVE-DIRECTORY(...) / *TAPE(...) / *LIBRARY
  |   *ARCHIVE-DIRECTORY(...)
  |     | NAME = <filename 1..54 without-gen-vers>
  |     | ,DEVICE-TYPE = <device>
  |     *TAPE(...)
  |     | VOLUME = <vsn>
  |     | ,DEVICE-TYPE = <device>
,CHECK-CONFIGURATION = *STD / *NO / *YES
,PARAMETER-FILE = *STD / *NO / <filename 1..54 without-gen-vers>

```

Operands

UNIT-NAME =

Source for supply units to be installed.

UNIT-NAME = *FROM-SOLIS2-DELIVERY(...)

Install supply units from a SOLIS2 delivery.

PACKAGE-NAME = <alphanum-name 1..12>

Package name (as stated in the footer of the delivery contents).

USER-CODE = <alphanum-name 1..8>

User code (as stated in the footer of the delivery contents).

DISTRIBUTION-MEDIUM =

The distribution medium: either IMON uses the information from the SCI or the distribution medium is specified explicitly.

DISTRIBUTION-MEDIUM = *REGISTERED-MEDIUM

The delivery information, the VSN and the device type of the tape or the name of the PLAM library are taken from the open SCI. Files that have already been installed are not read again from the MTC.

It is now possible to install the same supply unit for the same system multiple times using the *REGISTERED-MEDIUM functionality (e.g. after deleting a file of the supply unit by mistake). Moreover, the "Installed" status of a supply unit will now be set only in the SCI of the system where the supply unit is effectively installed and no longer in the SCI from where the installation is triggered (where it remains unchanged).

You can select the following for the installation:

- parked software
- software that has not (yet) been installed or partially installed software
- installed software with files that were not deleted by the park ID (multiple installation)
- software for which the delivery information was already printed.

DISTRIBUTION-MEDIUM = *LIBRARY(...)

The delivery is stored in a PLAM library called SOLFTR.<package-name>.<user-code> on the user ID SYSSAG.

<package-name> and <user-code> are the values of the PACKAGE-NAME and USER- CODE operands.

The delivery information is read from the library. The selected supply units are installed from the library and registered in the SCI.

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, then the new delivery documentation is added to its contents.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC, where <package name> is the value of the operand PACKAGE-NAME, <customer ID> is the value of the operand USER-CODE and <work file ID> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC, where <package name> is the value of the operand PACKAGE-NAME, <customer ID> is the value of the operand USER-CODE and <work file ID> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

An explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI

or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information from the library supplied.

Note

All information on previously installed supply units from this delivery will be lost.

These supply units are no longer available for an installation with UNIT-NAME=*FROM-SCI.

DISTRIBUTION-MEDIUM = *SOLIS2-VOLUME(...)

Read delivery information from a data volume shipped with Solis2 (tape cartridge, CD or DVD). The delivery information will be copied from data volume. All supply units of the specified delivery are available for selection. The selected supply units will be copied from data volume to the current user ID and registered in the SCI. Existing files with the same names will be overwritten.

VOLUME = <vsn>

VSN of the data volume which contains the delivery (see delivery contents).

DEVICE-TYPE = <device>

Device type of the data volume (see delivery contents).

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC, where <package name> is the value of the operand PACKAGE-NAME, <customer ID> is the value of the operand USER-CODE and <work file ID> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

An explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information read from the data medium.

Note

All information on previously installed supply units from this delivery will be lost.

These supply units are no longer available for an installation with UNIT-NAME=*FROM-SCI.

DISTRIBUTION-MEDIUM = *LOCAL-VOLUME(...)

Read delivery information from a data volume created by the customer tape cartridge or virtual volume in a CentricStor). The data volume information of the product movement file is ignored. All supply units of the specified delivery are available for selection. The selected supply units will be copied from data volume to the current user ID and registered in the SCI. Existing files with the same names will be overwritten.

i With LOCAL-VOLUME there is no support of continuation volume. So the maximal size of a copiable SOLIS2 delivery is limited by the capacity of the used LOCAL-VOLUME. The maximal necessary size for the delivery may be found on page 4 in the delivery documentation.

Refer to "[Creating a LOCAL-VOLUME](#)".

VOLUME = <vsn>

VSN of the data volume which contains the delivery.

DEVICE-TYPE = <device>

Device type of the data volume.

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC, where <package name> is the value of the operand PACKAGE-NAME, <customer ID> is the value of the operand USER-CODE and <work file ID> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

An explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information read from the data medium.

Note

All information on previously installed supply units from this delivery will be lost. These supply units are no longer available for an installation with UNIT-NAME=*FROM-SCI.

SUPPLY-UNITS =

The supply units to be installed.

SUPPLY-UNITS = *ALL-REMAINING(...)

Only supply units that do not have the installation status *Installed* or *Being Installed* are installed from the SOLIS2 delivery.

REPLACE-OLD-FILES =

Overwrite/do not overwrite an existing file during installation.

REPLACE-OLD-FILES = *YES

Overwrite existing files.

REPLACE-OLD-FILES = *MINIMUM

Do not overwrite existing files, if possible.

REPLACE-OLD-FILES = *NO

Do not overwrite existing files.

FORCE-LOCATION =

Determines how IMON behaves when placing release items if the desired location (determined by catalog ID, user ID and prefix) is not the same as the stipulated user ID or prefix of the release unit or of the release item.

FORCE-LOCATION = *NO

Abort the placement of the release item, if the location or the user ID is stipulated for the release unit or one of the release items. Otherwise the desired location is used.

FORCE-LOCATION = *MINIMUM

Place the release units and release items with the stipulated user ID or prefix under the resulting location. Otherwise, use the desired user ID or prefix.

FORCE-LOCATION = *YES

Use the desired location. Any stipulated user IDs or prefix are ignored.

SUPPLY-UNITS = *COMPLETE-DELIVERY(...)

Install all supply units in the SOLIS2 delivery.

REPLACE-OLD-FILES = *YES / *MINIMUM / *NO

See description at SUPPLY-UNITS=*ALL-REMAINING(...) on ["INSTALL-UNITS Install and register software"](#).

FORCE-LOCATION = *NO / *MINIMUM / *YES

See description at SUPPLY-UNITS=*ALL-REMAINING(...) on ["INSTALL-UNITS Install and register software "](#).

SUPPLY-UNITS = *BY-CONFIGURATION(...)

Install all supply units based on configuration.

All supply units from delivery will be installed if they are in higher or equal version than already installed in system.

If supply unit was not in system, then it's installation will be skipped.

REPLACE-OLD-FILES = *YES / *MINIMUM / *NO

See description at SUPPLY-UNITS=*ALL-REMAINING(...) on ["INSTALL-UNITS Install and register software"](#).

FORCE-LOCATION = *NO / *MINIMUM / *YES

See description at SUPPLY-UNITS=*ALL-REMAINING(...) on ["INSTALL-UNITS Install and register software"](#).

SUPPLY-UNITS = *BY-DIALOG

Define the selection criteria and the target supply units in a dialog (see ["Interactive selection of supply units"](#)).

SUPPLY-UNITS = list-poss(30): <text 1..30 without-sep>(...)

Name of the supply unit, see ["Supply unit"](#).

VERSION =

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

Use the highest version of the supply unit.

VERSION = *ALL

Use all versions of the supply unit.

VERSION = <product-version without-man-corr>

Version designation without the release and correction state. CORRECTION-STATE = Correction state of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Use the highest correction state of the supply unit.

CORRECTION-STATE = *ALL

Use all correction states of the supply unit

CORRECTION-STATE = *LOWEST-EXISTING

Use the lowest correction state of the supply unit.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

Explicitly specify the correction state of the supply unit.

Format: <aso>

REPLACE-OLD-FILES =

Overwrite/do not overwrite an existing file during installation.

REPLACE-OLD-FILES = *YES

Overwrite existing files.

REPLACE-OLD-FILES = *MINIMUM

Do not overwrite existing files, if possible.

REPLACE-OLD-FILES = *NO

Do not overwrite existing files.

FORCE-LOCATION =

Determines how IMON behaves when placing release items if the desired location (determined by catalog ID, user ID and prefix) is not the same as the stipulated user ID or prefix of the release unit or of the release item.

FORCE-LOCATION = *NO

Abort the placement of the release item, if the location or the user ID is stipulated for the release unit or one of the release items. Otherwise the desired location is used.

FORCE-LOCATION = *MINIMUM

Place the release units and release items with the stipulated user ID or prefix under the resulting location. Otherwise, use the desired user ID or prefix.

FORCE-LOCATION = *YES

Use the desired location. Any stipulated user IDs or prefix are ignored.

UNIT-NAME = *FROM-SCI(...)

The supply units are already installed in a system and registered in the SCI.

PACKAGE-NAME = <alphanum-name 1..12>

Statement of the package name (as stated in the footer of the supply information) to which selection of the supply units is restricted.

PACKAGE-NAME = *BY-SUPPLY-UNITS

*This operand may be used only together with USER-CODE=*BY-SUPPLY-UNITS.*

Selection of the supply units is made using the opened SCI and is independent of the package name.

USER-CODE = <alphanum-name 1..8>

Statement of the customer ID (as stated in the footer of the supply information).

USER-CODE = *BY-SUPPLY-UNITS

*This operand may be used only together with PACKAGE-NAME=*BY-SUPPLY-UNITS.*

Selection of the supply units is made using the opened SCI and is independent of the customer ID.

SUPPLY-UNITS =

The supply units to be installed.

SUPPLY-UNITS = *ALL(...)

Install all supply units in the SOLIS2 delivery.

REPLACE-OLD-FILES =

Overwrite/do not overwrite an existing file during installation.

REPLACE-OLD-FILES = *YES

Overwrite existing files.

REPLACE-OLD-FILES = *MINIMUM

Do not overwrite existing files, if possible.

REPLACE-OLD-FILES = *NO

Do not overwrite existing files.

FORCE-LOCATION =

Determines how IMON behaves when placing release items if the desired location (determined by catalog ID, user ID and prefix) is not the same as the stipulated user ID or prefix of the release unit or of the release item.

FORCE-LOCATION = *NO

Abort the placement of the release item, if the location or the user ID is stipulated for the release unit or one of the release items. Otherwise the desired location is used.

FORCE-LOCATION = *MINIMUM

Place the release units and release items with the stipulated user ID or prefix under the the resulting location. Otherwise, use the desired user ID or prefix.

FORCE-LOCATION = *YES

Use the desired location. Any stipulated user IDs or prefix are ignored.

SUPPLY-UNITS = *BY-DIALOG

Define the selection criteria and the target supply units in a dialog (see ["Interactive selection of supply units"](#)).

SUPPLY-UNITS = list-poss(30): <text 1..30 without-sep>(…)

Name of the supply unit, see ["Supply unit"](#).

VERSION =

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

Use the highest version of the supply unit.

VERSION = *ALL

Use all versions of the supply unit.

VERSION = <product-version without-man-corr>

Version designation without the release and correction state.

CORRECTION-STATE =

Correction state of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Use the highest correction state of the supply unit.

CORRECTION-STATE = *ALL

Use all correction states of the supply unit

CORRECTION-STATE = *LOWEST-EXISTING

Use the lowest correction state of the supply unit.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

Explicitly specify the correction state of the supply unit.

Format: <aso>

REPLACE-OLD-FILES =

Overwrite/do not overwrite an existing file during installation.

REPLACE-OLD-FILES = *YES

Overwrite existing files.

REPLACE-OLD-FILES = *MINIMUM

Do not overwrite existing files, if possible.

REPLACE-OLD-FILES = *NO

Do not overwrite existing files.

FORCE-LOCATION =

Determines how IMON behaves when placing release items if the desired location (determined by catalog ID, user ID and prefix) is not the same as the stipulated user ID or prefix of the release unit or of the release item.

FORCE-LOCATION = *NO

Abort the placement of the release item, if the location or the user ID is stipulated for the release unit or one of the release items. Otherwise the desired location is used.

FORCE-LOCATION = *MINIMUM

Place the release units and release items with the stipulated user ID or prefix under the the resulting location. Otherwise, use the desired user ID or prefix.

FORCE-LOCATION = *YES

Use the desired location. Any stipulated user IDs or prefix are ignored.

UNIT-NAME = *CUSTOMER-APPROVED(...)

The supply units are already installed in a system and registered in the SCI. In contrast to installation with UNIT-NAME=*FROM-SCI(...) only those supply units that have a customer approval ID are selected. A delta installation is then performed for the selected supply units; in other words, only those installation items that have an older installation timestamp in the target system are installed.

PACKAGE-NAME = <alphanum-name 1..12>

Statement of the package name (as specified in the footer of the letter accompanying the delivery) to which selection of the supply units is limited.

PACKAGE-NAME = *BY-SUPPLY-UNITS

*This operand may be used only together with USER-CODE=*BY-SUPPLY-UNITS.*

Selection of the supply units is made using the opened SCI and is independent of the package name.

USER-CODE = <alphanum-name 1..8>

Statement of the customer ID (as specified in the footer of the letter accompanying the delivery) to which selection of the supply units is limited.

USER-CODE = *BY-SUPPLY-UNITS

*This operand may be used only together with PACKAGE-NAME=*BY-SUPPLY-UNITS.*

Selection of the supply units is made using the opened SCI and is independent of the customer ID.

SUPPLY-UNITS = *ALL(...) / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...)

Specifies the supply units to be installed. For a detailed description of the operand values refer to UNIT-NAME=*FROM-SCI(...).

SUPPLY-UNITS = *ALL(...)

All supply units of the SOLIS2 delivery that have a customer approval ID are selected for the delta installation.

REPLACE-OLD-FILES = *YES / *MINIMUM / *NO

See description of SUPPLY-UNITS=*ALL(...) under UNIT-NAME=*FROM-SCI(...) in ["INSTALL-UNITS Install and register software"](#).

FORCE-LOCATION = *NO / *MINIMUM / *YES

See description of SUPPLY-UNITS=*ALL(...) under UNIT-NAME=*FROM-SCI(...) in ["INSTALL-UNITS Install and register software"](#).

SUPPLY-UNITS = *BY-DIALOG

Define the selection criteria and the target supply units in a dialog (see ["Interactive selection of supply units"](#)).

SUPPLY-UNITS = list-poss(30): <text 1..30 without-sep>(...)

Name of the supply unit, see ["Supply unit"](#).

VERSION = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>

Version of the supply unit. Default is *HIGHEST-EXISTING, i.e. the highest version of the supply unit is used.

*ALL causes all versions of the supply unit to be used. Explicit version specification is made without release and correction status.

REPLACE-OLD-FILES = *YES / *MINIMUM / *NO

See description of SUPPLY-UNITS=*ALL(...) under UNIT-NAME=*FROM-SCI(...) in ["INSTALL-UNITS Install and register software"](#).

FORCE-LOCATION = *NO / *MINIMUM / *YES

See description of SUPPLY-UNITS=*ALL(...) under UNIT-NAME=*FROM-SCI(...) in ["INSTALL-UNITS Install and register software"](#).

TARGET-SYSTEM = *PARAMETERS(...)

System for which the installation is to take place (target system).

VERSION =

Version of BS2000.

VERSION = *CURRENT

Use the current BS2000 version.

VERSION = <product-version without-man-corr>

Explicit specification of the version of BS2000.

Permitted entries: V19.0, V20.0 and V21.0

PUBSET =

Home pubset of the system for installation.

PUBSET = *DEFAULT

Perform the installation on the home pubset of the current system under the default catalog ID of the corresponding user ID.

PUBSET = *PARAMETERS(...)

Perform the installation on an imported pubset for another target system ("foreign system").

CATID = <cat-id 1..4>

Catalog ID of the pubset on which installation will take place.

DEFLUID =

Default user ID of the target system.

DEFLUID = *STD

The default ID of the current system is used as the system default ID if the parameter file does not contain a "PVS-INFO" parameter for the specified subset of the target system. Otherwise the catalog ID agreed in the "PVS-INFO" parameter for the specified subset of the target system is used. For more information refer to the [section "IMON parameter file for default installation parameters"](#).

DEFLUID = <name 1..8>

An explicitly specified default user ID of the target system.

PROCESSOR-TYPE =

Specifies the hardware type of the target system for installing hardware-dependent release items.

PROCESSOR-TYPE = *STD

Value found in the PVS-INFO record associated to the specified target subset in the IMON parameter file, if any specified (cfr [section "IMON parameter file for default installation parameters"](#) for more details). Hardware type depending on target BS2000 version otherwise (see VERSION operand):

Target system version	The presetting *STD corresponds to
BS2000 V19.0 – V21.0	*ANY; all values are permitted (*CISC-390, *X86)

PROCESSOR-TYPE = *CISC-390

The target system runs on a /390 machine. Only those release items that have the TARGET=A code (hardware-independent) or the TARGET=S (/390 variant) code are installed.

PROCESSOR-TYPE = **X86

Only those release items that have the TARGET=A code (hardware-independent) or the TARGET=K o (X86 variant) are installed.

PROCESSOR-TYPE = *ANY

The release items whose hardware variants are supported in the target system are installed.

Target system version	Supported hardware types
BS2000 V19.0 – V21.0	*CISC-390 and *X86

ITEM-SELECTION =

This presetting specifies whether only release items that are specified for the relevant target system version are installed (see VERSION operand).

ITEM-SELECTION =

This presetting specifies whether only release items that are specified for the relevant target system version are installed (see VERSION operand).

ITEM-SELECTION = *STD

Value found in the PVS-INFO record associated to the specified target subset in the IMON parameter file, if any specified (cfr [section "IMON parameter file for default installation parameters"](#) for more details). *TARGET-VERSION-ONLY is otherwise assumed.

ITEM-SELECTION = *ALL

All release items are installed regardless of the target system version.

ITEM-SELECTION = *TARGET-VERSION-ONLY

Only release items for which the relevant target system version is specified are installed. Release items that are specified for other target system versions are not installed (this refers in particular to the item types SSC, SSD, MSV, SDF, *DA and REP).

OLD-FILE-SAVING = *STD / *NO / *WITH-LMS / *WITH-ARCHIVE(...)

Save/do not save the files that are overwritten by the installation.

OLD-FILE-SAVING = *STD

Value specified in the OLD-FILE-SAVING record of the used IMON parameter file, if any specified; *NO assumed otherwise (cfr [section "IMON parameter file for default installation parameters"](#) ([IMON parameter file for default installation parameters](#)) for more details).

OLD-FILE-SAVING = *NO

Do not save the files.

OLD-FILE-SAVING = *WITH-LMS

The files are saved in a library with LMS/LMSCONV. The name of the save library for BS2000 is \$<work file ID>. IMON.SAVE.LIB.<package name>.<customer ID>, where <package name> is the value of the PACKAGE-NAME operand, <customer ID> is the value of the USER-CODE operand and <work file ID> is the current work file ID setting (see the MODIFY-IMON-OPTIONS statement on "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

OLD-FILE-SAVING = *WITH-ARCHIVE(...)

Save the files with ARCHIVE.

VOLUME = <vsn> / *FROM-OPERATOR(...)

VSN of the data volume to which the files are to be saved with ARCHIVE.

VOLUME = *FROM-OPERATOR(...)

The MAREN subsystem automatically selects the VSN of a free volume. If the MAREN subsystem is not loaded, the statement is rejected with the message IMO0449.

LOCATION = *STD / <alphanumeric-name 1..8>

Name of the location from which the free volume is to be selected. *STD is default, i.e. the ARCHIVE backup is carried out without MAREN support, with the default settings of ARCHIVE.

DEVICE-TYPE = <device>

Device type of the data volume.

PASSWORD-FILE =

The passwords that are needed for overwriting a file during installation. The installation is aborted in the event of an error.

PASSWORD-FILE = *NO

There are no passwords stored in a file.

PASSWORD-FILE = <filename 1..54 without-gen-vers>(…)

Name of the file containing the ADD-PASSWORD commands for password-protected files. These commands are integrated into the installation procedure. They must be syntactically correct.

READ-PASSWORD = <c-string 1..4>

Read password of the password file.

LOG-PRINTING = *STD / *NO / *YES

Print/do not print the installation logs.

LOG-PRINTING = *STD

Value specified in the PRINT-LOG-FILES record of the used IMON parameter file, if any specified; *NO assumed otherwise (cfr [section "IMON parameter file for default installation parameters"](#) for more details).

PLACEMENT-MODE =

The placement mode of the installation items.

PLACEMENT-MODE = *STD

Use the default settings for the placement mode, for overwriting existing files and for the behavior in the event of user ID conflicts.

PLACEMENT-MODE = *BY-DIALOG

You are prompted to enter details on “customizing” and optional specifications for the placement mode of the installation items of the supply unit in dialog boxes (similar to [figure 27 \(Edit menu\)](#)).

If the string *RU is specified for a supply unit in the “Userid” column of the “Supply unit placement parameters” dialog box, a follow-on mask appears in which you can enter specific details for the release units of the supply unit. Details of the release unit have priority over those of the supply unit.

Replace file

Overwrite/do not overwrite an existing file during installation.

Possible entries:

- Y Overwrite existing files.
- M Do not overwrite existing files, if possible.
- N Do not overwrite existing files.

Force Loc.

Determines how IMON behaves when placing release items if the desired user ID is not the same as the stipulated user ID of the release unit or of the release item.

Possible entries:

- N The placement of the release items is aborted, if the location or the user ID is stipulated for the release unit or one of the release items. Otherwise the desired location is used.

-
- M Place release units and release items with the stipulated user ID or prefix on the resulting location. Otherwise, use the desired user ID or prefix.
 - Y Use the desired location for the release items. Any stipulated user IDs or prefix are ignored.

Library

Determines how IMON handles the supply unit libraries. Possible entries:

- Y The supply unit libraries are cataloged on the target system under their original names. Their elements are also incorporated in alternative libraries via IMON.
- N The supply unit libraries are cataloged on the target system under their original names only.
- The supply unit does not contain a library file.

Call/Enter

Start/do not start installation items of the type DO or ENT during installation.

Possible entries:

- Y Start the item.
- N Do not start the item.

ACTIVATION-MODE = *STD / *PARAMETERS(...)

Defines which installation items are to be prepared for subsequent activation during the installation procedure.

ACTIVATION-MODE = *STD

For each activation process, the value which applies is the value declared in the corresponding parameter record ("SDF-PROCESSING", "MIP-PROCESSING" ...). In the absence of a declaration *YES applies to all parameters excepted for POSIX-PROCESSING for which *BY-CONFIGURATION-FILE applies.

ACTIVATION-MODE = *PARAMETERS(...)

Select the installation items to be processed using dialog boxes (similar to [figure 29 "Edit menu"](#)).

SYNTAX-FILE-PROCESS = *BY-DIALOG / *NO / *YES

Process/do not process syntax files during the installation procedure. You are prompted to enter the necessary specifications in dialog boxes.

SDF param file

Name of the SDF parameter file.

MESSAGE-FILE-PROCESS = *BY-DIALOG / *NO / *YES

Process/do not process message files during the installation procedure. You are prompted to enter the necessary specifications for specific versions in dialog boxes.

MIP param file

Name of the MIP parameter file.

MES file

Name of the message file into which the message files of the subsystems are to be merged using CREATION-TIME=BEFORE-DSSM-LOAD or AT-DSSM-LOAD.

DSSM-PROCESSING = *BY-DIALOG / *NO / *YES

Process/do not process the static DSSM subsystem catalog during the installation procedure. You are prompted to enter the necessary specifications in dialog boxes. The new DSSM subsystem catalog is only activated in the next system run

Catalog name

Defines which static DSSM subsystem catalog is processed. If you do not enter a catalog name here, the default name of the static catalog is used.

IMON saves an existing catalog as <old-catalog-name>.<time-stamp> before replacing it with the new catalog. If the name is too long, <old-catalog-name> is abbreviated.

Keep old version: Yes / No

Keep/do not keep old versions of the processed subsystems in the DSSM subsystem catalog for subsystems that allow to keep old versions.

REP-PROCESSING = *BY-DIALOG / *NO / *YES

Defines how the RMS depot is processed during the installation procedure.

REP-PROCESSING = *BY-DIALOG

You must enter the necessary specifications in dialog boxes.

RMS processing: 1. Depot+loaders 2. Depot only 3.No

The type of processing for RMS data.

If *Depot only* is specified, the RMS data is transferred into the RMS depot.

If *Depot+loaders* is specified, the REP loaders are also generated.

If *No* is specified, the depot is not updated and no REP loaders are generated.

Depot location: 1. Standard 2. Enforced

The name of the RMS depot for transferring data and generating the REP loaders.

If you specify *Standard*, the RMS depot specified during parking is used for parked software.

Otherwise, the RMS depot specified for "Location" is used.

If you specify *Enforced*, the RMS depot specified for "Location" is always used.

Location: [:<cat-id>:]\$[<userid>].

The catalog ID and/or user ID under which the RMS depot is stored. The default file name is "RMS.DEPOT".

REP-PROCESSING = *NO

Do not process the RMS files. The depot is not updated and no REP loaders are generated.

Note

If there are no REP loaders, the system may not be able to attain a "SYSTEM READY".

POSIX-PROCESSING = *BY-DIALOG / *BY-CONFIGURATION-FILE / *NO / *ALL

Specifies whether POSIX satellites are to be processed at installation (as installation calls in POSIX). *BY-CONFIGURATION-FILE / *ALL can be given even if POSIX configuration file does not exist.

POSIX-PROCESSING = *BY-DIALOG

Activation parameters are asked interactively using a further specific dialog box.

POSIX-PROCESSING = *BY-CONFIGURATION-FILE / *NO / *ALL

Specifies whether the POSIX processing must be performed or not.

POSIX-PROCESSING = *BY-CONFIGURATION-FILE

no customized activation, each POSIX satellite is processed only if the product to which it pertains is already registered into the POSIX configuration file.

POSIX-PROCESSING = *NO

no POSIX satellite is processed.

POSIX-PROCESSING = *ALL

each POSIX satellite is processed. The old value "*YES" is assumed to be "*ALL".

START = *IMMEDIATELY / *BY-USER

Installation is either immediate and automatic or deferred for start by user. The procedure name is reported after the file is generated or during the automatic start (start with the job name INSTALL).

START = *IMMEDIATELY

Installation is immediate and automatic.

START = *BY-USER

The installation process (placement and activation) is introduced by starting the installation procedure (ENTER-PROCEDURE). The installation procedure can be run interactively (CALL-PROCEDURE), but this is time-consuming.

WORK-FILE-DELETING = *STD / *YES / *NO

Delete/do not delete installation files that are no longer needed (files of the parked release items and SYSRMS files).

WORK-FILE-DELETING = *STD

Value specified in the WORK-FILE-DELETING record of the used IMON parameter file, if any specified; *YES assumed otherwise (cfr [section "IMON parameter file for default installation parameters"](#) for more details).

WORK-FILE-DELETING = *NO

Do not delete the files after installation. This means that parked software can be installed a number of times.

WORK-FILE-DELETING = *YES

Delete the specified files after they are processed successfully.

UNDO-PREPARATION = *STD / *NO / *PARAMETERS(...)

Specifies if all metadata required for restoration to the original state before installation (Undo function, see the UNDO-SUPPLY-UNITS statement) are to be saved in separate files (Undo files).

UNDO-PREPARATION = *STD

Value specified in the UNDO-PREPARATION record of the used IMON parameter file, if any specified; *NO assumed otherwise (cfr [section "IMON parameter file for default installation parameters"](#) for more details).

UNDO-PREPARATION = *PARAMETERS(...)

The metadata is saved in Undo files.

RETENTION-PERIOD = 30 / <integer 0..999 days>

Specifies after how many days the Undo files may be changed or deleted. The default value is a retention period of 30 days.

SAVE-MEDIUM = *ARCHIVE-DIRECTORY(...) / *TAPE(...)

Specifies the data media used to save the data with ARCHIVE.

SAVE-MEDIUM = *ARCHIVE-DIRECTORY(...)

The data media to be used are administered in an ARCHIVE directory. An accordingly free data medium from this directory is used.

NAME = <filename 1..54 without-gen-vers>

Name of the ARCHIVE directory.

DEVICE-TYPE = <device>

Device type of the data medium.

SAVE-MEDIUM = *TAPE(...)

The data medium to be used is specified explicitly.

VOLUME = <vsn>

VSN of the data medium.

DEVICE-TYPE = <device>

Device type of the data medium.

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC, where <package name> is the value of the operand PACKAGE-NAME, <customer ID> is the value of the operand USER-CODE and <work file ID> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement on ["MODIFY-IMON-OPTIONS Modify IMON parameter settings"](#)).

SAVE-MEDIUM = *LIBRARY

Data are saved with LMS/LMSCONV into a PLAM library Library name: <work-file-location>.IMON.UNDO.LIB.<package>, where <package name> is the value of the operand PACKAGE-NAME and <work file location> is the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement ["MODIFY-IMON-OPTIONS Modify IMON parameter settings"](#)).

CHECK-CONFIGURATION = *STD / *NO / *YES

Specifies whether the software configuration is to be checked.

CHECK-CONFIGURATION = *STD

Value specified for the specified subset of the target system in the CONFIGURATION-CHECK record of the used IMON parameter file, if any specified; *YES assumed otherwise. (cfr [section "IMON parameter file for default installation parameters"](#) for more details).

PARAMETER-FILE = *STD / *NO / <filename 1..54 without-gen-vers>

Specifies whether a parameter file with predefined default values for some installation parameters has to be used or not. See [section "IMON parameter file for default installation parameters"](#) for more details.

A customer-specific parameter file \$TSOS.SYSPAR.IMON.<kkz> or the file \$TSOS.SYSPAR.IMON are always evaluated (see [section "IMON parameter file for optional installation functions"](#)).

PARAMETER-FILE = *STD

If they exist, the standard parameter file \$TSOS.SYSPAR.IMON.LAST and the customerspecific parameter file \$TSOS.SYSPAR.IMON.<kkz> (resp. \$TSOS.SYSPAR.IMON) are used. In the case of entries which are contained in both files, the value from the parameter file \$TSOS.SYSPAR.IMON.LAST is used.

PARAMETER-FILE = *NO

The parameter file \$TSOS.SYSPAR.IMON.LAST is ignored. A customer-specific parameter file \$TSOS.SYSPAR.IMON.<kkz> or the file \$TSOS.SYSPAR.IMON is always evaluated.

PARAMETER-FILE = <filename 1..54 without-gen-vers>

Specification of a specific user parameter file.

Additionally, \$TSOS.SYSPAR.IMON.<user-code> or the file \$TSOS.SYSPAR.IMON will be evaluated. The parameter file \$TSOS.SYSPAR.IMON.LAST is ignored.

Notes

1. Delivery information files:

If DISTRIBUTION-MEDIUM=*SOLIS2-VOLUME/*LOCAL-VOLUME is specified, the delivery information from data volume is stored at the saving location for work files. Existing files with the same names are overwritten. The saving location is registered in the SCI so that they can be used again later. If DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM is specified, the distribution medium (*SOLIS2-VOLUME/*LOCAL-VOLUME /*LIBRARY) and the values for VOLUME and DEVICE-TYPE that also have to be specified for *SOLIS2-VOLUME/*LOCAL-VOLUME are restored from the SCI. The stored delivery information enables you to use cataloged files for saving accesses to data volume/library.

The acknowledge for systems support is printed automatically each time when the delivery is opened from the delivery medium (*SOLIS2-VOLUME/*LOCAL-VOLUME/*LIBRARY).

If DISTRIBUTION-MEDIUM=*LOCAL-VOLUME the delivery has already been saved on another data volume (e. g. virtual volume of a CentricStor). In this case, the specified VSN and device type are used for access to the installation files and the entries in the product movement file are ignored. However, only files from the specified volume can be processed.

When opening a new SOLIS2 delivery using INSTALL-UNITS, PARK-UNITS or PRINT-DOCUMENTATION, the delivery is registered in the SCI and the working files used by the current IMON version (e. g. SOLPAR.IMON.<ver>.GEN) are extracted on the workfile location.

When a SOLIS2 delivery which is already registered is opened again, the delivery information and work files used at the original open time are used again to save unnecessary accesses to the delivery medium (MTC /Plam library).

New IMON versions can use the AUSGEN file (SOLPAR.IMON.<ver>.GEN) from older versions without any problem, but in this case new functions provided in the new IMON version with the help of the AUSGEN file are not available.

For this reason, the following process may be necessary:

When a new version of IMON is installed (from the BS2GA.IMON *3 delivery) and supply units from the same delivery and using the same work file location are still to be processed (park or installation, process), it is useful to install BS2GA.IMON from this delivery again with the newly installed IMON version in order to extract the new IMON environment and so to activate possible new optional functions for these remaining units.

2. Operand UNIT-NAME=*FROM-SCI(...):

With this function, a supply unit which has already been installed and tested on a test system, is moved to the productive system and installed. It is a prerequisite that the original installation files (especially the product transfer file A.SOLIS.B.<paketname>.<kundenkennzeichen>) are present on the work file ID. This method of installation offers the same installation functions as the standard installation (from a SOLIS2 delivery), but is different in the following respects:

- For the target system version, only the version for which the supply unit has already been installed can be stated.
- The installation files are copied from the system which has already been installed.
- The RMS files are not processed. No depot update is carried out and no REP loaders are created. The REP loaders are copied from the system that has already been installed.

3. Operand UNIT-NAME=*CUSTOMER-APPROVED(...):

In contrast to UNIT-NAME=*FROM-SCI(...) only supply units that have a customer approval ID are selected. Customers set the approval ID using the SET-CUSTOMER-APPROVAL statement to release the supply unit with a defined quality standard for installation in production systems. IMON resets the customer approval ID if installation operations change an installation item of a released supply unit. A delta installation is then performed for the selected supply units; in other words, only those installation items that have an older installation timestamp in the target system are installed.

Installation cannot be performed unless the following conditions are fulfilled for all selected supply units.

- The change date of the supply unit must be earlier than the approval time (timestamp of the customer approval ID).
- Installation on the source pubset must have taken place after installation on the target pubset.
- The correction status on the source pubset must be higher than that on the target pubset.
- The product movement files must be available on the work file ID.

RMS processing is unnecessary during installation. Only the REP loaders are copied from the source pubset to the target system.

This installation function should only be used if the installation on the source and target pubset is for the same target version.

4. DELIVERY-MEDIUM=*REGISTERED-MEDIUM operand:

The files of a new delivery can therefore overwrite existing files and these files can differ from delivery to delivery. If the statement is called using the DELIVERY-MEDIUM=*REGISTERED-MEDIUM operand, problems can arise if the cataloged delivery information does not match the delivery. In this case, the files should be read from MTC.

5. FORCE-LOCATION operand:

If FORCE-LOCATION=*MINIMUM is specified, the stipulated user IDs are used, where necessary. A user ID specified for a non-standard installation is used for other items. Users must therefore accept the fact that their items do not necessarily appear under the user ID which they specified.

If FORCE-LOCATION=*YES is specified, stipulated user IDs are ignored. A user ID that is specified for a non-standard installation is used for all items. There is no guarantee that the installed products will operate properly.

6. The statement is rejected without action and an error message is displayed in the following cases:

- The value *BY-DIALOG was specified in batch mode.
- The target system version is not [V]19.0, [V]20.0 or [V]21.0.
- The user ID specified as the saving location for work files using MODIFY-IMON-OPTIONS cannot be accessed or does not have sufficient space.
- DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM was specified and there is no SOLIS2 delivery with the specified package name and customer ID in the SCI.
- The delivery information does not exist or cannot be accessed:
 - DISTRIBUTION-MEDIUM=*REGISTERED-MEDUM was specified, but the delivery information file does not exist or cannot be accessed.
 - DISTRIBUTION-MEDUM=*SOLIS2-VOLUME/*LOCAL-VOLUME was specified, but the specified data volume or the specified device does not exist or cannot be accessed, or the delivery information file cannot be copied from data volume to the current user ID.
 - DISTRIBUTION-MEDIUM=*LIBRARY was specified, but the corresponding PLAM library does not exist on the current user ID or cannot be accessed, or the delivery information file cannot be retrieved from the library and copied to the current user ID.
- The delivery information cannot be successfully processed by IMON.
- The product movement file belonging to the specified SOLIS2 delivery or to the registered SOLIS2 delivery does not contain any of the supply units that were specified by the user.
- The specified parameter file does not exist, cannot be opened, or has the wrong format.
- An installation procedure was already generated for one or more of the supply units specified by the user. The installation status is set to “being installed”.
- The input list contains duplicate supply units, i.e. units with the same values or key words for one of the identification parameters.
- An installation parameter is not correct (e.g. non-existing user ID)
- The IMON-BAS generation procedure required for the specified target system version is not available in the delivery. The following scenarios can arise:
 - The target system version [V]19.0, [V]20.0 requires a generation procedure with IMON-BAS >= V3.3.
 - The target system version [V]21.0 and higher requires a generation procedure with IMON-BAS >= V3.4.

7. An installation batch job is aborted in the following cases:

- An installation user ID is not defined (deleted after the batch job is created).
- REPLACE-OLD-FILES=*NO was specified and a supply unit must be installed under the name of an existing file.

- If there is already a file under the standard installation path in which a supply unit is to be stored, there are two possibilities: Either the supply unit can be installed anywhere and IMON generates a new installation path, or the supply unit cannot be installed anywhere and IMON does not install the supply unit. Existing files must either be deleted manually, or REPLACE-OLD-FILES=*MINIMUM must be specified.
 - A file that has to be overwritten is password-protected, but no password file was specified.
 - A file that has to be overwritten is password-protected, but a specified password file does not contain the necessary password commands or contains incorrect commands.
 - An existing file cannot be overwritten (e.g. it may be locked).
8. You should avoid running a number of installation processes at the same time since processes may be blocked when attempts are made to access shared resources. For example, global files are locked while they are being merged.
 9. If parked files are to be deleted after installation, these files are regarded as being no longer parked as soon as the batch job for the installation has been generated.
 10. If a number of installations are run at the same time, the “Being Installed” and “Installed” states can overwrite each other.
 11. IMON parameter file:
If there is a customer-specific IMON parameter file with the default name \$TSOS.SYSPAR.IMON.<customer ID>, then IMON also evaluates the parameters in this file (see also [section "IMON parameter files"](#)).
 12. Undo files are only generated when UNDO-PREPARATION=*PARAMETERS(...) is explicitly specified. The metadata saved in these files are required for the restoration to the original state (Undo function). A different volume is used for every UNDO preparation if data saving on a data volume is selected.

Creating a LOCAL-VOLUME

Because of the wide range of application options and customer-specific dependencies, conversion of a SOLIS2 delivery to a LOCAL-VOLUME must be performed manually. Please proceed as described in the example below:

1. Only when robots are used:

Import the SOLIS2-VOLUME to the foreign cartridge area of ROBAR using the ROBAR SDF statement ADD-ROBAR-VOLUME including the declaration of the position in the real archive (e.g. 05)

2. Only when MAREN is used:

Import the SOLIS2-VOLUME to the MAREN catalog using the MARENADM statement

```
//ADD-PRIVATE-VOLUME
```

3. Convert a SOLIS2 delivery to a LOCAL-VOLUME:

```
/START-ARCHIVE
FILES FROM=SV,(vsn1),DEVICE=TAPE-C4
FILES FROM=SV,(vsnx),DEVICE=TAPE-C4
SAVE DIR=NONE,TAPES=POOL,DEVICE=TAPE-C4,CH=NO,L=SYSLST
END
```

In the case of continuation tapes, a FILES statement must be issued for each MTC. Here vsn1 and vsnx are the Volume Serial Numbers (VSNs) of the SOLIS2 delivery (see delivery note).

4. If the delivery is to be copied to a predefined VSN (e.g. when a VSN is reserved using MAREN), the entry in the ARCHIVE job must be changed from “=POOL” to “=<vsn>”.
5. Only when robots are used:

Export the SOLIS2-VOLUME from the foreign cartridge area of ROBAR with the RO-BAR statement //REMOVE-ROBAR-VOLUME

Interactive selection of supply units

The operand value SUPPLY-UNITS= *BY-DIALOG in the INSTALL-UNITS statement displays a selection window. Subsequent selections are made interactively and in dialog boxes.

The choice of supply units depends on the delivery information. If the delivery information is from data volume, all supply units of a delivery can be selected.

If the delivery information is from an SCI, you can select only those supply units that have not yet been installed. Exception: If the supply units were installed without deleting the work files, they can be selected again for installation. Supply units are selected interactively by entering any character in front of each desired supply unit.

It is now possible to install the same supply unit for the same system multiple times using the *REGISTERED-MEDIUM functionality (e.g. after deleting a file of the supply unit by mistake). Moreover, the "Installed" status of a supply unit will now be set only in the SCI of the system where the supply unit is effectively installed and no longer in the SCI from where the installation is triggered (where it remains unchanged).

```
IMON: SOLIS2 delivery: Package name: <package-name> User code: SOL2P
-----
Unit name          SU selection      Units      1 through  5 of  5
                                Version    Corr state  More:
X EDT              17.0      D30
X OPENFT          12.1      C10
X OPENFT-CR       12.1      B10
X PERCON          02.9      C00
X SORT            08.0      A30
*** End of SU selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 57: Interactive selection of supply units

Notes

- Interactive selection is only part of the *BY-DIALOG operand value. It is different from the similar looking screen which is displayed when you select the *Open* option in the *File* menu in menu mode.
- Press [DUE] when you have finished selecting the supply units.
- Interactive selection is not available in batch mode.

4.4.7 MODIFY-IMON-OPTIONS Modify IMON parameter settings

Privileges: SUBSYSTEM-MANAGEMENT,
STD-PROCESSING

Function

The MODIFY-IMON-OPTIONS statement defines the following defaults for the current IMON programme run:

- the SCI to be processed (SCI-NAME operand)
- the location at which work files and software that need to be parked are temporarily stored (WORK-FILE-LOCATION operand)
- a reference file, which is to be used as the basis for a configuration check during installation (operand REFERENCE-FILE)

After IMON is called, the respective default values are valid immediately (corresponds to the respective operand value *STD). The currently set values are valid until a new value is explicitly set (with MODIFY-IMON-OPTIONS or the relevant menu function) or until the end of the IMON programme run.

Format

MODIFY-IMON-OPTIONS

SCI-NAME = *UNCHANGED / *STD / <filename 1..50 without-gen-vers>

,WORK-FILE-LOCATION = *UNCHANGED / *STD / <partial-filename 2..23>

,REFERENCE-FILE = *UNCHANGED / *STD / <filename 1..54 without-gen-vers>

Operands

SCI-NAME =

Selects an SCI. This SCI is retained until the MODIFY-IMON-OPTIONS statement is used to select a different SCI or IMON is ended. Both these actions implicitly close the SCI. Privileged users can use this operand to open a foreign SCI. If the foreign SCI defined here does not yet exist, it is generated under the specified file name on another pubset.

You have write access to the SCI when it is opened if you are a privileged user, whereas the SCI is read-only for nonprivileged users.

SCI-NAME = *UNCHANGED

The open SCI is left alone. If no SCI was opened beforehand, the standard SCI is opened.

SCI-NAME = *STD

The standard SCI \$TSOS.SYS.IMON.SCI is opened on the home pubset.

SCI-NAME = <filename 1..50 without-gen-vers>

Explicit specification of an SCI. The file name is restricted to a maximum of 50 characters, because 4 characters are reserved for the ".GPN" suffix.

WORK-FILE-LOCATION =

Specifies the location where the software is to be parked or where work files are to be created.

If the user ID SYSSAG is defined, then it is set by default to the home pubset after the call using the START-IMON command. If it is not defined, then the location is set to \$TSOS.IMON.

WORK-FILE-LOCATION = *UNCHANGED

The location selected in the preceding MODIFY-IMON-OPTIONS statement is retained.

WORK-FILE-LOCATION = *STD

The default location for work files to be used is the standard default one.

WORK-FILE-LOCATION = <partial-filename 2..23>

Explicit specification of the location. Temporary files may not be specified.

The specification is made in the form <partial-filename 2..23> and can contain a file name prefix in addition to a catalog ID and user ID. The file name prefix may be up to a maximum of 7 characters long (including the period at the end).

The specification of a prefix can lead to problems in the following statements when the name created exceeds the maximum file name length permitted. For this reason it is recommended not to exceed a total length of 18 characters (including catalog ID and user ID) when a prefix is used.

REFERENCE-FILE =

Selects a reference file which is to be used as the basis for a configuration check during installations. The selection is valid until a new reference file is specified with MODIFY-IMON-OPTIONS or IMON is ended.

REFERENCE-FILE = *UNCHANGED

The previously agreed reference file is kept. If no reference file had been specified, the default reference file is still valid (see *STD).

REFERENCE-FILE = *STD

The default reference file, \$TSOS.SYS.IMON.SCI.REF, is used on the home pubset.

REFERENCE-FILE = <filename 1..54 without-gen-vers>

Location of a reference file.

Notes

The statement is rejected and an error message is displayed in the following cases:

1. If one of the following errors occurred when the operand SCI-NAME was entered:
 - The file name that was specified without a catalog ID and/or a user ID cannot be extended to produce a path name of up to 50 characters.
 - The SCI was not cataloged using USER-ACCESS=WRITE.
 - The user has the SUBSYSTEM-MANAGEMENT privilege, but the specified SCI cannot be generated, and read and write access is not permitted for it.
 - The user has the STD-PROCESSING privilege, but the specified SCI cannot be generated, and read access is not permitted for it.

-
2. The user ID and/or catalog ID specified in the WORK-FILE-LOCATION operand is not defined in the system or cannot be accessed.
 3. The file name prefix specified in the WORK-FILE-LOCATION operand is longer than 7 characters (including the dot at the end).

4.4.8 PARK-UNITS Park Software

Privileges: SUBSYSTEM-MANAGEMENT

Function

The PARK-UNITS statement enables you to park supply units on a user ID (park ID). This statement prepares for the installation of software. At the same time, the delivery is registered in the SCI.

This statement generates a park procedure that is started automatically by IMON if the specified operands are successfully validated.

Parked software can be installed using the INSTALL-UNITS statement by specifying DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM. As a rule, the parked supply units are deleted after the park ID has been installed (WORK-FILE-DELETING=*YES). The supply units can be retained for a multiple installation.

If the delivery comes with a data volume, the delivery documentation is created and stored in a library.

Format

PARK-UNITS

UNIT-NAME = *FROM-SOLIS2-DELIVERY(...)

***FROM-SOLIS2-DELIVERY(...)**

| **PACKAGE-NAME** = <alphanum-name 1..12>

| **,USER-CODE** = <alphanum-name 1..8>

| **,DISTRIBUTION-MEDIUM** = *SOLIS2-VOLUME(...) / *LOCAL-VOLUME(...) / *REGISTERED-MEDIUM

| ***SOLIS2-VOLUME(...)**

| | **VOLUME** = <vsn>

| | **,DEVICE-TYPE** = <device>

| | **,DOCUMENT-LIBRARY** = *STD / <filename 1..54 without-gen-vers>

| | **,REGISTRATION** = *EXTEND / *REPLACE

| ***LOCAL-VOLUME(...)**

| | **VOLUME** = <vsn>

| | **,DEVICE-TYPE** = <device>

| | **,DOCUMENT-LIBRARY** = *STD / <filename 1..54 without-gen-vers>

| | **,REGISTRATION** = *EXTEND / *REPLACE

| **,SUPPLY-UNITS** = *ALL / *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(...

| <text 1..30 without-sep>(...

| | **VERSION** = *HIGHEST-EXISTING / *ALL / <product-version without-man-corr>

| | **,CORRECTION-STATE** = *HIGHEST-EXISTING / *ALL / *LOWEST-EXISTING / list-poss(15):
<alphanum-name 3..3>

,TARGET-SYSTEM = *PARAMETERS(...)

***PARAMETERS(...)**

| **VERSION** = *CURRENT / <product-version without-man-corr>

| **,PUBSET** = *DEFAULT / <cat-id 1..4>

,OLD-FILE-SAVING = *NO / *WITH-LMS / *WITH-ARCHIVE(...)

***WITH-ARCHIVE(...)**

| **VOLUME** = <vsn> / *FROM-OPERATOR(...)

```

|      *FROM-OPERATOR(...)
|      |      LOCATION = *STD / <alphanum-name 1..8>
|      ,DEVICE-TYPE = <device>
,RMS-DEPOT-UPDATING = *NO / *PARAMETERS(...)
  *PARAMETERS(...)
|      DEPOT-LOCATION = *STD / <partial-filename 2..16>
,FILE-LOCATION = <partial-filename 2..16>

```

Operands

UNIT-NAME =

Source of supply units to be parked.

UNIT-NAME = *FROM-SOLIS2-DELIVERY(...)

Park supply units from a SOLIS2 delivery.

PACKAGE-NAME = <alphanum-name 1..12>

Package name (as stated in the footer of the delivery contents).

USER-CODE = <alphanum-name 1..8>

User code (as stated in the footer of the delivery contents).

DISTRIBUTION-MEDIUM =

The distribution medium: either IMON uses the information from the SCI or the distribution medium is specified explicitly.

DISTRIBUTION-MEDIUM = *SOLIS2-SUPPORT(...)

The delivery is stored on a data volume shipped with Solis2 (tape cartridge, CD or DVD). All supply units of the specified delivery are available for selection. The delivery information is read from data volume. The selected supply units are copied from data volume to the current user ID and registered in the SCI. Existing files with the same names are overwritten.

VOLUME = <vsn>

VSN of the data volume which contains the delivery (see delivery contents).

DEVICE-TYPE = <device>

Device type of the data volume (see delivery contents).

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC where <package name> is the value of the PACKAGE-NAME operand, <customer ID> is the value of the USER-CODE operand and <work file ID> the current file location setting for work files (see the MODIFY-IMON-OPTIONS statement "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

An explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

Supply units that have already been parked may not be parked any more.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information read from the data medium.

Note

All information on previously installed supply units from this delivery will be lost.

DISTRIBUTION-MEDIUM = *LOCAL-SUPPORT(...)

The delivery is stored on a data volume created by the customer (tape cartridge, or virtual volume in a CentricStor). The data volume information of the product movement file is ignored. All supply units of the specified delivery are available for selection. The delivery information is read from data volume. The selected supply units are copied from data volume to the current user ID and registered in the SCI. Existing files with the same names are overwritten.

Restriction/Note:

With LOCAL-VOLUME there is no support of the continuation volume, so the maximum size of a copiable SOLIS2 delivery is limited by the capacity of the LOCAL-VOLUME used.

The maximum necessary size for the delivery may be found on page 4 of the delivery documentation.

Because of the wide range of application options and customer-specific dependencies, conversion of a SOLIS2 delivery to a LOCAL-VOLUME must be performed manually.

Please proceed as described in the example below:

1. Only when robots are used:
Import the SOLIS2-VOLUME to the foreign cartridge area of ROBAR using the ROBAR statement ADD-ROBAR-VOLUME including the declaration of the position in the real archive (e.g. 05)
2. Only when MAREN is used:
Registration of the SOLIS2 VOLUME in the MAREN catalog using the MAREN statement ADD-PRIVATE-VOLUME

3. Transfer of the SOLIS2 delivery onto a LOCAL-VOLUME:

```
FILES FROM=SV, ( vsnx ), DEVICE=TAPE-C4
SAVE DIR=NONE, TAPES=POOL, DEVICE=TAPE-C4, CH=NO, L=SYSLST
END
```

For multi-tape deliveries, one FILES statement must be specified per tape. vsn1-x are the Volume Serial Numbers (VSN) of the SOLIS2 delivery (see page 1 of the delivery documentation)

4. If the delivery is to be copied to a predefined VSN (e.g. when a VSN is reserved using MAREN), the entry in the ARCHIVE job must be changed from "=-POOL" to "=-POOL" to "=-volume number"
5. Only when robots are used:

Export the SOLIS2-VOLUME from the foreign cartridge area of ROBAR with the ROBAR statement //REMOVE-ROBAR-VOLUME

VOLUME = <vsn>

VSN of the data volume which contains the delivery.

DEVICE-TYPE = <device>

Device type of the data volume.

DOCUMENT-LIBRARY = *STD / <filename 1..54 without-gen-vers>

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it. For details refer to the operand description for DISTRIBUTION-MEDIUM=*SOLIS2-VOLUME(...) on "[PARK-UNITS Park Software](#)".

REGISTRATION = *EXTEND / *REPLACE

Specifies if the delivery is to be handled as a delivery already registered in the SCI or as a new delivery. For details refer to the operand description for DISTRIBUTION-MEDIUM=*SOLIS2-VOLUME(...) on "[PARK-UNITS Park Software](#)".

DISTRIBUTION-MEDIUM = *REGISTERED-MEDIUM

The delivery information, the VSN and the device type of the supplied data volume are taken from the open SCI. Files that have already been parked or installed are not read again from the data volume.

It is now possible to install the same supply unit for the same system multiple times using the *REGISTERED-MEDIUM functionality (e.g. after deleting a file of the supply unit by mistake). Moreover, the "Installed" status of a supply unit will now be set only in the SCI of the system where the supply unit is effectively installed and no longer in the SCI from where the installation is triggered (where it remains unchange).

SUPPLY-UNITS =

The supply units to be parked.

SUPPLY-UNITS = *ALL

Park all supply units of the SOLIS2 delivery.

SUPPLY-UNITS = *BY-DIALOG

Select the selection criteria and the associated supply units interactively (see "[Interactive selection of supply units](#)" ([INSTALL-UNITS Install and register software](#))).

SUPPLY-UNITS = list-poss(30): <text 1..30 without-sep>(…)

Name of the supply unit, see "Supply unit".

VERSION =

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

Use the highest version of the supply unit.

VERSION = *ALL

Use all versions of the supply unit.

VERSION = <product-version without-man-corr>

Version designation without the release and correction state.

CORRECTION-STATE =

Correction state of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Use the highest correction state of the supply unit.

CORRECTION-STATE = *ALL

Use all correction states of the supply unit.

CORRECTION-STATE = *LOWEST-EXISTING

Use the lowest correction state of the supply unit.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

Explicit specification of the correction state of the supply unit. Format: <aso>

TARGET-SYSTEM = *PARAMETERS(…)

Specifications on the target system for which the supply units are to be parked.

VERSION =

BS2000 version of the target system.

VERSION = *CURRENT

Use the current BS2000 version.

VERSION = <product-version without-man-corr>

Explicit specification of the version of BS2000.

Permitted entries: V19.0, V20.0 and V21.0

PUBSET =

HOME pubset of the system for installation.

This specification should only be evaluated if the RMS depot is to be updated (see RMS-DEPOT-UPDATING operand).

PUBSET = *DEFAULT

HOME pubset of the current system.

PUBSET = <cat-id 1..4>

HOME pubset of another target system (foreign system).

OLD-FILE-SAVING =

Save/do not save the files that are overwritten by parking.

OLD-FILE-SAVING = *NO

Do not save the files.

OLD-FILE-SAVING = *WITH-LMS

The files are saved in a library with LMS/LMSCONV .

The name of the save library is \$<work file ID>.IMON.SAVE.LIB.<package name>.<customer ID>, where <package name> is the value of the PACKAGE-NAME operand, <customer ID> is the value of the USER-CODE operand and <work file ID> is the current work file ID setting (see the MODIFY-IMON-OPTIONS statement "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

OLD-FILE-SAVING = *WITH-ARCHIVE(...)

Save the files with ARCHIVE.

VOLUME = <vsn> / *FROM-OPERATOR(...)

VSN of the data volume to which files are to be saved with ARCHIVE.

VOLUME = *FROM-OPERATOR(...)

The MAREN subsystem automatically chooses the VSN of a free volume. If the MAREN subsystem is not loaded, the statement is rejected with the message IMO0449.

LOCATION = *STD / <alphanum-name 1..8>

Name of the location, from which the free volume will be selected. The default is *STD, i.e. the ARCHIVE backup is carried out with the default settings, without MAREN support.

DEVICE-TYPE = <device>

Device type of the data volume.

RMS-DEPOT-UPDATING =

Transfer/do not transfer the RMS delivery set to the RMS depot.

RMS-DEPOT-UPDATING = *NO

Do not transfer the RMS delivery set to the RMS depot.

RMS-DEPOT-UPDATING = *PARAMETERS(...)

Transfer the RMS delivery set to the RMS depot.

DEPOT-LOCATION = *STD

Use the RMS depot with the standard file name <catid>\${<userid>}.RMS.DEPOT, where <catid> is the catalog ID of the pubset of the target system (see PUBSET operand) and <userid> is taken from the delivery information.

DEPOT-LOCATION = <partial-filename 2..18>

The RMS depot with the :<catid>:\${<userid>}. specified in DEPOT-LOCATION and the standard file name RMS.DEPOT are used.

FILE-LOCATION = <partial-filename 2..16>

The location at which the release items are to be parked.

Notes

1. Work files

After the supply units have been successfully parked with no errors, IMON generates a cleanup procedure for the park ID that contains all DELETE-FILE commands for deleting the parked supply units. The park ID can be cleaned using this procedure after the installation of the park ID has been completed:

```
${<park ID>}.IMONDEI.PRK.<name of the supply unit>.<package name>
```

2. The statement is rejected and an error message is displayed in the following cases:

- The user ID specified as the saving location for work files using WORK-FILE-LOCATIONS in the MODIFY-IMON-OPTIONS statement cannot be accessed.
- The user ID specified as the saving location using FILE-LOCATIONS cannot be accessed or has a memory overflow.
- DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM was specified, but the SCI does not contain a SOLIS2 delivery with the specified package name and user code.
- The delivery information does not exist or cannot be accessed:
 - DISTRIBUTION-MEDIUM=*REGISTERED-MEDUM is specified, but the delivery information file does not exist or cannot be accessed.
 - DISTRIBUTION-MEDUM=*SOLIS2-VOLUME/*LOCAL-VOLUME is specified, but the specified data volume or the specified device does not exist or cannot be accessed, or the delivery information file cannot be copied from tape to the current user ID.
- The delivery information cannot be processed successfully by IMON.
- The target system version is not [V]19.0, [V]20.0 and [V]21.0

3. Execution of the park order is aborted if:

- The user ID specified as the storage location cannot be accessed or has a memory overflow.
- A data backup with LMS was requested, however LMS/LMSCONV cannot be accessed.
- A data backup with ARCHIVE was requested, however ARCHIVE cannot be accessed.
- A file that was already parked cannot be overwritten (because it is locked).

4. If DISTRIBUTION-MEDIUM=*LOCAL-VOLUME the delivery has already been saved on another data volume (e. g. virtual volume of a CentricStor). In this case, the specified VSN and device type are used for access to the installation files and the entries in the product movement file are ignored. However, only files from the specified volume can be processed.

-
5. In a heterogeneous disk environment (K and NK disks), the same storage location must be used for work files (WORK-FILE-LOCATION) when installing the park ID as for parking. IMON requires a few work files which were created when parking in order to generate the installation procedure.

4.4.9 PRINT-DOCUMENTATION Print delivery documentation

Privileges: SUBSYSTEM-MANAGEMENT

Function

The PRINT-DOCUMENTATION statement enables you to print delivery documentation. At the same time, the open SCI is updated so that DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM can be selected in a subsequent installation.

You can print the following documentation:

- return letter, confirmation of delivery for system support
- delivery contents, a summary of the delivery
- release notices of the supply units

If the delivery comes from a data volume or a library (DISTRIBUTION-MEDIUM= *SOLIS2-VOLUME / *LOCAL-VOLUME / *LIBRARY), the delivery documentation is created and stored in a library.

Format

PRINT-DOCUMENTATION

DOCUMENTATION = *FROM-SOLIS2-DELIVERY(...)

***FROM-SOLIS2-DELIVERY(...)**

| **PACKAGE-NAME** = <alphanum-name 1..12>

| **,USER-CODE** = <alphanum-name 1..8>

| **,DISTRIBUTION-MEDIUM** = ***REGISTERED-MEDIUM** / ***LIBRARY(...)** / ***SOLIS2-VOLUME(...)**
| /

| ***LOCAL-VOLUME(...)**

| ***LIBRARY(...)**

| | **DOCUMENT-LIBRARY** = ***STD** / <filename 1..54 without-gen-vers>

| | **,REGISTRATION** = ***EXTEND** / ***REPLACE**

| ***SOLIS2-VOLUME(...)**

| | **VOLUME** = <vsn>

| | **,DEVICE-TYPE** = <device>

| | **,DOCUMENT-LIBRARY** = ***STD** / <filename 1..54 without-gen-vers>

| | **,REGISTRATION** = ***EXTEND** / ***REPLACE**

| ***LOCAL-VOLUME(...)**

| | **VOLUME** = <vsn>

| | **,DEVICE-TYPE** = <device>

| | **,DOCUMENT-LIBRARY** = ***STD** / <filename 1..54 without-gen-vers>

| | **,REGISTRATION** = ***EXTEND** / ***REPLACE**

| **,INFORMATION** = ***PARAMETERS(...)**

| ***PARAMETERS(...)**

| | **RETURN-LETTER** = ***YES** / ***NO**

| | **,DELIVERY-CONTENTS** = ***YES** / ***NO**

| | **,RELEASE-NOTICES** = ***YES** / ***NO**

Operands

DOCUMENTATION =

The scope of documentation to be printed.

DOCUMENTATION = *FROM-SOLIS2-DELIVERY(...)

The documentation is from a SOLIS2 delivery.

PACKAGE-NAME = <alphanum-name 1..12>

Package name (as stated in the footer of the delivery contents).

USER-CODE = <alphanum-name 1..8>

User code (as stated in the footer of the delivery contents).

DISTRIBUTION-MEDIUM =

The distribution medium.

DISTRIBUTION-MEDIUM = *REGISTERED-MEDIUM

Read the delivery information from the open SCI.

DISTRIBUTION-MEDIUM = *LIBRARY(...)

Read the delivery information from the PLAM library SOLFTR.<package-name>.<user-code> on the user ID SYSSAG, where <package-name> and <user-code> are the values of the PACKAGE-NAME and USER-CODE operands.

The supply units of the delivery are registered in the open SCI.

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC where <package name> is the value of the PACKAGE-NAME operand, <customer ID> is the value of the USER-CODE operand and <work file ID> the current file location setting or work files (see the MODIFY-IMON-OPTIONS statement "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

Explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI

or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information from the library supplied.

Note

All information on previously installed supply units from this delivery will be lost. These supply units are no longer available for an installation with UNIT-NAME=*FROM-SCI.

DISTRIBUTION-MEDIUM = *SOLIS2-VOLUME(...)

Read the delivery information from a data volume shipped with Solis2. The delivery information is copied from the data volume to the current user ID and registered in the SCI. Existing files with the same name are overwritten.

VOLUME = <vsn>

VSN of the data volume which contains the delivery.

DEVICE-TYPE = <device>

Device type of the data volume.

DOCUMENT-LIBRARY =

Name of the library in which the delivery documentation is stored. If the library already exists, the new delivery documentation is added to it.

DOCUMENT-LIBRARY = *STD

The delivery documentation is stored in the library with the standard name \$<work file ID>.<package name>.<customer ID>.DOC where <package name> is the value of the PACKAGE-NAME operand, <customer ID> is the value of the USER-CODE operand and <work file ID> the current file location setting or work files (see the MODIFY-IMON-OPTIONS statement "[MODIFY-IMON-OPTIONS Modify IMON parameter settings](#)").

DOCUMENT-LIBRARY = <filename 1..54 without-gen-vers>

An explicitly specified library name.

REGISTRATION =

Specifies if the delivery is to be handled as a delivery already registered in the SCI or as a new delivery.

REGISTRATION = *EXTEND

The delivery is handled like a delivery that has already been registered. If there are already entries for this delivery in the SCI, then they are not to be overwritten.

REGISTRATION = *REPLACE

The delivery is handled like a new delivery. If there are already entries for this delivery in the SCI, then they will be recreated using the information read from the data medium.

Note

All information on previously installed supply units from this delivery will be lost.

These supply units are no longer available for an installation with UNIT-

NAME=*FROM-SCI.

DISTRIBUTION-MEDIUM = *LOCAL-VOLUME(...)

Read the delivery information from a data volume created by the customer. The delivery information is copied from the data volume to the current user ID and registered in the SCI. Existing files with the same name are overwritten. For information on the VOLUME, DEVICE-TYPE, DOCUMENT-LIBRARY and REGISTRATION operands refer to the corresponding operand descriptions for DISTRIBUTION-MEDIUM=*SOLIS2-VOLUME (...).

INFORMATION = *PARAMETERS(...)

Defines which documentation will be printed.

RETURN-LETTER = *YES / *NO

Print/do not print the return letter.

DELIVERY-CONTENTS = *YES / *NO

Print/do not print the delivery contents

RELEASE-NOTICES = *YES / *NO

Print/do not print the release notices of the supply units.

Notes

1. Work files

The function creates work files that are stored at the location defined by the WORK-FILE-LOCATION operand of the MODIFY-IMON-OPTIONS statement. These work files are subsequently deleted.

2. Delivery information files

If DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM is specified, the storage location of the distribution medium is taken from the SCI.

If DISTRIBUTION-MEDIUM=*SOLIS2-VOLUME/*LOCAL-VOLUME/*LIBRARY is specified, the delivery information is copied from tape/library to the specified document library.

3. The statement is rejected and an error message is displayed in the following cases:

- The work file ID cannot be accessed or does not have enough memory.
- DISTRIBUTION-MEDIUM=*REGISTERED-MEDIUM was specified and there is no SOLIS2 delivery with the specified package name and customer ID in the SCI.
- The delivery information does not exist or cannot be accessed:
 - DISTRIBUTION-MEDIUM=*REGISTERED-MEDUM was specified, but the delivery information file does not exist or cannot be accessed.
 - DISTRIBUTION-MEDUM=*SOLIS2-VOLUME/*LOCAL-VOLUME was specified, but the specified data volume or the specified device does not exist or cannot be accessed, or the delivery information file cannot be copied from data volume to the current user ID.
- The delivery information cannot be successfully processed by IMON.

4.4.10 REMOVE-INSTALLATION-UNITS Remove installation units from SCI

Privileges: SUBSYSTEM-MANAGEMENT

Function

The REMOVE-INSTALLATION-UNITS statement enables you to remove management information about installation units from the open SCI.

Once installation units have been removed from the SCI, they can no longer be accessed by IMON-GPN. Files or elements associated with the elements removed by this statement are not deleted.

Format

REMOVE-INSTALLATION-UNITS

UNIT-NAME = *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(…)

<text 1..30 without-sep>(…)

| **VERSION = *HIGHEST-EXISTING** / *LOWEST-EXISTING / *ALL / <product-version without-man>

| **,CORRECTION-STATE = *HIGHEST-EXISTING** / *LOWEST-EXISTING / *ALL /

| list-poss(15): <alphanum-name 3..3>

Operands

UNIT-NAME =

The installation units to be selected.

UNIT-NAME = *BY-DIALOG

The installation units are listed in a selection window and can be selected in an interactive dialog (see the section "[Interactive selection of installation units](#)"). Interactive selection is not available in batch mode.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Name of the installation unit, see "[Installation unit \(IU\)](#)".

VERSION =

Version of the installation unit.

VERSION = *HIGHEST-EXISTING

Use the highest version of the installation unit.

VERSION = *LOWEST-EXISTING

Use the lowest version of the installation unit.

VERSION = *ALL

Use all versions of the installation unit.

VERSION = <product-version 3..7 without-man>

Version designation without the release and correction state.

CORRECTION-STATE =

Correction state of the installation unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Use the highest correction state of the installation unit.

CORRECTION-STATE = *LOWEST-EXISTING

Use the lowest correction state of the installation unit.

CORRECTION-STATE = *ALL

Use all correction states of the installation unit

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

Explicit specification of the correction state. Format: <aso>.

Notes

1. The statement is rejected and an error message is displayed in the following cases:

- No write access is permitted for the SCI.
- The value *BY-DIALOG was specified in batch mode.
- The SCI does not contain any of the specified installation units.
- The input list contains duplicate supply units, i.e. units with the same values or key words for one of the identification parameters.

2. The selected installation unit is deleted.

If the installation unit is found in the “Lost+Found” group, it is deleted even if the version and correction state specifications do not match. If you are using the corresponding key words, it is a good idea to check this first by issuing a SHOW statement with the same criteria.

Interactive selection of installation units

If you enter the operand value UNIT-NAME=*BY-DIALOG in the REMOVE-INSTALLATION-UNITS statement, a selection window appears. Further selections can then be made interactively and in dialog boxes. You can select the installation units interactively by entering any character in front of the desired installation unit.

```

IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
Unit name      Installation units selection  Units 1 through 13 of 168
                  Version          Corr state  More: +
ACS             21.0          A00
AID             03.5          A00
AIDSYS         21.0          A00
AIDSYSA        21.0          A00
ANITA          21.0          A00
APACHE         02.4          A00
ARCHIVE        12.0          C00
ASE            21.0          A00
ASSEMBH        01.4          A10
ASSEMBH-GEN    01.4          A10
ASSTRAN        05.4          A00
ASTI           21.0          A00
ATS            05.3          A20
-----
Command ==>
F1=Help F3=Exit F5=Previous F6=Next F7=Backward F8=Forward F10=Menu ...

```

Figure 58: Interactive selection of installation units

Notes

- Interactive selection is only part of the *BY-DIALOG operand value. It is different from the similar looking screen which is displayed when you select the *Open* option in the *File* menu in menu mode.
- Installation units that could not be assigned (no SYSSII file) are indicated by a hash character (#).
- Once you have selected the supply units, press [DUE] to exit the selection and the window. You are then returned to the SDF interface of IMON, where you can enter additional IMON statements.
- Interactive selection is not available in batch mode.

4.4.11 REMOVE-PACKAGES Deleting a software delivery from an SCI

Privilege: SUBSYSTEM-MANAGEMENT

Function Description

With the statement REMOVE-PACKAGES management information about software deliveries, which are no longer needed, can be deleted from the open SCI. Supply units, which are part of a delivery, which is to be deleted, are deleted along with it, if they are in the installation status "Being installed", "In Library", "On SOLIS2 Volume" or "On Local Volume".

Format

REMOVE-PACKAGES

```
PACKAGE-NAME = *BY-DIALOG / list-poss(30): <alphanum-name 1..12>
,USER-CODE = *ALL / <alphanum-name 1..8>
,WORK-FILE-DELETION = *YES / *NO
,OUTPUT = *SYSOUT / *SYSLST(...)
    *SYSLST(...)
        | SYSLST-NUMBER = *STD / <integer 1..99>
```

Operands

PACKAGE-NAME =

Selects the software delivery to be deleted by the package name (as stated in the footer of the supply information). The statement can be limited to a specific customer ID with the operand USER-CODE.

PACKAGE-NAME = *BY-DIALOG

The deliveries are listed in the display window and can be selected interactively (see section "[Interactive selection of deliveries](#)"). Interactive selection is not available in batch mode.

PACKAGE-NAME = list-poss(30): <alphanum-name 1..12>

Package name of the software delivery, see "[Delivery](#)". Up to 30 package names can be stated in a list.

USER-CODE =

Selection of the software deliveries to be displayed by customer ID (as stated in the footer of the supply information).

USER-CODE = *ALL

Output for all customer IDs.

USER-CODE = <alphanum-name 1..8>

Output only for software deliveries with the stated customer ID.

WORK-FILE-DELETION = *YES / *NO

Specifies whether all work files, which are related to a software delivery which is to be deleted, should be deleted. The work files are only deleted after error-free processing.

Work files, which are related to a software delivery, which is to be deleted, are deleted with the following command:

```
/DELETE-FILE <work-file-location>.<package-name> ,  
          IGNORE-PROTECTION=( *ACCESS , *EXPIRATION-DATE )
```

The work file ID (if applicable with prefix) is removed from the product movement file, which is registered in the SCI for the delivery. In the interactive dialog, the DELETE-FILE command is carried out in a control dialog with the user.

OUTPUT =

Specifies the aim of the output.

OUTPUT = *SYSLST(...)

Data output from SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Data output from default SYSLST.

SYSLST-NUMBER = <integer 1..99>

Output of SYSLST number.

Notes

1. In the following cases the statement is rejected without taking any action and an error message is output:
 - The SCI cannot be accessed with write privileges.
 - The value *BY-DIALOG was specified in the batch mode.
 - The SCI does not contain the specified supply units.
 - There are double deliveries entries in the input list, i.e. deliveries with the same package name for one of the identification parameters.
2. The selection criteria are applied to the specified deliveries. When using the corresponding keywords, it is recommended to issue a SHOW statement with the same criteria beforehand as a check.

Interactive selection of deliveries

When the operand UNIT-NAME=*BY-DIALOG is entered in the statement REMOVE-PACKAGES, a window appears. The further selection is interactive and controlled by masks.

The deliveries are interactively selected and the sign selected is entered in front of the desired deliveries.

```

-----
Package name      User code      Package selection      1 through      3 of      3
x <package-name-1> SOL2p
  <package-name-2> SOL2p
  <package-name-3> SOL2P
                        *** End of Package selection ***

-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 59: Interactive selection of deliveries

Notes

- Interactive selection is only part of the *BY-DIALOG operand value. It is different from the similar looking screen which is displayed when you select the *Open* option in the *File* menu in menu mode.
- Once you have selected the deliveries, press [DUE] to exit the selection and the window. You are then returned to the SDF interface of IMON, where you can enter additional IMON statements.
- Interactive selection is not available in batch mode.

4.4.12 REMOVE-SUPPLY-UNITS Remove supply units from SCI

Privileges: SUBSYSTEM-MANAGEMENT

Function

The administration information on supply units not needed anymore can be deleted from the open SCI with the REMOVE-SUPPLY-UNITS statement. Supply units in the "Parked" or "Installed" installation state cannot be deleted.

Format

REMOVE-INSTALLATION-UNITS

UNIT-NAME = *BY-DIALOG / list-poss(30): <text 1..30 without-sep>(…)

<text 1..30 without-sep>(…)

| **VERSION = *HIGHEST-EXISTING** / *LOWEST-EXISTING / *ALL / <product-version without-man>

| **,CORRECTION-STATE = *HIGHEST-EXISTING** / *LOWEST-EXISTING / *ALL /

| list-poss(15): <alphanum-name 3..3>

Operands

UNIT-NAME =

Selection of the supply units to be deleted.

UNIT-NAME = *BY-DIALOG

The supply units can be listed in a selection window and selected in an interactive dialog (see ["Interactive selection of supply units"](#)). Interactive selection is not available in the batch mode.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(…)

Name of the supply unit, see ["Installation unit \(IU\)"](#). Up to 30 supply units can be specified in a list.

VERSION =

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

The highest version of the supply unit will be used.

VERSION = *LOWEST-EXISTING

The lowest version of the supply unit will be used.

VERSION = *ALL

All versions of the supply unit will be used.

VERSION = <product-version 3..7 without-man>

Version designation without the release and correction status.

CORRECTION-STATE =

Correction state of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

The highest correction state of the supply unit will be used.

CORRECTION-STATE = *LOWEST-EXISTING

The lowest correction state of the supply unit will be used.

CORRECTION-STATE = *ALL

All correction states of the supply unit will be used.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

The correction state of the supply unit is specified explicitly.

Format: <aso>

Notes

1. In the following cases the statement is rejected without taking any action and an error message is output:
 - The SCI cannot be accessed with write privileges.
 - The value *BY-DIALOG was specified in the batch mode.
 - The SCI does not contain the specified supply units.
 - There are double supply unit entries in the input list, i.e. supply units with the same values or keywords for one of the identification parameters.
 - One of the selected supply units is in the "Installed" or "Parked" state.
2. The selection criteria are applied to the specified supply units. When using the corresponding keywords, it is recommended to issue a SHOW statement with the same criteria beforehand as a check.

Interactive selection of supply units

If the operand value UNIT-NAME=*BY-DIALOG is entered in the REMOVE-SUPPLY-UNITS statement, then a selection window appears. Additional supply units are selected interactively in a dialog. The supply units are interactively selected by entering any character in front of the desired supply unit.

IMON: SCI: :I29A:\$TSOS.SYS.IMON.SCI

```
-----  
Units      1 through      13 of      168  
SU selection  
Unit name  Vers Corr Package name  User code  Inst. Status  CAP Act  
ACS        21.0 A00 <package-name> SOL2P    Installed    N   Y  
AID        03.5 A00 <package-name> SOL2P    Installed    N   Y  
AIDSYS     21.0 A00 <package-name> SOL2P    Installed    N   Y  
AIDSYSA    21.0 A00 <package-name> SOL2P    Installed    N   Y  
ANITA      21.0 A00 <package-name> SOL2P    Installed    N   Y  
APACHE     02.4 A00 <package-name> SOL2P    Installed    N   Y  
ARCHIVE    12.0 C00 <package-name> SOL2P    Installed    N   Y  
ASE        21.0 A00 <package-name> SOL2P    Installed    N   Y  
ASSEMBH    01.4 A10 <package-name> SOL2P    Installed    N   Y  
ASSEMBH-GEN 01.4 A10 <package-name> SOL2P    Installed    N   Y  
ASSTRAN    05.4 A00 <package-name> SOL2P    Installed    N   Y  
ASTI       21.0 A00 <package-name> SOL2P    Installed    N   Y  
ATS        05.3 A20 <package-name> SOL2P    Installed    N   Y  
-----  
Command ==>  
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...
```

Figure 60: Interactive selection of supply units

Notes

- After the supply units have been selected, the selection process and the window are terminated with [DUE]. The user is then back in IMON's SDF interface and additional IMON statements can be entered.
- Interactive selection is not available in the batch mode.

4.4.13 REQUEST-CORRECTION-DELIVERY Request correction deliveries for registered supply units

Privileges: SUBSYSTEM-MANAGEMENT

Function

The REQUEST-CORRECTION-DELIVERY statement enables you to request new correction versions for a supply unit registered in the SCI. The scope of the request is defined using the DELIVERY-SCOPE operand.

- Delta delivery with only the modified release items of the supply unit (default)
- Correction delivery for the entire supply unit
- Only information on the available corrections

Using the parameter file for mail configuration (see "[Parameter file for mail configuration](#)") and the user input, IMON creates a correction request (with the file name \$TSOS.IMON.DELREQ.<timestamp>) that is sent by mail to the software distribution center of Fujitsu.

Format

REQUEST-CORRECTION-DELIVERY

DELIVERY-SCOPE = *MODIFIED-ITEMS / *ALL-ITEMS / *INFORMATION-ONLY

,USER-CODE = <alphanum-name 1..8>

,SUPPLY-UNITS = *BY-DIALOG / list-poss(30): <text 1..15 without-sep>(…)

<text 1..30 without-sep>(…)

| **VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>**

,DISTRIBUTION-MEDIUM = *LIBRARY(…) / *SOLIS2-VOLUME(…)

*LIBRARY(…)

| **DELIVERY-METHOD = *FTP-VIA-WWW / *OPENFT-VIA-WWW / *FT-BS2000**

*SOLIS2-VOLUME(…)

| **DEVICE-TYPE = <device>**

,DELIVERY-DATE = *STD / <date with-compl>(…)

<date with-compl>(…)

| **TIME = *ANY / <time>**

,SEND-MAIL = *IMMEDIATELY(…) / *BY-USER

*IMMEDIATELY(…)

| **SMTP-SERVER = *LOCAL / <c-string 1..256>**

Operands

DELIVERY-SCOPE =

Specifies the scope of the correction request.

DELIVERY-SCOPE = *MODIFIED-ITEMS

Requests a correction delivery that contains only the modified release items of the supply unit.

DELIVERY-SCOPE = *ALL-ITEMS

Requests a correction delivery for the entire supply unit.

DELIVERY-SCOPE = *INFORMATION-ONLY

Requests only information on corrections that are available for the supply unit.

USER-CODE = <alphanum-name 1..8>

Customer ID for which the correction delivery is to be made.

SUPPLY-UNITS =

Selection of the supply units for which corrections are being requested.

SUPPLY-UNITS = *BY-DIALOG

The supply units can be listed in a selection window and selected in an interactive dialog (see "[Interactive selection of supply units](#)"). Interactive selection is not available in the batch mode.

SUPPLY-UNITS = list-poss(30): <text 1..15 without-sep>(…)

Name of the supply unit, see "[Installation unit \(IU\)](#)". Up to 30 supply units can be specified in a list.

VERSION = *HIGHEST-EXISTING / <product-version without-man-corr>

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

The highest version of the supply unit will be used.

VERSION = <product-version without-man-corr>

Version designation without the release and correction status.

DISTRIBUTION-MEDIUM =

Specifies the distribution medium for the requested corrections.

DISTRIBUTION-MEDIUM = *LIBRARY(…)

Distribution takes place in a PLAM library.

DELIVERY-METHOD = *FTP-VIA-WWW / *OPENFT-VIA-WWW / *FT-BS2000

Specifies how the supply library is to be transferred to the customer.

DELIVERY-METHOD = *FTP-VIA-WWW

The correction delivery is to be made available to the customer on the WWW for collection using FTP.

DELIVERY-METHOD = *OPENFT-VIA-WWW

The correction delivery is to be made available to the customer on the WWW for collection using *openFT*.

DELIVERY-METHOD = *FT-BS2000

Only for customers with a direct BS2000 network connection

The correction delivery is to be transferred directly to the customer system using *openFT*.

DISTRIBUTION-MEDIUM = *SOLIS2-VOLUME(…)

The correction delivery is to be distributed on a data volume.

DEVICE-TYPE = <device>

Device type of the data volume to be delivered.

DELIVERY-DATE = *STD / <date with-compl>(…)

Specifies the desired delivery date.

DELIVERY-DATE = *STD

The correction delivery should be distributed or made available within two working days. WWW deliveries are generally made available one or two hours after request receipt.

DELIVERY-DATE = <date with-compl>(…)

Specifies the desired delivery or provision date.

TIME = *ANY / <time>

Time specification.

SEND-MAIL = *IMMEDIATELY(…)/ *BY-USER

Specifies how the mail with the correction request is to be sent to the software distribution center of Fujitsu.

SEND-MAIL = *IMMEDIATELY(…)

The mail is sent automatically and immediately. A prerequisite is use of the INETSERV software Version 3.1 or higher and entry of the necessary mail addresses in the parameter file for mail configuration (see "[Parameter file for mail configuration](#)").

SMTP-SERVER = *LOCAL / <c-string 1..256>

Name of the mail server via which mail is to be sent. Default is *LOCAL (local mail server).

SEND-MAIL = *BY-USER

The mail is sent manually by the customer.

Notes

1. In the following situations the statement is rejected, no action is taken, and an error message is output.
 - The value *BY-DIALOG was specified in batch mode.
 - The SCI does not contain all the specified supply units.
 - There are duplicate supply units in the input list, i.e. units with the same values or key words for one of the identification parameters.
 - There is no parameter file for mail configuration for the specified customer ID (see section "[Parameter file for mail configuration](#)") or the parameter file is not complete.
2. If the automatic sending of mail fails for SEND-MAIL=*IMMEDIATELY, the system administrator is prompted to manually send the generated correction request with the file name \$TSOS.IMON.DELREQ.<timestamp>. A second file with the same name and the suffix ".LOG" contains further information on the failed send attempt.

Interactive selection of supply units

If the operand value SUPPLY-UNITS=*BY-DIALOG is entered in the REQUEST-CORRECTION-DELIVERY statement, then a selection window appears. Additional supply units are selected interactively in a dialog. The supply units are interactively selected by entering any character in front of the desired supply unit.

```

IMON: SCI: :I29A:$TSOS.SYS.IMON.SCI
-----
                Units      1 through      13 of      168
                SU selection      More:      +
Unit name      Vers Corr Package name      User code      Inst. Status      CAP Act
ACS            21.0 A00 <package-name> SOL2P      Installed              N      Y
AID            03.5 A00 <package-name> SOL2P      Installed              N      Y
AIDSYS        21.0 A00 <package-name> SOL2P      Installed              N      Y
AIDSYSA       21.0 A00 <package-name> SOL2P      Installed              N      Y
ANITA         21.0 A00 <package-name> SOL2P      Installed              N      Y
APACHE        02.4 A00 <package-name> SOL2P      Installed              N      Y
ARCHIVE       12.0 C00 <package-name> SOL2P      Installed              N      Y
ASE           21.0 A00 <package-name> SOL2P      Installed              N      Y
ASSEMBH       01.4 A10 <package-name> SOL2P      Installed              N      Y
ASSEMBH-GEN   01.4 A10 <package-name> SOL2P      Installed              N      Y
ASSTRAN       05.4 A00 <package-name> SOL2P      Installed              N      Y
ASTI          21.0 A00 <package-name> SOL2P      Installed              N      Y
ATS           05.3 A20 <package-name> SOL2P      Installed              N      Y
-----
Command ==>
F1=Help  F3=Exit  F5=Previous  F6=Next  F7=Backward  F8=Forward  F10=Menu ...

```

Figure 61: Interactive selection of supply units

Notes

- After the supply units have been selected, the selection process and the window are terminated with [DUE]. The user is then back in IMON’s SDF interface and additional IMON statements can be entered.
- Interactive selection is not available in the batch mode.

4.4.14 RESET-CUSTOMER-APPROVAL Reset customer approval ID of a supply unit

Privileges: SUBSYSTEM-MANAGEMENT

Function

The RESET-CUSTOMER-APPROVAL statement enables the customer approval ID to be explicitly reset for a specific version of a supply unit registered in the SCI. The ID is reset for the most recently installed correction status of the specified version. This version of the supply unit is then ignored when installing released supply units.

IMON implicitly resets the customer approval ID of a supply unit if an installation item of the “released” supply unit is changed during installation.

Format

RESET-CUSTOMER-APPROVAL

```
SUPPLY-UNITS = *BY-DIALOG / list-poss(30): <c-string 1..15>(…) / <text 1..15 without-sep>(…)
    <c-string 1..15>(…)
        |   VERSION = <product-version without-man-corr>
    <text 1..30 without-sep>(…)
        |   VERSION = <product-version without-man-corr>
```

Operands

SUPPLY-UNITS =

Selection of the supply units for which the customer approval ID is to be removed.

SUPPLY-UNITS = *BY-DIALOG

All released supply units of the opened SCI are listed in a selection window and can be selected in an interactive dialog (see ["Interactive selection of supply units" \(REMOVE-SUPPLY-UNITS Remove supply units from SCI \)](#)).

Interactive selection is not available in the batch mode.

SUPPLY-UNITS = list-poss(30): <c-string 1..15>(…) / <text 1..15 without-sep>(…)

Name of the supply unit, see ["Installation unit \(IU\)"](#). Up to 30 supply units can be specified in a list.

VERSION = <product-version without-man-corr>

Version of the supply unit (version designation without the release and correction status).

Notes

- The customer approval ID cannot be reset for a specific correction status of a supply unit but only for a specific version.
- The creation and modification date of the supply unit stored in the SCI are retained unchanged when the customer approval ID is reset.
- The statement is terminated without error if a non-released version of a supply unit is specified.

4.4.15 SAVE-SOFTWARE-INVENTORY Copy current SCI

Privileges: SUBSYSTEM-MANAGEMENT

Function

The SAVE-SOFTWARE-INVENTORY statement enables you to make a copy of the current SCI during an IMON run. This function creates consistent backups for the two physical SCI files (IMON-SCI and GPN-SCI).

You can specify an explicit name for the backup. The corresponding IMON-GPN-SCI is saved under the same file name with the suffix “.GPN”.

Format

SAVE-SOFTWARE-INVENTORY

TO-SCI = <u>*STD</u> / <filename 1..50>
--

Operands

TO-SCI =

Name of the target file (backup copy).

TO-SCI = *STD

Use the file name of the currently open SCI suffixed by the time stamp (date and time) in the form <yyyymmddhhmmss>.

TO-SCI = <filename 1..50>

Explicit specification of the name of the target file.

Note

The statement is rejected and an error message is displayed if the length of the SCI name, including the catalog ID and user ID, exceeds 50 characters.

4.4.16 SET-CUSTOMER-APPROVAL Set customer approval ID for a supply unit

Privileges: SUBSYSTEM-MANAGEMENT

Function

The SET-CUSTOMER-APPROVAL statement lets you set a customer approval ID for a specific version of a supply unit registered in the SCI. The approval ID is set for the most recently installed correction status of the specified version. If appropriate, the approval IDs of previously installed versions of the supply unit are implicitly reset.

The “released” version of the supply unit is now available for installation of released supply units.

The customer approval ID can be explicitly reset using the RESET-CUSTOMER-APPROVAL statement. IMON implicitly resets the approval ID of a supply unit if an installation item of the “released” supply unit is changed for an installation.

Format

SET-CUSTOMER-APPROVAL

```
SUPPLY-UNITS = *BY-DIALOG / list-poss(30): <c-string 1..15>(…) / <text 1..15 without-sep>(…)
```

```
  <c-string 1..15>(…)
```

```
    |   VERSION = <product-version without-man-corr>
```

```
  <text 1..30 without-sep>(…)
```

```
    |   VERSION = <product-version without-man-corr>
```

Operands

SUPPLY-UNITS =

Selection of the supply units that are to be given a customer approval ID.

SUPPLY-UNITS = *BY-DIALOG

The supply units can be listed in a selection window and selected in an interactive dialog

(see ["Interactive selection of supply units" \(REMOVE-SUPPLY-UNITS Remove supply units from SCI \)](#)). Interactive selection is not available in the batch mode.

SUPPLY-UNITS = list-poss(30): <c-string 1..15>(…) / <text 1..30 without-sep>(…)

Name of the supply unit, see ["Installation unit \(IU\) "](#). Up to 30 supply units can be specified in a list.

VERSION = <product-version without-man-corr>

Version of the supply unit (version designation without the release and correction status).

Notes

- The customer approval ID cannot be reset for a specific correction status of a supply unit but only for a specific version.

-
- The creation and modification date of the supply unit stored in the SCI are retained unchanged when the customer approval ID is set.

4.4.17 SHOW-FORMATTED-FILE Show formatted file

Privileges: STD-PROCESSING

Function

The SHOW-FORMATTED-FILE statement enables you to show information contained in a formatted file in a structured way.

The advantage of a formatted file is that the output information from a number of SHOW statements can be collected in a file. The formatted file can then be output.

For example, a user specifies the SHOW-INSTALLATION-UNITS INSTALLATION-ITEMS=*YES statement with output to a formatted file. The result file is specified as an input and output file in another SHOW statement in order to show the installation items. The formatted file generated from this contains all the information about installation units and installation items. The contents of the formatted file can be output to SYSLST using the SHOW-FORMATTED-FILE statement.

Format

SHOW-FORMATTED-FILE

FROM-FILE = <filename1..54 without-gen-vers>

,OUTPUT = *SYSLST(...)

*SYSLST(...)

| **SYSLST-NUMBER** = *STD / <integer 1..99>

Operands

FROM-FILE = <filename 1..54 without-gen-vers>

Name of the formatted file.

OUTPUT =

The target for output.

OUTPUT = *SYSLST(...)

Output the information to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specifies the SYSLST number.

Notes

1. The statement is rejected without action and an error message is displayed if the input file does not exist or cannot be accessed.

-
2. The statement is aborted and a message is displayed if a format error occurs.
 3. If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

4.4.18 SHOW-INSTALLATION-ITEMS Show installation items

Privileges: STD-PROCESSING, SUBSYSTEM-MANAGEMENT

Function

The SHOW-INSTALLATION-ITEMS statement enables you to show information about the installation items in the open SCI (see ["Installation item \(I\)"](#)).

If you are a nonprivileged user, this statement only shows the path names for which DMS access is possible.

The statement supports structured output in S variables (see ["Output in S variables"](#) ["SHOW-INSTALLATION-ITEMS Show installation items"](#)).

Format

SHOW-INSTALLATION-ITEMS

ITEM-NAME = ***BY-INSTALLATION-PATH**(...) / ***FROM-FORMATTED-FILE**(...) /

list-poss(30): <filename 1..30 without-cat-user-gen-vers>(…)

***BY-INSTALLATION-PATH**(…)

| **PATH-NAME** = <filename 1..54 without-gen-vers>

***FROM-FORMATTED-FILE**(…)

| **FILE-NAME** = <filename 1..54 without-gen-vers>

<filename 1..30 without-cat-user-gen-vers>(…)

| **VERSION** = ***ALL** / ***HIGHEST-EXISTING** / <c-string 1..5>

| **,UNIT-NAME** = ***ALL** / list-poss(5): <text 1..30 without-sep>(…)

| <text 1..30 without-sep>(…)

| | **VERSION** = ***ALL** / ***HIGHEST-EXISTING** / <product-version without-man-corr>

| | **,CORRECTION-STATE** = ***ALL** / ***HIGHEST-EXISTING** / ***LOWEST-EXISTING**

| | list-poss(3): <alphanum-name 3..3>

,INFORMATION = ***PARAMETERS**(…)

***PARAMETERS**(…)

| **REPORT-LEVEL** = ***MINIMUM** / ***ALL-ATTRIBUTES**

,OUTPUT = ***SYSOUT** / ***SYSLST**(…) / ***INPUT-FORMATTED-FILE** / <filename 1..54 without-gen-vers>(…) / ***NONE**

***SYSLST**(…)

| **SYSLST-NUMBER** = ***STD** / <integer 1..99>

<filename 1..54 without-gen-vers>(…)

| **WRITE-MODE** = ***REPLACE** / ***EXTEND**

,STRUCTURE-OUTPUT = ***NONE** / ***SYSINF** / <composed-name 1..255>(…)

<composed-name 1..255>(…)

| **WRITE-MODE** = ***REPLACE** / ***EXTEND**

Operands

ITEM-NAME =

Select the installation item.

ITEM-NAME = *BY-INSTALLATION-PATH(...)

The item is already installed and will be selected by its path name.

PATH-NAME = <filename 1..54 without-gen-vers>

Path name of the installation item.

ITEM-NAME = *FROM-FORMATTED-FILE(...)

The installation items will be taken from the formatted file.

FILE-NAME = <filename 1..54 without-gen-vers>

Name of the formatted file.

ITEM-NAME = list-poss(30): <filename 1..30 without-cat-user-gen-vers>(...)

Name of the installation item, see "[Installation item \(II\)](#)".

VERSION =

Version of the installation item.

VERSION = *ALL

Select all versions of the installation item.

VERSION = *HIGHEST-EXISTING

Select the highest version of the installation item.

VERSION = <c-string 1..5>

Explicit specification of the version of the installation item.

UNIT-NAME =

Name of the installation unit searched for the installation item.

UNIT-NAME = *ALL

Search for the installation item in all registered installation units.

UNIT-NAME = list-poss(5): <text 1..30 without-sep>(...)

Specifies the installation unit searched for the installation item, see "[Installation item \(II\)](#)".

VERSION =

Version of the installation unit.

VERSION = *ALL

Search all versions of the installation unit.

VERSION = *HIGHEST-EXISTING

Search the highest version of the installation unit.

VERSION = <product-version without-man-corr>

Version designation without release and correction states.

CORRECTION-STATE =

Correction state of the specified installation unit.

CORRECTION-STATE = *ALL

Search all correction states of the installation unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Search the highest correction state of the installation unit.

CORRECTION-STATE = *LOWEST-EXISTING

Search the lowest correction state of the installation unit.

CORRECTION-STATE = list-poss(3): <alphanum-name 3..3>

Explicit specification of the correction state of the installation unit.

Format: <aso>.

INFORMATION = *PARAMETERS(...)

The scope of information to be shown.

REPORT-LEVEL =

The reporting level for installation item attributes.

REPORT-LEVEL = *MINIMUM

Show the names of the installation items.

REPORT-LEVEL = *ALL-ATTRIBUTES

Show the names and attributes of the installation items.

OUTPUT =

The target for output.

OUTPUT = *SYSOUT

Output the information to SYSOUT.

OUTPUT = *INPUT-FORMATTED-FILE

Output the information to the formatted file specified in the operand

ITEM-NAME=*FROM-FORMATTED-FILE(FILE-NAME=).

OUTPUT = *NONE

No output is generated except structured output when requested.

OUTPUT = *SYSLST(...)

Output the information to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specifies the SYSLST number.

OUTPUT = <filename 1..54 without-gen-vers>(…)

Output the information to the formatted file. Use the SHOW-FORMATTED-FILE statement to output the collected information in a formatted file.

WRITE-MODE =

Write or extend the file.

WRITE-MODE = *REPLACE

Generate the formatted file or overwrite an existing file.

WRITE-MODE = *EXTEND

Generate the formatted file or extend an existing file.

STRUCTURE-OUTPUT =

Target of structured output when requested.

STRUCTURE-OUTPUT = *NONE

No structured output is requested.

STRUCTURE-OUTPUT = *SYSINF

The SYSINF stream is to be used for structured output.

STRUCTURE-OUTPUT = <composed-name 1..255>(…)

Name of the S-variable (of the “list” type) to be used for the structured output.

WRITE-MODE = *REPLACE

The S-variable is created or overwritten if already existing.

WRITE-MODE = *EXTEND

S-variable is completed with the new information.

Notes

1. Input

- If ITEM-NAME=*FROM-FORMATTED-FILE is specified, the ID of the objects to be shown is taken from the formatted file records that contain an identifier for the desired object type.
- For the sake of consistency, we recommend that you only specify a SHOW statement for a formatted file if it was generated with the same SCI status. However, if the information in the SCI has changed since the formatted file was created, the information that is output is not longer current.

2. Output

- If the user does not have the SUBSYSTEM-MANAGEMENT privilege, the path name of the installation items is only output when the user has DMS access to it (e.g. SHOW-FILE possible). Otherwise, an asterisk (*) is output.
 - If the output is directed to SYSOUT or SYSLST, the information is edited so as to be legible.
 - If the information is directed to a formatted file, the output is compressed and is not legible for the user.
3. In the following cases execution of the statement is continued with a warning:
- An object, for which information is to be supplied, does not exist in the SCI.
 - If, when displaying an installation item, the file name returned by the path name manager is not the same as that specified during registration, both file names are displayed.
4. In the following cases, a warning is issued and execution of the statement resumes:
- The explicitly specified alphanumeric correction state is not a letter followed by two numbers.
 - ITEM-NAME=*FROM-FORMATTED-FILE was specified, but the corresponding file does not exist or cannot be accessed.
 - OUTPUT=*INPUT-FORMATTED-FILE was specified, but the ITEM-NAME parameter is not *FROM-FORMATTED-FILE.
 - There are duplicate units in the input list, i.e. units with the same values or key words for one of the identification parameters.

Output in S variables

Output information	Name of the S variable	T	Contents
current container access right (ACCESS)	var(*LIST).II-CONTAINER(*LIST). CURRENT-ACCESS	S	Read/Write
current container access control list	var(*LIST).II-CONTAINER(*LIST). CURRENT-ACL	S	Yes/No
current container backup class	var(*LIST).II-CONTAINER(*LIST). CURRENT-BACK-CLASS	S	<c-string 1..1>
current container basic access control list	var(*LIST).II-CONTAINER(*LIST). CURRENT-BASIC-ACL	S	<c-string 1..9>
current container block control information	var(*LIST).II-CONTAINER(*LIST). CURRENT-BLK-CONTR	S	None / No / Pamkey / Data / Data-2K / Data-4K
current container change date	var(*LIST).II-CONTAINER(*LIST). CURRENT-CHANGE-DATE	S	yyyy-mm-dd
current container change time	var(*LIST).II-CONTAINER(*LIST). CURRENT-CHANGE-TIME	S	hh:mm:ss

current container creation date	var(*LIST).II-CONTAINER(*LIST). CURRENT-CREA-DATE	S	yyyy-mm-dd
current container creation time	var(*LIST).II-CONTAINER(*LIST). CURRENT-CREA-TIME	S	hh:mm:ss
current container migration identifier	var(*LIST).II-CONTAINER(*LIST). CURRENT-MIGRATE	S	Allowed / Special / Inhibit
current container access right (USER-ACCESS)	var(*LIST).II-CONTAINER(*LIST). CURRENT-USER-ACC	S	All / Special / Owner-Only
initial container access right (ACCESS)	var(*LIST).II-CONTAINER(*LIST). INIT-ACCESS	S	Read/Write
initial container access control list	var(*LIST).II-CONTAINER(*LIST).INIT-ACL	S	Yes/No
initial container backup class	var(*LIST).II-CONTAINER(*LIST). INIT-BACK-CLASS	S	<c-string 1..1>
initial container basic access control list	var(*LIST).II-CONTAINER(*LIST) .INIT-BASIC-ACL	S	<c-string 1..9>
initial container block control information	var(*LIST).II-CONTAINER(*LIST). INIT-BLK-CONTR	S	None / No / Pamkey / Data / Data-2K / Data-4K
initial container change date	var(*LIST).II-CONTAINER(*LIST). INIT-CHANGE-DATE	S	yyyy-mm-dd
initial container change time	var(*LIST).II-CONTAINER(*LIST). INIT-CHANGE-TIME	S	hh:mm:ss
initial container creation date	var(*LIST).II-CONTAINER(*LIST). INIT-CREA-DATE	S	yyyy-mm-dd
initial container creation time	var(*LIST).II-CONTAINER(*LIST). INIT-CREA-TIME	S	hh:mm:ss
initial container migration identifier	var(*LIST).II-CONTAINER(*LIST). INIT-MIGRATE	S	Allowed / Special / Inhibit
initial container access right (USER-ACCESS)	var(*LIST).II-CONTAINER(*LIST). INIT-USER-ACC	S	All / Special / Owner-Only

current access right (ACCESS)	var(*LIST).II-CURR-ACCESS	S	Read/Write
current access control list	var(*LIST).II-CURR-ACL	S	Yes/No
current backup class	var(*LIST).II-CURR-BACK-CLASS	S	<c-string 1..1>
current basic access control list	var(*LIST).II-CURR-BASIC-ACL	S	<c-string 1..9>
current block control information	var(*LIST).II-CURR-BLK-CONTR	S	None / No / Pamkey / Data / Data-2K / Data-4K
current change date	var(*LIST).II-CURR-CHANGE-DATE	S	yyyy-mm-dd
current change time	var(*LIST).II-CURR-CHANGE-TIME	S	hh:mm:ss
current creation date	var(*LIST).II-CURR-CREA-DATE	S	yyyy-mm-dd
current creation time	var(*LIST).II-CURR-CREA-TIME	S	hh:mm:ss
current migration identifier	var(*LIST).II-CURR-MIGRATE	S	Allowed / Special / Inhibit
current access right (USER-ACCESS)	var(*LIST).II-CURR-USER-ACC	S	All / Special / Owner-Only
initial access right (ACCESS)	var(*LIST).II-INIT-ACCESS	S	Read/Write
initial access control list	var(*LIST).II-INIT-ACL	S	Yes/No
initial backup class	var(*LIST).II-INIT-BACK-CLASS	S	<c-string 1..1>
initial basic access control list	var(*LIST).II-INIT-BASIC-ACL	S	<c-string 1..9>
initial block control information	var(*LIST).II-INIT-BLK-CONTR	S	None / No / Pamkey / Data / Data-2K / Data-4K
initial container change date	var(*LIST).II-INIT-CHANGE-DATE	S	yyyy-mm-dd
initial change time	var(*LIST).II-INIT-CHANGE-TIME	S	hh:mm:ss
initial creation date	var(*LIST).II-INIT-CREA-DATE	S	yyyy-mm-dd
initial creation time	var(*LIST).II-INIT-CREA-TIME	S	hh:mm:ss
initial migration identifier	var(*LIST).II-INIT-MIGRATE	S	Allowed / Special / Inhibit
initial access right (USER-ACCESS)	var(*LIST).II-INIT-USER-ACC	S	All / Special / Owner-Only

change date	var(*LIST).II-CHANGE-DATE	S	yyyy-mm-dd
change time	var(*LIST).II-CHANGE-TIME	S	hh:mm:ss
current container path	var(*LIST).II-CONTAINER(*LIST). CURRENT-PATH	S	<filename 1..54>
current container path status flag	var(*LIST).II-CONTAINER(*LIST). CURRENT-PATH-PLACED	B	FALSE / TRUE
initial container path name	var(*LIST).II-CONTAINER(*LIST).INIT-PATH	S	<filename 1..54>
initial container path status flag	var(*LIST).II-CONTAINER(*LIST). .INIT-PATH-PLACED	B	FALSE / TRUE
creation date	var(*LIST).II-CREA-DATE	S	yyyy-mm-dd
creation time	var(*LIST).II-CREA-TIME	S	hh:mm:ss
current path name	var(*LIST).II-CURR-PATH	S	<filename 1..54>
current path status flag	var(*LIST).II-CURR-PATH-PLACED	B	FALSE / TRUE
function level	var(*LIST).II-FU-LEVEL	S	BOTH / TPR / TU
initial path	var(*LIST).II-INIT-PATH	S	<filename 1..54>
initial path status flag	var(*LIST).II-INIT-PATH-PLACED	B	FALSE / TRUE
name of installation unit	var(*LIST).II-IU-NAME	S	<text 1..30>
version of installation unit	var(*LIST).II-IU-VERSION	S	<product-version 7..7>
logical name	var(*LIST).II-LOGICAL-ID	S	<filename 1..30>
name of the installation item	var(*LIST).II-NAME	S	<filename 1..30>
hardware variant	var(*LIST).II-TARGET	S	A/S/R/P/U
item type	var(*LIST).II-TYPE	S	<c-string 1..3>
user ID	var(*LIST).II-USID	S	<name 1..8>
version of the installation item	var(*LIST).II-VERSION	S	<c-string 1..5>

4.4.19 SHOW-INSTALLATION-UNITS Show installation units

Privileges: STD-PROCESSING, SUBSYSTEM-MANAGEMENT

Function

The SHOW-INSTALLATION-UNITS statement enables you to show information about the installation units in the open SCI (see ["Installation unit \(IU\)"](#)).

Possible inputs are an installation unit or a formatted file generated earlier by IMON. In the dialog mode installation units can also be selected interactively with the operand UNIT-NAME=*BY-DIALOG .

If you are a nonprivileged user, this statement only shows the path names for which DMS access is possible.

The statement supports structured output in S variables (see ["Output in S variables"](#) ["SHOW-INSTALLATION-UNITS Show installation units"](#)).

Format

SHOW-INSTALLATION-UNITS

UNIT-NAME = ***ALL** / ***BY-DIALOG** / ***FROM-FORMATTED-FILE(...)** / list-poss(30): <text 1..30 without-sep>
(...)

***FROM-FORMATTED-FILE(...)**

| **FILE-NAME** = <filename 1..54 without-gen-vers>

<text 1..30 without-sep>(…)

| **VERSION** = ***ALL** / ***HIGHEST-EXISTING** / <product-version without-man-corr>

| **,CORRECTION-STATE** = ***ALL** / ***HIGHEST-EXISTING** / ***LOWEST-EXISTING** / list-poss(15):
<alphanum-name 3..3>

,INFORMATION = ***PARAMETERS(...)**

***PARAMETERS(...)**

| **INSTALLATION-ITEMS** = ***YES** / ***NO**

| **,REPORT-LEVEL** = ***MINIMUM** / ***ALL-ATTRIBUTES**

,OUTPUT = ***SYSOUT** / ***SYSLST(...)** / ***INPUT-FORMATTED-FILE** / <filename 1..54 without-gen-vers>(…)
***NONE**

***SYSLST(...)**

| **SYSLST-NUMBER** = ***STD** / <integer 1..99>

<filename 1..54 without-gen-vers>(…)

| **WRITE-MODE** = ***REPLACE** / ***EXTEND**

,STRUCTURE-OUTPUT = ***NONE** / ***SYSINF** / <composed-name 1..255>(…)

<composed-name 1..255>(…)

WRITE-MODE = ***REPLACE** / ***EXTEND**

Operands

UNIT-NAME =

Name of the installation unit to be shown.

UNIT-NAME = ***ALL**

Show all installation units registered in the SCI.

UNIT-NAME = ***BY-DIALOG**

The supply units are listed in a selection window and can be selected interactively in a dialog (see also "Selection of installation units" in the menu mode on ["View menu"](#)). The information requested is output after the installation units have been selected by entering any character in front of the corresponding installation units and the window has been closed with [DUE]. Interactive selection is not available in the batch mode.

UNIT-NAME = *FROM-FORMATTED-FILE(...)

Take the names of the installation units from the formatted file.

FILE-NAME = <filename 1..54 without-gen-vers>

Name of the formatted file containing the list of the installation units.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(...)

Name of the installation unit, see ["Installation unit \(IU\)"](#).

VERSION =

Version of the installation unit.

VERSION = *ALL

Select all versions of the installation unit.

VERSION = *HIGHEST-EXISTING

Select the highest version of the installation unit.

VERSION = <product-version without-man-corr>

Version designation without release and correction states.

CORRECTION-STATE =

Correction state of the installation unit.

CORRECTION-STATE = *ALL

Select all correction states of the installation unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Select the highest correction state of the installation unit.

CORRECTION-STATE = *LOWEST-EXISTING

Select the lowest correction state of the installation unit.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

The correction state of the installation unit is specified explicitly.

Format: <aso>

INFORMATION = *PARAMETERS(...)

The scope of information to be shown.

INSTALLATION-ITEMS = *YES / *NO

Show/do not show the installation items in the installation units.

REPORT-LEVEL =

The reporting level for the installation unit attributes.

REPORT-LEVEL = *MINIMUM

Show the names of the installation units only.

REPORT-LEVEL = *ALL-ATTRIBUTES

Show the names and attributes of the installation units.

OUTPUT =

The target for output.

OUTPUT = *SYSOUT

Output the information to SYSOUT.

OUTPUT = *SYSLST(...)

Output the information to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specifies the SYSLST number.

OUTPUT = *INPUT-FORMATTED-FILE

Output the information to the formatted file specified in the operand UNIT-NAME=*FROM-FORMATTED-FILE(FILE-NAME=).

OUTPUT = <filename 1..54 without-gen-vers>(...)

Output the information to a formatted file. Use the SHOW-FORMATTED-FILE statement to view the contents of the formatted file.

WRITE-MODE =

Write or extend the file.

WRITE-MODE = *REPLACE

Generate the formatted file or overwrite an existing file.

WRITE-MODE = *EXTEND

Generate the formatted file or extend an existing file.

OUTPUT = *NONE

No output is generated except structured output when requested.

STRUCTURE-OUTPUT =

Target of structured output when requested.

STRUCTURE-OUTPUT = *NONE

No structured output is requested.

STRUCTURE-OUTPUT = *SYSINF

The SYSINF stream is to be used for structured output.

STRUCTURE-OUTPUT = <composed-name 1..255>(…)

Name of the S-variable (of the “list” type) to be used for the structured output.

WRITE-MODE = *REPLACE

The S-variable is created or overwritten if already existing.

WRITE-MODE = *EXTEND

S-variable is completed with the new information.

Notes

1. Input

- If UNIT-NAME=*FROM-FORMATTED-FILE is specified, the ID of the objects to be shown is taken from the formatted file records that contain an identifier for the desired object type.
- For the sake of consistency, we recommend that you only specify a SHOW statement for a formatted file if it was generated with the same SCI status. However, if the information in the SCI has changed since the formatted file was created, the information that is output is not longer current.

2. Output

- If the user does not have the SUBSYSTEM-MANAGEMENT privilege, the path name of the installation items is only output when the user has DMS access to it (e.g. SHOW-FILE possible). Otherwise, an asterisk (*) is output.
- If the output is directed to SYSOUT or SYSLST, the information is edited so as to be legible.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.

3. The statement is rejected and an error message is displayed in the following cases:

- The value *BY-DIALOG was specified in batch mode.
- The specified alphanumeric correction state is not a letter followed by two numbers.
- UNIT-NAME=*FROM-FORMATTED-FILE was specified, but the corresponding file does not exist or cannot be accessed.
- OUTPUT=*INPUT-FORMATTED-FILE was specified, but the UNIT-NAME parameter is not *FROM-FORMATTED-FILE.
- There are duplicate units in the input list, i.e. units with the same values or key words for one of the identification parameters.

4. In the following cases, a warning is issued and execution of the statement resumes:

- An object, for which information is to be supplied, does not exist in the SCI.
- If, when displaying an installation item, the file name returned by the path name manager is not the same as that specified during registration, both file names are displayed.

Output in S variables

Output information	Name of the S variable	T	Contents
activable level	var(*LIST).IU-ACTIV-LEVEL	B	FALSE/TRUE
activable status	var(*LIST).IU-ACTIVABLE	S	<text 1..2>
target version	var(*LIST).IU-BS2000-OSD	S	190/200/210
change date	var(*LIST).IU-CHANGE-DATE	S	yyyy-mm-dd
change time	var(*LIST).IU-CHANGE-TIME	S	hh:mm:ss
creation date	var(*LIST).IU-CREA-DATE	S	yyyy-mm-dd
creation time	var(*LIST).IU-CREA-TIME	S	hh:mm:ss
function level	var(*LIST).IU-FU-LEVEL	S	BOTH/TPR/TU
name of the installation item	var(*LIST).IU-II(*LIST).II-NAME	S	<filename 1..30>
path name of the item	var(*LIST).IU-II(*LIST).II-PATH	S	<filename 1..54>
version of the installation item	var(*LIST).IU-II(*LIST).II-VERSION	S	<c-string 1..5>
number of contained items	var(*LIST).IU-II-COUNT	I	<integer 0..2147483647>
installation counter	var(*LIST).IU-INST-COUNTER	I	<integer 0..2147483647>
name of the installation unit	var(*LIST).IU-NAME	S	<text 1..30>
user ID	var(*LIST).IU-USID	S	<name 1..8>
version of the installation unit	var(*LIST).IU-VERSION	S	<product-version 7..7>

4.4.20 SHOW-PACKAGES Output information on software deliveries

Privileges: SUBSYSTEM-MANAGEMENT

Function

Information on software deliveries that are registered in the open SCI (see ["Software Configuration Inventory \(SCI\)"](#)) are output with the SHOW-PACKAGES statement. Deliveries can also be selected interactively in the dialog mode with the PACKAGE-NAME=*BY-DIALOG operand.

The statement supports structured output in S variables (see ["Output in S variables" "SHOW-PACKAGES Output information on software deliveries"](#)).

Format

SHOW-PACKAGES

```
PACKAGE-NAME = *ALL / *BY-DIALOG / list-poss(30): <alphanum-name 1..12>
,USER-CODE = *ALL / <alphanum-name 1..8>
,INFORMATION = *PARAMETERS(...)
    *PARAMETERS(...)
        | SUPPLY-UNITS = *YES / *NO
        | ,REPORT-LEVEL = *MINIMUM / *ALL-ATTRIBUTES
,OUTPUT = *SYSOUT / *SYSLST(...) / *NONE
    *SYSLST(...)
        | SYSLST-NUMBER = *STD / <integer 1..99>
,STRUCTURE-OUTPUT = *NONE / *SYSINF / <composed-name 1..255>(…)
    <composed-name 1..255>(…)
        | WRITE-MODE = *REPLACE / *EXTEND
,SELECT = *BY-ATTRIBUTES(…)
    *BY-ATTRIBUTES(…)
        | CREATION-DATE = *ANY / *INTERVAL(…)
        | *INTERVAL(…)
        |           | FROM = *ANY / <date with-compl>(…)
        |           | ,TO = *ANY / <date with-compl>(…)
        | ,CHANGE-DATE = *ANY / *INTERVAL(…)
        | *INTERVAL(…)
        |           | FROM = *ANY / <date with-compl>(…)
        |           | ,TO = *ANY / <date with-compl>(…)
        | ,CONTAINED-SU-STATUS = *ANY / *ALL-INSTALLED / *PARTIALLY-INSTALLED / *NOTHING-
        | INSTALLED
```

Operands

PACKAGE-NAME =

Selection of the software deliveries to be displayed using the package name (as specified in the footer of the delivery notification). The output can be restricted to a certain customer ID in the USER-CODE operand.

PACKAGE-NAME = *ALL

All software deliveries registered in the SCI are selected.

PACKAGE-NAME = *BY-DIALOG

The software deliveries are listed in a selection window and can be selected there interactively (see also "Selection of deliveries" in the menu mode on "[View menu](#)"). The information requested is output after the software deliveries have been selected by entering any character in front of the corresponding package names and the window has been closed with [DUE].

The data must be specified for all customer IDs (USER-CODE=*ALL).

PACKAGE-NAME = list-poss(30): <alphanum-name 1..12>

Only software deliveries with the specified package name are selected.

USER-CODE =

Selection of the software deliveries to be displayed using the customer ID (as specified in the footer of the delivery notification).

USER-CODE = *ALL

Data is output for all customer IDs.

USER-CODE = <alphanum-name 1..8>

Data is output only for software deliveries with the specified customer ID.

INFORMATION = *PARAMETERS(...)

Specifies the scope of the information to be output.

SUPPLY-UNITS = *YES / *NO

Specifies if the supply units contained in the specified deliveries are also to be output.

REPORT-LEVEL =

Controls the output of the attributes of the supply units.

REPORT-LEVEL = *MINIMUM

Only the names of the supply units are output.

REPORT-LEVEL = *ALL-ATTRIBUTES

The names and attributes of the supply units are output.

OUTPUT =

Specifies the destination of the output.

OUTPUT = *SYSOUT

The information is output to SYSOUT.

OUTPUT = *SYSLST(...)

The information is output to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

The information is output to the standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specification of the SYSLST number.

OUTPUT = *NONE

No output is generated except structured output when requested.

STRUCTURE-OUTPUT =

Target of structured output when requested.

STRUCTURE-OUTPUT = *NONE

No structured output is requested.

STRUCTURE-OUTPUT = *SYSINF

The SYSINF stream is to be used for structured output.

STRUCTURE-OUTPUT = <composed-name 1..255>(…)

Name of the S-variable (of the “list” type) to be used for the structured output.

WRITE-MODE = *REPLACE

The S-variable is created or overwritten if already existing.

WRITE-MODE = *EXTEND

S-variable is completed with the new information.

SELECT = *BY-ATTRIBUTES(…)

Limits the selection of deliveries with specific criteria.

CREATION-DATE = *ANY / *INTERVAL(…)

All deliveries are shown according to creation date (time when the delivery was first entered in the SCI).

CREATION-DATE = *ANY

The selection of deliveries is independent of creation date.

CREATION-DATE = *INTERVAL(…)

All the deliveries which were first entered into the SCI within the stated time period are displayed.

Area specifications include the area boundaries.

A meaningful output with area boundaries is only possible, if a lower boundary <= the upper boundary is selected. If no upper or lower boundary is selected (corresponding to the default *ANY), the selection area has no upper or lower limit.

FROM = *ANY / <date with-compl>

All deliveries which were first entered in the SCI after the stated date, are displayed (CREATION-DATE >= stated date).

TO = *ANY / <date with-compl>

All deliveries which were first entered in the SCI before the stated date, are displayed (CREATION-DATE <= stated date).

CHANGE-DATE = *ANY / *INTERVAL(...)

All deliveries are shown, according to change date (the time at which the delivery was last modified in the SCI).

CHANGE-DATE = *ANY

The selection of deliveries is independent of change date.

CHANGE-DATE = *INTERVAL(...)

All deliveries which have been modified in the SCI within the stated time period, are displayed.

Area specifications include the area boundaries.

A meaningful output with area boundaries is only possible, if a lower boundary <= the upper boundary is selected. If no upper or lower boundary is selected (corresponding to the default *ANY), the selection area has no upper or lower limit.

FROM = *ANY / <date with-compl>

All deliveries which were last changed in the SCI after the stated date, are displayed (CREATION-DATE >= stated date).

TO = *ANY / <date with-compl>

All deliveries which were last changed in the SCI before the stated date, are displayed (CREATION-DATE <= stated date).

CONTAINED-SU-STATUS = *ANY / *ALL-INSTALLED / *PARTIALLY-INSTALLED / *NOTHING-INSTALLED

All deliveries are displayed, according to installation status of the supply units that belong to it.

CONTAINED-SU-STATUS = *ANY

The selection of deliveries is independent of the supply units that belong to it.

CONTAINED-SU-STATUS = *ALL-INSTALLED

All deliveries are shown in which all supply units that belong to it have been installed (status "Installed").

CONTAINED-SU-STATUS = *PARTIALLY-INSTALLED

All deliveries are shown in which only a portion of the supply units that belong to it has been installed (status "Installed").

CONTAINED-SU-STATUS = *NOTHING-INSTALLED

All deliveries are shown in which none of the supply units that belong to it have been installed (status "Installed") yet.

Notes

1. In the following cases the statement is rejected without taking any action and an error message is output:
 - Selections are to be made interactively (PACKAGE-NAME=*BY-DIALOG) and an operand value not equal to *ALL was specified in the USER-CODE operand.
 - PACKAGE-NAME=*BY-DIALOG was specified in the batch mode.
 - Package names were specified more than once in a list of package names in the PACKAGE-NAME operand.
2. If there is no delivery for one of the specified package names and customer IDs, then a warning is output and execution of the statement is continued.
3. Supply units already installed using a different package/customer ID are marked with an asterisk after the installation status in the output for a specified package name /customer ID.

Output in S variables

Output information	Name of the S variable	T	Contents
change date	var(*LIST).PACK-CHANGE-DATE	S	yyyy-mm-dd
change time	var(*LIST).PACK-CHANGE-TIME	S	hh:mm:ss
creation date	var(*LIST).PACK-CREA-DATE	S	yyyy-mm-dd
creation time	var(*LIST).PACK-CREA-TIME	S	hh:mm:ss
device type of the delivery (if MEDIUM=A)	var(*LIST).PACK-DEL-DEV-TYPE	S	<device>
delivery library (if MEDIUM=L)	var(*LIST).PACK-DEL-LIB	S	<filename 1..54>
VSN of the delivery (if MEDIUM=A)	var(*LIST).PACK-DEL-VSN	S	<vsnn>
documentation library	var(*LIST).PACK-DOC-LIB	S	<filename 1..54>
delivery medium type L=library A=data volume	var(*LIST).PACK-MEDIUM	S	L / A
name of the package	var(*LIST).PACK-NAME	S	<alphanum-name 1..12>

installation status	var(*LIST).PACK-SU(*LIST).SU-INST-STATUS	S	Installed Being installed Parked In Library On SOLIS2 Volume On Local Volume Being deinstalled Partially installed
name of the supply unit	var(*LIST).PACK-SU(*LIST).SU-NAME	S	<text 1..30>
version of the supply unit	var(*LIST).PACK-SU(*LIST).SU-VERSION	S	<product-version 7..7>
number of contained supply units	var(*LIST).PACK-SU-COUNT	I	<integer 0..2147483647>
user code	var(*LIST).PACK-USER-CODE	S	<alphanumeric 1..8>
user ID	var(*LIST).PACK-USID	S	<name 1..8>
work file location	var(*LIST).PACK-WORK-FILE-LOC	S	<filename 1..22>

4.4.21 SHOW-SUPPLY-UNITS Show supply units

Privileges: SUBSYSTEM-MANAGEMENT

Function

The SHOW-SUPPLY-UNITS statement enables you to show information about the supply units in the open SCI (see "[Software Configuration Inventory \(SCI\)](#)").

Possible entries are the supply units of an SOLIS2 delivery or a formatted file generated earlier by IMON. Supply units can also be selected interactively with the SUPPLY-UNITS=*BY-DIALOG operand in the dialog mode.

The statement supports structured output in S variables (see "[Output in S variables](#)" "[SHOW-SUPPLY-UNITS Show supply units](#)").

Format

SHOW-SUPPLY-UNITS

UNIT-NAME = ***ALL** / ***BY-DIALOG** / ***FROM-SOLIS2-DELIVERY(...)** / ***FROM-FORMATTED-FILE(...)** /list-poss(30): <text 1..30 without-sep>(…)

***FROM-SOLIS2-DELIVERY(...)**

| **PACKAGE-NAME** = <alphanum-name 1..12>

| **,USER-CODE** = <alphanum-name 1..8>

| **,SUPPLY-UNITS** = ***ALL** / ***BY-DIALOG** / list-poss(30): <text 1..30 without-sep>(…)

| <text 1..30 without-sep>(…)

| **VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <product-version without-man-corr>/ **,CORRECTION-STATE** = ***HIGHEST-EXISTING** / ***ALL** / ***LOWEST-EXISTING** / list-poss(15): <alphanum-name 3..3>

***FROM-FORMATTED-FILE(...)**

| **FILE-NAME** = <filename 1..54 without-gen-vers>

<text 1..30 without-sep>(…)

| **VERSION** = ***ALL** / ***HIGHEST-EXISTING** / <product-version without-man-corr>

| **,CORRECTION-STATE** = ***ALL** / ***HIGHEST-EXISTING** / ***LOWEST-EXISTING** /list-poss(15): <alphanum-name 3..3>

,INFORMATION = ***PARAMETERS(...)**

***PARAMETERS(...)**

| **INSTALLATION-UNITS** = ***YES** / ***NO**

| **,REPORT-LEVEL** = ***SUMMARY** / ***MINIMUM** / ***FOR-REFERENCE-FILE** / ***ALL-ATTRIBUTES**

,OUTPUT = ***SYSOUT** / ***SYSLST(...)** / ***INPUT-FORMATTED-FILE** / <filename 1..54 without-gen-vers>(…) / ***NONE**

***SYSLST(...)**

| **SYSLST-NUMBER** = ***STD** / <integer 1..99>

<filename 1..54 without-gen-vers>(…)

| **WRITE-MODE** = ***REPLACE** / ***EXTEND**

,STRUCTURE-OUTPUT = ***NONE** / ***SYSINF** / <composed-name 1..255>(…)

<composed-name 1..255>(…)

| **WRITE-MODE** = ***REPLACE** / ***EXTEND**

,SELECT = ***BY-ATTRIBUTES(...)**

***BY-ATTRIBUTES(...)**

- | **INSTALLATION-STATUS = *ANY / list-poss(20): *INSTALLED / *BEING-INSTALLED / *PARKED / *IN-LIBRARY / *ON-SOLIS2-VOLUME / *ON-LOCAL-VOLUME / *BEING-DEINSTALLED / *PARTIALLY-INSTALLED**
- | **,CUSTOMER-APPROVED = *ANY / *YES / *NO**

Operands

UNIT-NAME =

Name of the supply unit to be shown.

UNIT-NAME = *ALL

Show all supply units registered in the SCI.

UNIT-NAME = *BY-DIALOG

The supply units are listed in a selection window and can be selected interactively in a dialog (see also "Selection of supply units" in the menu mode on "[View menu](#)"). The information requested is output after the supply units have been selected by entering any character in front of the corresponding supply units and the window has been closed with [DUE].

UNIT-NAME = *FROM-SOLIS2-DELIVERY(...)

The supply units are from a SOLIS2 delivery.

PACKAGE-NAME = <alphanum-name 1..12>

Package name (as stated in the footer of the delivery contents).

USER-CODE = <alphanum-name 1..8>

User code (as stated in the footer of the delivery contents).

SUPPLY-UNITS =

The supply units to be shown.

SUPPLY-UNITS = *ALL

Show all the supply units of a SOLIS2 delivery registered in the SCI.

SUPPLY-UNITS = *BY-DIALOG

The supply units are listed in a selection window and can be selected interactively in a dialog (see also "Selection of supply units" in the menu mode on "[View menu](#)"). The information requested is output after the supply units have been selected by entering any character in front of the corresponding supply units and the window has been closed with [DUE].

SUPPLY-UNITS = list-poss(30): <text 1..30 without-sep>(...)

Explicit specification of the supply units, see "[Supply unit](#)".

VERSION =

Version of the supply unit.

VERSION = *HIGHEST-EXISTING

Select the highest version of the supply unit.

VERSION = *ALL

Select all versions of the supply unit.

VERSION = <product-version without-man-corr>

Version designation without release and correction states.

CORRECTION-STATE =

Correction state of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Select the highest correction state of the supply unit.

CORRECTION-STATE = *ALL

Select all correction states of the supply unit.

CORRECTION-STATE = *LOWEST-EXISTING

Select the lowest correction state of the supply unit.

CORRECTION-STATE = list-poss(3): <alphanum-name 3..3>

Explicit specification of the correction state of the supply unit.

UNIT-NAME = *FROM-FORMATTED-FILE(...)

The names of the supply units will be taken from a formatted file.

FILE-NAME = <filename 1..54 without-gen-vers>

Name of the formatted file containing the list of supply units.

UNIT-NAME = list-poss(30): <text 1..30 without-sep>(...)

Name of the supply unit, see "[Supply unit](#)".

VERSION =

Version of the supply unit.

VERSION = *ALL

Select all version designations of the supply unit.

VERSION = *HIGHEST-EXISTING

Select the highest version designation of the supply unit.

VERSION = <product-version without-man-corr>

Version designation without release and correction states.

CORRECTION-STATE =

Correction state of the supply unit.

CORRECTION-STATE = *ALL

Select all correction states of the supply unit.

CORRECTION-STATE = *HIGHEST-EXISTING

Select the highest correction state of the supply unit.

CORRECTION-STATE = *LOWEST-EXISTING

Select the lowest correction state of the supply unit.

CORRECTION-STATE = list-poss(15): <alphanum-name 3..3>

Select the specified correction state of the supply unit.

INFORMATION = *PARAMETERS(...)

The scope of information to be shown.

INSTALLATION-UNITS = *YES / *NO

Show/do not show the installation units in the supply unit.

REPORT-LEVEL =

The reporting level of the supply unit attributes.

REPORT-LEVEL = *SUMMARY

The attributes of the supply units are not output.

The last version of the selected supply unit is displayed regardless of the customer ID or package name.

REPORT-LEVEL = *MINIMUM

Show the names of the supply units only. If the delivery is a delta delivery, show only the supplied installation unit.

REPORT-LEVEL = *FOR-REFERENCE-FILE

The names of the supply units and the creation date (CREATION-DATE) are displayed.

With this setting, the installation of the reference file is carried out under the user ID SERVICE.

REPORT-LEVEL = *ALL-ATTRIBUTES

Show the names and attributes of the supply units for each correction state. If the delivery is a delta delivery, show only the supplied installation unit.

OUTPUT =

The target for output.

OUTPUT = *SYSOUT

Output the information to SYSOUT.

OUTPUT = *SYSLST(...)

Output the information to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Output the information to standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specifies the SYSLST number.

OUTPUT = *INPUT-FORMATTED-FILE

Output the information to the formatted file specified in the operand UNIT-NAME=*FROM-FORMATTED FILE(FILE-NAME).

OUTPUT = <filename 1..54 without-gen-vers>(…)

Output the information to a formatted file. Use the SHOW-FORMATTED-FILE statement to view the contents of the formatted file.

WRITE-MODE =

Write or extend the file.

WRITE-MODE = *REPLACE

Generate the formatted file or overwrite an existing file.

WRITE-MODE = *EXTEND

Generate the formatted file or extend an existing file.

OUTPUT = *NONE

No output is generated except structured output when requested.

STRUCTURE-OUTPUT =

Target of structured output when requested.

STRUCTURE-OUTPUT = *NONE

No structured output is requested.

STRUCTURE-OUTPUT = *SYSINF

The SYSINF stream is to be used for structured output.

STRUCTURE-OUTPUT = <composed-name 1..255>(…)

Name of the S-variable (of the “list” type) to be used for the structured output.

WRITE-MODE = *REPLACE

The S-variable is created or overwritten if already existing.

WRITE-MODE = *EXTEND

S-variable is completed with the new information.

SELECT = *BY-ATTRIBUTES(…)

Restricts the selection to supply units with certain criteria.

INSTALLATION-STATUS =

All supply units that are in the specified installation state are displayed. Several installation states can be specified in a list as selection criterion (logically ORed).

INSTALLATION-STATUS = *ANY

The supply units are selected regardless of their installation state.

INSTALLATION-STATUS = *INSTALLED

Supply units in the “Installed” state are displayed.

INSTALLATION-STATUS = *BEING-INSTALLED

Supply units in the “Being Installed” state are displayed.

INSTALLATION-STATUS = *PARKED

Supply units in the “Parked” state are displayed.

INSTALLATION-STATUS = *IN-LIBRARY

Supply units in the “In Library” state are displayed.

INSTALLATION-STATUS = *ON-SOLIS2-VOLUME

Supply units in the “On SOLIS2 Volume” state are displayed.

INSTALLATION-STATUS = *ON-LOCAL-VOLUME

Supply units in the “On Local Volume” state are displayed.

INSTALLATION-STATUS = *BEING-DEINSTALLED

Supply units in the “Being Deinstalled” state are displayed.

INSTALLATION-STATUS = *PARTIALLY-INSTALLED

Supply units in the “Partially Installed” state are displayed.

CUSTOMER-APPROVED =

Display of supply units depends on an approval timestamp. An approval timestamp is set with the SET-CUSTOMER-APPROVAL statement to release an installed supply unit for further installations in the production system.

CUSTOMER-APPROVED = *ANY

Display of supply units is independent of a timestamp.

CUSTOMER-APPROVED = *YES

Supply units with an approval timestamp are displayed.

CUSTOMER-APPROVED = *NO

Supply units without an approval timestamp are displayed.

Notes

1. Input

- If UNIT-NAME=*FROM-FORMATTED-FILE is specified, the ID of the objects to be shown is taken from the formatted file records that contain an identifier for the desired object type.
- For the sake of consistency, we recommend that you only specify a SHOW statement for a formatted file if it was generated with the same SCI status. However, if the information in the SCI has changed since the formatted file was created, the information that is output is not longer current.

2. Output

- If the output is directed to SYSOUT or SYSLST, the information is edited so as to be legible.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.

3. Execution of the statement is continued with a warning when an object for which information is to be supplied does not exist in the SCI.

4. In the following cases, the statement is rejected and an error message is displayed :

- The value *BY-DIALOG was specified in batch mode.
- The explicitly specified alphanumeric correction state is not a letter followed by two numbers.
- UNIT-NAME=*FROM-FORMATTED-FILE was specified, but the corresponding file does not exist or cannot be accessed.
- OUTPUT=*INPUT-FORMATTED-FILE was specified, but the UNIT-NAME parameter is not *FROM-FORMATTED-FILE.
- There are duplicate units in the input list, i.e. units with the same values or key words for one of the identification parameters.

Output in S variables

Output information	Name of the S variable	T	Contents
date of customer approval	var(*LIST).SU-CUST-APPR-DATE	S	yyyy-mm-dd
time of customer approval	var(*LIST).SU-CUST-APPR-TIME	S	hh:mm:ss
activable status	var(*LIST).SU-ACTIVABLE	B	FALSE/TRUE
change date	var(*LIST).SU-CHANGE-DATE	S	yyyy-mm-dd
change time	var(*LIST).SU-CHANGE-TIME	S	hh:mm:ss
creation date	var(*LIST).SU-CREA-DATE	S	yyyy-mm-dd
creation time	var(*LIST).SU-CREA-TIME	S	hh:mm:ss
installation counter	var(*LIST).SU-INST-COUNTER	I	

installation status	var(*LIST).SU-INST-STATUS	S	Installed Being installed Parked In Library On SOLIS2 Volume On Local Volume Being deinstalled Partially installed
name of the installation unit	var(*LIST).SU-IU(*LIST).IU-NAME	S	<text 1..30>
version of the installation unit	var(*LIST).SU-IU(*LIST).IU-VERSION	S	<product-version 7..7>
number of contained installation units	var(*LIST).SU-IU-COUNT	I	<integer 0..2147483647>
name of the supply unit	var(*LIST).SU-NAME	S	<text 1..30>
name of the package	var(*LIST).SU-PACK-NAME	S	<alphanum-name 1..12>
user code	var(*LIST).SU-USER-CODE	S	<alphanum-name 1..8>
user ID	var(*LIST).SU-USID	S	<name 1..8>
version of the supply unit	var(*LIST).SU-VERSION	S	<product-version 7..7>

4.4.22 SWITCH-TO-FHS Switch to FHS interface of IMON

Privileges: STD-PROCESSING, SUBSYSTEM-MANAGEMENT

Function

The SWITCH-TO-FHS statement enables you to switch from the SDF interface to menu mode. The current SCI remains open. This statement is not available in batch mode.

Format

SWITCH-TO-FHS

This statement has no operands.

4.4.23 UNDO-SUPPLY-UNITS Undo installation

Privileges: SUBSYSTEM-MANAGEMENT

Function

The last installation performed of a supply unit is undone with the UNDO-SUPPLY-UNITS statement. The supply unit is removed from the system with the Undo function (deinstalled). In addition, the original state before installation of the supply unit is restored based on the Undo files saved during installation. The following actions are performed when the Undo function is called:

- Installed files are deleted if necessary.
- “New” syntax files, message files and subsystems are deactivated.
- The SCI is cleaned.
- “Old” syntax files, message files and subsystems are reactivated.
- Changed or deleted files are reconstructed.

A requirement for this is that all changed or deleted files were saved during installation and the metadata of the installation was recorded in Undo files (see the INSTALL-UNITS statement, UNDO-PREPARATION operand). Only a SCI with the standard name \$TSOS.SYS.IMON.SCI on the home pubset or an imported pubset is open. The Undo function is rejected for another SCI.

The test mode can be set using the operand EXECUTION=*NO . In this case only the preliminary analysis used to monitor the successful execution of the Undo function is performed.

Format

UNDO-SUPPLY-UNITS

```
UNIT-NAME = *BY-DIALOG / list-poss(30): <text 1..30 without-sep>
,DSSM-CATALOG = *DEFAULT / <filename 1..54 without-gen-vers>
,EXECUTION = *YES / *NO
,OUTPUT = *SYSOUT / *SYSLST(...)
      *SYSLST(...)
      |   SYSLST-NUMBER = *STD / <integer 1..99>
```

Operands

UNIT-NAME =

Selection of the supply units for which the Undo function is to be executed.

UNIT-NAME = *BY-DIALOG

The selection criteria and the corresponding supply units are selected interactively (see ["Interactive selection of supply units"](#) (INSTALL-UNITS Install and register software)).

UNIT-NAME = list-poss(30): <text 1..30 without-sep>

Name of the supply unit, see "Supply unit".

DSSM-CATALOG = *DEFAULT / <filename 1..54 without-gen-vers>

Name of the static DSSM catalog to be reconstructed. The catalog ID is specified by the open SCI.

EXECUTION =

Specifies if the actual Undo actions are to be executed after performing the preliminary analysis. During the preliminary analysis a check is performed to see if the most important requirements to perform the deinstallation are fulfilled: :

1. Subsystem to be deleted must be stopped.
2. Files whose activation is to be undone and that are not reconstructed while the UNDO actions are executing must be accessible.

EXECUTION = *YES

The actual Undo actions are executed after a successful preliminary analysis. If the preliminary analysis determines that a requirement is not fulfilled, then an error is logged and error handling is initiated:

- In the procedure or batch mode the processing mode is changed to EXECUTION=*NO, i.e. the preliminary analysis is continued and all errors detected are recorded. The actual Undo actions are not executed anymore, however.
- In the interactive dialog processing is interrupted with a message requiring a response. Depending on the response of the user, processing is simply continued (the error is ignored) processing is continued with EXECUTION=*NO or the current step of the check is repeated.
When an error is ignored IMON assumes during processing after ignoring the error that no error was detected during this step of the check. Repeating this step of the check only makes sense when the cause of the error was eliminated in the meantime.

EXECUTION = *NO

The call is made in the test mode, i.e. only the preliminary analysis for executing the Undo function is performed and any errors detected during analysis are recorded.

OUTPUT = *SYSOUT

Error messages are output to SYSOUT.

OUTPUT = *SYSLST(...)

Error messages are output to SYSLST.

SYSLST-NUMBER =

SYSLST number.

SYSLST-NUMBER = *STD

Error messages are output to the standard SYSLST.

SYSLST-NUMBER = <integer 1..99>

Specification of the SYSLST number.

Notes

1. In the following cases the statement is rejected without taking any action and an error message is output:
 - No SCI with the standard name \$TSOS.SYS.IMON.SCI is open.
 - There is no Undo file for one of the selected supply units. The Undo files are expected under the following name:
:<catid of the open sci>:\$SYSSAG.IMON.UNDO.<name of the supply unit>
 - One of the following situations arises while interactively selecting the supply unit (UNIT-NAME=*BY-DIALOG):
 - More than one version was selected for a given supply unit .
 - The version of a supply unit found in the Undo file does not match the selected version.
 - The value *BY-DIALOG was specified in the batch mode.
2. In the following cases execution of the statement is continued with a warning:
 - The supply unit found in the Undo file is not registered in the SCI.
 - The supply unit found in the Undo file is registered in the SCI, but is not in the “Installed” state.

4.5 The IMON-BAS macros

The IMON-BAS macros enable you to use the program interface for the convenient, automated invocation of the IMON-BAS functions. The following macros are invoked directly from within an application program (Assembler) and executed in real time.

List of functions

Macro	Function
IMOSHFF	Show formatted file
IMOSHII	Show installation items
IMOSHIU	Show installation units
IMOSHSU	Show supply units

The macros are described in alphabetical order; each description is structured as follows:

- Macro name and function
- Description of macro function
- Presentation of macro format
An empty line separates the format operands from the function operands.
- Description of operands
The format operands and metasyntax are described on "[Macro syntax](#)".
- Macro return codes
- Notes

4.5.1 IMOSHFF Show formatted file

The IMOSHFF macro enables you to view in structured form the information contained in a formatted file.

The advantage of a formatted file is that it can be used as the target for the output of piped SHOW functions. You can then output the formatted file and view the preprocessed information.

Macro	Operands
IMOSHFF	MF= <u>D</u> / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= <u>MOE</u> / <char (3)> ,PARAM= <adr> / (<reg>) ,INFF= <c-string: 1..54> /<var: char:54> ,SYSLST#= <u>Q</u> / <integer 1..99> / <var: int:1>

Operands

The MF, PREFIX, MACID, and PARAM parameters are described in “Macro forms” on ["Macro syntax "](#).

INFF

Formatted file to be used for output.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

SYSLST#

SYSLST number.

=0

Output the information to standard SYSLST.

=<integer 1..99>

Specifies the SYSLST number.

=<var: int:1>

Symbolic address containing the SYSLST number.

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0003'	SYSLST number invalid
X'00'	X'01'	X'0004'	Parameter list invalid
X'00'	X'40'	X'0015'	Error accessing input file, function not executed
X'00'	X'40'	X'0019'	SHOW output written to SYSLST only partially or not at all
	X'20'	X'0100'	System error
	X'20'	X'0101'	Internal error
X'00'	X'01'	X'FFFF'	Function is not supported
X'00'	X'03'	X'FFFF'	Version of the interface is not supported.

Notes

- The function is aborted without action if the input file does not exist, cannot be accessed or a format error occurs.
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

4.5.2 IMOSHII Show installation items

The IMOSHII macro enables you to fetch information about the installation items in the standard SCI, see ["Installation item \(II\) "](#).

Possible entries are the name of an installation item, the installation path name of the item (file or library element) or a formatted file generated beforehand with the SHOW functions of IMON.

The output is directed to SYSOUT, to SYSLST, to the formatted file specified during input, of to another formatted file that can be generated, extended or overwritten.

If you are an nonprivileged user, this macro will return only the path names to which DMS access is possible.

Macro	Operands
IMOSHII	MF= <u>D</u> / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= <u>MOI</u> / <char (3)> ,PARAM= <adr> / (<reg>) ,ITYPE= * <u>ITEM</u> / *INSPATH / *FILE ,INAME= <c-string: 1..30> / <var: char:30> ,IVERS= * <u>ALL</u> / *HIGH / <c-string 1..5> / <var: char:5> ,UNAME= * <u>ALL</u> / <c-string: 1..30> / <var: char:30> ,UVERS= * <u>ALL</u> / *HIGH / <c-string: 3..5 > / <var: char:5> ,UCORSTA= * <u>ALL</u> / *HIGH / *LOW / <c-string: 3..3> / <var: char:3> ,INSPATH= <c-string: 1..54> / <var: char:54> ,INFF= <c-string: 1..54> / <var: char:54> ,REPLEV= * <u>MINIMUM</u> / *ALL ,OUTTYPE= * <u>SYSOUT</u> / *INFILE / *FILE / *SYSLST ,OUTNAME= <c-string: 1..54> / <var: char:54> ,WRMODE= * <u>REPLACE</u> / *EXTEND ,SYSLST#= <u>Q</u> / <integer 1..99> / <var: int:1>

Operands

The MF, PREFIX, MACID, and PARAM parameters are described in "Macro forms" on ["Macro syntax "](#).

ITYPE

Input mode used to select the installation item.

=*ITEM

The installation item is identified by its name and version (INAME and IVERS). The assignment of the installation item to an installation unit (UNAME, UVERS and UCORSTA) is taken into account.

=*INSPATH

The installation item is identified by its path name (INSPATH).

=*FILE

The installation item will be taken from a formatted file (INFF).

INAME

Name of the installation item.

=<c-string: 1..30>

Specification of the name of the installation item.

=<var: char:30>

Symbolic address containing the specification for the name of the installation item.

IVERS

Version of the installation item.

=*ALL

Select all version of the installation item.

=*HIGH

Select the highest version designation of the installation item.

=<c-string 1..5>

Specification of the version of the installation item.

=<var: char:5>

Symbolic address containing the specification for the version of the installation item.

UNAME

Name of the installation unit searched for the installation item.

=*ALL

Search all installation units for the installation item.

=<c-string: 1..30>

Specification of the name of the installation unit.

=<var: char:30>

Symbolic address containing the specification for the name of the installation unit.

UVERS

Version of the installation unit where the search will be performed for the installation item.

=*ALL

Search all versions of the installation unit.

=*HIGH

Search the highest version of the installation unit.

=<c-string: 3..5>

Specifies the version of the installation unit.

=<var: char:5>

Symbolic address containing the specification for the version of the installation unit.

UCORSTA

Correction state of the installation unit where the search will be performed for the installation item.

=*ALL

Search all correction states of the installation unit.

=*HIGH

Search the highest correction state of the installation unit.

=*LOW

Search the lowest correction state of the installation unit.

=<c-string: 3..3>

Specification of the correction state of the installation unit. Format: <aso>.

=<var: char:3>

Symbolic address containing the specification for the correction state of the installation unit.

INSPATH

Path name of the installation item.

=<c-string: 1..54>

Specification of the path name.

=<var: char:54>

Symbolic address containing the path name.

INFF

Name of the formatted file to be used as input.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

REPLEV

Reporting level for the installation item attributes.

=*MINIMUM

Show the names of the installation items only.

=*ALL

Show the names and attributes of the installation items.

OUTTYPE

The target for output.

=*SYSOUT

Output the information to SYSOUT.

=*INFILE

Output the information to the formatted file used as input.

=*FILE

Output the information to a formatted file (OUTNAME).

=*SYSLST

Output the information to SYSLST.

OUTNAME

Name of the formatted file to be used as output.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

WRMODE

Overwrite or extend the file.

=*REPLACE

Generate the formatted file or overwrite an existing file.

=*EXTEND

Generate the formatted file or extend an existing file.

SYSLST#

SYSLST number.

=Q

Output the information to standard SYSLST.

=<integer 1..99>

Specifies the SYSLST number.

=<var: int:1>

Symbolic address containing the specification for the SYSLST number.

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	Function not permitted in batch mode
X'00'	X'01'	X'0002'	Output invalid
X'00'	X'01'	X'0003'	SYSLST number invalid
X'00'	X'01'	X'0004'	Parameter list invalid
X'00'	X'40'	X'0013'	Cannot access the SCI, function not executed
X'00'	X'40'	X'0014'	Specified unit not found in the SCI, function not executed, no formatted file created
X'00'	X'40'	X'0015'	Error accessing input file (library element), function not executed
X'00'	X'40'	X'0016'	Error accessing output file, function not executed
X'00'	X'40'	X'0017'	Information missing. Units contained in the formatted file (ITYPE=*FILE) not present in the SCI
X'00'	X'40'	X'0018'	Formatted file contains no information for action
X'00'	X'40'	X'0019'	SHOW output written to SYSLST only partially or not at all
	X'20'	X'0100'	System error
	X'20'	X'0101'	Internal error
X'00'	X'01'	X'FFFF'	Function is not supported
X'00'	X'03'	X'FFFF'	Interface version is not supported

X'03'	X'00'	X'0000'	Warning: Information output is partial, user does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0014'	User does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0017'	User does not have SUBSYSTEM-MANAGEMENT privilege

Notes

- An MNOTE is generated if:
 - ITYPE =*ITEM and INAME was not specified.
 - ITYPE =*INSPATH and INSPATH was not specified.
 - ITYPE =*FILE and INFF was not specified.
 - ITYPE =*ITEM / *INSPATH and OUTTYPE =*INFILE.
 - OUTTYPE =*FILE and OUTNAME was not specified.
 - OUTTYPE =*SYSOUT and REPLEV =*ALL.If ITYPE =*ITEM is specified, the values specified for INSPATH and INFF are ignored.
- If ITYPE =*INSPATH is specified, the values specified for INAME, IVERS, UNAME, UVERS, UCORSTA and INFF are ignored.
- If ITYPE =*FILE is specified, the values specified for INAME, IVERS, UNAME, UVERS, UCORSTA and INSPATH are ignored.
- If OUTTYPE =*SYSOUT / *SYSLST / *INFILE is specified, the value specified for WRMODE is ignored.
- If OUTTYPE =*SYSOUT / *FILE / *INFILE is specified, the value specified for SYSLST# is ignored.
- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.
- An asterisk (*) is substituted for each path name that the user is not authorized to read.
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

4.5.3 IMOSHIU Show installation units

The IMOSHIU macro enables you to view information about the installation units contained in the standard SCI, see "[Installation unit \(IU\)](#)".

Possible inputs are an installation unit or a formatted file generated beforehand with the SHOW functions of IMON. You also have the option of selecting installation units interactively in interactive mode (see also "Selection of installation units" in the menu mode on "[View menu](#)").

The output is directed to SYSOUT, to SYSLST, to the formatted file specified during input, or to another formatted file that can be generated, extended or overwritten.

If you are a nonprivileged user, this macro will show only the installation units to which nonprivileged DMS access is possible (i.e. no TPR installation units).

Macro	Operands
IMOSHIU	,PREFIX= I / <char (1)> ,MACID= MOU / <char (3)> ,PARAM= <adr> / (<reg>) ,UTYPE= *UNIT / *ALL / *DIALOG / *FILE ,UNAME= <c-string: 1..30> / <var: char:30> ,UVERS= *ALL / *HIGH / <c-string: 3..5> / <var: char:5> ,UCORSTA= *ALL / *HIGH / *LOW / <c-string: 3..3> / <var: char:3> ,INFF= <c-string: 1..54 > / <var: char:54> ,INSITEM= *YES / *NO ,REPLEV= *MINIMUM / *ALL ,OUTTYPE= *SYSOUT / *INFILE / *FILE / *SYSLST ,OUTNAME= <c-string: 1..54> / <var: char:54> ,WRMODE= *REPLACE / *EXTEND ,SYSLST#= 0 / <integer 1..99> / <var: int:1>

Operands

The MF, PREFIX, MACID, and PARAM parameters are described in "Macro forms" on "[Macro syntax](#)".

UTYPE

Input mode used to select installation units.

=*UNIT

Select installation units by UNAME, UVERS and UCORSTA

=*ALL

Select all installation units of the SCI.

=*DIALOG

The installation units will be selected interactively (see also "Selection of installation units" in the menu mode on "[View menu](#)").

=*FILE

Select the installation units from a formatted file (INFF).

UNAME

Name of the installation unit.

=<c-string: 1..30>

Specification of the name of the installation unit.

=<var: char:30>

Symbolic address containing the specification for the name of the installation unit.

UVERS

Version of the installation unit.

=*ALL

Select all versions of the installation unit.

=*HIGH

Select the highest version of the installation unit.

=<c-string: 3..5>

Specifies the version of the installation unit.

=<var: char:5>

Symbolic address containing the specification for the version of the installation unit.

UCORSTA

Correction state of the installation unit.

=*ALL

Select all correction states of the installation unit.

=*HIGH

Select the highest correction state of the installation unit.

=*LOW

Select the lowest correction state of the installation unit.

=<c-string: 3..3>

Specification of the correction state of the installation unit. Format: <aso>.

=<var: char:3>

Symbolic address containing the specification for the correction state of the installation unit.

INFF

Name of the formatted file to be used as input.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

INSITEM

Show/do not show installation items.

=*YES

Show installation items.

=*NO

Do not show installation items.

REPLEV

Reporting level for installation unit attributes.

=*MINIMUM

Show the names of the installation units only.

=*ALL

Show the names and attributes of the installation items.

OUTTYPE

The target for output.

=*SYSOUT

Output the information to SYSOUT.

=*INFILE

Output the information to the formatted file used as input.

=*FILE

Output the information to a formatted file (OUTNAME).

=*SYSLST

Output the information to SYSLST.

OUTNAME

Name of the formatted file to be used as output.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

WRMODE

Overwrite or extend the file.

=*REPLACE

Generate the formatted file or overwrite an existing file.

=*EXTEND

Generate the formatted file or extend an existing file.

SYSLST#

SYSLST number.

=0

Output the information to standard SYSLST.

=<integer 1..99>

Specifies the SYSLST number.

=<var: int:1>

Symbolic address containing the specification for the SYSLST number.

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	Function not permitted in batch mode
X'00'	X'01'	X'0002'	Output invalid
X'00'	X'01'	X'0003'	SYSLST number invalid

X'00'	X'01'	X'0004'	Parameter list invalid
X'00'	X'40'	X'0013'	Cannot access the SCI, function not executed
X'00'	X'40'	X'0014'	Specified unit not found in the SCI, function not executed, no formatted file created
X'00'	X'40'	X'0015'	Error accessing input file (library element), function not executed
X'00'	X'40'	X'0016'	Error accessing output file, function not executed
X'00'	X'40'	X'0017'	Information missing. Units contained in the formatted file (UTYPE=*FILE) not present in the SCI.
X'00'	X'40'	X'0019'	SHOW output written to SYSLST only partially or not at all
	X'20'	X'0100'	System error
	X'20'	X'0101'	Internal error
X'00'	X'01'	X'FFFF'	Function is not supported
X'00'	X'03'	X'FFFF'	Interface version is not supported
X'03'	X'00'	X'0000'	Warning: Information output is partial, user does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0014'	User does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0017'	User does not have SUBSYSTEM-MANAGEMENT privilege

Notes

- An MNOTE is generated if:
 - UTYPE =*UNIT and UNAME was not specified.
 - UTYPE =*FILE and INFF was not specified.
 - UTYPE =*ALL / *DIALOG and UNAME / INFF was specified.
 - UTYPE =*UNIT / *ALL / *DIALOG and OUTTYPE =*INFILE.
 - OUTTYPE =*FILE and OUTNAME was not specified.
 - OUTTYPE =*SYSOUT and REPLEV =*ALL.
- If UTYPE =*ALL / *DIALOG / *FILE is specified, the values specified for UNAME, UVERS and UCORSTA are ignored.
- If OUTTYPE =*SYSOUT / *SYSLST / *INFILE is specified, the value specified for WRMODE is ignored.
- If OUTTYPE =*SYSOUT / *FILE / *INFILE is specified, the value specified for SYSLST# is ignored.
- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

4.5.4 IMOSHSU Show supply units

The IMOSHSU macro enables you to view information on the supply units contained in the standard SCI, see ["Supply unit "](#).

Possible inputs are the supply units of a SOLIS2 delivery or a formatted file generated beforehand with the SHOW functions of IMON. You also have the option of selecting installation units interactively in interactive mode (see also "Selection of supply units" in the

menu mode auf ["View menu "](#)).

The output is directed to SYSOUT, to SYSLST, to the formatted file specified during input, or to another formatted file that can be generated, extended or overwritten.

This macro requires the SUBSYSTEM-MANAGEMENT privilege.

Macro	Operands
IMOSHSU	MF= D / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= MOS / <char (3)> ,PARAM= <adr> / (<reg>) ,UTYPE= *UNIT / *ALL / *DIALOG / *FILE ,UNAME= <c-string: 1..30 > / <var: char:30> ,UVERS= *ALL / *HIGH / <c-string: 3..5> /<var: char:5> ,UCORSTA= *ALL / *HIGH / *LOW / <c-string: 3..3> / <var: char:3> ,INFF= <c-string: 1..54> /<var: char:54> ,INSUNIT= *YES / *NO ,REPLEV= *MINIMUM / *ALL ,OUTTYPE= *SYSOUT / *INFILE / *FILE / *SYSLST ,OUTNAME= <c-string: 1..54> / <var: char:54> ,WRMODE= *REPLACE / *EXTEND ,SYSLST#= 0 / <integer 1..99> / <var: int:1>

Operands

The MF, PREFIX, MACID, and PARAM parameters are described in "Macro forms" on

["Macro syntax "](#).

UTYPE

Input mode used to select supply units.

=*UNIT

Select supply units by UNAME, UVERS and UCORSTA

=*ALL

Select all supply units of the SCI.

=*DIALOG

The supply units are selected interactively (see also "Selection of supply units" in the menu mode on "[View menu](#)").

=*FILE

Take the supply unit from a formatted file (INFF).

UNAME

Name of the supply unit to be shown.

=<c-string: 1..30>

Name of the supply unit.

=<var: char:30>

Symbolic address containing the specification for the name of the supply unit.

UVERS

Version of the supply unit.

=*ALL

Select all version designations of the supply unit.

=*HIGH

Select the highest version designation of the supply unit.

=<c-string: 3..5>

Specifies the version of the installation unit.

=<var: char:5>

Symbolic address containing the specification for the version of the supply unit.

UCORSTA

Correction state of the supply unit.

=*ALL

Select all correction states of the supply unit.

=*HIGH

Select the highest correction state of the supply unit.

=*LOW

Select the lowest correction state of the supply unit.

=<c-string: 1..3>

Specifies the correction state of the supply unit. Format: <aso>.

=<var: char:3>

Symbolic address containing the specification for the correction state of the supply unit.

INFF

Name of the formatted file to be used as input.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

INSUNIT

Show/do not show the installation units.

=*YES

Show the installation units.

=*NO

Do not show the installation units.

REPLEV

Reporting level for supply unit attributes.

=*MINIMUM

Show the names of the supply units only. If the delivery is a delta delivery, show only the supplied installation unit.

=*ALL

Show the names of the supply units for each correction state. If the delivery is a delta delivery, show only the supplied installation unit.

OUTTYPE

The target for output.

=*SYSOUT

Output the information to SYSOUT.

=*INFILE

Output the information to the formatted file used as input.

=*FILE

Output the information to a formatted file (OUTNAME).

=*SYSLST

Output the information to SYSLST.

OUTNAME

Name of the formatted file to be used as output.

=<c-string: 1..54>

Name of the formatted file.

=<var: char:54>

Symbolic address containing the name of the formatted file.

WRMODE

Overwrite or extend the file.

=*REPLACE

Generate the formatted file or overwrite an existing file.

=*EXTEND

Generate the formatted file or extend an existing file.

SYSLST#

SYSLST number used.

=0

Output the information to standard SYSLST.

=<integer 1..99>

Specifies the SYSLST number.

=<var: int:1>

Symbolic address containing the specification for the SYSLST number.

Macro return codes

(SC2)	SC1	Maincode	Meaning
-------	-----	----------	---------

X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	Function not permitted in batch mode
X'00'	X'01'	X'0002'	Output invalid
X'00'	X'01'	X'0003'	SYSLST number invalid
X'00'	X'01'	X'0004'	Parameter list invalid
X'00'	X'40'	X'0013'	Cannot access the SCI, function not executed
X'00'	X'40'	X'0014'	Specified unit not found in the SCI, function not executed, no formatted file created
X'00'	X'40'	X'0015'	Error accessing input file, function not executed
X'00'	X'40'	X'0016'	Error accessing output file, function not executed
X'00'	X'40'	X'0017'	Information missing. Units contained in the formatted file (UTYPE=*FILE) not present in the SCI
X'00'	X'40'	X'0019'	SHOW output written to SYSLST only partially or not at all
X'00'	X'40'	X'001A'	No SUBSYSTEM-MANAGEMENT privilege
	X'20'	X'0100'	System error
	X'20'	X'0101'	Internal error
X'00'	X'01'	X'FFFF'	Function is not supported
X'00'	X'03'	X'FFFF'	Interface version is not supported
X'03'	X'00'	X'0000'	Warning: Information output is partial, user does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0014'	User does not have SUBSYSTEM-MANAGEMENT privilege
X'03'	X'40'	X'0017'	User does not have SUBSYSTEM-MANAGEMENT privilege

Notes

- An MNOTE is generated if:
 - UTYPE =*UNIT and UNAME was not specified.
 - UTYPE =*FILE and INFF was not specified.
 - UTYPE =*ALL / *DIALOG and UNAME / INFF was specified.
 - UTYPE =*UNIT / *ALL / *DIALOG and OUTTYPE =*INFILE.
 - OUTTYPE =*FILE and OUTNAME was not specified.
 - OUTTYPE =*SYSOUT and REPLEV =*ALL.
- If UTYPE =*ALL / *DIALOG / *FILE is specified, the values specified for UNAME, UVERS and UCORSTA are ignored.
- If OUTTYPE =*SYSOUT / *SYSLST / *INFILE is specified, the value specified for WRMODE is ignored.
- If OUTTYPE =*SYSOUT / *FILE / *INFILE is specified, the value specified for SYSLST# is ignored.
- The information is edited so as to be legible if the output is directed to SYSOUT or SYSLST.
- If the information is directed to a formatted file, the output is compressed and is not legible for the user.

-
- If the information in the SCI changes after the formatted file is generated, the information that is output is no longer current.

4.5.5 Example illustrating output with macros

The program in this example outputs the data of one or more installation units in the SCI to SYSOUT. The products for which the data is output can be selected in an interactive dialog.

The corresponding examples using the FHS and SDF interfaces begin on [chapter "Installation sequences under BS2000"](#), respectively.

```
        ICTL 1,71,18
TCASM   CSECT
        PRINT GEN,NOREF
        ...
*
* //show-installation-units unit-name=*by-dialog, -
* //   information=*par(installation-items=*yes,report-level=*minimum), -
* //   output=test.ff(write-mode=*replace)
*
        IMOSHIU MF=E,PARAM=SHIUPL      call IMOSHIU
*
* //show-installation-items item-name=*from-formatted-file( -
* //   file-name=test.ff,information=*par(report-level=*all-attributes), -
* //   output=*input-formatted-file
*
        IMOSHII MF=E,PARAM=SHI IPL     call IMOSHII
*
* //show-formatted-file from-file=test.ff,output=*syslst
*
        IMOSHFF MF=E,PARAM=SHFFPL     call IMOSHFF
*
        ...
*
        DS    0F
*
SHIUPL  IMOSHIU MF=L,UTYPE=*DIALOG,INSITEM=*YES,REPLEV=*MINIMUM,
        OUTTYPE=*FILE,OUTNAME=TEST.FF,WRMODE=*REPLACE
```

*

SHIPL IMOSHII MF=L, ITYPE=*FILE, INFF=TEST.FF, REPLEV=*ALL,
OUTTYPE=*INFILE

*

SHFFPL IMOSHFF MF=L, INFF=TEST.FF, SYSLST#=0

*

END TCASM

5 Administering software

This chapter describes the files used by IMON to administer the software and how to administer the path names and product versions using IMON-GPN. Commands and macros are available to administer the path names and product versions.

It also describes the commands and macro calls for administering the path names and product versions.

5.1 The files of IMON

The result of each installation is stored in the Software Configuration Inventory (SCI). The SCI is used to administer the installed software.

The information for an installation is stored in diverse files:

- The IMON parameter files contain parameters for controlling the installation. These parameters can be customized by the user.
- The SYSSII file contains information on the structure of a product. The structure of a product corresponds to the layout of a release unit.
- The IDF file contains information about private software and BS2000 software that is already installed in the system and is to be registered in the SCI.
- The IMON reference files contain information for the software configuration check.

5.1.1 Software Configuration Inventory (SCI)

The Software Configuration Inventory (SCI) is the central database for IMON. The SCI contains the information detailing the supply units/installation units administered by IMON and their installation items.

From the logical point of view IMON administers a single SCI. In physical terms the SCI is divided into two files:

- The IMON-SCI is managed by IMON-BAS.
- The IMON-GPN-SCI is managed by IMON-GPN.

Internal interfaces ensure that the two files remain consistent at all times.

The SCI contains information (attributes)

- on each installed supply unit (see "[Supply unit](#) ")
- on each installation unit (see "[Installation unit \(IU\)](#) ")
- on each installation item (see "[Installation item \(II\)](#) ")
- for each delivery (package), which is processed.

The [section "Administering installation units"](#) describes how installation units are registered in the SCI, how information is output from the SCI, and how installation units are removed from the SCI.

Standard and foreign SCIs

IMON can process a standard SCI or a foreign SCI. The standard SCI is the SCI of the current system in which IMON was started. It is located on the HOME pubset and is always called :<home-catid>:\$TSOS.SYS.IMON.SCI, unless it is the SCI of IMON-GPN, in which case the name is :<home-catid>:\$TSOS.SYS.IMON.SCI.GPN. The standard SCI contains information on the products installed in the current system.

If the standard SCI does not already exist, it is created by IMON on startup.

In addition to the standard SCI, IMON can also be used to generate or process a foreign SCI with a freely selectable name. This is the case, for example, when creating backup copies of the SCIs or if a separate SCI is to be used for parking. If the foreign SCI is the standard SCI of another BS2000 system and it prepares the installation of products on this system, the fixed file names :<catid>:\$TSOS.SYS.IMON.SCI or :<catid>:\$TSOS.SYS.IMON.SCI.GPN are used, where <catid> is the catalog ID of an imported pubset.

Only one SCI can be open at any time in an IMON session. When IMON is started in SDF mode, the standard SCI of the running system is opened. When started in menu mode, no SCI is opened initially, but the standard SCI of the running system is used implicitly when a delivery is opened.

The SCI to be edited can be opened using the *Open* option in the *File* menu or with the MODIFY-IMON-OPTIONS statement. In menu mode, any SCI already opened before must be closed explicitly (*Close* option in the *File* menu). During installation, the standard SCI of the target system is updated. If there is another SCI open (e.g. a special park SCI), then the user can select whether the currently open SCI or the standard SCI of the target system is updated.

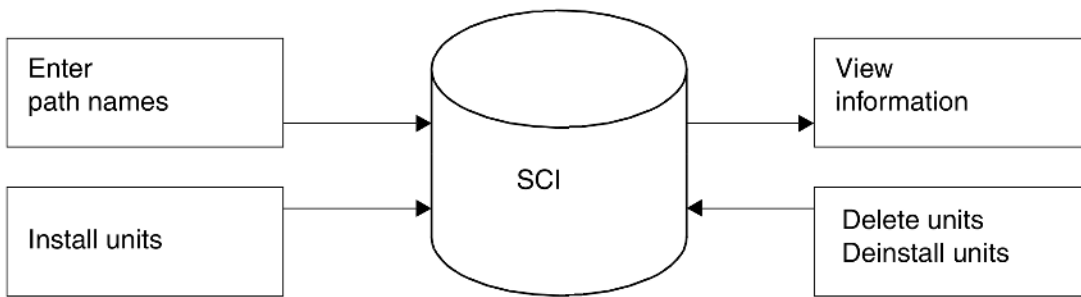


Figure 62: Ways of accessing the SCI

Backing up the SCI

The SCI files should be backed up at regular intervals. A backup copy should also be created before extensive changes are made (in particular the deletion of entries).

The current status of the open SCI can be backed up via the *Save as* menu option in the *File* menu or using the SAVE-SOFTWARE-INVENTORY statement. This creates consistent backup copies for both files (IMON-SCI and IMON-GPN-SCI). Copying the files would not guarantee this.

Entries in the SCI can be written to an IDF file via the export/import function (see "[Exporting and importing SCI entries](#)"). In this process, information from the SCI on supply units or installation units can be saved and inserted into another SCI.

If the IMON subsystem is unloaded, the standard SCI can be backed up with the command SAVE-SOFTWARE-INVENTORY or rebuilt from a backup copy with the command RESTORE-SOFTWARE-INVENTORY.

Automatic backup

The SCI files are backed up automatically in the following cases:

- The standard SCI is backed up each time the system starts. The backup copies of the SCI files are created with the suffix SAV (i.e. \$TSOS.SYS.IMON.SCI.SAV and \$TSOS.SYS.IMON.SCI.SAV.GPN). Existing backup copies are overwritten.
- The SCI current open is backed up during the installation (see "[Important files in the installation \(The troubleshooting process\)](#)") on "[The troubleshooting process](#)"). The automatic backup function can be suppressed explicitly if you are using an IMON parameter file (see "[IMON parameter files](#)").

Protecting the SCI from unauthorized access

The SCI is a system file and, like any other file assigned to a user ID, it is protected against unauthorized access by DMS, the file management system in BS2000. The lowest protection level is USER-ACCESS=*OWNER-ONLY, which IMON automatically sets when it creates the SCI.

IMON is the only product which has read and write access to the SCI.

- Users with the SUBSYSTEM-MANAGEMENT privilege and applications in TPR mode can fetch information on all products installed in the system and read all path names.
- Users with the STD-PROCESSING privilege and applications in TU mode can only fetch information on the nonprivileged products in the system. These users and applications receive information only on the path names to which they are permitted access.

Neither can they create new SCIs.

5.1.2 IMON parameter files

IMON uses the following parameter files:

- [IMON parameter file for optional installation functions](#)
- [IMON parameter file for default installation parameters](#)
- [Parameter file for mail configuration](#)

5.1.2.1 IMON parameter file for optional installation functions

The user can control installation functions using an IMON parameter file.

An example of a IMON parameter file is supplied with IMON under the standard name SYSPAR.IMON-BAS.<version>. This file contains all IMON parameters with their default settings. In addition the file contains comments in the German and English languages that describe the function and optional values of each of the parameters.

To generate the installation procedure, IMON first evaluates the parameters that the user specified when calling the installation function. If a customer-specific IMON parameter file with the default name \$TSOS.SYSPAR.IMON.<customer_code> exists, IMON also evaluates the parameters in this file. If it does not exist IMON evaluates the file \$TSOS.SYSPAR.IMON. The parameter file is evaluated for all installations of deliveries with the customer code defined in the name.

The user can create the customer-specific IMON parameter file by copying the standard parameter file and modifying the parameters to meet his or her requirements. If there is no customer-specific parameter file, the installation procedure can be generated with the default settings documented in the IMON parameter file supplied.

The standard parameter file has been extended and reorganized. It now consists of a dynamic and a static section:

The dynamic section

In the dynamic section the user can enter the name of the global DSSM catalog and installation user ID(s) for certain supply units. The dynamic section also aids in automating the installation procedure even for nonstandard installations. The preset values for this automation are displayed when the corresponding query screens are output and can be changed by the user.

The dynamic section should be located at the beginning of the IMON parameter file for performance reasons, i.e. before the static section. It starts with the keyword START-DYN-PAR and ends with the keyword END-DYN-PAR. The lines of the dynamic section begin with the character string "****". The standard parameter file supplied already contains a dynamic section that consists only of the beginning and end lines. You can enter input lines for the desired settings between these two lines:

- Preset for the DSSM catalog:

The line containing the new default value for the name of the DSSM catalog (SYS.SSD.CAT.X is the preset value) begins with the keyword DSSM-CAT. The new name is specified in the following form:

```
****DSSM-CAT <filename 1..48 without-gen-vers>
```

- Preset value for the installation ID for a supply unit:

The preset value for an installation ID for a supply unit begins with the keyword SU. The name of the supply unit and the name of the installation ID follow. The preset value is specified in the following form:

```
****SU <text 1..30 without-sep> <name 1..8>
```

If the preset values for the installation ID(s) for several supply units are to be changed, the corresponding number of input lines must be added.

See "[IMON parameter file for optional installation functions](#)" for an example.

The static section

This section contains all optional IMON parameters with the default settings, which can only be modified using the parameters file. It also contains comments in English and German explaining the functions of the individual parameters and their optional values.

Comment lines start with the character "***". Input lines that are evaluated during generation of the installation procedure start with "*" and contain statements for the generation program. Parameter values are defined in the following format using the SET statement:

```
SET <parameter_name> "<value>"
```

Example of a parameter from the default parameter file

```
* SET ZVUNLOAD "Y"
```

"Y" is defined by default for the ZVUNLOAD parameter, i.e. the volume is unloaded automatically after installation.

The following functions are preset in the static part of the IMON parameter file and can, if desired, be changed. The presettings in the default parameter file correspond also to the behavior of IMON V2.2:

- The reference file \$SERVICE.SOLREF.IMON.SW-CONF is generated.
Generation of this reference file can be suppressed.
- Data volumes used for installation are unloaded automatically.
Unloading of the data volumes can be suppressed.
- The SCI files are backed up automatically before they are registered (see ["Backing up the SCI" \(Software Configuration Inventory \(SCI\)\)](#)). Backing up of the SCI files can be suppressed.
- The source file for creating the DSSM catalog is created or expanded under the name <dssm-catalog>.SRC.
In addition you can specify that the source file will not be appended to and is to be recreated for each installation instead.
The name of the DSSM catalog to be expanded can be specified explicitly or creation of the source file can be suppressed.
- The installation result is registered in the standard SCI of the target system.
You have the option of defining that the installation result is to be registered in the SCI currently open. If the SCI currently open is set, the dialog query is suppressed (see ["Example with SDF statements"](#) for an example of the dialog query).
- The SYSSII files are incorporated into the target system and incorporated into the SOLLIB.IMON.SYSSII library under the work file ID (possibly with a prefix).
You have the option of suppressing transfer of the SYSSII files.
- There is no special storage location defined for documentation files (Release Notices and readme files; item type *FE, *FG and files whose names begin with SYSSGM. or SYSSRME.). They are installed under the installation ID.
You have the option of setting a central storage location for documentation files.
- There is no special storage location defined for subsystem declarations (item type SSC, SSD and files whose names begin with SYSSSC. or SYSSSD.). They are installed under the installation ID.
You have the option of setting a central storage location for subsystem declarations.
- The save data media created by IMON during installation and parking do not have deletion or overwrite protection. You can also optionally specify a retention period from 1 to 999 days.
- REP loaders can be generated when parking a delivery. You can also optionally specify that the REP files are only to be added to the RMS depot.
- If the files to be installed already exist, then their previous file attributes (ACCESS, USER-ACCESS and MIGRATE) are kept. You can also optionally specify that the file attributes are set to match the data specified for the SOLIS2 delivery.

- The product movement file (PMF) for each delivery is saved in the customer system in the PLAM library \$<work file ID>.SOLLIB.IMON.PBD. You can also optionally specify that this file is to be deleted after installation without making a backup copy.
- The product movement file (PBD) for each delivery is saved in the customer system in the PLAM library \$<work file ID>.SOLLIB.IMON.PBD. You can optionally specify that this PBD file is to be deleted at the end of the installation process without saving it in the PLAM library.
- Optionally it can be agreed that in customer systems which have a default user ID which is not TSOS, \$.TASKLIB and \$.MACROLIB should be processed instead of \$TSOS.TASKLIB and \$TSOS.MACROLIB.
- Optionally a specific tape repository can be employed for robot use. The tape repository must be specified in the /SECURE-RESOURCE-ALLOCATION command for processing the SOLIS delivery medium.

Information on using central storage locations

The default for each of the parameters for specifying central storage locations is a null string (""). If you intend to use a central storage location, the corresponding parameter value must be changed in "<storage_location>" in the customer-specific IMON parameter file, where <storage_location> must be specified as a <partial-filename 1..23>.

Input format: [:<catid>:][\${<userid>}.][<prefix>.]

IMON expands partial storage locations (i.e. where the catalog or user ID is missing) in accordance with the following rules:

- The TSOS user ID is assumed if no user ID is specified.
- The catalog ID is specified as a function of the target system if not catalog ID is specified:
 - In the case of an installation on the home pubset, the default catalog ID of the specified or an expanded user ID is assumed.
 - In the case of an installation on an imported pubset, the catalog ID of the imported pubset is assumed.

Example

The user with the customer ID CUST01 wishes to automate an installation procedure. The user does not use the standard DSSM catalog, but instead uses a DSSM catalog named SYS.SSD.CAT.SYS1. For some supply units the user has defined some of his or her own installation user IDs to reserve some storage space for the TSOS user ID. The special user IDs are TOOLS and COMPIL in this example \$TOOLS for the supply units PERCON and LMS \$COMPIL for the supply units COBOL85 and PLI1. Furthermore, the documentation files are to be stored centrally on the HOME pubset under the RZINFO user ID.

Procedure

1. Create a copy of the standard parameter file supplied with IMON V3.2 (file name SYSPAR.IMON-BAS.032) under the user-specific name (\$TSOS.SYSPAR.IMON.<customer ID>):

```
/copy-file from-file=$.SYSPAR.IMON-BAS.032,  
           to-file=$TSOS.SYSPAR.IMON.CUST01
```

2. Edit the user-specific parameter file:

The presets to be changed for the DSSM catalog and the desired installation IDs are entered in the dynamic section:

```
****START-DYN-PAR
****DSSM-CAT  $TSOS.SYS.SSD.CAT.SYS1
****SU  PERCON      TOOLS
****SU  LMS         TOOLS
****SU  COBOL85     COMPIL
****SU  PLI1        COMPIL
****END-DYN-PAR
```

The changes to the preset for the storage location of the documentation files are made in the dynamic section. To do this, the preset for the ZVDOCPRF parameter

```
* SET ZVDOCPRF  " "
```

must be changed to:

```
* SET ZVDOCPRF  " :HOME:$RZINFO. "
```

5.1.2.2 IMON parameter file for default installation parameters

The majority of the non standard installation options get through the different dialog boxes may be defined in the parameter file for default installation parameters and passed to IMON at installation time.

This parameter file (default name is \$TSOS.SYSPAR.IMON.LAST) can be produced by the customer itself using any standard editor (e.g. EDT) or may be produced automatically by IMON at installation time when using the FHS interface.

This file is used by IMON (on user request) to assign to the *STD keyword of the //INSTALL-UNITS statement (or to prefill the associated input fields in the FHS interface) default values either defined/modified by the customer or saved by IMON from the last performed installation. Irrespective of this, IMON always evaluates the customer-specific

IMON parameter file \$SYSPAR.IMON.<kkz> which optionally exists (see [section "IMON parameter file for optional installation functions"](#)). If the entries are identical, IMON uses the values from the parameter file for default installation parameters.

Structure of the parameter file

```
*** IMON INSTALLATION PARAMETERS *** (mandatory identification line)
PVS-INFO= <catid 1..4> <defluid 1..8/*STD> <S/R/P/A> <Y/N> _____ (1)
OLD-FILE-SAVING= N/L/M <devtype> <maren loc> _____ (2)
UNDO-PREPARATION= N/Y
PRINT-LOG-FILES= Y/N
CONFIGURATION-CHECK= Y/N
WORK-FILE-DELETING= Y/N
MIP-PROCESSING= Y/N
SDF-PROCESSING= Y/N
POSIX-PROCESSING= C/A/Y/N
DSSM-PROCESSING= N/Y <catalogname/*STD> _____ (3)
RMS-PROCESSING= N/Y <depot-loc./*STD>/D <depot-loc./*STD> _____ (4)
SU-INFO= <name> <vers/*ALL> <catid/*DEF> <userid/*STD> <pref/*NONE> <Y/N> (5)
```

- (1) 1 line per target system with catid, defluid value, HW variant, all items to install
- (2) no file saving, or with LMS, or with MAREN (with specification of the device type and the MAREN location)
- (3) <dssm catalog name> = <filename 1..48-without-catid>, *STD means \$TSOS.SYS.SSD.CAT.X
- (4) <depot-loc.> = [:catid:]\$.userid, *STD = :<catid of target system>.\$TSOS
- (5) 1 line per supply unit with SU name, main version or *ALL, delivery installation pubset, installation userid, installation prefix, keep old version in DSSM or not.

Observe the following rules:

- All lines are optional (except the first one) but when IMON creates the parameter file, it will fill and write all lines.
- New SU-INFO line will be always written with version = *ALL
- Any unknown keyword is ignored without any message
- All keywords must start on first column
- No blank must stay between the keyword and the "=" sign but at least one blank must be used after this "=" and as separator between different values for a same keyword.

-
- Any duplicate keyword will overwrite info from a previous one without any message.
 - All entered values will be syntactically and semantically validated.
 - A first occurred error, the validation will stop with an error message indicating the line number where the error occurred.

5.1.2.3 Parameter file for mail configuration

When the “request correction deliveries” function is called (using the REQUEST-CORRECTION-DELIVERY statement or the *Edit:Request correction delivery* option), IMON creates a mail that contains the request and the desired data. IMON uses the following file as the parameter file for mail configuration.

\$TSOS.SYSPAR.IMON.<customer_ID>.MAIL.CONFIG

As of Version 2.9 IMON creates this parameter file by default on the basis of the delivery information. If necessary, the parameter file can also be created or edited manually. It is then important to note that the parameter file is updated or created anew when a new delivery is installed. This means that the data in the parameter file always reflects the current status as registered with Fujitsu.

Structure of the parameter file

```
REGIO:<customer region code> _____ (1)
TO:<Fujitsu Siemens Computer delivery center email address> _____ (2)
FROM:<customer email address>| _____ (3)
DEVICES:<device1> [<devicen>] _____ (4)
```

- (1) The first line is mandatory (subsequent lines are optional). “REGIO:” is followed by the region code (string comprising 3 letters) of the regional distribution center of Fujitsu that is responsible for the customer. IMON needs this information to generate the formatted request with the file name \$TSOS.IMON.DELREQ.<timestamp>.
- (2) “TO:” is followed by the mail address of the recipient, i.e. the mail address of the relevant distribution center of Fujitsu.
- (3) “FROM:” is followed by the mail address of the sender. If the correction delivery is to be made available via the WWW, the mail address of the system administrator responsible for the installation must specified here because notification information including necessary access data is sent to this mail address once the correction delivery has been made available.
- (4) “DEVICES:” is followed by one to eight volume types (blank-separated) that can be used for distribution.

IMON only needs the mail addresses if the generated request is to be sent automatically.

Changes to the parameter file should be made only if the mail address of the sender is missing or needs to be corrected. The corrected mail address should then also be communicated to the distribution center of Fujitsu so that this information is correct for future deliveries and is correctly generated in the parameter file.

5.1.3 Structure and Installation Information file (SYSSII file)

The SYSSII file is a product file (release item) supplied with each software product (release unit) from Fujitsu. A SYSSII file contains information on the structure of a product.

During installation for a target system version, the information about all SYSSII files of a delivery is contained in the product movement file, and is evaluated from there. The SYSSII file of the product is not installed in the system, but only included in the library SOLLIB.IMON.SYSSII with the currently set work file ID (with a prefix where applicable) (this can be suppressed in a user-defined IMON parameter file).

The SYSSII file contains information that is of significance as regards the installation of a product and the assignment of path names to logical IDs.

The information in the SYSSII file is as follows:

- name and version of the release unit
- list of the release items in the release unit
- definition of the logical ID for each individual release item.

A SYSSII file is always a description of a complete release unit with all its release items. This also applies to a correction delivery containing only new or modified items. IMON evaluates the SYSSII file or the product movement file when it registers release units and release items in the SCI as installation units and installation items.

A SYSSII file can be used as an input file with the *Add...* menu option in the *Edit* menu or with the ADD-INSTALLATION-UNITS statement in order to register the installation units and installation items it contains in the SCI.

You can use IMON-SIC to generate a SYSSII file for your private software or BS2000 software that was not installed with IMON. IMON-SIC enables you to assign a logical ID to each release item in this software: these assignments are written into the SYSSII file. Once this software has been registered in the SCI, you can employ the logical IDs to utilize the functions of IMON-GPN for this software.

5.1.4 Installation Definition File (IDF file)

Products can be registered in the SCI even if they were not installed with IMON. The user must generate an IDF (Installation Definition File) for these products.

An installed product corresponds to an installation unit. The individual installed product files correspond to the installation items.

The IDF file generated in this way is the input file for the *Add...* menu option in the *Edit* menu or the ADD-INSTALLATION-UNITS statement for registering in the SCI the installation units and installation items contained in the file.

Structure of the IDF file

The IDF is a SAM or ISAM file with a maximum record length of 2032 bytes. Any suitable editor can be used to generate and edit the IDF.

The individual records are structured in accordance with the syntax rules described below.

The syntax consists of keywords (in uppercase letters) and parameters. The rules are as follows:

- Blanks and end-of-record markers are separators. Multiple consecutive blanks or end-of-record markers are interpreted as one separator.
- A comment is introduced by the keyword *REMARK.
- Square brackets [...] enclose optional information.
- Uppercase and lowercase letters are permissible.
- You can specify multiple *ITEM ... *FILE lines for a *UNIT . Multiple *UNIT ... *ITEM ... *FILE specifications can be strung together.
- An installation unit (*UNIT) cannot be described more than once in an IDF.
- An installation item (*ITEM) can be defined only once in an installation unit (*UNIT).

Syntax in the IDF

```
*UNIT <IU-name>'BLANK'<IU-version>'BLANK'<IU-correction-state>
  *ITEM [ <II-name>'BLANK'<II-version>'BLANK'<II-type> ]
    *FILE <filename 1..54 with-cat-user without-gen-vers>
```

Description of the keywords and parameter values

*UNIT	Description of an installation unit (see "Installation unit (IU)")
<IU-name>	Name of the installation unit Syntax: <text 1..30 without-sep>.
<IU-version>	Version designation of the installation unit. format: "[m]m.n"
<IU-correction-state>	Correction state of the installation unit. Must consist of one alpha character followed by two digits Syntax: <alphanum-name 3..3> Example: A00.

*ITEM	Description of an installation item (see " Installation item (II) ") belonging to the preceding installation unit.
<II-name>	Name of the installation item Syntax: <filename 1..30 without-cat-user-gen-vers>
<II-version>	Version of the installation item. The value set can be up to 5 characters in length Syntax: <char 1..5>
<II-type>	Type of the installation item (see " Path name, logical ID and type of an installation item ") Syntax: <char 3..3>
*FILE	The path name of the installation item with catalog and user IDs Syntax: <filename 1..54 with-cat-user without-gen-vers>
	The following substitutions are automatic:
	<ul style="list-style-type: none"> • If the path name has an alias, the alias is substituted • A default is inserted if the user ID is omitted (user ID of the current task or TSOS, if the TSOS privilege is available) • \$.<name> causes the system default user ID (DEFLUID) to be inserted • The standard catalog ID (DEFCAT) of the user ID is inserted if the catalog ID is omitted

Example of an IDF file for ARCHIVE

ARCHIVE V12.0 was installed without IMON and is now to be registered in the SCI file via an IDF file using the ADD-INSTALLATION-UNITS statement (or the corresponding *Add ...* option in the *Edit* menu):

*REMARK Example of an IDF file for ARCHIVE

*UNIT ARCHIVE 12.0 C00

```
*ITEM SKMLNK.ARCHIVE.120.TPR      001  DAT
      *FILE :AB01:$TSOS.SKMLNK.ARCHIVE.120.TPR
*ITEM SYSFGM.ARCHIVE.120.D        001  *FG
      *FILE :AB01:$TSOS.SYSFGM.ARCHIVE.120.D
*ITEM SYSFGM.ARCHIVE.120.E        001  *FE
      *FILE :AB01:$TSOS.SYSFGM.ARCHIVE.120.E
*ITEM SYSLIB.ARCHIVE.120          001  DAT
      *FILE :AB01:$TSOS.SYSLIB.ARCHIVE.120
*ITEM SYSLNK.ARCHIVE.120          001  DAT
      *FILE :AB01:$TSOS.SYSLNK.ARCHIVE.120
*ITEM SYSLNK.ARCHIVE.120.TPR      001  DAT
      *FILE :AB01:$TSOS.SYSLNK.ARCHIVE.120.TPR
*ITEM SYSMES.ARCHIVE.120          001  MES
      *FILE :AB01:$TSOS.SYSMES.ARCHIVE.120
*ITEM SYSNRF.ARCHIVE.120          001  DAT
      *FILE :AB01:$TSOS.SYSNRF.ARCHIVE.120
*ITEM SYSPAR.ARCHIVE.120          001  *NW
      *FILE :AB01:$TSOS.SYSPAR.ARCHIVE.120
*ITEM SYSPRG.ARCHIVE.120          001  DAT
      *FILE :AB01:$TSOS.SYSPRG.ARCHIVE.120
*ITEM SYSPRG.ARCHIVE.120.DIRCONV  001  DAT
      *FILE :AB01:$TSOS.SYSPRG.ARCHIVE.120.DIRCONV
*ITEM SYSREP.ARCHIVE.120          001  *DF
      *FILE :AB01:$TSOS.SYSREP.ARCHIVE.120
*ITEM SYSRME.ARCHIVE.120.D        001  *FG
      *FILE :AB01:$TSOS.SYSRME.ARCHIVE.120.D
*ITEM SYSRME.ARCHIVE.120.E        001  *FE
      *FILE :AB01:$TSOS.SYSRME.ARCHIVE.120.E
*ITEM SYSRMS.ARCHIVE.120          001  REP
      *FILE :AB01:$TSOS.SYSRMS.ARCHIVE.120
*ITEM SYSSDF.ARCHIVE.120          001  SDF
      *FILE :AB01:$TSOS.SYSSDF.ARCHIVE.120
*ITEM SYSSII.ARCHIVE.120          001  DAT
      *FILE :AB01:$TSOS.SYSSII.ARCHIVE.120
*ITEM SYSSSC.ARCHIVE.120          001  SSC
      *FILE :AB01:$TSOS.SYSSSC.ARCHIVE.120
```

5.1.5 Installation definition file (IDF file, internal format)

IMON uses an internal format of an IDF file for the “Exporting/importing SCI entries” function.

In the case of an export operation (*Generate installation definition file* menu option in the *Edit* menu or GENERATE-IDF statement), IMON generates an IDF file in which the SCI entries of selected supply components or installation units are described in full. The data records of the internal IDF file are part of the generated import procedure and are read there by SYSDDTA using the ADD-INSTALLATION-UNITS statement.

The individual records have a structured format in accordance with the following syntax. The syntax consists of keywords (in uppercase letters) and parameters. Blanks and end of record characters are used as separators; a number of blanks or end of record characters are interpreted as one separator.

An IDF file in internal format should only be created using the export function as manual processing could result in structural inconsistencies that could then lead to a defective SCI being produced during the import operation.

Syntax in the generated IDF file

Syntax when selecting one or more supply units:

```
*GEN-IDF

*GEN-IDF

*DEL-ID <package name> <user code>

*SU <SU name1> <SU version> <SU correction state>
:
:      Description of the relevant installation units
:

*DEL-ID <package name> <user code>

*SU <SU name2> <SU version> <SU correction state>
:
:      Description of the relevant installation units
:

*END
```

Syntax when selecting one or more installation units:

*GEN-IDF
*GEN-IDF
*IU <IU namel> <IU version> <IU correction state> <lost+found>
*IU-ATTR <functional level> <BS2000 version>
:
: *Description of the relevant installation items*
:
*IU <IU name2> <IU version> <IU correction state>
<lost+found>
*IU-ATTR <functional level> <BS2000 version>
:
: *Description of the relevant installation items*
:
*END

Subsyntax for describing an installation unit:

*IU <IU namel> <IU version> <IU correction state>
<lost+found>
*IU-ATTR <functional level> <BS2000 version>
:
: *Description of the relevant installation items*
:

Subsyntax for describing an installation item:

*ITEM <II name> <II version> <II type>
*II-ATTR <functional level> <user-access> <migrate> <access>
<format>
<target>
*LOG-ID <logical id> <GPN-path>
*LOG-ID-ATTR <mandatory> <updatable> :

depends on the item type:

*FILE <filename 1..54 with-cat-user-without-gen-vers>

or

*MERGED <filename 1..54 with-cat-user-without-gen-vers>

or

*DF <filename 1..54 with-cat-user-without-gen-vers>

Description of the keywords and parameter values

*GEN-IDF	Identifies the start of the generated IDF file (data record 1 and 2)
*DEL-ID	Identifies the delivery for the supply unit described subsequently (*SU)
<package name>	Package name
<user code>	User code
*SU	Description of the supply unit (consists of one or more installation units)
<SU name>	Name of the supply unit
<SU version>	Version of the supply unit
<SU correction-state >	Modification state of the supply unit
*IU	Description of an installation unit
<IU name>	Name of the installation unit
<IU version>	Version of the installation unit
<IU correction-state>	Modification state of the installation unit
<lost+found>	Assignment identifier: Y = Assigned to Lost+Found N = Installation unit could not assigned

*IU-ATTR	Attribute description of the installation unit
<functional level>	Functional state in which the installation unit is running:
	U TU
	P TPR or SIH
	B Can run in both functional states
<BS2000 version>	BS2000 version:
	*NONE No version information
	190 V19.0
	200 V20.0
	210 V21.0
*ITEM	Description of an installation item
<II name>	Name of the installation item: <filename-without-cat-user-gen-vers 1..30>
<II version>	Version of the installation item
<II type>	Type of the installation item

*II-ATTR

Attribute description of the installation item

<functional level>

Functional statement in which the installation item is running:

- U TU
- P TPR or SIH
- B Can run in both functional states
- * Undefined

<user access>

Access right (USER-ACCESS) for the assigned file:

- A All users
- O Owner
- S All users including Service department
- * Undefined

<migrate>

Migration identifier and backup class of the assigned file:

- S MIGRATE=*STD and BACKUP-CLASS=*STD
- I MIGRATE=*INHIBIT and BACKUP-CLASS=*STD
- E MIGRATE=*INHIBIT and BACKUP-CLASS=E
- * Undefined

<access>

Access right (ACCESS) for the assigned file:

- R Read only
- W Write
- * Undefined

<format>

Format of the assigned file:

- K PAMKEY
- 2 NK2
- 4 NK4
- * Undefined

<target>

Hardware dependencies:

- K X86 mode
- A hardware-independent (ANY)
- S /390 mode
- * Undefined

*LOG-ID	Logical name of the installation item and assigned path name
<logical-id>	Logical name of the installation item
<GPNpath>	GPN path name of the installation item, if it exists and is fully qualified: <i><filename 1..54 with-cat-user-without-gen-vers></i> If it does not exist: *NONE
*LOG-ID-ATTR	Attribute description of the logical name
<mandatory>	Identifies whether the path name is mandatory
	Y Yes
	N No
<updatable>	Identifies whether the path name can be updated
	Y Yes
	N No
*FILE	Name of the installation item
*MERGED	Name of a merged installation item (library elements mixed in)
*DF	Dummy item of the type *DF
<filename>	Fully qualified filename
*END	Identifies the end of the generated IDF file (last data record).

Example of a generated IDF file (in an import procedure)

```

/SET-PROCEDURE-OPTIONS DATA-ESCAPE-CHAR='&&'
/DECL-PARAMETER NAME=( SCI -
/ ( INITIAL-VALUE= ' *STD' ) )
/ASSIGN-SYSOUT TRANSFERSCI.MAR301501492021.GI
/ASSIGN-SYSDTA *SYSCMD
/REMARK :4H21:$TSOS.SYS.IMON.SCI
/START-IMON
//MODIFY-IMON-OPTIONS SOFTWARE-INVENTORY = &SCI
//ADD-INSTALLATION-UNITS FROM-FILE = *SYSDTA
*GEN-IDF
*GEN-IDF
*IU IMON-BAS 03.4 A00 N
installation unit IMON-BAS

```

Definition of the

*IU-ATTR B *NONE
*IU-ACT NS 255 N
*ITEM SINLIB.IMON-BAS.034 001 *NP
*II-ATTR U A S R 4 A
*LOG-ID SINLIB :4H21:\$TSOS.SINLIB.IMON-BAS.034
*LOG-ID-ATTR Y N
*FILE :4H21:\$TSOS.SINLIB.IMON-BAS.034

.
.
Not all of the output is shown here.
.

*ITEM SYSSSC.IMON-BAS.034 001 SSC
*II-ATTR P O S R 4 A
*LOG-ID SYSSSC :4H21:\$TSOS.SYSSSC.IMON-BAS.034
*LOG-ID-ATTR Y N
*FILE :4H21:\$TSOS.SYSSSC.IMON-BAS.034

*IU IMON-GPN 03.4 A00 N
installation unit IMON-GPN

Definition of the

*IU-ATTR B *NONE
*IU-ACT NS 255 N
*ITEM SKMLNK.IMON-GPN.034 001 DAT
*II-ATTR P O I R 4 K
*LOG-ID SYSLNK :4H21:\$TSOS.SKMLNK.IMON-GPN.034
*LOG-ID-ATTR Y N
*FILE :4H21:\$TSOS.SKMLNK.IMON-GPN.034
ITEM SYSLIB.IMON-GPN.034 001 PL
*II-ATTR B A S R 4 A
*LOG-ID SYSLIB :4H21:\$TSOS.SYSLIB.IMON-GPN.034
*LOG-ID-ATTR Y Y
*FILE :4H21:\$TSOS.SYSLIB.IMON-GPN.034

.
.
Not all of the output is shown here.
.

*ITEM SYSSSC.IMON-GPN.034 001 SSC
*II-ATTR U O S R 4 A
*LOG-ID SYSSSC :4H21:\$TSOS.SYSSSC.IMON-GPN.034
*LOG-ID-ATTR Y N
*FILE :4H21:\$TSOS.SYSSSC.IMON-GPN.034

*IU IMON-SIC 03.4 A00 N
installation unit IMON-SIC

Definition of the

*IU-ATTR U *NONE
*IU-ACT NS 255 N
*ITEM SYSFHS.IMON-SIC.034 001 *DF
*II-ATTR U A S R 4 A
*LOG-ID SYSFHS *NONE
*LOG-ID-ATTR N Y
*ITEM SYSFHS.IMON-SIC.034.D 001 DAT
*II-ATTR U A S R 4 A

```
*LOG-ID SYSFHS.D :4H21:$TSOS.SYSFHS.IMON-SIC.034.D
*LOG-ID-ATTR Y Y
*FILE :4H21:$TSOS.SYSFHS.IMON-SIC.034.D
*ITEM SYSFHS.IMON-SIC.034.E 001 DAT
*II-ATTR U A S R 4 A
*LOG-ID SYSFHS.E :4H21:$TSOS.SYSFHS.IMON-SIC.034.E
*LOG-ID-ATTR Y Y
*FILE :4H21:$TSOS.SYSFHS.IMON-SIC.034.E
.
.      Not all of the output is shown here.
.
*FILE :4H21:$TSOS.SYSSII.IMON-SIC.034
*END
//END
/ASSIGN-SYSOUT *PRIMARY
/ASSIGN-SYSDTA *PRIMARY
/ENDP
```

5.1.6 IMON reference files

Reference file for system customer service

During installation the reference file `$SERVICE.SOLREF.IMON.SW-CONF` is generated (this can be optionally suppressed, see [section "IMON parameter files"](#)). This reference file is exclusively for the analysis of the software configuration through the system customer service. It is generated using the following statement.

```
//SHOW-SUPPLY-UNITS INFORMATION=*PAR(REPORT-LEVEL=*FOR-REFERENCE-FILE)
```

It contains, per `USER-CODE`, the last installed correction state (based on the registered installation time information) of each main version of each supply unit.

Example of reference file layout

```
      SCI Name = :Q80K:$TSOS.SYS.IMON.SCI
=====
Supply Unit Name          Version Package      Code      Date      Time
-----
VM2000                    12.0A00 <package-name> SOL2P    2020-11-25 11:43:53
AID                       03.5A00 <package-name> 80QA2    2020-07-01 16:02:44
ASSEMBH                   01.4A00 <package-name> 80QA2    2020-07-01 16:02:44
ASSTRAN                   05.4A00 <package-name> 80QA2    2020-07-01 16:02:44
AVAS                      08.6A00 <package-name> 80QA2    2020-07-01 16:02:44
AVAS-SV-BS2              08.6A00 <package-name> 80QA2    2020-07-01 16:02:44
BS2GA.APACHE             02.4A00 <package-name> 80QA2    2020-07-01 16:02:44
BS2GA.CRTE-BAS          21.0A00 <package-name> 80QA2    2020-07-01 16:02:44
BS2GA.DSSM               21.0A00 <package-name> 80QA2    2020-07-01 16:02:44
...
UDS/SQL                   02.9B60 <package-name> 80QA2    2020-07-01 16:02:57
```

Note

All dates and times in the `$SERVICE.SOLREF.IMON.SW-CONF` file, as well as in the global SCI, are always GMT ones and may so differ from the real date and time from customer system.

Reference file for the IMON configuration check

For the software configuration check during the installation, IMON uses a further reference file with the standard name `$TSOS.SYS.IMON.SCI.REF`. Another name can be set in the IMON options (see operand `REFERENCE-FILE` in the statement `MODIFY-IMON-OPTIONS` or the field *Reference File* in the dialog box *IMON Options*).

In the configuration check a log (`<workfile location>.<package>.<timestamp>.DP`) lists the dependencies of the supply units to be installed on the supply units already installed.

Example of the content of a log file

Target system informations

```

-----
Version          : 170
SCI              : :CATID:$TSOS.SYS.IMON.SCI
Reference File   : :CATID:$TSOS.SYS.IMON.SCI.REF
Report for Supply Unit : SWLETEST6          01.0A00
-----

```

Installable on the target system : yes

Dependence				Status	Can be resolved by Supply Unit of Package			
I RU01	01.0	A00	ADD 120* A	I Unres	I SWLETEST2	01.0A00	02DEZ10410	RAUER
				I	I SWLETEST4	01.0A00	02DEZ10410	RAUER
I RU11	01.0	A00	ADD 120* A	I Res				
I RU12	01.0	A00	ADD 120* A	I Unres	I SWLETEST8	01.0A00	02DEZ10410	RAUER
I RU13	01.0	A00	ADD 120* A	I Unres				
I RU14	01.0	A00	ADD 120* A	I Unres	I SWLETEST9	01.0A00	02DEZ10410	RAUER

The *Status* data column shows if a dependency can be resolved. With *Status=Unres* (unresolvable dependency), the next column shows, where applicable, which supply units must be installed to resolve the dependency.

5.2 Administering path names and product versions (IMON-GPN)

The chapter contains an introduction to working with IMON-GPN, the installation path manager, and an overview of the user interfaces (command and macros).

IMON-GPN is a privileged subsystem that is activated when the BS2000 system is being loaded (activation point MANDATORY-AT-STARTUP).

Administering installation paths

A product installed in BS2000 comprises multiple installation items (files) which may be stored under one or more user IDs. Within the product, each installation item is unambiguously identified by its logical ID and the path name of the associated file.

Hardwaredependent installation items have a logical name and if necessary a number of path names with different identifiers for the hardware dependency (TARGET, see also "[Path name, logical ID and type of an installation item](#) ").

IMON-GPN supports you in the task of assigning logical IDs to file paths. IMON-GPN saves this assignment information in the SCI. The logical ID and the path are registered in the SCI for the first time during installation.

A logical name can be assigned to any path name using IMON-GPN, thus allowing the storage location to be decoupled from the logical name. The assignment of logical names to path names can be modified, canceled and /or displayed.

In the case of internal calls, the IMON-GPN interface is used to determine the path names of the files at runtime.

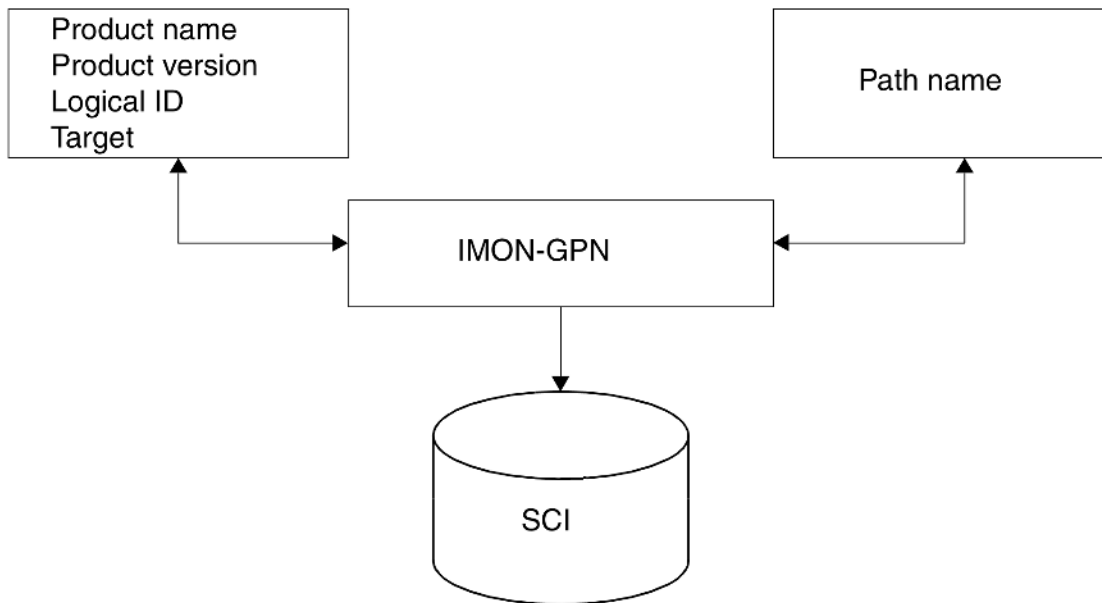


Figure 63: How IMON-GPN decouples paths and logical IDs

Administering product versions

IMON-GPN supports the user in selecting a product version, if a number of versions of a product are installed in the system.

IMON-GPN can be used to define different product versions for different users with different periods of validity.

With IMON GPN, product versions can be selectively locked for the users of the system, or existing locks can be lifted.

In the case of internal calls, the IMON-GPN interface is used to determine the version of the products or the subsystem to be loaded at runtime.

Users of IMON-GPN

- The privileged user (SUBSYSTEM-MANAGEMENT privilege)
 - uses the SHOW functions of IMON-GPN and can execute these functions globally
 - performs global administration of the installation paths (enter, change, view) and the product-version defaults (set, view)
 - enters name changes of pubsets in the SCI
 - can select the version of any product to be loaded and define the validity of this selection
 - locks specific product versions for the users of the system or lifts an existing lock
 - makes a backup copy of the standard SCI or rebuilds the standard SCI from a backup copy.
- The nonprivileged user
 - can view the installation paths to which he/she has access
 - can view the defaults for product versions and override these defaults for specific program sessions or tasks.
- Internal system functions

Internal functions of the BS2000 basic configuration and internal software executing in the TPR mode use IMON-GPN to ascertain the logical IDs of the path names (see “How file names are decoupled”) and to identify the product or subsystem versions to be loaded.

List of functions of IMON-GPN

Function	Command	Subroutine call
Lock produkt version	LOCK-PRODUCT-VERSION	
Modify SCI	MODIFY-IMON-SCI	
Restore SCI	RESTORE-SOFTWARE- INVERTORY	
Save SCI	SAVE-SOFTWARE-INVERTORY	
Select product version	SELECT-PRODUCT-VERSION	SELPROV
Enter or modify installation path	SET-INSTALLATION-PATH	SETINSP
Show installation path	SHOW-INSTALLATION-PATH	GETINSP / GETINSV
Show product version	SHOW-SELECTED-PRODUCT- VERSION	GETPROV
Unlock product version	UNLOCK-PRODUCT-VERSION	

The commands of IMON-GPN are described in the „Commands“ [[4 \(Related publications \)](#)] manual. The macros are described from "[The macros of IMON-GPN](#) " on.

5.2.1 Example with commands

Change installation path and select product version

For the purposes of this example, let us assume that the product SDF-I has been successfully installed and that standard installation was selected. Standard installation is illustrated on ["Default installation under BS2000"](#).

The purpose in this example is to render all installed versions of the product SDF-I centrally accessible under test ID RZTEST. The specified version (V4.1C10) will be set as the system-wide standard product version.

Part 1. Copy the product files

As the user with the SUBSYSTEM-MANAGEMENT privilege under the user ID TSOS, you begin by fetching information on the installed versions of the product or program file.

```
/show-installation-path installation-unit=sdf-i(version=*all),-
/
/          logical-identifier=sysprg,-
/          output=*sysout

INSTALLATION UNIT: SDF-I                      VERSION : 04.1B10
                                              FU-LEVEL: TU
                                              LOCKED  : NO

LOGICAL-ID: SYSPRG                          TARGET: A (ANY)

PATH-NAME: :1QHE:$TSOS.SYSPRG.SDF-I.041

MANDATORY: YES  UPDATE: YES  DUMMY: NO  FILENAME: FULL  FU-LEVEL:
TU

INSTALLATION UNIT: SDF-I                      VERSION : 04.1C10
                                              FU-LEVEL: TU
                                              LOCKED  : NO

LOGICAL-ID: SYSPRG                          TARGET: A (ANY)

PATH-NAME: :1QHE:$TSOS.SYSPRG.SDF-I.041

MANDATORY: YES  UPDATE: YES  DUMMY: NO  FILENAME: FULL  FU-LEVEL:
TU

% IMO9001 Command successfully processed
```

Copy the program file for SDF-I V4.1B10 to the user ID of your choice.

```
/copy-file from-file=$tsos.sysprg.sdf-i.041,-
/ to-file=$rztest.sysprg.sdf-i.041
```

Part 2. Set assignment of path name to logical ID

You must now set the installation path for this version of SDF-I in the SCI.

```
/set-installation-path path-name=$rztest.sysprg.sdf-i.041,-  
/  
/          logical-identifier=sysprg,-  
/          installation-unit=sdf-i(version=v04.1)
```

Part 3. Select product version

Select V4.1C10 as the standard version for the system.

```
/select-product-version product-name=sdf-i,version=v04.1c10,scope=*system
```

All users now use version V4.1C10 of SDF-I by default. A nonprivileged user (with SCOPE=*PROGRAM or *TASK) can use the SELECT-PRODUCT-VERSION command to select another available version of SDF-I (4.1B10 is still possible in the example) for a particular program or task session.

You should then check the result.

```
/show-installation-path installation-unit=sdf-i(version=V04.0A65),-  
/  
/          logical-identifier=sysprg,-  
/          output=*sysout
```

```
INSTALLATION UNIT: SDF-I                      VERSION : 04.1B10  
  
FU-LEVEL: TU  
  
LOCKED : NO  
  
LOGICAL-ID: SYSPRG                          TARGET: A (ANY)  
  
PATH-NAME: :1QHE:$RZTEST.SYSPRG.SDF-I.041  
  
MANDATORY: YES  UPDATE: YES  DUMMY: NO  FILENAME: FULL  FU-LEVEL: TU  
  
% IMO9001 Command successfully processed
```

```
/show-selected-product-version product-name=sdf-i,output=*sysout
```

```
PRODUCT NAME          PROGRAM  TASK      SYSTEM  
-----  
SDF-I                SDF-I    SDF-I    04.1B10
```

```
% IMO9001 Command successfully processed
```

```
/start-sdf-i
```

```
% BLS0500 PROGRAM 'SDF-I', VERSION 'V04.1B10' OF '2009-07-29' LOADED  
% BLS0552 COPYRIGHT (C) FUJITSU TECHNOLOGY SOLUTIONS 2009. ALL RIGHTS  
RESERVED  
  
*...  
  
*end
```

5.2.2 The macros of IMON-GPN

The macros of IMON-GPN enable you to use the program interface for the convenient, automated invocation of the IMON-GPN functions. The following IMON functions are invoked directly from within an application program (Assembler) and executed in real time.

List of functions

Macro	Function
GETINSP	Get installation path
GETINSV	Get version of installation unit
GETPROV	Get selected product version
SELPROV	Select product version
SETINSP	Set assignment of path to logical ID

The macros are described in alphabetical order; each description is structured as follows:

- Macro name and function
- Description of the macro function
- Presentation of the macro format
The format operands are separated from the function operands by an empty line.
- Description of operands
The format operands and metasyntax are described on "[Macro syntax](#)".
- Layout of the output area
- Macro return codes
- Notes

5.2.3 GETINSP Get installation path

GETINSP supplies information on the path name of an installation item. GETINSP enables you to view the relationship between the logical IDs and path names of installation items belonging to an installation unit.

Macro	Operands
GETINSP	MF= D / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= MOG / <char (3)> ,PARAM= <adr> / (<reg>) ,XPAND= PARAM / OUTPUT ,CALLER= USER / SYSTEM ,IUNAME= <c-string 1..30> / <var: char:30> ,UVERS= <c-string 7..7> / <var: char:7> ,LOGID= *ALL / <c-string 1..30> / <var: char:30> ,TARGET= '_' / <c-string 1..1> / <var: char:1> ,OUT@= <adr> / (<reg>) ,OUTLEN= <integer 1..2147483647> / <var: int:4>

The MF, PREFIX, MACID, PARAM and XPAND parameters are described in "Macro forms" in section ["Macro syntax"](#).

CALLER

Controls the type of system call if MF = E.

=USER

System call via SVC for nonprivileged caller (TU).

=SYSTEM

Direct system call for privileged caller (TPR).

IUNAME

Name of the installation unit containing the installation item or items, see ["Installation unit \(IU\)"](#).

=<c-string 1..30>

Explicit specification of the installation unit.

=<var: char:30>

Symbolic address containing the specification for the name of the installation unit. May not be specified with MF=L.

UVERS

Version of the installation unit. Format: <mm.naso>

=<c-string 7..7>

Explicit specification of the version of the installation unit.

=<var: char:7>

Symbolic address containing the specification for the version of the installation unit. May not be specified with MF=L.

LOGID

The logical name of the installation item for which the path name will be output.

=*ALL

Output the logical names of all installation items.

=<c-string 1..30>

Explicit specification of the logical ID of the installation item.

=<var: char:30>

Symbolic address containing the specification for the logical ID of the installation item. May not be specified with MF=L.

TARGET='_' / <c-string 1..1> / <var: char:1>

Specifies the hardware form belonging to the installation item (specify for IUNAME and IUVERS).

Possible values:

' ' : The installation item is part of the hardware variant of the current system.

A : The installation item is independent of the hardware variant.

S : The installation item is part of the /390 variant only.

K : The installation item is only part of the X86 variant.

OUT@= adr> / (<reg>)

Address of the area that is to accept the output information. The address must be aligned on a word boundary. This parameter has no significance unless MF = M.

OUTLEN=<integer 1..2147483647> / <var: int:4>

Length of the output area in bytes, the minimum length is 4. Only output information with the specified length is transferred. The first 4 bytes contain the length of the returned output information. This is then followed by one or more installation items.

Layout of the output area

Distance	Length	Format	Contents
X'00'	4	<integer>	Length of the output information
X'04'	30	<char>	Logical name of the installation item no. 1
X'22'	54	<char>	File name of the installation item no. 1
X'58'	1	<char>	Hardware model of the installation item no. 1
X'59'	1	<char>	Indicator of the installation item no. 1: X'80' File cannot be accessed X'40' File not initialized
X'5A'	2	<char>	Reserved
X'5C'	30	<char>	Logical name of the installation item no. 2
X'7A'	54	<char>	File name of the installation item no. 2
X'B0'	1	<char>	Hardware model of the installation item no. 2
X'B1'	1	<char>	Indicator of the installation item no. 2: X'80' File cannot be accessed X'40' File not initialized
X'B2'	2	<char>	Reserved
...			Further installation items

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'01'	X'00'	X'0000'	No path name assigned
X'02'	X'00'	X'0000'	User not privileged to read the path name. "*" is substituted for the path
X'00'	X'01'	X'0001'	Name of installation unit invalid
X'00'	X'01'	X'0002'	Version of installation unit invalid
X'00'	X'01'	X'0003'	Logical ID invalid
X'00'	X'01'	X'0008'	Illegal use of fields reserved for the parameter area
X'00'	X'40'	X'0011'	Installation unit not found
X'00'	X'40'	X'0012'	Version of installation unit not found
X'03'	X'40'	X'0012'	Access to version of installation unit not permitted
X'00'	X'40'	X'0013'	Logical ID not found
X'03'	X'40'	X'0013'	Access to logical ID not permitted

X'00'	X'40'	X'0018'	SCI invalid
X'00'	X'40'	X'001A'	SCI version invalid
X'00'	X'40'	X'001B'	SCI does not exist
X'00'	X'40'	X'001E'	Installation item not initialized
X'00'	X'01'	X'0021'	Output field address not assigned
X'00'	X'01'	X'0022'	Output field length insufficient (minimum=4)
X'00'	X'01'	X'0023'	Output field too small
X'00'	X'01'	X'0025'	Operand TARGET invalid
X'00'	X'01'	X'0026'	SYSTEM RC cannot be written.
X'00'	X'20'	X'00FF'	DMS error in SCI access
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

Notes

- You can use GETINSV to ascertain a version designation which you do not know in full.
- If a logical name does not contain any assigned path names, an item with an empty file name (blanks) is output in the output area and the indicator contains the value X'40'.
- If you are a nonprivileged user, this macro will fetch information only on the installation items of functional level TU (see the section "[Attributes of an installation unit](#)").
- The scope of information output is the name and version of the installation unit, plus the logical ID, the path name and the associated attributes for each installation item.
The character "*" is substituted for the path name if it cannot be output for the user. The indicator contains the value X'80'.
- The first four bytes of the output area indicate the length of the information returned.

5.2.4 GETINSV Get version of installation unit

GETINSV supplies the version of an installation unit.

Macro	Operands
GETINSV	MF= D / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= MOV / <char (3)> ,PARAM= <adr> / (<reg>) ,XPAND= PARAM / OUTPUT ,CALLER= USER / SYSTEM ,IUNAME= <c-string 1..30> / <var: char:30> ,IUVERS= *STD / *ALL / <c-string 4..10> / <var: char:10> ,SDFVSYN *NO / *YES ,SCOPE= *ANY / *SYSTEM / *LOCAL / <var: enum-of scope:1> ,ACTIVE= *ANY / *YES / <var: enum-of active:1> ,OUT@= <adr> / (<reg>) ,OUTLEN= <integer 1..2147483647> / <var: int:4>

The MF, PREFIX, MACID, PARAM and XPAND parameters are described in “Macro forms” on ["Macro syntax"](#).

CALLER

Defines the type of system call if MF = E.

=USER

System call via SVC for nonprivileged caller (TU).

=SYSTEM

Direct system call for privileged caller (TPR).

IUNAME

Name of the installation unit, see ["Installation unit \(IU\)"](#).

=<c-string 1..30>

Explicit specification of the installation unit.

=<var: char:30>

Symbolic address containing the specification for the name of the installation unit. May not be specified with MF=L.

IUVERS

Version of the installation unit. Format: see SDFVSYN operand.

=*STD

The version selected with the SELECT-PRODUCT-VERSION command is output. If no version was selected, the highest version is output.

= ALL

All version designations of the installation unit are output.

=<c-string 4..10>

Explicit specification of version of the installation unit.

=<var: char:10>

Symbolic address containing the specification for the version of the installation unit. May not be specified with MF=L.

SDFVSYN

Syntax format of the version.

=*NO

Version is specified in IUVERS as "mm.n[aso]".

=*YES

Version is specified in IUVERS as ['] [V][m]m.n[aso]['].

SCOPE

How the installation unit can be loaded.

=*ANY

The installation unit can be loaded in any way.

=*SYSTEM

The installation unit can be loaded as a DSSM subsystem; it is entered in the DSSM catalog.

=*LOCAL

The installation unit can be loaded as a program.

=<var: enum-of scope:1>

Name of the field together with the manner in which the installation unit is loaded.

ACTIVE

Specifies whether the installation unit is already active. This parameter is ignored if SCOPE = *LOCAL was specified. It is relevant only for installation units that are DSSM subsystems.

=*ANY

The installation unit can be activated, but activation is not mandatory.

=*YES

The installation unit must be activated, i.e. the subsystem is started.

=<var: enum-of active:1>

Name of the field together with the value signifying whether the installation unit is activated.

OUT@=<adr> / (<reg>)

Address of the area that will accept the output information. The address must be aligned on a word boundary. This parameter has no significance unless MF = M.

OUTLEN=<integer 1..2147483647> / <var: int:4>

Length of the output area in bytes, the minimum length is 4. Only output information with the specified length is transferred. The first 4 bytes contain the length of the returned output information. This is then followed by one or more installation units.

Layout of the output area

Distance	Length	Format	Contents
X'00'	4	<integer>	Length of output information
X'04'	7	<char>	Version designation, format nn.n[ann]
X'0B'	1	S / L / U	SCOPE defines how the installation unit is loaded. S = SYSTEM, L = LOCAL, U = UNDEFINED
X'0C'	1	Y / N / U	ACTIVE indicates whether the installation unit is already active. Y = YES, N = NO, U = UNDEFINED (not a subsystem)
X'0D'	1	Y / N / U	Selected by SELECT-PRODUCT-VERSION: Y = YES, N = NO, U = UNDEFINED
X'0E'	1	Y / N / U	Logical name exists Y = YES, N = NO, U = UNDEFINED
...			Other versions of the installation unit

Macro return codes

(SC2)	SC1	Maincode	Meaning
-------	-----	----------	---------

X'00'	X'00'	X'0000'	Function successfully executed
X'03'	X'00'	X'0000'	Function successfully executed (partial information)
X'09'	X'00'	X'0000'	Installation unit locked
X'00'	X'01'	X'0001'	Name of installation unit invalid
X'00'	X'01'	X'0002'	Partial version of installation unit invalid
X'00'	X'01'	X'0005'	SCOPE invalid
X'00'	X'01'	X'0006'	ACTIVE invalid
X'00'	X'01'	X'0008'	Illegal use of fields reserved for parameter area
X'00'	X'40'	X'0011'	Installation unit not found
X'00'	X'40'	X'0012'	No matching version available
X'03'	X'40'	X'0012'	Access to specified version not permitted
X'00'	X'40'	X'0018'	SCI invalid
X'00'	X'40'	X'001A'	SCI version invalid
X'00'	X'40'	X'001B'	SCI does not exist
X'00'	X'01'	X'0021'	Output field address not assigned
X'00'	X'01'	X'0022'	Output field length insufficient (minimum=4)
X'00'	X'01'	X'0023'	Output field too small
X'00'	X'01'	X'0029'	Too many subsystem versions
X'00'	X'20'	X'00FF'	DMS error in SCI access
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

Notes

- The first four bytes of the output area indicate the length of the information returned.
- If you are a nonprivileged user, this macro will fetch information only on the installation items of functional level TU (see the section "[Attributes of an installation unit](#)").
- SCOPE = *SYSTEM is not evaluated unless the name of the subsystem and the name for the specified installation unit are the same.
- If the version was defined for different settings (SCOPE, see the SELPROV macro), IUVERS=*STD returns the "lowest" SCOPE (PROGRAM < TASK < SYSTEM).

5.2.5 GETPROV Get selected product version

GETPROV returns the version selected beforehand for a product with a particular scope.

A product can be either a DSSM subsystem (TU) or an installation unit of IMON.

Macro	Operands
GETPROV	MF= D / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= MOO / <char (3)> ,PARAM= <adr> / (<reg>) ,CALLER= USER / SYSTEM ,PRODNAM= <c-string 1..30> / <var: char:30> ,EXSTCHK= *YES / *NO

The MF, PREFIX, MACID, and PARAM parameters are described in "Macro forms" on ["Macro syntax "](#).

CALLER

Defines the type of system call if MF = E.

=USER

System call via SVC for nonprivileged caller (TU).

=SYSTEM

Direct system call for privileged caller (TPR).

PRODNAM

Name of the product, see ["Supply unit "](#).

=<c-string1..30>

Explicit specification of the product name.

=<var: char:30>

Symbolic address containing the specification for the name of the product. May not be specified with MF=L.

EXSTCHK

Determines whether the existence of the product is checked.

=*YES

A check is performed as to whether the product exists.

=*NO

No check is performed as to whether the product exists.

Output operands

Data field <prefix><macid>PRGV

This is the output data field for the version selected for a product with scope PROGRAM.

Data field <prefix><macid>TSKV

This is the output data field for the version selected for a product with scope TASK.

Data field <prefix><macid>SYSV

This is the output data field for the version selected for a product with scope SYSTEM.

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	Name of product invalid
X'00'	X'01'	X'0008'	Illegal use of fields reserved for the parameter area
X'00'	X'40'	X'0011'	Product not found
X'00'	X'20'	X'00FF'	DMS error in SCI access
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

Notes

- No information is available, the version output contains only blanks.
- EXSTCHK=*NO must be specified if a high level of performance is required or if the caller is DSSM.

5.2.6 SELPROV Select product version

SELPROV enables you to select the product version of an installation unit or DSSM subsystem (TU, DSSM V3.5 and higher). If multiple versions of this product are installed and accessible, you can define the version that the user will access. The version defined here is selected when the program starts or when the link is made to a subsystem.

Macro	Operands
SELPROV	,PREFIX= I / <char (1)> ,MACID= MOL / <char (3)> ,PARAM= <adr> / (<reg>) ,CALLER= USER / SYSTEM ,PRODNAM= <c-string 1..30> / <var: char:30> ,PRODVER= *STD / <c-string 4..7> / <var: char:7> ,SCOPE= *PROGRAM / *TASK / *SYSTEM / <var: enum-of scope:1>

The MF, PREFIX, MACID, and PARAM parameters are described in “Macro forms” on ["Macro syntax "](#).

CALLER

Defines the type of system call if MF = E.

=USER

System call via SVC for nonprivileged caller (TU).

=SYSTEM

Direct system call for privileged caller (TPR).

PRODNAM

Name of the product, see ["Supply unit "](#).

=<c-string1..30>

Explicit specification of the product name

=<var: char:30>

Symbolic address containing the specification for the name of the product. May not be specified with MF=L.

PRODVER

Defines the version of the product to be loaded. Partially qualified entries are permissible: the highest matching version or (DSSM subsystem) the matching version is selected.

=*STD

Undo a version selection and enable the system default (the highest version is used).

=<c-string 4..7>

Explicit specification of the product version.

=<var: char:7>

Symbolic address containing the specification for the product version. May not be specified with MF=L.

SCOPE

Time for which the selection is valid.

=*PROGRAM

The selection is valid until the current or subsequent program is terminated.

=*TASK

Selection is valid until the task is terminated.

=*SYSTEM

Selection is valid system-wide for this system session (until shutdown).

This operand can be used for the functional level TPR only by a user with the SUBSYSTEM-MANAGEMENT privilege .

=<var: enum-of scope:1>

Name of the field together with the period of validity.

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	Name of product invalid
X'00'	X'01'	X'0002'	Product version invalid
X'00'	X'01'	X'0005'	SCOPE invalid
X'00'	X'01'	X'0008'	Illegal use of fields reserved for the parameter area
X'00'	X'40'	X'0011'	Product not found
X'00'	X'40'	X'0012'	No matching version available
X'03'	X'40'	X'0012'	Access to specified version illegal
X'00'	X'40'	X'0015'	User without SUBSYSTEM-MANAGEMENT privilege

X'00'	X'40'	X'0019'	No version can be selected for the specified product
X'00'	X'40'	X'001D	DSSM subsystem cannot be connected.
X'00'	X'40'	X'001F	Version cannot be selected using DSSM.
X'00'	X'20'	X'00FF'	DMS error in SCI access
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

Notes

- If a version is specified for PROGRAM and for TASK or SYSTEM, the PROGRAM specification is preemptive.
- Only one version can be selected for a product. If the macro is executed twice for a product with different versions, the most recent selection is valid.
- Selection with SCOPE=*SYSTEM is valid only until shutdown. The selection must be repeated in a new system session.
- As a non privileged user, this macro shows you only those installation items of the TU functional status (see the section "[Attributes of an installation unit](#)" ([Installation unit \(IU\)](#))).

5.2.7 SETINSP Enter or modify installation path

The SETINSP macro enables you to change the correlation between path name and logical ID after product installation or to cancel the assignment. Note that the change is not implemented if it is not permitted by the attributes.

SETINSP requires the presence of a SYSSII file for the product.

Privilege required: **SUBSYSTEM-MANAGEMENT**

Macro	Operands
SETINSP	MF= D / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= <u>MOS</u> / <char (3)> ,PARAM= <adr> / (<reg> ,CALLER= <u>USER</u> / SYSTEM ,SCINAME= ' ' / <c-string 1..54> / <var: char:54> ,IUNAME= <c-string 1..30> / <var: char:30> ,IUVERS= <c-string 7..7> / <var: char:7> ,TARGET= ' ' / <c-string 1..1> / <var: char:1> ,LOGID= <c-string 1..30> / <var: char:30> ,PATHNAM= *NONE / <c-string 1..54> / <var: char:54> ,ENFORCE= <u>*NO</u> / *YES

The MF, PREFIX, MACID, and PARAM parameters are described in “Macro forms” on ["Macro syntax"](#).

CALLER

Defines the type of system call if MF = E.

=USER

System call via SVC for nonprivileged caller (TU).

=SYSTEM

Direct system call for privileged caller (TPR).

SCINAME

Name of the SCI.

= ' ' .

The standard SCI will be processed (the files \$TSOS.SYS.IMON.SCI and \$TSOS.SYS.IMON.SCI.GPN).

=<c-string 1..54> / <var: char:54>

Name of a foreign SCI.

IUNAME

Name of the installation unit to be processed, see "[Installation unit \(IU\)](#)".

=<c-string 1..30>

Explicit specification of the installation unit.

=<var: char:30>

Symbolic address containing the specification for the name of the installation unit. May not be specified with MF=L.

IUVERS

Version of the installation unit. Format: <mm.naso>

=<c-string 7..7>

Explicit specification of the version.

=<var: char:7>

Symbolic address containing the specification for the version of the installation unit. May not be specified with MF=L.

TARGET= ' ' / <c-string 1..1> / <var: char:1>

Specifies the hardware form belonging to the installation item (specify for IUNAME and IUVERS).

Possible values:

' ' : The installation item is part of the hardware variant of the current system.

A : The installation item is independent of the hardware variant.

S : The installation item is part of the /390 variant only.

K : The installation item is only part of the X86 variant.

LOGID

The logical name of the installation item to be updated.

=<c-string 1..30>

Explicit specification of logical name of the installation item.

=<var: char:30>

Symbolic address containing the specification for the logical ID of the installation item.
May not be specified with MF=L.

PATHNAM

The path name assigned to the logical name of the installation item.

=*NONE

No path name assigned to the logical name.

An existing assignment is cancelled. This specification will be rejected if the logical ID was defined with "Mandatory Y" (see "[Installation item \(II\)](#)").

=<c-string1..54>

Explicit specification of the path name (fully or partially qualified).

The partially qualified file name refers to all files assigned to the logical ID.

=<var: char:54>

Symbolic address containing the specification for the path name of the installation item.
May not be specified with MF=L.

ENFORCE

Specifies whether the path name can be assigned.

=*NO

The path name is not assigned.

=*YES

The path name will be assigned even though the logical name was defined with "Update N" (see "[Installation item \(II\)](#)").

Macro return codes

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'05'	X'00'	X'0000'	Enforced function successfully executed
X'06'	X'00'	X'0000'	Warning: File does not exist
X'07'	X'00'	X'0000'	Warning: Forced file does not exist
X'00'	X'01'	X'0001'	Name of installation unit invalid
X'00'	X'01'	X'0002'	Version of installation unit invalid
X'00'	X'01'	X'0003'	Logical ID invalid

X'00'	X'01'	X'0004'	Path name invalid
X'00'	X'01'	X'0008'	Illegal use of fields reserved for the parameter area
X'00'	X'40'	X'0011'	Installation unit not found
X'00'	X'40'	X'0012'	Version of installation unit not found
X'00'	X'40'	X'0013'	Logical ID not found
X'00'	X'40'	X'0014'	Path name cannot be changed as requested
X'00'	X'40'	X'0015'	User without SUBSYSTEM-MANAGEMENT privilege
X'00'	X'40'	X'0018'	SCI invalid
X'00'	X'40'	X'001A'	SCI version invalid
X'00'	X'40'	X'001B'	SCI does not exist
X'00'	X'40'	X'001C'	Path name incomplete
X'00'	X'01'	X'0025	Target operand invalid
X'00'	X'01'	X'0026'	SYSTEM RC cannot be written.
X'00'	X'20'	X'00FF'	DMS error in SCI access
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

5.2.8 Example with macros

Change installation path and select product version

In this example, one of several installed versions of the SDF-I product is copied to the test user ID RZTEST and defined as the standard product version with system-wide validity.

The corresponding example with commands starts on ["Example with commands"](#).

```
EXAMPLE CSECT
COPY      OPSYN
*
*   Dsects
*
GETIP     GETINSP MF=D,XPAND=PARAM
GETIPOUT  GETINSP MF=D,XPAND=OUTPUT
*
GETIV     GETINSV MF=D,XPAND=PARAM
GETIVOUT  GETINSV MF=D,XPAND=OUTPUT
*
SETIP     SETINSP MF=D
*
SELPR     SELPROV MF=D
*
COPYD     IDCOP D
*
*
PROG      ...
*
*   Init work register
*
        LA    R3,OUTPATH+4           Output area for GETINSP
        USING GETIPOUT,R3
        LA    R4,OUTAREA+4          Output area for GETINSV
        USING GETIVOUT,R4
*
        LA    R5,GETIPSK            PL skeleton for GETINSP
        USING GETIP,R5
        LA    R6,SETIPSK            PL skeleton for SETINSP
        USING SETIP,R6
        LA    R7,GETIVSK            Pl skeleton for GETINSV
        USING GETIV,R7
        LA    R8,COPYSK             PL skeleton for COPY
        USING COPYD,R8
        LA    R9,SELPRSK            PL skeleton for SELPROV
        USING SELPR,R9
```

```

*
* Get all available versions of the installation unit SDF-I
*
*
*       LA      R2,OUTAREA           Output area for GETINSV
*       GETINSV MF=M,OUT@=(R2)      Fill OUT@ field
*       GETINSV MF=E,PARAM=(R7)     Call function
*
*       CLC     IMOGRET(4),=A(IMOVSUCC) Function successful ?
*       BNE     TERM                 NO -> Terminate
*
*       LA      R2,OUTPATH           Output area for GETINSP
*       GETINSP MF=M,OUT@=(R2)      Fill OUT@ field
*
* For all returned versions, get the path name associated to the
* Logical-id SYSPRG, copy the file under RZTEST userid and update
* the path name
*
*       LA      R2,4                 for loop limit
LOOP    C       R2,OUTAREA           End of area reached ?
*       BNL     SELECT               Yes, terminate
*       GETINSP MF=M,IUVERS=IMOVOVER Take current IU version
*       GETINSP MF=E,PARAM=(R5)      Get path name
*       CLC     IMOGRET,=A(IMOGSUCC) OK ?
*       BNE     TERM                 NO, terminate
*
* path name is available in IMOGOPAT field (in our example, it looks
* like this : :SOLX:$TSOS.SYSPRG.SDF-I.vvv)
*
*       MVC     IDBCFN1L,IMOGOPAT     Initial path name
*       MVC     IDBCFN2L(8),=C'$RZTEST.' New userid
*       MVC     IDBCFN2L+8(42),IMOGOPAT+12 Move path name part
*       COPY    MF=(E,(R8))
*       LTR     R15,R15               Copy OK ?
*       BNZ     TERM                 NO, terminate
*
* Update the path name for the corresponding logical-id
*
*       SETINSP MF=M,IUVERS=IMOVOVER,PATHNAM=IDBCFN2L
*       SETINSP MF=E,PARAM=(R6)
*       CLC     IMOSRET,=A(IMOSSUCC) SETINSP OK ?
*       BNE     TERM                 NO, terminate
*
*       LA      R2,IMOVOMD#(R2)      Next Installation unit version
*       B       LOOP
*

```

```

SELECT  DS    0H
*
* Select the first returned version for the system
*
      LA    R4,OUTAREA+4      Point on first returned version
      SELPROV MF=M,PRODNAM='SDF-I',PRODVER=IMOVOVER,SCOPE=*SYSTEM
      SELPROV MF=E,PARAM=(R9)
*
TERM    DS    0H
      ...
*
* Constants for IMON calls
*
      DS    0F
GETIPSK GETINSP MF=L,IUNAME='SDF-I',LOGID='SYSPRG',OUTLEN=88
*
      DS    0F
SETIPSK SETINSP MF=L,IUNAME='SDF-I',LOGID='SYSPRG'
*
      DS    0F
GETIVSK GETINSV MF=L,IUNAME='SDF-I',IUVERS=*ALL,OUTLEN=50
*
      DS    0F
SELPRSK SELPROV MF=L,PRODNAM='SDF-I',SCOPE=*SYSTEM
*
      DS    0F
COPYSK COPY MF=L,INNAME,OUTNAME
*
OUTAREA DS    0F
      DC    50CL1' '
*
OUTPATH DC    88CL1' '
*
      ...
      END    EXAMPLE

```

5.3 Administering installation units

This section describes how to register installation units in the SCI, how to transfer information from the SCI to a different SCI (exporting and importing), how to remove installation units from the SCI, and how to obtain information from the SCI.

5.3.1 Registering installation units in the SCI

Registering means entering installed software in the Software Configuration Inventory.

Registration can be triggered in two ways:

- automatically after successful installation or parking
- explicitly by the user for software already installed in the system by means of the IMON-BAS installation function (ADD-INSTALLATION-UNITS statement or *Add...* option in the *Edit* menu).

Automatic registration of installation units

Once installation is complete, IMON creates an internal IDF file and registers the installation units and installation items described in it in the SCI (the same as registration from an IDF file with the ADD-INSTALLATION-UNITS statement).

The registration is logged

(File name: <work_file_ID>.<package_name>.<time_stamp>.ll).

Registration of installation units by the user

You register your installed software in the SCI using the IMON-BAS installation function (ADD-INSTALLATION-UNITS or the *Add...* option in the *Edit* menu).

As input, the statement expects an IDF file or a SYSSII file containing a description of the installation units that you would like to register together with their installation items.

The information (attributes) of the installation unit and installation items registered in the SCI is described on "[Installation unit \(IU\)](#)".

During registration from an IDF file, the information relating to the structure of the installation unit is taken from various files if necessary:

- The installation unit does not contain a SYSSII file:

The information on the structure of the installation unit is taken from the IDF. The ID of the installation unit (name and version) is simultaneously registered in IMON-GPN. The subordinate installation items are not registered by IMON-GPN.
- The installation unit contains a SYSSII file:

The information on the structure of the installation unit is taken from the SYSSII file.

The structure registered in the SYSSII file must match the structure in the IDF. Consequently, the following conditions apply:

 - Only one SYSSII file is permissible for each installation unit.
 - Each item described in the IDF must also be present in the SYSSII file.

If these two conditions are satisfied, the installation unit and its associated installation items are registered in the SCI. IMON-GPN takes the logical IDs from the SYSSII file.

The appropriate installation paths are then set in IMON-GPN.

An installation item described only in the SYSSII file (and not in the IDF) is not registered unless it is flagged as a dummy installation item. The file attributes of the installation item (including the name) are not registered in the SCI.

5.3.2 Exporting and importing SCI entries

SCI entries from selected supply units or installation units can be copied (exported) to another SCI using the “Export /import SCI entries” function.

The export operation is initiated using the *Edit: Generate installation definition file* menu or the GENERATE-IDF statement. IMON generates an import procedure in which the required target SCI is opened and into which SYSDTA reads a generated IDF file containing the structure of the selected supply components or installation units (see "[Installation definition file \(IDF file, internal format\)](#) ") using the ADD-INSTALLATION-UNITS statement.

The selected supply components or installation units are imported by calling the generated import procedure.

The installation path (complete path or only the catalog ID, user ID or file name prefix) can also be changed during the export operation. If you are changing the installation path, you can also generate a copy procedure containing all COPY-FILE commands for the affected installation items and the import procedure call. When you call the copy procedure, the installation items are first transferred to the new location (placement) and then registered in the specified SCI.

5.3.3 Removing installation units from the SCI

The entries for the installation units of an installed software product can be removed once again from the SCI, however this is restricted to only that information that IMON registered in the SCI. Installed files are retained on the system and measures that were performed to activate the software (e.g. activation of the subsystem and syntax file) are not reset. This means that the software product remains active. Since IMON-GPN cannot supply any information from the SCI, the default path names of the product are used. These may differ from the actual path names.

The IMON-BAS remove function (REMOVE-INSTALLATION-UNITS statement or *Edit: Remove*) enables you to remove the administrative information about superfluous installation units and their installation items from the open SCI.

Installation units that have been removed are no longer available to IMON-GPN.

Note, however, that the files corresponding to the installation items you removed are not removed from the system.

Removing an installed software product in the sense of a deinstallation can be performed with the IMON-BAS deinstallation function (DEINSTALL-SUPPLY-UNITS statement or the *Deinstall* option in the *Edit* menu, see [section "Deinstallation"](#)).

5.3.4 Fetching information from the SCI

The SHOW functions of IMON-BAS enable you to fetch information from the SCI.

The attributes of the deliveries, supply units, installation units and installation items and the relationships between the objects can be output.

If a formatted file is used as the input for another SHOW statement, the names of the installation units and installation items are taken from the input file named in the *FROM-FORMATTED-FILE operand.

If the input file is the same as the output file (OUTPUT=*INPUT-FORMATTED-FILE), the information from the SCI is added to the information already available in the formatted input file.

By piping SHOW statements, you can place all the information you want in a single formatted file. Use the SHOW-FORMATTED-FILE statement to output the formatted file in readable form.

Placement status of installation items

The placement status is the installation state of a file. It is included in the SHOW output.

The following values are possible:

PLACED:

- The item was defined in the IDF and found in the SCI. The file is cataloged.

NOT PLACED:

- The item was defined in the IDF file and found in the SCI. However the file is not cataloged or cannot be accessed.

UNDEFINED:

- The item is in the SYSSII file. The associated installation unit was registered with the *Edit:Add...* menu function or the ADD-INSTALLATION-UNITS statement, but the item is not included in the IDF.

Installation state of supply units

A supply unit can be in one of the following installation states:

Installed Installation of the supply unit is complete.

Being
Installed Installation of the supply unit is not yet complete.

Parked The supply unit is parked.

In Library The supply unit is only registered and is still located in the library of a file transfer delivery.

On
SOLIS2
Volume The supply unit is only registered and is still located on the data volume shipped with Solis2 (tape cartridge CD, DVD).

On Local
Volume The supply unit is only registered and is still located on the copy of the Solis2 data volume created by the customer (tape cartridge, CD, DVD or virtual volume in a CentricStor).

Being
Deinstalled The supply unit is currently being deinstalled.

Partially The supply unit was only partially installed.
Installed

The SHOW output for supply units can be limited to certain installation states.

Release status of supply units

A supply unit that is already installed in a system and is registered in the SCI can have a customer approval ID. This means it is available for installation on other systems; installation is then based exclusively on the customer approval ID. Using information that is output on the supply unit, the system administrator can obtain information in advance on which supply units are available for this installation function.

5.3.5 Removing obsolete information from the SCI

The SCI contains a great deal of metadata, such as information about opened deliveries, installed supply items, added installation units and installation items. Some of this information is obsolete and should be removed from the SCI. Obsolete information can be identified and deleted on the following levels and in the order shown:

1. Delivery
2. Installation unit
3. Installation item

Identifying and removing obsolete deliveries

The information about a delivery is needed as long as a supply unit can be installed from it.

This applies in the following cases:

- One or more supply units in the delivery are in the installation status “In Library”, “On SOLIS2 Volume”, “On Local Volume” or “Parked”.
- The user wants to reinstall a supply unit again.

As soon as no more supply units are to be installed from it, all information from the delivery can be removed from the SCI and optionally all work files belonging to it can be deleted, with the statement REMOVE-PACKAGES or the menu function *Edit:Remove*. All supply units, which are not in the installation status “Installed”, are deleted with the delivery.

To locate deliveries, which are no longer needed by users and can be deleted, further selection criteria are available to the user in the SELECT operand of the SHOW-PACKAGES statement or the *Filter packages* dialog box of the menu function *View:Packages (deliveries)*, as well as from creation date, change date of the delivery and the installation status of the supply units which belong to it.

Note

A supply unit of a delivery can be in an installation status other than “Installed”, and still be installed, if the installation has taken place from another delivery. In this case, a “*” is displayed in the marker column of the supply unit.

Identifying and removing obsolete supply units

The information about a supply unit is obsolete in the following cases:

- The supply unit is in the installation status “Installed” and the user does not wish to execute the installation again.
- New versions of the supply units have been installed and the user no longer needs the older versions.
- The supply unit is in the installation status “Installed”, but it contains no installation units.

Supply units, which are in the installation status “Installed” or “Parked” and are no longer needed, can be deleted through the deinstallation function. All other supply units can be deleted with the statement REMOVE-SUPPLY-UNITS or the menu function *Edit:Remove*.

To identify and locate deliveries, which are not in the installation status “Installed”, the user can employ installation status as a selection criterion in the SHOW-SUPPLY-UNITS statement in the SELECT operand, or the *Filter supply units* dialog box in the *View:Supply units* menu function.

Deliveries in the installation status “Installed” or “Parked”, which can only be deinstalled, can be identified and located in a similar way. As a check, the deinstallation should be carried out first in test mode and then if applicable started normally. During deinstallation, all files belonging to it are deleted.

Identifying and removing obsolete installation units

The information about an installation unit is obsolete if a higher version is installed and the other versions are no longer used.

In menu mode, all versions of an installation unit from the open SCI are displayed in the work area. The user can select the versions that are no longer needed and then call the function *Edit:Remove*.

6 Working with IMON-SIC

This chapter is an introduction to IMON-SIC. It describes how to start IMON-SIC and how to use the menu functions and subroutine calls supported by IMON-SIC. The subroutine calls (Assembler and C interface) read information from a SYSSII file.

This chapter contains an introduction to working with IMON-SIC, describes how IMON-SIC is started and which functions IMON-SIC offers through menu options, SDF statements and macro calls. The SDF statements and the subprogram calls (Assembler and C interface) each only read information from a single SYSSII file.

Information can also be output from a SYSSII file using the menu functions. A SYSSII file can also be created and edited:

- Opening a SYSSII file for processing (generating it first if necessary)
- Modifying the attributes of a release unit
- Modifying the structure of a release unit
- Modifying the attributes of a release item
- Recording or deleting logging information
- Displaying information
- Backing up edited SYSSII files

6.1 Starting and terminating IMON-SIC

Starting IMON-SIC

The command for calling IMON-SIC is START-IMON-SIC.

START-IMON-SIC

Alias: **IMON-SIC**

VERSION = *STD / <product-version>

,MONJV = *NONE / <filename 1..54 without-gen-vers>

,CPU-LIMIT = *JOB-REST / <integer 1..32767 seconds>

,INPUT-INTERFACE = *STD / *SDF

VERSION =

Use the selected version of IMON-SIC.

VERSION = *STD

Use the version selected with the SELECT-PRODUCT-VERSION command as the standard version. If no version was set, IMON-SIC will be loaded with the highest available version.

VERSION = <product-version>

Explicit specification of the version in the format mm.n[a[so]], e.g. 3.4 or 3.4A or 3.4A00.

MONJV =

A job variable for monitoring the IMON-SIC session.

MONJV = *NONE

No job variable.

MONJV = <filename 1..54 without-gen-vers>

Explicit specification of the job variables for monitoring the IMON-SIC session.

CPU-LIMIT =

Maximum CPU time in seconds allowed for the program at runtime.

CPU-LIMIT = *JOB-REST

Use the remaining CPU time for the task.

CPU-LIMIT = <integer 1..32767 seconds >

The maximum time to be used.

INPUT-INTERFACE =

Specifies with which interface IMON-SIC will be started.

INPUT-INTERFACE = *STD

In dialog mode, IMON-SIC is started with the menu interface.

In procedure or batch mode, IMON-SIC is started with the SDF interface.

INPUT-INTERFACE = *SDF

IMON-SIC is started with the SDF interface.

Command return code

(SC2)	Maincode	Meaning/Guaranteed messages
SC1		
0	CMD0001	Command executed successfully
1	CMD0202	Syntax error
32	CMD0221	Command terminated abnormally. System error
32	IMO9101	Command terminated abnormally. Internal error in IMON-GPN
64	IMO9100	Command not executed. Installation unit, version or logical name not found, etc.; the exact cause can be taken from the SYSOUT message.

Note

When the program terminates, the monitoring job variable can assume the following values:

Status display (byte 0 - 2)	Return code (byte 3 - 7)	Meaning
\$T	0000	No error
\$T	1010	Statement or function rejected, program is resumed.
\$A	2010	Statement or function rejected, program is aborted, Spin-off takes place.
\$A	2015	Unexpected EOF on SYSDTA.
\$A	3020	Internal inconsistency.

Terminating IMON-SIC

IMON-SIC is terminated differently depending on which interface was set:

- IMON-SIC is terminated with the [F3] key in the menu interface.
- When the SDF interface was specified, IMON-SIC is terminated with the END statement.

6.2 The menu functions of IMON-SIC

In interactive mode, IMON-SIC offers a mask-driven user interface (FHS masks).

The screen is output in the German or English language depending on the message language setting. The English language screens are described in the following.

6.2.1 Mask layout

The masks of IMON-SIC share a common layout. [figure 64](#) shows the layout of the IMON-SIC masks in schematic form.



Figure 64: Mask layout in IMON-SIC

Menu bar (1)

In the menu bar <title> stands for the current mask name. The mask name in abbreviated form appears in PANEL.

Global information (2)

The global information area consists of a single line which is displayed whenever a release unit or a release item is processed. The name and main version of the current release unit or the name of the release item appears in this line.

Body (3)

The body contains information and/or prompts for user input.

Instruction area (4)

The instruction area shows you what to do in the mask.

Function key area (5)

This area contains a list of the function keys with their current functions. The function keys which may be listed here are as follows:

[K1] Cancel the current function. Inputs since the last [DUE] are lost. Return to the preceding mask.
=cancel

[K2] Press [K2] to interrupt IMON-SIC and switch to the BS2000 command shell, where you can enter
=interrupt BS2000 commands. Return to IMON-SIC with the BS2000 command RESUME-PROGRAM. The
 mask in which you pressed [K2] appears. Prior entries in this mask are lost.

[K3] The mask is restored to the state which preceded the last [DUE].
=refresh

[DUE] / Press [DUE] or [ENTER] to confirm your entries. The next mask is displayed.
[ENTER]
=
continue

[F2] = An action requiring page up/down movement in the body. Press [F2] to move down one page and
Prev / [F3] to move up one page. Press [F3] in the main menu to terminate IMON.
[F3] =
Next

Message area (6)

Information, warnings and messages are displayed in the message area. Messages from IMON-SIC start with the prefix `IM08`. A warning begins with the word `Warning`. If an error message is output, the cursor is positioned on the first invalid field.

6.2.2 The sequence of masks

When you call IMON-SIC, the first mask which appears is the main menu. This menu enables you to select the SYSSII file you want to process.

If all you want to do is view the contents of the SYSSII file, you are returned to the main menu after you have set the output parameters and viewed the file.

If you want to edit the SYSSII file, the next mask is the SYSSII menu. This mask enables you to select the editing steps.

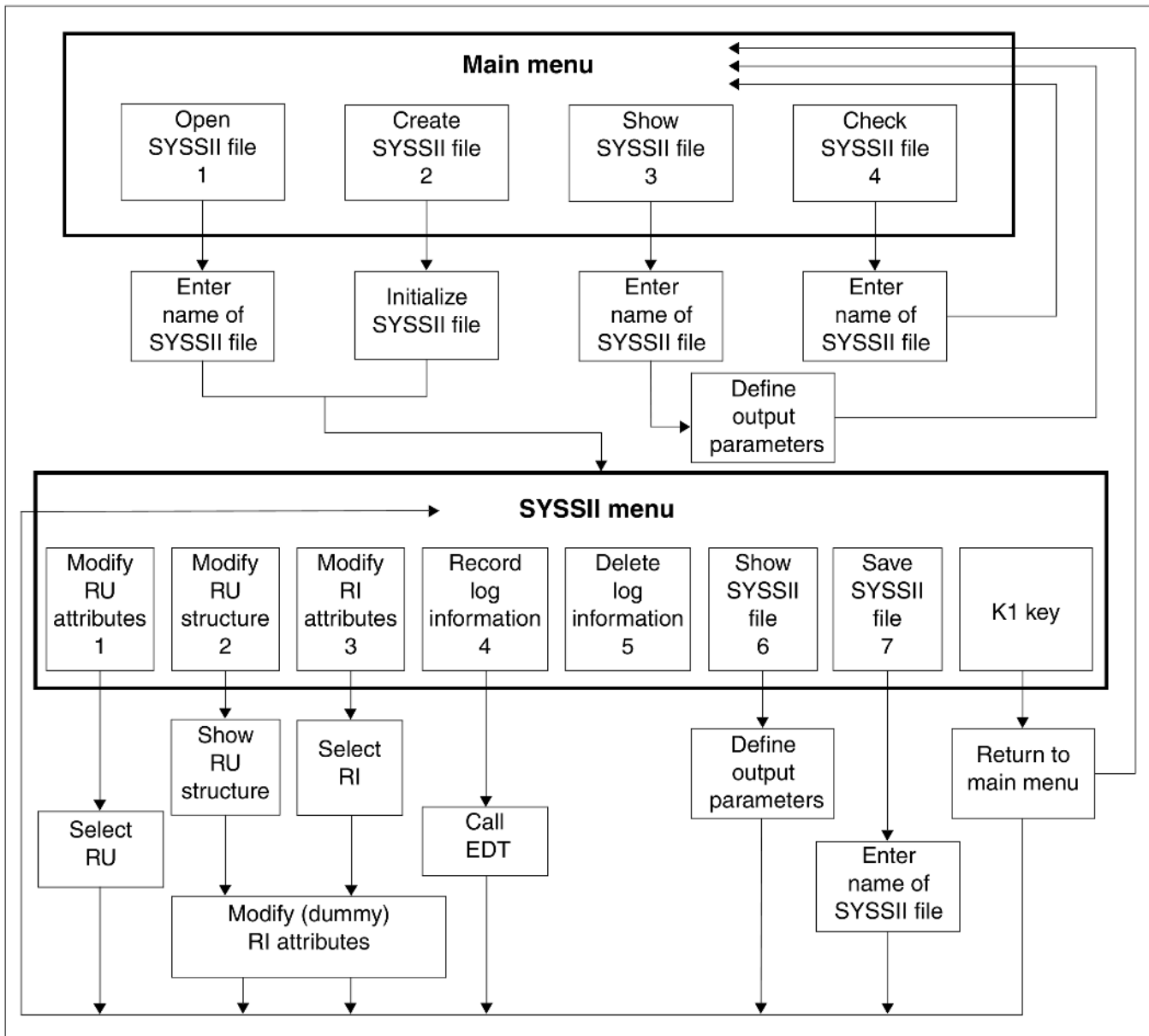


Figure 65: Sequence of IMON-SIC mask sequence (RU = release unit, RI = release item)

6.2.3 Description of the menu functions of the main menu

The main menu of IMON-SIC (see [figure 66](#)) offers you a choice of four actions:

1. *Open SYSSII ...*
open an existing SYSSII file for processing
2. *New SYSSII ...*
create a new SYSSII file
3. *Show SYSSII ...*
view the contents of a SYSSII file
4. *Check SYSSII ...*
check the contents of a SYSSII file (SYSSII files in the customer format are not checked)

Enter the digit (1, 2, 3 or 4) corresponding to the action of your choice.

```
PROGRAM : IMON-SIC V03.4          MAIN MENU          PANEL : MAIME
-----
                                1. Open SYSSII ...
                                2. New SYSSII ...
                                   Type : S (C=Customer / S=SU BS2000)
                                3. Show SYSSII ...
                                4. Check SYSSII ...

Choose the action; set the view type and press DUE
-----
K2=interrupt  K3=refresh  F3=exit
-----
```

Figure 66: Main menu of IMON-SIC

- Type Specifies the format of the new created SYSSII file.
- C Preset: A new SYSSII file is created in the customer format. The release unit described in the file is user software that cannot be installed with IMON.
 - S This format is reserved exclusively for BS2000 software development. The release unit described in the file is system software delivered with SOLIS2.

6.2.4 Open, display or check SYSSII file

If you choose action 1 (*Open SYSSII ...*), 3 (*Show SYSSII ...*) or 4 (*Check SYSSII ...*), you must enter the name of the SYSSII file.

```
PROGRAM : IMON-SIC          SYSSII File Selection          PANEL : SIIFI
-----
SYSSII File name :

Fill the SYSSII file name and press DUE
-----
K1=cancel  K2=interrupt  K3=refresh
-----
```

Figure 67: Select SYSSII file

Notes on user entries

- Uppercase and lowercase letters are permissible
- If the file you specify already exists and is not a SYSSII file, an error message is output and the program prompts for another name for the SYSSII file.

The next step depends on what you selected in the main menu:

- If you selected 1 (*Open SYSSII ...*), the SYSSII menu appears (see [figure 71 "Menu functions of the SYSSII menu"](#)).
- If you selected 3 (*Show SYSSII ...*), you must now define the parameters for SYSSII file output (see [figure 68](#)).
- If you selected 4 (*Check SYSSII ...*), you must now define the parameters for SYSSII file check.

Define output parameters for the SYSSII file

```

PROGRAM : IMON-SIC          Show Option Selection          PANEL : SHATT
-----
SYSSII : SYSSII.TESTPROG.010

Information : 1 1. Minimum
              2. All attributes

Output : 1 1. SYSOUT
          2. SYSLST
          3. File
              Name :
              Overwrite : 1 1. Yes
                          2. No

Fill the Attribute and options fields and press DUE
-----
K1=cancel K2=interrupt K3=refresh
-----

```

Figure 68: Define output parameters for the SYSSII file

- SYSSII** IMON-SIC automatically enters the name of the SYSSII file you selected in this field.
- Information** Defines the scope of the SHOW output. The options are:
- 1 (Minimum) Output consists of the name of the release unit, its main version, correction state and functional level, plus the associated release items with functional level and logical IDs.
 - 2 (All attributes) Over and above the scope for “Minimum” the output includes the attributes of the release items plus comments, dependencies and parameter text, as applicable.
- These attributes are assigned to the release items in the “Modify Release Item Attributes” mask (see [figure 72 \(Modify release unit attributes \)](#)).
- A hash (#) in the first column indicates a dummy release item.
- Output** Defines the output medium. The options are:
- 1 (SYSOUT) Write the contents of the SYSSII file to SYSOUT.
 - 2 (SYSLST) Write the contents of the SYSSII file to SYSLST.
 - 3 (File) Write the contents of the SYSSII file to a file.
- Name** Specify the name of the output file (max. 54 characters).
- Overwrite** The attribute for overwriting an existing output file.
- 1 (Yes) Default: A already existing file will be overwritten.
 - 2 (No) The output file will be created. If it already exists, then an error message is output, the file name is highlighted in the mask and new input is expected.

Once the SYSSII file has been output, IMON-SIC returns to the main menu.

Examples of the show output can be found in the [section "Example with masks from the main menu"](#).

6.2.5 Create SYSSII file

If you select 2 (*New SYSSII ...*), you must begin by initializing the SYSSII file, in other words entering the name and the attributes of the release unit ([figure 69](#)).

The attributes of a release unit are the type, the main-version number, the correction state, the functional level on which the release unit runs and comments if applicable.

```
PROGRAM : IMON-SIC          Modify Release Unit Attributes          PANEL : MRUAT
-----
Name :
Main version : .           Correction state :           Functional level : U (U/P/B)
Comments : N (Y/N)

Fill the attributes and press DUE
-----
K1=cancel K2=interrupt K3=refresh
-----
```

Figure 69: Initialize a SYSSII file

Name	Enter the release unit name as <text 1..30 without-sep> (see " Installation unit (IU) ").
Main version	The main version must be entered in the form <nn.n>, where n is a digit from 0 to 9.
Correction state	Enter the correction state as a three-character code <aso> (a=A..Z, so=00..99).
Functional Level	The functional level of the release unit. The options are: U Task Unprivileged, TU P Task Privileged, TPR B Both, if the release unit or parts of the release unit can be executed in both TU and TPR.
Comments	Comments as applicable. A comment can consist of up to 3 lines of 72 characters per line.
N (No)	Default: No comments are to be entered.
Y (Yes)	A comment is to be entered. The "Edit comment lines" screen is output. (for SYSSII files in the customer format with a maximum of 3 lines, i.e. 70 characters)

6.2.6 Example with masks from the main menu

The example below illustrates how to obtain the SHOW output for the IMON-BAS release unit.

Begin by starting IMON-SIC with the BS2000 command START-IMON-SIC.

The main menu appears as soon as the program starts.

```
PROGRAM : IMON-SIC V03.4          MAIN MENU          PANEL : MAIME
-----
                                     1. Open SYSSII ...
                                     2. New SYSSII ...
                                       Type : S (C=Customer / S=SU BS2000)
                                     3. Show SYSSII ...
                                     4. Check SYSSII ...

Choose the action; set the view type and press DUE
-----
K2=interrupt  K3=refresh  F3=exit
-----
```

Enter 3 in the main menu to select “Show SYSSII ...”, because you want to view a SYSSII file.

Press [DUE] to confirm your choice: the next mask appears and you are prompted for the name of your SYSSII file. In our example, the name of the file is SYSSII.IMON-BAS.034.

```
PROGRAM : IMON-SIC          SYSSII File Selection          PANEL : SIIFI
-----
SYSSII File name : SYSSII.IMON-BAS.034

Fill the SYSSII file name and press DUE
-----
K1=cancel  K2=interrupt  K3=refresh
-----
```

Press [DUE] to confirm your entry: the mask for defining the output parameters appears.

PROGRAM : IMON-SIC Show Option Selection PANEL : SHATT

SYSSII : SYSSII.IMON-BAS.034

Information : 1 1. Minimum
 2. All attributes

Output : 1 1. SYSOUT
 2. SYSLST
 3. File
 Name :
 Overwrite : 1 1. Yes
 2. No

Fill the Attribute and options fields and press DUE

K1=cancel K2=interrupt K3=refresh

You want to view the SYSSII file on screen with the minimum scope of information. This is the default, so all you have to do is press [DUE] to confirm: the contents of the SYSSII file are displayed on screen:

Release Unit	Version	F.lev	
Release Item	Target	F.lev	Logical-ID

IMON-BAS	03.4A00	BOTH	
SINLIB.IMON-BAS.034	A (COM)	TU	SINLIB
SKMLNK.IMON-BAS.034.TPR	K (SKM)	TPR	SYSLNK.TPR
SYSDAT.IMON-BAS.034.IDF	A (COM)	TU	SYSDAT.IDF
SYSDAT.IMON-BAS.034.POSIX	A (COM)	TU	SYSDAT.POSIX
#SYSFHS.IMON-BAS.034		TU	SYSFHS
SYSFHS.IMON-BAS.034.D	A (COM)	TU	SYSFHS.D
SYSFHS.IMON-BAS.034.E	A (COM)	TU	SYSFHS.E
SYSLIB.IMON-BAS.034	A (COM)	TU	SYSLIB
SYSLNK.IMON-BAS.034.TPR	S (390)	TPR	SYSLNK.TPR
SYSLNK.IMON-BAS.034.TU	A (COM)	TU	SYSLNK.TU
SYSMES.IMON-BAS.034	A (COM)	TU	SYSMES
SYSNRF.IMON-BAS.034	A (COM)	TU	SYSNRF
SYSPAR.IMON-BAS.034	A (COM)	TU	SYSPAR
SYSPRC.IMON-BAS.034	A (COM)	TU	SYSPRC
#SYSREP.IMON-BAS.034		TU	SYSREP
SYSRMS.IMON-BAS.034	A (COM)	TU	SYSRMS
SYSSDF.IMON-BAS.034	A (COM)	TU	SYSSDF
SYSSII.IMON-BAS.034	A (COM)	TU	SYSSII
SYSSSC.IMON-BAS.034	A (COM)	TPR	SYSSSC

This output is followed by a return to the main menu. Press [F3] to terminate IMON-SIC.

The listing below shows the maximum scope of information on the SYSSII file, which would have been supplied if you had selected 2 for "Information" in the above example:

----- Release Unit -----
Release Unit Name : IMON-BAS Version : V03.4A00
Functional Level : BOTH

User-ID : *DEFLUID Mandatory : No
----- Dependences :
Release Unit : *SYSTEM (**.*) **** *** 210* A
Release Unit : IMON-GPN (03.4*) A00* ADD 210* A

----- Release Item -----
Release Item Name : SINLIB.IMON-BAS.034 Dummy : No
Language :
Target System : A (COM)
----- Utilization :
Functional Level : TU
Logical-ID : SINLIB
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No
----- Installation :
Type : *NP
Filename : SINLIB.IMON-BAS.034 Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SKMLNK.IMON-BAS.034.TPR Dummy : No
Language :
Target System : K (SKM)
----- Utilization :
Functional Level : TPR
Logical-ID : SYSLNK.TPR
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No
----- Installation :
Type : DAT
Filename : SKMLNK.IMON-BAS.034.TPR Mandatory : No
User-ID : *STD Mandatory : No
User-access : Owner-Only Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SYSDAT.IMON-BAS.034.IDF Dummy : No
Language :
Target System : A (COM)
----- Utilization :
Functional Level : TU
Logical-ID : SYSDAT.IDF
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : Yes
----- Installation :
Type : *NW
Filename : SYSDAT.IMON.IDF Mandatory : Yes
User-ID : TSOS Mandatory : Yes
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SYSDAT.IMON-BAS.034.POSIX Dummy : No
Language :
Target System : A (COM)
----- Utilization :
Functional Level : TU

Logical-ID : SYSDAT.POSIX
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No
----- Installation :
Type : DAT
Filename : SYSDAT.IMON-BAS.034.POSIX Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SYSFHS.IMON-BAS.034 Dummy : Yes
----- Utilization :
Functional Level : TU
Logical-ID : SYSFHS
Filename type : Full

----- Release Item -----
Release Item Name : SYSFHS.IMON-BAS.034.D Dummy : No
Language : D
Target System : A (COM)
----- Utilization :
Functional Level : TU
Logical-ID : SYSFHS.D
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No
----- Installation :
Type : DAT
Filename : SYSFHS.IMON-BAS.034.D Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SYSFHS.IMON-BAS.034.E Dummy : No
Language : E
Target System : A (COM)
----- Utilization :
Functional Level : TU
Logical-ID : SYSFHS.E
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No
----- Installation :
Type : DAT
Filename : SYSFHS.IMON-BAS.034.E Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----
Release Item Name : SYSLIB.IMON-BAS.034 Dummy : No
Language :
Target System : A (COM)
----- Utilization :
Functional Level : TU
Logical-ID : SYSLIB
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : Yes
----- Installation :
Type : PLM

Filename : SYSLIB.IMON-BAS.034 Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Parameter Text :
LIB MACROLIB

----- Release Item -----

Release Item Name : SYSLNK.IMON-BAS.034.TPR Dummy : No
Language :

Target System : S (390)

----- Utilization :

Functional Level : TPR
Logical-ID : SYSLNK.TPR
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : DAT
Filename : SYSLNK.IMON-BAS.034.TPR Mandatory : No
User-ID : *STD Mandatory : No
User-access : Owner-Only Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----

Release Item Name : SYSLNK.IMON-BAS.034.TU Dummy : No
Language :

Target System : A (COM)

----- Utilization :

Functional Level : TU
Logical-ID : SYSLNK.TU
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : DAT
Filename : SYSLNK.IMON-BAS.034.TU Mandatory : No
User-ID : *STD Mandatory : No
User-access : All-Users Access : Read
IPL file : No Format : NK4
Migrate : Standard

----- Release Item -----

Release Item Name : SYSMES.IMON-BAS.034 Dummy : No
Language :

Target System : A (COM)

----- Utilization :

Functional Level : TU
Logical-ID : SYSMES
Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : MES
Filename : SYSMES.IMON-BAS.034 Mandatory : No
User-ID : *STD Mandatory : No
User-access : Special Access : Read
IPL file : No Format : NK4
Migrate : Inhibit

----- Release Item -----

Release Item Name : SYSNRF.IMON-BAS.034 Dummy : No
Language :

Target System : A (COM)

----- Utilization :

Functional Level : TU

Logical-ID : SYSNRF

Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : DAT

Filename : SYSNRF.IMON-BAS.034 Mandatory : No

User-ID : *STD Mandatory : No

User-access : All-Users Access : Read

IPL file : No Format : NK4

Migrate : Inhibit

----- Release Item -----

Release Item Name : SYSPAR.IMON-BAS.034 Dummy : No

Language :

Target System : A (COM)

----- Utilization :

Functional Level : TU

Logical-ID : SYSPAR

Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : DAT

Filename : SYSPAR.IMON-BAS.034 Mandatory : No

User-ID : *STD Mandatory : No

User-access : All-Users Access : Read

IPL file : No Format : NK4

Migrate : Standard

----- Release Item -----

Release Item Name : SYSPRC.IMON-BAS.034 Dummy : No

Language :

Target System : A (COM)

----- Utilization :

Functional Level : TU

Logical-ID : SYSPRC

Path-Name Attributes in IMON-GPN : Mandatory : Yes Update : No

----- Installation :

Type : DAT

Filename : SYSPRC.IMON-BAS Mandatory : Yes

User-ID : TSOS Mandatory : Yes

User-access : All-Users Access : Read

IPL file : No Format : NK4

Migrate : Standard

----- Release Item -----

Release Item Name : SYSREP.IMON-BAS.034 Dummy : Yes

----- Utilization :

Functional Level : TU

Logical-ID : SYSREP

Filename type : Full

----- Installation :

Default Pathname : \$TSOS.SYSREP.IMON-BAS.034

----- Release Item -----

Release Item Name : SYSRMS.IMON-BAS.034 Dummy : No

Language :

Target System : A (COM)

----- Utilization :

```

Functional Level      : TU
Logical-ID           : SYSRMS
Path-Name Attributes in IMON-GPN : Mandatory : Yes   Update   : No
----- Installation :
Type                : REP
Filename            : SYSRMS.IMON-BAS.034           Mandatory : No
User-ID             : *STD                          Mandatory : No
User-access         : Owner-Only                    Access    : Read
IPL file            : No                            Format     : NK4
Merged in CP: No

```

----- Release Item -----

```

Release Item Name   : SYSSDF.IMON-BAS.034           Dummy    : No
Language           :
Target System      : A (COM)

```

```

----- Utilization :
Functional Level    : TU
Logical-ID         : SYSSDF
Path-Name Attributes in IMON-GPN : Mandatory : Yes   Update   : No
----- Installation :
Type               : SDF
Filename           : SYSSDF.IMON-BAS.034           Mandatory : No
User-ID            : *STD                          Mandatory : No
User-access        : Special                       Access    : Read
IPL file           : No                            Format     : NK4
Migrate           : Inhibit

```

```

----- Parameter Text :
DYN IMON

```

----- Release Item -----

```

Release Item Name   : SYSSII.IMON-BAS.034           Dummy    : No
Language           :
Target System      : A (COM)

```

```

----- Utilization :
Functional Level    : TU
Logical-ID         : SYSSII
Path-Name Attributes in IMON-GPN : Mandatory : Yes   Update   : No
----- Installation :
Type               : DAT
Filename           : SYSSII.IMON-BAS.034           Mandatory : No
User-ID            : *STD                          Mandatory : No
User-access        : Owner-Only                    Access    : Read
IPL file           : No                            Format     : NK4
Migrate           : Standard

```

----- Release Item -----

```

Release Item Name   : SYSSSC.IMON-BAS.034           Dummy    : No
Language           :
Target System      : A (COM)

```

```

----- Utilization :
Functional Level    : TPR
Logical-ID         : SYSSSC
Path-Name Attributes in IMON-GPN : Mandatory : Yes   Update   : No
----- Installation :
Type               : SSC
Filename           : SYSSSC.IMON-BAS.034           Mandatory : No
User-ID            : *STD                          Mandatory : No
User-access        : Owner-Only                    Access    : Read
IPL file           : No                            Format     : NK4

```

Migrate : Standard

----- ETPND -----

Name : IMOSII

Domain : IMON-BAS

Version : 034

6.2.7 Menu functions of the SYSSII menu

The SYSSII menu ([figure 71](#)) contains the functions for modifying, creating and saving a SYSSII file. These functions apply to the SYSSII file you selected or defined in the main menu. IMON-SIC automatically inserts the name and version of the release unit.

```
PROGRAM : IMON-SIC          SYSSII Menu          PANEL : SIIME
-----
RU Name : TESTPROG          RU Main version : 01.0

    1. Modify RU Attributes ...
    2. Modify RU Structure ...
    3. Modify RI Attributes ...

    4. Add logging ...
    5. Erase logging

    6. Show SYSSII ...
    7. Save SYSSII ...

Choose the action and press DUE
-----
K1=return to MAIN menu  K2=interrupt  K3=refresh
-----
```

Figure 71: SYSSII menu

Notes on filling in the mask

- You are expected to input a number (1, 2, ..., 7).
- The sequence of masks depends on the selection, see “Sequence of IMON-SIC masks” ([figure 64 \(Mask layout\)](#)).
- When option “5” or “7” is selected, a message indicating success is output.
- A warning is output if you select 3 and attempt to modify release item attributes for a SYSSII file that contains no release items.

6.2.8 Modify release unit attributes

Selection 1 (*Modify RU Attributes ...*) enables you to modify the name and attributes of a release unit. The attributes of a release unit include the type of release unit, the mainversion number, the correction state, the functional level on which the release unit executes, and comments if applicable.

Any changes you make to the name and/or main version of a release unit in this mask are also implemented in the name of the SYSSII file you generate.

```
PROGRAM : IMON-SIC          Modify Release Unit Attributes          PANEL : MRUAT
-----
Name : TESTPROG
Main version : 01.0      Correction state : A00      Functional level : B (U/P/B)
Comments : N (Y/N)

Fill the attributes and press DUE
-----
K1=cancel  K2=interrupt  K3=refresh
-----
```

Figure 72: Modify release unit attributes

All fields in this mask are assigned preset values by IMON-SIC, however they can be changed.

- | | |
|------------------|---|
| Name | Enter the release unit name as <text 1..30 without-sep> (see " Installation unit (IU) "). |
| Main version | The main version must be entered in the form <nn.n>, where n is a digit from 0 to 9. |
| Correction state | Enter the correction state as a three-character code <aso> (a=A..Z, so=00..99). |
| Functional Level | The functional level of the release unit. The options are:

U Task Unprivileged, TU
P Task Privileged, TPR
B Both, if the release unit or parts of the release unit can be executed in both TU and TPR. |
| Comments | Comments can be specified if desired. The preset is N (No). Three lines with up to 70 characters each are allowed for each comment. |

The SYSSII menu reappears when you confirm the settings.

6.2.9 Modify release unit structure

Selection 2 (*Modify RU Structure ...*) enables you to modify the structure of a release unit by:

- adding new release items
- removing release items
- renaming a release item
- declaring release items as dummy release items.

```
PROGRAM : IMON-SIC          Modify Release Unit structure          PANEL : MRUST
-----
RU Name : IMON-BAS          RU Main Version : 03.4
Op      Release Items      Dummy Target No more delivered
SINLIB.+
SKMLNK.+ .TPR              A
SYSDAT.+ .IDF              A
SYSDAT.+ .POSIX            A
SYSFHS.+                   Y   A
SYSFHS.+ .D                A
SYSFHS.+ .E                A
SYSLIB.+                   A
SYSLNK.+ .TPR              S
SYSLNK.+ .TU               A
SYSMES.+                   A
SYSNRF.+                   A

Set Op (D/M); Set Name and Dummy and press DUE.
-----
K1=cancel K2=interrupt K3=refresh F2=prev page F3=next page
-----
```

Figure 73: Modify release unit structure

Notes on user entries

- Specify the release-item name as <filename 1..30 without-cat-user-gen-vers> (see "[Installation item \(II\)](#) "). The length of the name is calculated once the metacharacter has been replaced by the appropriate string.
- To delete a release item, enter "d" (for delete) in the "Op" field of the line corresponding to the release item.
- To rename a release item, change the "Release Item" field. The RI attributes belonging to the item are not affected by the change.
- To define a release item as a dummy release item, enter "y" in the "Dummy" column. A blank in this column resets the value for dummy. Attributes which do not belong to the new type are lost (RI becomes a dummy RI and vice versa).
- A plausibility check is run when you press [DUE]. If no adjustments are necessary after this check, press [DUE] again to register the new structure.
- Incorrect entries in the "Op" field are highlighted.

-
- You can use the IMON-SIC metacharacter '+' in RI names. The metacharacter stands for “<RU name>.<RU main version>” (e.g. : SYSSSC.+CL4 corresponds to SYSSSC.<RU name>.<RU main version>.CL4).

The RU version is generated without the period: for example version 10.0 yields 100.

The metacharacter reduces the number of changes required when a new main version is introduced. Changing the main version in the “Modify RU Attributes” mask leads to a great deal of changes, even if the RU structure and the RI attributes do not change from version to version. The metacharacter is stored in the SYSSII file.

Instead of being replaced automatically in the mask, the metacharacter is retained so that the mechanism remains available for future changes to the RI name. If an RI name contains the string “<RU name>.<RU main version>”, this string is not converted to the metacharacter.

- Once all changes have been completed, a check is run to ascertain that two release items do not have the same name.

All metacharacters are evaluated. For example, an RI with the name “RI.+” and another RI with the name “RI.<RU name>.<RU version>” are recognized as duplicates.

If duplicates are found, the double RI names are underscored in the mask and a message is issued.

The cursor is positioned on the first error in the item list.

- Once the checks have been completed and all errors rectified, the attributes mask is output for each new or modified release item and dummy release item (see [figure 74 \(Modify release item attributes \)](#) for release items and [figure 75 \(Modify release item attributes \)](#) for dummy release items).

If you press [K1] into the next mask “Modify (Dummy) Release Item Attributes”, the “Modify Release Unit Structure” mask described here again appears, showing all the modifications to the structure defined to date.

If you press [K1] again at this point, the new structure is not saved.

6.2.10 Modify release item attributes

Selection 3 (*Modify RI Attributes ...*) enables you to modify the attributes of release items.

Begin by selecting the release items or dummy release items whose attributes you want to change (see [figure 74](#)).

```
PROGRAM : IMON-SIC          Release Item Selection          PANEL : RISEL
-----
RU Name : TESTPROG          RU Main Version : 01.0
Release Items              Dummy Target No more delivered
SINLIB.+                  A (COM)
SKMLNK.+..TPR             K (SKM)
SYSDAT.+..IDF             A (COM)
SYSDAT.+..POSIX           A (COM)
SYSPHS.+                  Y A (COM)
SYSPHS.+..D              A (COM)
SYSPHS.+..E              A (COM)
SYSLIB.+                  A (COM)
SYSLNK.+..TPR            S (390)
SYSLNK.+..TU             A (COM)
SYSMES.+                  A (COM)
SYSNRF.+                  A (COM)

Select RI(s) with '/' for modification and press DUE
-----
K1=cancel K2=interrupt K3=refresh F2=prev page F3=next page
-----
```

Figure 74: Select release items

Notes on user entries

- Enter a slash (/) in the first column to select an item.
- A “Y” in the “Dummy” column indicates a dummy release item.
- The metacharacter (+) is not replaced.

The mask for modifying attributes is then displayed for each release item or dummy release item in turn (see [figure 75](#) for release items and [figure 76](#) for dummy release items).

Modify release item attributes

```
PROGRAM : IMON-SIC          Modify Release Item Attributes    PANEL : MRIOT
-----
RI Name : SYSSDF.TESTPROG.010      Functional level : U (U/P/B)
Language : (E=English / D=Deutsch / ...)
Logical Id definition
Logical Id :
Update      : N (Y/N)      Mandatory : Y (Y/N)
Default file attributes at installation time
User-Access : A (A/O/S)    Access      : R (W/R)    NK4-Disk : Y (Y/N)
Migrate     : A (A/I)      NK2-Disk   : Y (Y/N)
Comments

Fill the attributes and press DUE
-----
K1=cancel K2=interrupt K3=refresh
-----
```

Figure 75: Modify release item attributes

An entry in the “Logical Id” field is mandatory. All other fields are assigned values by IMON-SIC, however you can change them at your discretion.

RI-Name

Release item name.

Functional level

The functional level in which a release item can execute depends on the functional level of the higher-order release unit.

You cannot change this attribute unless B (for “both”) was entered as the functional level in the “Modify Release Unit Attributes” mask ([figure 71 \(Menu functions of the SYSSII menu \)](#)). Otherwise, the release item has the same functional level as the release unit to which it belongs.

Possible values:

U: Task Unprivileged (TU)

P: Task Privileged (TPR)

B: Both function states when the release unit or a part of it can run as TU as well as TPR.

Logical Id definition

Describes the logical ID of the release item.

Logical Id

The logical ID assigned to a release item. An entry in this field is mandatory. This logical ID is evaluated by IMON-GPN

Update

Permit/do not permit updating of the file name belonging to the logical ID after installation (Y for yes, N for no).

Mandatory

Mandatory assignment of a logical ID to a file name during installation (Y for yes, N for no).

“Update” and “Mandatory” cannot both be set to N for no. If N is entered for both, the “Mandatory” field in the mask is underscored and must be changed.

Default file attributes

The following file attributes can be assigned for a release item

User-Access	A	All-Users
	O	Owner-Only
	S	Special
Migrate	I	Migrate=Inhibited and Backup-Class=Any
	E	Migrate=Inhibited and Backup-Class=E
	S	Standard setting of the system
Access	R	Read
	W	Write
NK2-Disk	Y	Yes
	N	No
NK4-Disk	Y	Yes
	N	No

Comments

Comments as applicable. A comment can consist of up to 3 lines of 70 characters per line.

If you press [K1], the changes in the current mask are lost and the preceding mask is displayed again.

Invalid entries are underscored in the mask and you are prompted for a correct entry.

Modify dummy release item attributes

```
PROGRAM : IMON-SIC          Modify Dummy Item Attributes          PANEL : MDIAT
-----
Name : SYSFHS.TESTPROG.010
Functional level : U (U/P/B)
Logical Id :
Comments : N (Y/N)
Default Pathname:

Fill the attributes and press DUE
-----
K1=cancel K2=interrupt K3=refresh
-----
```

Figure 76: Modify dummy release item attributes

An entry in the “Logical Id” field is mandatory. All other fields are assigned values by IMON-SIC, however you can change them at your discretion.

Functional Level

The functional level in which a dummy release item can execute depends on the functional level of the higher-order release unit.

You cannot change this attribute unless B (for “both”) was entered as the functional level in the “Modify Release Unit Attributes” mask ([figure 71 \(Menu functions of the SYSSII menu \)](#)). Otherwise, the dummy release item has the same functional level as the release unit to which it belongs.

Possible values:

- U: Task Unprivileged (TU)
- P: Task Privileged (TPR)
- B: Both (TU and TPR)

Logical Id

The logical ID assigned to a dummy release item. An entry in this field is mandatory. This logical ID is evaluated by IMON-GPN.

Comments

Comments can optionally be specified. A comment can consist of up to 3 lines with 70 characters per line.

Possible values:

- N: Preset. No comment present (No)
- Y: Comment present (Yes)

Default path name

Default path name in the form \$<user-id>.<filename> that is assigned to the dummy release item.

If you press [K1], all changes are lost.

If you called this mask after the “Modify RU Structure” mask, you are returned to this preceding mask. The structure modifications implemented in this mask are retained.

Otherwise, you are returned to the “Release Item Selection” mask, which contains the list of selected release items.

6.2.11 Add logging information

Selection 4 (*Add logging ...*) enables you to

- add logging information to the SYSSII file
- change logging information in the SYSSII file.

This action calls the EDT file processing program.

```
1.00 *** SYSSII Logging ***
2.00
3.00
4.00
5.00
6.00
7.00
8.00
9.00
10.00
11.00
12.00
13.00
14.00
15.00
16.00
17.00
18.00
19.00
20.00
21.00
22.00
Update or add Logging then type RET or HALT at EDT command prompt to end
```

Figure 77: Add logging information

The logging information can now be recorded or edited. After editing, the EDT is terminated with the EDT statement RETURN or HALT (refer also to the information in the EDT screen status line). The SYSSII menu is then output once more.

The logging information is saved to the SYSSII file. IMON-SIC creates a copy under the name “#T.<x>.IMON-SIC.<tsn>” for editing (<x> is either a letter or a number, <tsn> is the Task Sequence Number). This file is not deleted automatically after the file is edited and the EDT terminated.

6.2.12 Erase logging

Selection 5 (*Erase logging*) enables you to erase logging information from a SYSSII file.

Erasure of the information is confirmed by a message.

6.2.13 Show SYSSII file

Selection 6 (*Show SYSSII ...*) requires you to enter the parameters for the SHOW output before the file can be shown.

The mask for these parameters corresponds to that called for the SHOW function of the main menu (see [figure 68 \(Open, display or check SYSSII file \)](#)). The mask is described on "[Description of the menu functions of the main menu](#)".

6.2.14 Save SYSSII file

Selection 7 (*Save SYSSII ...*) enables you to save your SYSSII file. All modifications are saved in the SYSSII file.

Figure 78: Save SYSSII file

```
PROGRAM : IMON-SIC          SYSSII File Selection          PANEL : SIIFISA
-----
SYSSII File name : SYSSII.TESTPROG.010
                                Overwrite ? Y (Y/N)

Fill the SYSSII file name and press DUE
-----
K1=cancel  K2=interrupt  K3=refresh
-----
```

- SYSSII file name** Name of the file in which the backup will be written.
The field contains the name of the open SYSSII file by default. The standard file name SYSSII.<RUname>.<RUversion> is used for a new SYSSII file.
- Overwrite** Specifies if an already existing file may be overwritten.
- N (No) Preset. If a file with the specified name already exists, then an error message appears and the screen is output again for correction purposes. Another file name can be entered now or *Overwrite=Y* is set.
 - Y (Yes) A already existing file will be overwritten.

If the file was saved correctly, you are returned to the SYSSII menu and a message confirming the save is issued.

6.2.15 Return to main menu

If the SYSSII file was not changed, then you return directly to the main menu by pressing the [K1] key.

If you have modified the SYSSII file in any way, you must first confirm the changes before you are returned to the main menu:

```
PROGRAM : IMON-SIC          Return Confirmation          PANEL : RETCO
-----
Current SYSSII : SYSSII.TESTPROG.010

Do you want to return to the Main Menu
without saving the current SYSSII ?

N (Y/N)

Respond by Yes or No and press DUE
-----
K1=cancel K2=interrupt K3=refresh
-----
```

Figure 79: Return to the main menu

Notes on user entries

- Answer Y if you *do not* want to save the changes before returning to the main menu.
- If you answer in the negative you are returned to the SYSSII menu ([figure 71 "Menu functions of the SYSSII menu "](#)). Select 7 (*Save SYSSII...*) to save your SYSSII file and then return to the main menu.
- Pressing [K1] has the same effect as answering in the negative.

6.3 IMON-SIC statements

The SDF interface from IMON-SIC is always started with the START-IMON-SIC command in procedure and batch mode. IMON-SIC expects statements in the SDF format as additional input.

The SDF interface can be started directly with the operand INPUT-INTERFACE=*SDF in the START-IMON-SIC command.

Overview of the IMON-SIC statements and the SDF standard statements

Statement	Function
CHECK-SII	Check the contents of a SYSSII file
SHOW-SII	Output the contents of a SYSSII file
END	Close files and terminate IMON-SIC

The following SDF standard statements can also be specified.

Statement	Function
EXECUTE-SYSTEM-CMD	Execute command while the program is running
HOLD-PROGRAM	Switch to the command mode
HELP-MSG-INFORMATION	Output the text of a system message to SYSOUT
MODIFY-SDF-OPTIONS	Change SDF settings
REMARK	Output comment
RESET-INPUT-DEFAULTS	Reset task-specific default values
RESTORE-SDF-INPUT	Redisplay the last input
SHOW-INPUT-DEFAULTS	Display task-specific default values
SHOW-INPUT-HISTORY	Output input buffer to SYSOUT
SHOW-SDF-OPTIONS	Display SDF settings
SHOW-STMT	Output syntax description of a statement

STEP	Mark section for error handling (only in procedure or ENTER file)
WRITE-TEXT	Output text to SYSOUT/SYSLST

The SDF standard statements are not described in this manual. You will find a detailed description in the “SDF Dialog Interface” manual [[2 \(Related publications \)](#)].

The IMON-SIC statements are described in alphabetical order according to the following format:

- Statement name and function
- Privileges
- Description the function
- Representation of the statement format

A description of the SDF syntax can be found in the Appendix starting on "[SDF syntax representation](#)".

- Description of the operands
- Notes

6.3.1 CHECK-SII Check contents of SII file

Privileges: STD-PROCESSING

Function

The contents of a SII file can be checked for errors (e.g. item name, version specification, completeness of the items) with the CHECK-SII statement. The check is only performed for SII files (Format=S) supplied with SOLIS2.

Format

CHECK-SII

SII-FILE = <filename 1..30 without-gen-vers>

,RANGE = *PARAMETERS(...)

*PARAMETERS(...)

| **FROM-VERSION** = 140 / <integer 140..210>

| **,TO-VERSION** = 210 / <integer 140..210>

Operands

SII-FILE = <filename 1..30 without-gen-vers>

Name of the SII file.

RANGE = *PARAMETERS(...)

Specifies for which BS2000 versions the check will be performed. The specification is made in the form of an interval with the lowest and highest versions to be checked:

FROM-VERSION = 140 / <integer 140..210>

BS2000 version with which the check starts. Possible specifications are 140, 150, 160, 170, 180, 190, 200 and 210.

TO-VERSION = 210 / <integer 140..210>

BS2000 version with which the check ends. Possible specifications are 140, 150, 160, 170, 180, 190, 200 and 210.

6.3.2 SHOW-SII Output contents of SYSSII file

Privileges: STD-PROCESSING

Function

The contents of a SYSSII file are output with the SHOW-SII statement. Either just the release unit and the release items are listed or all information on the release unit and the release items is output. The output can be sent to SYSOUT, SYSLST or a file.

The layout of the output corresponds to the layout of the output requested in menu mode using option 3 (*Show SYSSII*) in the main menu ("[Open, display or check SYSSII file](#)", see "[Example with masks from the main menu](#)" for an example).

Format

SHOW-SII

```
SII-FILE = <filename 1..30 without-gen-vers>  
,INFORMATION = *ITEM-LIST / *ALL-ATTRIBUTES  
,OUTPUT = *SYSOUT / *SYSLST / <filename 1..54 without-gen-vers>(…)  
           <filename 1..54 without-gen-vers>(…)  
           | WRITE-MODE = *NEW-FILE / *REPLACE-FILE
```

Operands

SII-FILE = <filename 1..30 without-gen-vers>

Name of the SYSSII file.

INFORMATION =

Specifies which information is to be output.

INFORMATION = *ITEM-LIST

Only the structure information for the release unit and the release items belonging to it are output. The output contains the name of the release unit, main version, correction state and function state as well as the corresponding release items with their function states and logical names.

INFORMATION = *ALL-ATTRIBUTES

All information (including the attributes, dependencies, parameter text and comments) on the release unit and the release items belonging to it are output.

OUTPUT =

Specifies the destination of the output.

OUTPUT = *SYSOUT

The information is output to SYSOUT.

OUTPUT = *SYSLST

The information is output to SYSLST.

OUTPUT = <filename 1..54 without-gen-vers>(…)

The information is output to the specified file.

WRITE-MODE = *NEW-FILE / *REPLACE-FILE

Specifies if an already existing output file is to be overwritten.

The preset value is *NEW-FILE, i.e. the output file is created if necessary, but it is not overwritten.

The output file is created if necessary or overwritten with *REPLACE-FILE.

6.4 The macros of IMON-SIC

The user is provided with an easy to use program interface to execute IMON-SIC functions for evaluating SYSSII files with the IMOKIT, IMOKIA and IMOKIS macros.

SYSSII files are created and edited through the IMON-SIC menu interface only.

The following functions are invoked directly from within an Assembler program and executed in real time. A function is called using the corresponding macro where the function name with a leading asterisk is to be specified in the FUNCT parameter.

List of functions

Function	Macro	Meaning
SIIOOPEN	IMOKIT	Open SYSSII file
SIIREAD	IMOKIT	Read information from SYSSII file
SIICLOSE	IMOKIT	Close SYSSII file
SIIRCOM	IMOKIA	Read comments on the release unit or on a release item from the SYSSII file
SIIRDEP	IMOKIA	Read the old dependencies of the release unit from the SYSSII file
SIIRKON	IMOKIA	Read the new dependencies of the release unit from the SYSSII file
SIIRPAR	IMOKIA	Read the parameter text of a release items from the SYSSII file
SIISHOW	IMOKIS	Display the contents of a SYSSII file

Notes

1. Internal data fields (e.g. the file descriptor <prefix><macid>FD) are prepared for additional function calls and may not be deleted or overwritten.
2. When strings are explicitly specified in the parameter list (e.g. a SYSSII file name), then at least one space character (X'40') or X'00' must be added at the end of the valid string as the last character of the string.
3. X'00' is the preset value for the last character of the string when outputting strings in the parameter list (e.g. unit name) for the IMOKIT and IMOKIA macro functions SIIOOPEN, SIIREAD, SIIRCOM, SIIRDEP, SIIRKON and SIIRPAR. The last character of the string can be explicitly set to the space character (X'40') in the corresponding parameter list of the IMOKIT or IMOKIA macro in the IREADEL or IAREDEL field.

6.4.1 IMOKIT Read information from SYSSII file

Information can be read from a SYSSII file with the IMOKIT macro. The SYSSII file must first be opened with the SIIOPEN function. Information on release units defined in the file are read upon opening. With the SIIREAD function information on the release items can be read sequentially from an open SYSSII file. The SYSSII file is closed with the function SIICLOSE.

A SYSSII file can only be edited with the SIIRCOM, SIIRDEP, SIIRKON and SIIRPAR functions of the IMOKIA macro ("[IMOKIA Read comments, dependencies and parameter text](#)") when the SYSSII file was opened with the SIIOPEN function.

Macro	Operands
IMOKIT	MF= <u>D</u> / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= <u>REA</u> / <char (3)> ,PARAM= <adr> / (<reg> ,FUNCT= <u>*SIIOPEN</u> / *SIIREAD / *SIICLOSE ,SYSSII= <u>X'40'</u> / <c-string 1..54> / <var: char:55>

The MF, PREFIX, MACID, and PARAM parameters are described in "Macro forms" on "[Macro syntax](#)".

FUNCT	The desired function.
= <u>*SIIOPEN</u>	Open a SYSSII file.
= *SIIREAD	Read information from a SYSSII file.
= *SIICLOSE	Close the SYSSII file.
SYSSII	Name of the SYSSII file. The name only needs to be specified when opening the file (FUNCT=*SIIOPEN).
= <u>X'40'</u>	No specification.
=<c-string 1..54>	Explicit specification of the SYSSII filename.
=<var: char:55>	Symbolic address containing the name of the SYSSII file. May not be specified together with MF=L.

Parameter list

```

IMOKIT  IMOKIT MF=D
1 IMOKIT  MFTST MF=D,PREFIX=I,MACID=REA,ALIGN=F,          C
1          DMACID=REA,SUPPORT=(E,D,C,M,L),DNAME=REA_PL
2 IMOKIT  DSECT ,
2          *,##### PREFIX=I, MACID=REA #####
1 IREAPVER      EQU   23          value for parameter list
1 *
1 IREAFOPN      EQU   150         value for SIIOPEN
1 *
1 IREAFCLO      EQU   151         value for SIICLOSE
1 *
1 IREAFRED      EQU   152         value for SIIREAD
1 *
1 *  PARAMETER LIST OF IMON-SIC
1 IREASTANDARD_HEADER FHDR MF=(C,IREA),EQUATES=NO
2 IREASTANDARD_HEADER DS      0A
2 IREAFHE DS      0XL8          0  GENERAL PARAMETER AREA HEADER
2 *
2 IREAIIFID DS      0A          0  INTERFACE IDENTIFIER
2 IREAFCTU DS      AL2          0  FUNCTION UNIT NUMBER
2 *
2 *                                BIT 15  HEADER FLAG BIT,
2 *                                MUST BE RESET UNTIL FURTHER NOTICE
2 *                                BIT 14-12 UNUSED, MUST BE RESET
2 *                                BIT 11-0  REAL FUNCTION UNIT NUMBER
2 IREAFCT DS      AL1          2  FUNCTION NUMBER
2 IREAFCTV DS      AL1          3  FUNCTION INTERFACE VERSION NUMBER
2 *
2 IREARET DS      0A          4  GENERAL RETURN CODE
2 IREASRET DS      0AL2         4  SUB RETURN CODE
2 IREASR2 DS      AL1          4  SUB RETURN CODE 2
2 IREASR1 DS      AL1          5  SUB RETURN CODE 1
2 IREAMRET DS      0AL2         6  MAIN RETURN CODE
2 IREAMR2 DS      AL1          6  MAIN RETURN CODE 2
2 IREAMR1 DS      AL1          7  MAIN RETURN CODE 1
2 IREAFHL EQU      8          8  GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *                                STD HEADER
1 *  main return codes
1 IREAOK      EQU   0          Success
1 IREAESIN    EQU   1          Invalid SYSSII
1 IREAEIPL    EQU   2          Invalid Parameter List
1 *
1 IREAEIFC    EQU   7          Invalid Function Called
1 IREAESIF    EQU   20         Bad SYSSII format
1 IREAEEDMS   EQU   21         DMS error
1 IREAEFNO    EQU   22         File not open

```

1	IREAEOF	EQU	30	End of file
1	IREAES	EQU	256	System error
1	IREAEI	EQU	257	Internal error
1	*			
1	IREAFD	DS	A	FILE DESCRIPTOR
1	IREADMS	DS	H	DMS ERROR
1	IREADEL	DS	X	STRING DELIMITOR
1	IREASII	DS	CL55	SYSSII FILE NAME
1	IREASIV	DS	CL7	SYSSII FILE VERSION
1	IREACN	DS	CL9	ETPND COMPONENT NAME
1	IREACV	DS	CL4	ETPND COMPONENT VERSION
1	IREACD	DS	CL14	ETPND COMPONENT DOMAIN
1	IREACPM	DS	20CL9	ETPND COMPONENT PM NUMBER
1	IREACPM#	EQU	20	
1	IREAUN	DS	CL31	UNIT NAME
1	IREAUV	DS	CL8	UNIT VERSION
1	IREAUFL	DS	X	UNIT FUNCTIONAL LEVEL
1	IREAAU	DS	CL9	UNIT USER ID
1	IREAUMU	DS	X	UNIT MANDATORY USER ID
1	IREAR1	DS	XL2	filler 1
1	IREAODEP	DS	F	UNIT OLD DEPENDENCE NUMBER
1	IREADEP	DS	F	UNIT DEPENDENCE NUMBER
1	IREAUC	DS	F	UNIT COMMENT NUMBER
1	IREAIN	DS	CL31	ITEM NAME
1	IREAIDUM	DS	X	DUMMY ITEM FLAG
1	IREAIT	DS	X	ITEM TARGET
1	IREAR1A	DS	XL3	filler 1a
1	IREAILG	DS	X	ITEM LANGUAGE
1	IREAR2	DS	XL3	filler 2
1	IREAIC	DS	F	ITEM COMMENT NUMBER
1	IREAIFL	DS	X	ITEM FUNCTIONAL LEVEL
1	IREAIL	DS	CL31	ITEM LOGICAL ID
1	IREAILM	DS	X	MANDATORY PATHNAME
1	IREAILU	DS	X	UPDATE PATHNAME
1	IREAILF	DS	X	FILENAME TYPE
1	IREAITP	DS	CL4	ITEM TYPE
1	IREAIF	DS	CL39	ITEM FILENAME
1	IREAIFM	DS	X	MANDATORY FILENAME
1	IREAIUI	DS	CL9	ITEM USER ID
1	IREAIMU	DS	X	ITEM MANDATORY USER ID
1	IREASH	DS	X	USER-ACCESS ATTRIBUTE
1	IREAAC	DS	X	ACCESS ATTRIBUTE
1	IREAMI	DS	X	MIGRATE ATTRIBUTE
1	IREAFO	DS	X	FORMAT ATTRIBUTE
1	IREAIPL	DS	X	IPL FILE
1	IREAR3	DS	XL2	filler 3
1	IREAPAR	DS	F	INSTALLATION PARAMETER NUMBER
1	IREA#	EQU	*-IREASTANDARD_HEADER	

Programming notes

The functions SIIOPEN, SIIREAD and SIICLOSE are related to each other. They use the same parameter list even when SIIREAD is called a multiple number of times.

The information on the first release item of the opened SYSSII file is output in the parameter list during the first SIIREAD call . The information on the rest of the release items is obtained by calling SIIREAD again and again until the return code “EOF reached in SYSSII file” is returned.

The SIIREAD function evaluates the metacharacter “+” in the SYSSII file and returns the complete release item name.

When several parameter lists are used, several SYSSII files can be processed in parallel.

The name of the SYSSII file (see SYSSII parameter) can also be passed in the parameter list in the IREASII field to call the SIIOPEN function.

Additional data for evaluating the parameter list is passed in addition to the return code for the SIIOPEN and SIIREAD functions:

- The attributes of the release unit are returned for FUNCT=*SIIOPEN:

Assembler	C interface	Meaning
IREAFD	sii_fd	Internal file descriptor (may not be changed)
IREADMS	dms_error	Denotes the DMS error (if there was one) that occurred when the SYSSII file was opened
IREASII	sii_name	Name of the SYSSII file
IREASIV	sii_file_version	Version of the SYSSII file
IREAUN	ru_name	Name of the release unit
IREAUV	ru_version	Version of the release unit e.g. V01.2A00 is represented as C'012A00' followed by the end character setting X'00' or X'40'
IREAUFL	ru_funclev	Function state in which the release unit runs (U=TU, P=TPR, B=Both)
IREAUU	ru_userid	Default user ID
IREAUMU	ru_mandatory_uid	Indicates if the user ID is mandatory (Y=Yes, N=No)
IREADEP	ru_dependence	Number of new dependencies to other release units. For values > 0 the dependencies can be read with the IMOKIA macro (FUNCT=*SIIRKON).
IREAODEP	ru_old_dependence	Number of old dependencies to other release units. For values > 0 the dependencies can be read with the IMOKIA macro (FUNCT=*SIIRDEP).
IREAUC	ru_comment	Number of comment lines . For values > 0 the comment lines can be read with the IMOKIA macro (FUNCT= *SIIRCOM).

- The attributes of the release items are returned for FUNCT=*SIIREAD:

Assembler	C interface	Meaning
IREAFD	sii_fd	Internal file descriptor (may not be changed)
IREAIN	ri_name	Name of the release items
IREAIDUM	ri_dummy	Indicates if there is a dummy release item present (Y=Yes, N=No)
IREAIT	ri_target	Hardware variant (A, R, U or S)
IREAILG	ri_language	Language code for the comment
IREAIC	ri_comment	Number of comment lines. For values > 0 the comment lines can be read with the IMOKIA macro (FUNCT=*SIIRCOM).
IREAIFL	ri_funclev	Function state in which the release item runs (U=TU, P=TPR, B=Both)
IREAIL	ri_logid	Logical name of the release item
IREAILM	ri_logmand	Indicates if the path name is mandatory (Y=Yes, N=No)
IREAILU	ri_logupd	Indicates if the path name can be changed (Y=Yes, N=No)
IREAILF	ri_logfiltyp	Fully or partially qualified file name (F=Full, P=Partial)
IREAITP	ri_type	Installation type
IREAIF	ri-file	Default path name
IREAIFM	ri_mandatory_file	Indicates if the path name is mandatory (Y=Yes, N=No)
IREAIUI	ri_userid	Default user ID
IREAIMU	ri_mandatory_uid	Indicates if the user ID is mandatory (Y=Yes, N=No)
IREASH	ri_user_access	File attribute USER-ACCESS (O=Owner-Only, A=All-Users, S=Special)
IREAAC	ri_access	File attribute ACCESS (R=Read, W=Write)
IREAMI	ri_migrate	File attributes MIGRATE and BACKUP-CLASS I: Migrate=Inhibited and Backup-Class=Any E: Migrate=Inhibited and Backup-Class=E S: default setting of the system
IREAFO	ri_format	Block format of the file (K=Key, 2=NK2, 4=NK4)
IREAIPL	ri_ipl	Indicates if a IPL file is present (Y=Yes, N=No)

IREAPAR	ri_par	Number of lines for installation parameters. For values > 0 the parameter text lines can be read with the IMOKIA macro (FUNCT= *SIIRPAR).
---------	--------	---

Macro return codes

(SC2)	SC1	Maincode	Meaning	SII OPEN	SII READ	SII CLOSE
X'00'	X'00'	X'0000'	Function successfully executed	x	x	x
X'00'	X'01'	X'0001'	SYSSII file name invalid	x		
X'00'	X'01'	X'0007'	Function (Parameter FUNCT) invalid	x	x	x
X'00'	X'03'	X'0003'	Invalid parameter list	x	x	x
X'00'	X'40'	X'0014'	SYSSII file format invalid	x		
X'00'	X'40'	X'0015'	DMS error (see IREADMS field in the parameter list)	x		
X'00'	X'40'	X'0016'	No SYSSII file open		x	x
X'00'	X'40'	X'001E'	EOF reached in SYSSII file		x	
X'00'	X'20'	X'0100'	System error	x	x	x
X'00'	X'20'	X'0101'	Internal error	x	x	x

The return codes which can occur depend on the function of the IMOKIT macro you select.

6.4.2 IMOKIA Read comments, dependencies and parameter text

The information that was read from a SYSSII file with the SIIOPEN or SIIREAD functions of the IMOKIT macro can be evaluated further with the IMOKIA macro. The comments on the release unit or on a release item, the old and new dependencies to release units or the parameter text for a release item can be read from the SYSSII file with the SIIRCOM, SIIRDEP and SIIRPAR functions of the IMOKIA macro.

Macro	Operands
IMOKIA	MF= <u>D</u> / L / C / M / E ,PREFIX= I / <char (1)> ,MACID= <u>ARE</u> / <char (3)> ,PARAM= <adr> / (<reg>) / <name> ,FUNCT= <u>*SIIRCOM</u> / *SIIRDEP / *SIIRKON / *SIIRPAR ,FDESCR= <u>NULL</u> / <var: pointer> ,LENGTH= <u>Q</u> / <integer 1..2147483647> / <var: int: 4> ,AREA= <u>NULL</u> / <var: pointer>

See the "Macro forms" section on "[Macro syntax](#)" for descriptions of the MF, PREFIX, MACID and PARAM parameters. Symbolic addresses may not be specified together with MF=L.

- FUNCT Specifies the desired function.
- =*SIIRCOM Read the comments of a release unit or of a release item.
 - =*SIIRDEP Read the old dependencies of a release unit.
 - =*SIIRKON Read the new dependencies of a release unit.
 - =*SIIRPAR Read the parameter text of a release item.

FDESCR=NULL / <var: pointer>

Address of the file descriptor of the open SYSSII file. This value must be obtained from the parameter list of the corresponding IMOKIT call (FUNCT=*SIIOPEN or *SIIREAD).

LENGTH=Q / <integer 1..2147483647> / <var: int: 4>

Length of the output area in which the information read is to be stored.

AREA=NULL / <var: pointer>

Address of the output area in which the information read is to be

stored.

Parameter list

```
IMOKIA    IMOKIA MF=D
1 IMOKIA    MFTST MF=D,PREFIX=I,MACID=ARE,ALIGN=F,          C
1          DMACID=ARE,SUPPORT=(E,D,C,M,L),DNAME=ARE_PL
2 IMOKIA    DSECT ,
2          *,##### PREFIX=I, MACID=ARE #####
1 IAREPVER      EQU    21          value for parameter list
1 *            version
1 *
1 IAREFCOM      EQU    155         value for SIIRCOM
1 *
1 IAREFDEP      EQU    156         value for SIIRDEP
1 *
1 IAREFPAR      EQU    157         value for SIIRPAR
1 *
1 IAREFKON      EQU    158         value for SIIRKON
1 *
1 *    PARAMETER LIST OF IMON-SIC
1 IARESTANDARD_HEADER FHDR MF=(C,IARE),EQUATES=NO
2 IARESTANDARD_HEADER DS    0A
2 IAREFHE DS    0XL8            0    GENERAL PARAMETER AREA HEADER
2 *
2 IAREIFID DS    0A            0    INTERFACE IDENTIFIER
2 IAREFCTU DS    AL2           0    FUNCTION UNIT NUMBER
2 *
2 *            BIT 15    HEADER FLAG BIT,
2 *            MUST BE RESET UNTIL FURTHER NOTICE
2 *            BIT 14-12 UNUSED, MUST BE RESET
2 *            BIT 11-0  REAL FUNCTION UNIT NUMBER
2 IAREFCT DS    AL1            2    FUNCTION NUMBER
2 IAREFCTV DS    AL1            3    FUNCTION INTERFACE VERSION NUMBER
2 *
2 IARERET DS    0A            4    GENERAL RETURN CODE
2 IARESRET DS    0AL2          4    SUB RETURN CODE
2 IARES2 DS    AL1            4    SUB RETURN CODE 2
2 IARES1 DS    AL1            5    SUB RETURN CODE 1
2 IAREMRET DS    0AL2          6    MAIN RETURN CODE
2 IAREMR2 DS    AL1            6    MAIN RETURN CODE 2
2 IAREMR1 DS    AL1            7    MAIN RETURN CODE 1
2 IAREFHL EQU    8            8    GENERAL OPERAND LIST HEADER LENGTH
2 *
1 *            STD HEADER
1 *    main return codes
1 IAREOK      EQU    0          Success
1 IAREEIPL    EQU    2          Invalid Parameter List
1 *            Version
1 IAREEATO    EQU    3          Memory area too short
1 IAREEIFC    EQU    7          Invalid Function Called
```

1	IAREEFNO	EQU	22	File not open
1	IAREEOF	EQU	30	End of file
1	IAREES	EQU	256	System error
1	IAREEI	EQU	257	Internal error
1	*			
1	IAREFD	DS	A	FILE DESCRIPTOR
1	IAREDEL	DS	X	STRING DELIMITOR
1	IARERO	DS	XL3	filler 0
1	IARELEN	DS	F	LENGTH OF MEMORY AREA
1	IAREAREA	DS	A	POINTER ON MEMORY AREA
1	IARE#	EQU	*-IARESTANDARD_HEADER	

Programming notes

The following parameters can also be passed in the parameter list for calls of functions of the IMOKIA macro:

- The file descriptor in the IAREFD field (see the FDESCR parameter).
The value from the parameter list (IAREFD field) of the corresponding IMOKIT call (SIIOOPEN or SIIREAD function) must be used.
- The length of the output area in the IARELEN field (see the LENGTH parameter).
- The address of the output area in the IAREAREA field (see the AREA parameter).

*Read comments (FUNCT=*SIIRCOM)*

The function SIIRCOM reads sequential comments on a release unit or on a release item.

The SYSSII file must be opened. Each call returns a comment line with a maximum of 73 characters consisting of the language code (D or E), the specified delimiter character and the text of the comment line.

The information returned depends on the last IMOKIA macro call made:

- After a call with FUNCT=*SIIOOPEN the first comment line on the release unit is returned if there is one. The number of comment lines present is stored in the IREAUC field of the IMONKIT parameter list.
- After a call with FUNCT=*SIIREAD the first comment line on the “read” release item is returned if there is one. The number of comment lines present is stored in the IREAIC field of the IMONKIT parameter list.

Additional comment lines can be read by calling the SIIRCOM function again. The return code “EOF reached in SYSSII file” is returned once the last comment line has been read.

*Read old dependencies (FUNCT=*SIIRDEP)*

The function SIIRDEP sequentially reads the old dependencies existing for a release unit.

The SYSSII file must be opened. Each call returns dependency information in the output area. The information consists of two strings in a row that are separated by the delimiter character set earlier. The first string is up to 16 characters long and contains the name of the release unit to which there is a old dependency. The second string contains the version of the release unit.

The number of old dependencies present is stored in the IREAODEP field of the IMONKIT parameter list. The return code “EOF reached in SYSSII file” is returned once the last dependency has been read.

*Read new dependencies (FUNCT=*SIIRKON)*

The function SIIRKON sequentially reads the new dependencies existing for a release unit.

The SYSSII file must be opened. Each call returns dependency information in the output area. The information consists of two strings in a row that are separated by the delimiter character set earlier. The first string is up to 16 characters long and contains the name of the release unit to which there is a new dependency. The second string contains the version of the release unit.

The number of new dependencies present is stored in the IREADEP field of the IMONKIT parameter list. The return code "EOF reached in SYSSII file" is returned once the last dependency has been read.

*Read parameter text (FUNCT=*SIIRPAR)*

The function SIIRPAR sequentially reads parameter text that contains installation parameters for a release item. The SYSSII file must be open and the information on a release item must have been read during the last call of the IMOKIA macro (FUNCT=*SIIREAD). If there is parameter text for this release item, then the first call returns the first line (up to a maximum of 71 characters). The number of parameter text lines present is stored in the IREAPAR field of the IMONKIT parameter list. The return code "EOF reached in SYSSII file" is returned once the last line has been read.

Macro return code

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'03'	X'0002'	Invalid parameter list
X'00'	X'01'	X'0003'	Output area too small (LENGTH parameter)
X'00'	X'01'	X'0007'	Function (FUNCT parameter) invalid
X'00'	X'40'	X'0016'	SYSSII file not open
X'00'	X'40'	X'001E'	EOF reached in SYSSII file
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

6.4.3 IMOKIS Output contents of SYSSII file

The contents of a SYSSII file, i.e. all information on the release unit and the release items belonging to it, can be output with the IMOKIS macro (SIISHOW function). The scope of the output can be limited to the structure information only in the SHINFO parameter. The output can be sent to SYSOUT, SYSLST or a file.

The layout the output corresponds to the layout of the output requested in the menu mode using option 3 (*Show SYSSII*) in the main menu ("[Open, display or check SYSSII file](#) ") or via the SHOW-SII statement ("[SHOW-SII Output contents of SYSSII file](#) ", see example on "[Example with masks from the main menu](#) ").

Macro	Operand	Meaning
IMOKIS	MF=	<u>D</u> / L / C / M / E
	,PREFIX=	I / <char (1)>
	,MACID=	<u>SHQ</u> / <char (3)>
	,PARAM=	<adr> / (<reg>)
	,FUNCT=	<u>*SIISHOW</u>
	,SYSSII=	<u>X'40'</u> / <c-string 1..54> / <var: char:55>
	,SHINFO=	<u>*ALL</u> / *MIN
	,SHOOUT=	<u>*SYSLST</u> / *SYSOUT / *FILE
	,SHOFIL=	<u>X'40'</u> / <c-string 1..54> / <var: char:55>
	,OVERW=	<u>*NO</u> / *YES

See the "Macro forms" section on "[Macro syntax](#) " for descriptions of the MF, PREFIX, MACID and PARAM parameters.

FUNCT Specifies the desired function.

= Output the contents of a SYSSII file.

*SIISHOW Name of the SYSSII file.

SYSSII No specification.

=X'40' Explicit specification of the SYSSII file name.

=<c-string 1..54> Symbolic address containing the name of the SYSSII file. May not

=<var: char:55> be specified together with MF=L.

SHINFO Determines the scope of the information to be output.

=*ALL All information (including the attributes, dependencies, parameter text and comments) on the release unit and the release items belonging to it are output.

=*MIN	Only the structure information for the release unit and the release items belonging to it are output. The output contains the name of the release unit, main version, correction state and function state as well as the corresponding release items with their function states and logical names.
SHOOUT	Specifies the output destination.
=*SYSLST	The information is output to SYSLST.
=*SYSOUT	The information is output to SYSOUT.
=*FILE	The information is output to a file. The name of the file must be specified in the SHOFIL parameter.
SHOFIL	Specifies the name of the output file (only for SHOOUT=*FILE).
=X'40'	No specification.
=<c-string 1..54>	Explicit specification of the file name.
=<var: char:55>	Symbolic address containing the file name. May not be specified together with MF=L.
OVERW	Specifies if an existing output file may be overwritten.
=*NO	An existing file may not be overwritten.
=*YES	An existing file may be overwritten.

Parameter list

```
IMOKIS  IMOKIS MF=D
1 IMOKIS  MFTST MF=D,PREFIX=I,MACID=SHO,ALIGN=F,          C
1          DMACID=SHO,SUPPORT=(E,D,C,M,L),DNAME=SHO_PL
2 IMOKIS  DSECT ,
2          * ,##### PREFIX=I, MACID=SHO #####
1 *      for show output destination
1 ISHOSLST      EQU  0          show on syslst
1 ISHOSOUT      EQU 100        show on sysout
1 ISHOSFIL      EQU 101        show in file
1 *
1 ISHOPVER      EQU  21        value for parameter list
1 *            version
1 *
1 ISHOFSHO      EQU 153        value for SIISHOW
1 *
1 ISHOSIAC      EQU  65        show all
1 *
1 ISHOSIMC      EQU  77        minimum show
1 *
1 ISHOSOWF      EQU  89        overwrite show file
1 *
1 ISHOYES      EQU  1          overwrite show file
1 *
1 ISHONO      EQU  0          overwrite not show file
1 *
1 *      PARAMETER LIST OF IMON-SIC
1 ISHOSTANDARD_HEADER FHDR MF=(C,ISHO),EQUATES=NO
2 ISHOSTANDARD_HEADER DS  0A
2 ISHOFHE DS  0XL8          0  GENERAL PARAMETER AREA HEADER
2 *
2 ISHOIFID DS  0A          0  INTERFACE IDENTIFIER
2 ISHOFCTU DS  AL2        0  FUNCTION UNIT NUMBER
2 *
2 *                      BIT 15  HEADER FLAG BIT,
2 *                      MUST BE RESET UNTIL FURTHER NOTICE
2 *                      BIT 14-12 UNUSED, MUST BE RESET
2 *                      BIT 11-0  REAL FUNCTION UNIT NUMBER
2 ISHOFCT DS  AL1          2  FUNCTION NUMBER
2 ISHOFCTV DS  AL1          3  FUNCTION INTERFACE VERSION NUMBER
2 *
2 ISHORET DS  0A          4  GENERAL RETURN CODE
2 ISHOSRET DS  0AL2        4  SUB RETURN CODE
2 ISHOSR2 DS  AL1          4  SUB RETURN CODE 2
2 ISHOSR1 DS  AL1          5  SUB RETURN CODE 1
2 ISHOMRET DS  0AL2        6  MAIN RETURN CODE
2 ISHOMR2 DS  AL1          6  MAIN RETURN CODE 2
2 ISHOMR1 DS  AL1          7  MAIN RETURN CODE 1
```

```

2 ISHOFHL EQU 8 8 GENERAL OPERAND LIST HEADER LENGTH
2 *
1 * STD HEADER
1 * main return codes
1 ISHOOK EQU 0 Success
1 ISHOESIN EQU 1 Invalid SYSSII
1 ISHOEIPL EQU 2 Invalid Parameter List
1 * Version
1 ISHOEIOS EQU 4 Invalid Output Specification
1 ISHOEISF EQU 5 Invalid Show File Name
1 ISHOEIFC EQU 7 Invalid Function Called
1 ISHOESIF EQU 20 Bad SYSSII format
1 ISHOEDMS EQU 21 DMS error
1 ISHOEDOF EQU 23 DMS error on Output File
1 ISHOEFEX EQU 24 File existing
1 ISHOES EQU 256 System error
1 ISHOEI EQU 257 Internal error
1 *
1 ISHODMS DS H DMS ERROR
1 ISHOSII DS CL55 SYSSII FILE NAME
1 ISHOSIN DS X SHOW INFORMATION
1 ISHOSOT DS X SHOW OUTPUT TYPE
1 ISHOSFI DS CL55 SHOW OUTPUT FILE
1 ISHOOWF DS X FLAG FOR OVERWRITE SHOW FILE
1 ISHOR1 DS XL1 filler 1
1 ISHO# EQU *-ISHOSTANDARD_HEADER

```

Programming notes

The following parameters can also be passed in the parameter list for the SIISHOW function

call:

- The name of the SYSSII file in the ISHOSII field (see the SYSSII parameter).
- The scope of information to be output in the ISHOSIN field (see the SHINFO parameter).
The value C'A' is to be set to obtain the maximum scope of output (SHINFO=*ALL), and the value C'M' is to be set to obtain a limited amount of output (SHINFO=*MIN) .
- The output destination in the ISHOOT field (see the SHOOUT parameter). The following values must be set depending on the output destination:
 - The output to SYSLST corresponds to the value X'00'.
 - The output to SYSOUT corresponds to the value X'64'.
 - The output to a file corresponds to the value X'65'.
- The name of the output file in the ISHOSFI field (see the SHOFIL parameter). This is ignored when the output is sent to SYSLST or SYSOUT.
- The specification if an existing file is to be overwritten in the ISHOOWF field (see the OVERW parameter). The value X'01' allows overwriting and the value X'00' prevents it.

The DMS error code (in the ISHODMS field) is passed in the parameter list for the SIISHOW function in addition to the return code when a DMS error occurs.

Macro return code

(SC2)	SC1	Maincode	Meaning
X'00'	X'00'	X'0000'	Function successfully executed
X'00'	X'01'	X'0001'	SYSSII filename (SYSSII parameter) invalid
X'00'	X'03'	X'0002'	Invalid parameter list
X'00'	X'01'	X'0004'	Output destination invalid (SHOOUT parameter)
X'00'	X'01'	X'0005'	Name of the output file invalid (SHOFIL parameter)
X'00'	X'01'	X'0007'	Function (FUNCT parameter) invalid
X'00'	X'40'	X'0014'	SYSSII file (SYSSII parameter) in the wrong format
X'00'	X'40'	X'0015'	DMS error
X'00'	X'40'	X'0017'	DMS error while writing to the output file (SHOFIL parameter)
X'00'	X'40'	X'0018'	Output file (SHOFIL parameter) already exists
X'00'	X'20'	X'0100'	System error
X'00'	X'20'	X'0101'	Internal error

6.5 The C programming interface of IMON-SIC

The C program interface provides the same functionality as the Assembler interface. For each macro there is a corresponding header files (imokit.h, imokia.h and imokis.h) in the library SYSLIB.IMON-SIC.<version>. The corresponding header file is used when a macro function is called via the C program interface.

Each header file contains:

- the declarations of the C functions required for IMON-SIC
- the declarations of the parameter list and the return information for the C functions
- internal types and definitions of constants.

Unlike the Assembler interface, the C interface requires the user to fill the parameter list before calling the corresponding functions.

If an optional operand is omitted, fill the corresponding field in the parameter list with blanks or null bytes.

The basic C function call is described in the following. The C functions, divided according to the corresponding macro calls, are described after that with their call parameters and parameter lists.

C function calls

The C functions are called according to the following schema:

```
#define <macro>PLVERSION _____ (1)
#define <macro_fct> _____ (2)
void SIIENTR(struct <macro>_pl *); _____ (3)
```

- (1) Set the symbolic constant for the version of the parameter list.
- (2) Set the symbolic constant for the macro function to be called in the parameter list.
- (3) Function call with the corresponding parameter list. Additional input parameters may need to be supplied with data using the parameter list before the function is called. These parameters are listed in the description of the function.

Meaning

- <macro> designates the corresponding macro call that provides the function. Possible values are imokit, imokia and imokis.
- <macro>_pl designates the parameter list defined in the header file for this macro call (e.g. IMOKIT_pl for functions of the IMOKIT macro).
- <macro_fct> designates one of the following macro functions:

<macro_fct>	Function of the macro	Meaning
IMOKIT_OPEN	SIIOPEN IMOKIT	Open SYSSII file
IMOKIT_READ	SIIREAD IMOKIT	Read information from the SYSSII file

IMOKIT_CLOSE	SIICLOSE	IMOKIT	Close SYSSII file
IMOKIA_RCOM	SIIRCOM IMOKIA		Read comments on release unit or release item from the SYSSII file
IMOKIA_RDEP	SIIRDEP IMOKIA		Read old dependencies of the release unit from the SYSSII file
IMOKIA_RKON	SIIRKON IMOKIA		Read new dependencies of the release unit from the SYSSII file
IMOKIA_RPAR	SIIRPAR IMOKIA		Read parameter text of a release item from the SYSSII file
IMOKIS_SHOW	SIISHOW IMOKIS		Display contents of SYSSII file

The function call uses the following standard header:

```

struct std_hdr_t
{
    short unit ;
    char function;
    char version;
    char subcode2;
    char subcode1;
    short maincode;
};

```

Notes

1. Internal data fields (e.g. the file descriptor `sii_fd`) are prepared for additional function calls and may not be deleted or overwritten.
2. When strings are explicitly specified in the parameter list (e.g. a SYSSII file name), then at least one space character (`X'40'`) or `X'00'` must be added at the end of the valid string as the last character of the string.
3. `X'00'` is the preset value for the last character of the string when outputting strings in the parameter list (e.g. unit name) for the IMOKIT and IMOKIA macro functions `SIIOOPEN`, `SIIREAD`, `SIIRCOM`, `SIIRDEP`, `IMOKIA_RKON` and `SIIRPAR`. The last character of the string can be explicitly set to the space character (`X'40'`) in the corresponding parameter lists (`IMOKIT_pl` or `IMOKIA_pl`) in the “delimiter” variable.

6.5.1 IMOKIT Read information from SYSSII file

SIIOPEN function – Open SYSSII file

Function call

```
#define IMOKITPLVERSION      /* value for parameter list version */
#define IMOKIT_OPEN         /* value for SIIOPEN function      */
void SIIENTR(struct IMOKIT_pl *);
```

Programming notes

The name of the SYSSII file must be passed in the parameter list in the `sii_name` variable when calling the function.

The attributes of the release unit are returned in addition to the return code (see ["IMOKIT Read information from SYSSII file "](#)).

SIIREAD function – Read information from the SYSSII file

Function call

```
#define IMOKITPLVERSION      /* value for parameter list version */
#define IMOKIT_READ         /* value for SIIREAD function      */
void SIIENTR(struct IMOKIT_pl *);
```

Programming notes

The attributes of the last release item read are returned in addition to the return code (see ["IMOKIT Read information from SYSSII file "](#)).

SIICLOSE function – Close SYSSII file

Function call

```
#define IMOKITPLVERSION      /* value for parameter list version */
#define IMOKIT_CLOSE        /* value for SIICLOSE function      */
void SIIENTR(struct IMOKIT_pl *);
```

Programming notes

Only the return code is returned.

Parameter list

```
struct IMOKIT_pl {
    /* STD HEADER */
    struct ESMFHDR STANDARD_HEADER;
    void* sii_fd; /* FILE DESCRIPTOR */
    unsigned short dms_error; /* DMS ERROR */
    unsigned char delimiter; /* STRING DELIMITOR */
    char sii_name[55]; /* SYSSII FILE NAME */
    char sii_file_version[7]; /* SYSSII FILE VERSION */
    char etpnd_c_name[9]; /* ETPND COMPONENT NAME */
    char etpnd_c_version[4]; /* ETPND COMPONENT VERSION */
    char etpnd_c_domain[14]; /* ETPND COMPONENT DOMAIN */
    char etpnd_c_pm[20][9]; /* ETPND COMPONENT PM NUMBER */
    char ru_name[31]; /* UNIT NAME */
    char ru_version[8]; /* UNIT VERSION */
    unsigned char ru_funclev; /* UNIT FUNCTIONAL LEVEL */
    char ru_userid[9]; /* UNIT USER ID */
    unsigned char ru_mandatory_uid; /* UNIT MANDATORY USER ID */
    char R1[2]; /* filler 1 */
    unsigned long ru_old_dependence; /* UNIT OLD DEPENDENCE NUMBER */
    unsigned long ru_dependence; /* UNIT DEPENDENCE NUMBER */
    unsigned long ru_comment; /* UNIT COMMENT NUMBER */
    char ri_name[31]; /* ITEM NAME */
    unsigned char ri_dummy; /* DUMMY ITEM FLAG */
    unsigned char ri_target; /* ITEM TARGET */
    char R1a[3]; /* filler 1a */
    unsigned char ri_language; /* ITEM LANGUAGE */
    char R2[3]; /* filler 2 */
    unsigned long ri_comment; /* ITEM COMMENT NUMBER */
    unsigned char ri_funclev; /* ITEM FUNCTIONAL LEVEL */
    char ri_logid[31]; /* ITEM LOGICAL ID */
    unsigned char ri_logmand; /* MANDATORY PATHNAME */
    unsigned char ri_logupd; /* UPDATE PATHNAME */
    unsigned char ri_logfiltyp; /* FILENAME TYPE */
    char ri_type[4]; /* ITEM TYPE */
    char ri_file[39]; /* ITEM FILENAME */
    unsigned char ri_mandatory_file; /* MANDATORY FILENAME */
    char ri_userid[9]; /* ITEM USER ID */
    unsigned char ri_mandatory_uid; /* ITEM MANDATORY USER ID */
    unsigned char ri_user_access; /* USER-ACCESS ATTRIBUTE */
    unsigned char ri_access; /* ACCESS ATTRIBUTE */
    unsigned char ri_migrate; /* MIGRATE ATTRIBUTE */
    unsigned char ri_format; /* FORMAT ATTRIBUTE */
    unsigned char ri_ipl; /* IPL FILE */
    char R3[2]; /* filler 3 */
    unsigned long ri_par; /* INSTALLATION PARAMETER NUMBER */
};
```

6.5.2 IMOKIA Read comments, dependencies and parameter text

SIIRCOM function – Read comments of the release unit or of a release item

Function call

```
#define IMOKIAPLVERSION          /* value for parameter list version */
#define IMOKIA_RCOM              /* value for SIIRCOM function      */
void SIIENTR(struct IMOKIA_pl *);
```

SIIRDEP function – Read old dependencies of a release unit

Function call

```
#define IMOKIAPLVERSION          /* value for parameter list version */
#define IMOKIA_RDEP              /* value for SIIRDEP function      */
void SIIENTR(struct IMOKIA_pl *);
```

SIIRKON function – Read new dependencies of a release unit

Function call

```
#define IMOKIAPLVERSION          /* value for parameter list version */
#define IMOKIA_RKON              /* value for SIIRDEP function      */
void SIIENTR(struct IMOKIA_pl *);
```

SIIRPAR function – Read parameter text of a release item

Function call

```
#define IMOKIAPLVERSION          /* value for parameter list version */
#define IMOKIA_RPAR              /* value for SIIRPAR function      */
void SIIENTR(struct IMOKIA_pl *);
```

Programming notes

The following variables must be passed in the parameter list when calling the SIIRCOM, SIIRDEP and SIIRPAR functions:

- sii_fd The file descriptor must receive the value from the imokit_pl parameter list of the corresponding function call (IMOKIT_OPEN or IMOKIT_READ).
- len Length of the output area.
- output_area Pointer to the output area.

The information read is returned in addition to the return code for all three functions (see "[IMOKIA Read comments, dependencies and parameter text](#)" for more information).

Parameter list

```
struct IMOKIA_pl {
    /* STD HEADER */
    struct ESMFHDR STANDARD_HEADER;
    void* sii_fd; /* FILE DESCRIPTOR */
    unsigned char delimiter; /* STRING DELIMITOR */
    char R0[3]; /* filler 0 */
    unsigned long len; /* LENGTH OF MEMORY AREA */
    void* output_area; /* POINTER ON MEMORY AREA */
};
```

6.5.3 IMOKIS Output contents of SYSSII file

Function call

```
#define IMOKISPLVERSION      /* value for parameter list version */
#define IMOKIS_SHOW        /* value for SIISHOW function      */
void SIIENTR(struct IMOKIS_pl *);
```

Programming notes

The following variables must be passed in the parameter list when calling the function:

sii_name	Name of the SYSSII file.
show_info	Scope of the information output. The value C'A' is to be set to obtain the maximum amount of output, and the value C'M' is to be set to obtain a limited amount of output.
show_output_type	Output destination. The following values are to be set depending on the output destination: <ul style="list-style-type: none">• The output to SYSLST corresponds to the value X'00'.• The output to SYSOUT corresponds to the value X'64'.• The output to a file corresponds to the value X'65'.
show_output_file	Name of the output file. It is ignored when the output is sent to SYSLST or SYSOUT.
overwrite_file	Specifies if an existing file is to be overwritten. The value X'01' allows overwriting and the value X'00' prevents it.

The DMS error code (in the ISHODMS field) is passed in the parameter list (dms_error variable) in addition to the return code when a DMS error occurs.

The layout of the output information matches the layout of the output requested in the menu mode using option 3 (*Show SYSSII*) in the main menu ("[Open, display or check SYSSII file](#) ") or via the SHOW-SII statement ("[SHOW-SII Output contents of SYSSII file](#) ", see the example on "[Example with masks from the main menu](#) ").

Parameter list

```
struct IMOKIS_pl {
    /* STD HEADER */
    struct ESMFHDR STANDARD_HEADER;
    unsigned short dms_error; /* DMS ERROR */
    char sii_name[55]; /* SYSSII FILE NAME */
    unsigned char show_info; /* SHOW INFORMATION */
    unsigned char show_output_type; /* SHOW OUTPUT TYPE */
    char show_output_file[55]; /* SHOW OUTPUT FILE */
    unsigned char overwrite_file; /* FLAG FOR OVERWRITE SHOW */
    /* FILE */
    char R1[1]; /* filler 1 */
};
```

7 Troubleshooting

This chapter describes how IMON alerts the user to errors that occur during the installation and gives information on analyzing the error situation and rectifying the error.

It describes the most important files in the installation, e.g. the various log files and backup files. It then describes the contents and structure of the console log.

It describes the circumstances under which an aborted installation can be continued by means of a restart.

Error handling and restart during the activation procedure are also described.

Installation errors are handled when the POSIX subsystem is started in the case of POSIX satellites installed together with a supply unit.

7.1 The troubleshooting process

IMON generates the <work>.<package>.<date>.IE installation procedure from the user input during the installation. This installation procedure is then started as a background procedure. The procedure executes individual isolated processing steps, the installation steps (e.g. backing up files, RMS processing).

The beginning of each installation step is reported on the console.

The principle sequence of an installation step is illustrated in [figure 80](#).

If an error occurs during an installation step, a message that must be answered is output on the console. The installation procedure is interrupted until the message is answered.

The following are the possible response alternatives:

Answer	Effect	Notes
<tsn>.0	REPEAT: The installation step is repeated.	The installation step should only be repeated if the error was eliminated in the meantime (e.g. cancelation of a file lock) as otherwise the same error situation will arise.
<tsn>.1	CANCEL: The installation is canceled. A restart is possible	The installation should be aborted if the error cannot be eliminated immediately. Once the error is eliminated, the installation can be restarted at the installation step that was interrupted (see " Restarting an aborted installation "). Note that the installation steps that have already been carried out are not reconstructed, i.e. the initial status is not recreated for files that were changed.
<tsn>.2	IGNORE: The error is ignored and must be rectified manually. Processing is continued at the next installation step.	The error should only be ignored in exceptional cases as during subsequent processing it will be assumed that the "defective" installation step was terminated without an error. This assumption can cause other errors under some circumstances. A restart is not possible.

The cause of the error can be determined using the <work>.<package>.<date>.IL installation log. An overview of the course of the installation can be found in the <work>.<package>.IC console log (for information on evaluating the console log, see the section "[Contents and structure of the console log](#)").

More detailed information on the error situation and possible methods of rectifying it can be obtained using the HELP-MSG-INFORMATION command.

Principle sequence of an installation step with error handling

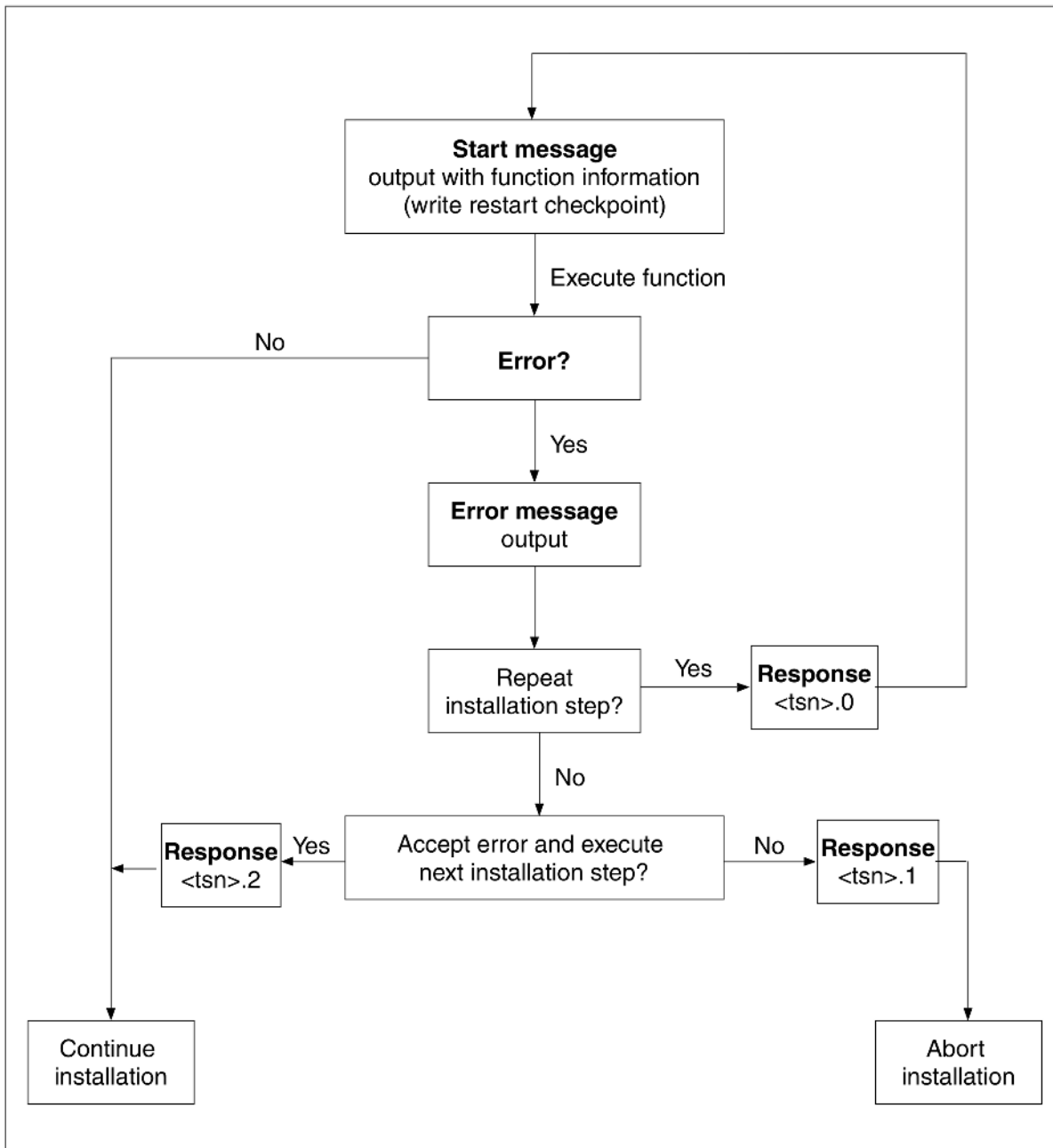


Figure 80: Principle sequence of an installation step with error handling

Important files in the installation

In addition to the installation procedure, during installation logging files that classify the generation parameters and the sequence of the installation are created. The following table shows the most important files:

File	Filename
Installation procedure	<work>.<package>.<date>.IE

Console log Contains all of the installation procedure console messages	<work>.<package>.IC
Installation log SYSLSST log of the installation procedure	<work>.<package>.<date>.IL
Registration log Logs the registration in the SCI	<work>.<package>.<time-stamp>.II
RMS log Information on the RMS processing	<work>.<package>.<date>.IR
Installation actions	<work>.<package>.<date>.IA
Installation parameters Parameters that were specified for the installation	<work>.<package>.<date>.IP
File provision	<work>.<package>.<date>.CD
Log of changes to file attributes Warning if the current file attributes of an installed file differ from the default file attributes	<work>.<package>.<date>.IW
Restart file Information for restarting the installation procedure	<work>.<package>.<date>.RS
Backup of the current SCI A backup copy of both SCI files is generated automatically during the installation.	<work>.<package>.<date>.SCI <work>.<package>.<date>.SCI.GPN
Backup of the DSSM catalog	SOLSAV.<date>
Procedure for deleting parked files	<park>.IMONDEI.PRK.<su>. <package>
Backup of the original library Comparative basis used in the execution of the Undo function	:<target-catid>:\$SYSSAG.IMON. UNDO.<su>

Explanation of the name components

<work>

Work file ID and a prescribed prefix (if one has been assigned)

(e.g. \$SYSSAG. or \$SYSSAG.W1.)

<package>

Package name

<dssm>

Name of the DSSM catalog

<park>

Park ID (e.g. \$PARKSW1.)

<su>

Name of a supply unit

<date>

Time stamp of the installation in the form <month><day><time><year>, where

<month>

is the first three letters of the month name

<day>

a one-letter or two-letter day of the month

<time>

time at which the installation starts in the form hhmmss

<year>

four-digit year

7.2 Contents and structure of the console log

The console log protocols all messages that are output to the console during the installation procedure together with the date and time of their occurrence. If the message is one that requires a response, the response is also logged.

The file can be read at any time during the course of the installation process using any editor. If messages are waiting for a response on the console at that time, this is indicated by a “?” in the response field.

Note

The console log should not be opened using the SHOW-FILE command during the installation process as then no messages can be updated or inserted.

An error message (error when accessing the logging file) is output to the console if the file is opened nonetheless).

You are then allowed to eliminate the error: The file opened using SHOW-FILE is closed using the END statement or [K1].

The file can then be accessed once more and the message displayed on the console can be responded to with “<tsn>”.

Explanation of the output format

The following example shows a console log as generated during a default installation (displayed on screen using SHOW-FILE). The logged information is prepared in tabular format.

The information columns have the following meaning:

Column	Meaning						
Time	Time at which the message was output in the form hh:mm:ss						
Date	Date on which the message was output in the form yy-mm-dd						
R	Response that was entered on the console for this message. Possible values: <table border="1" data-bbox="289 1339 1044 1520"><tr><td><i></td><td>The message was answered with <tsn>.<i> (i = 0 / 1 / 2)</td></tr><tr><td>.</td><td>The message was answered with <tsn>.</td></tr><tr><td>?</td><td>The message was not answered</td></tr></table>	<i>	The message was answered with <tsn>.<i> (i = 0 / 1 / 2)	.	The message was answered with <tsn>.	?	The message was not answered
<i>	The message was answered with <tsn>.<i> (i = 0 / 1 / 2)						
.	The message was answered with <tsn>.						
?	The message was not answered						
MSG####	7 digit message number						
E	If an error occurs, this positions contains a call character for the purpose of information.						
Info	Brief item of information. If an error occurs, this item of information starts with the number of the installation step.						

Example

```
/show-file $sysag.<package-name>.IC
Information columns:
| Date          | Time          | MSG###      | E      | Info
|              |              |            |       |
| 21-04-10    | 15:27:19    | IMO2019    |        | INSTALLATION FUNCTION EXECUTED FOR PACKAGE '<packag
| 21-04-10    | 15:27:22    | IMO2001    |        | FSTAT-RENAME : analyse of target system
| 21-04-10    | 15:27:43    | IMO2002    |        | SAVE-OLD-FILES procedure generation
| 21-04-10    | 15:27:44    | IMO2003    |        | SAVE-OLD-FILES procedure call
| 21-04-10    | 15:28:03    | IMO2004    |        | RESERVE-FILE : preparation of target system
| 21-04-10    | 15:28:07    | IMO2005    |        | IMPORT-FILE : import-procedure generation
| 21-04-10    | 15:28:08    | IMO2024    |        | IMPORT-FILE : import-procedure call
| 21-04-10    | 15:29:32    | IMO2006    |        | UPDATE-CATALOG-ENTRY : set file attributes
| 21-04-10    | 15:29:45    | IMO2007    |        | Library processing
| 21-04-10    | 15:29:48    | IMO2008    |        | ADD-INSTALLATION-UNITS : register installation unit
| 21-04-10    | 15:30:01    | IMO2015    |        | Reference file generation
| 21-04-10    | 15:30:12    | IMO2011    |        | RMS-PROCESSING: Revision packet put into RMS Depot
| 21-04-10    | 15:30:21    | IMO2016    |        | Processing of delivered SYSSII files
| 21-04-10    | 15:30:27    | IMO4009    |        | SUPPLY UNIT 'EDT' '17.0' 'D30' PROCESSED
| 21-04-10    | 15:30:27    | IMO2018    |        | ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'EDT
| 21-04-10    | 15:30:27    | IMO2009    |        | RMS-PROCESSING: Revision packet put into RMS depot
| 21-04-10    | 15:30:29    | IMO4010    |        | ACTIVATE-UNIT 'EDT' '17.0' 'D30'
| 21-04-10    | 15:30:32    | IMO4009    |        | SUPPLY UNIT 'OPENFT' '12.1' 'C10' PROCESSED
| 21-04-10    | 15:30:32    | IMO2018    |        | ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'OPE
| 21-04-10    | 15:30:32    | IMO2009    |        | RMS-PROCESSING: Revision packet put into RMS depot
| 21-04-10    | 15:30:34    | IMO4010    |        | ACTIVATE-UNIT 'OPENFT' '12.1' 'C10'
| 21-04-10    | 15:30:40    | IMO4009    |        | SUPPLY UNIT 'OPENFT-CR' '12.1' 'B10' PROCESSED
| 21-04-10    | 15:30:40    | IMO4009    |        | SUPPLY UNIT 'PERCON' '02.9' 'C00' PROCESSED
+
                                     S*SOF+          1(      1)
```

```
21-04-10 15:30:40 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'PER
21-04-10 15:30:40 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:43 IMO4010 ACTIVATE-UNIT 'PERCON' '02.9' 'C00'
21-04-10 15:30:46 IMO4009 SUPPLY UNIT 'SORT' '08.0' 'A30' PROCESSED
21-04-10 15:30:46 IMO2018 ACTIVATION PROCESS INITIATED FOR RELEASE UNIT: 'SOR
21-04-10 15:30:46 IMO2009 RMS-PROCESSING: Revision packet put into RMS depot
21-04-10 15:30:50 IMO4010 ACTIVATE-UNIT 'SORT' '08.0' 'A30'
21-04-10 15:30:53 IMO2010 RMS-PROCESSING: loader generation
21-04-10 15:31:05 IMO2027 SSCM global processing
21-04-10 15:31:10 IMO2021 INSTALLATION NORMALLY TERMINATED FOR PACKAGE '<pack
```

```
% SHO0301 WARNING: END OF FILE REACHED
```

```
e
```

```
S*SOF+
```

```
24( 1)
```

7.3 Restarting an aborted installation

The relevant number of the current installation step is stored in the restart file during the installation procedure. This means that if the procedure is aborted, the restart file contains the number of the installation step that was not fully executed. When the aborted procedure is restarted, the procedure run starts with the last installation step that was not concluded.

An installation that was not concluded correctly can be repeated in the following two cases:

1. If the installation was aborted with the response <tsn>.1 when an error occurred.
2. If the installation procedure was not checked before it was terminated (e.g. using CANCEL-JOB or SHUTDOWN).

The restart is activated by restarting the installation procedure with ENTER-PROCEDURE.

The message `SOL0012 EXECUTE RESTART FUNCTION ?` is output on the console.

This message must be answered with <tsn>.1 or <tsn>.2.

- Answer <tsn>.1
The installation starts at the interrupted installation step.
- Answer <tsn>.2
The entire installation is repeated.

Notes

- You must ensure that the logging files (IL, IR, and IC suffix) are closed before the installation is restarted.
- The installation can, in principle, be restarted at any installation step if the installation step number is entered in the restart file (RS suffix).
- Changes to the installation procedure only become active if the changes were made to the file under the original name.

7.4 Error handling and activation restart

At activation IMON generates the activation procedure

`$SYSSAG.<prefix>.<time-stamp>.DA` from user input. This procedure is then started automatically or manually as a background procedure. It performs individual, self-contained processing steps known as activation steps (see "[Structure of the activation procedure](#)"). The step currently being processed is logged to the file `$SYSSAG.<prefix>.<time-stamp>.RS`.

If an error occurs during an activation step, a message that must be answered is output on the console. The installation procedure is interrupted until the message is answered. The following are the possible response alternatives:

Answer	Effect and Notes
<code><tsn>.0</code>	<p>(=REPEAT)</p> <p>The activation procedure is continued and the aborted activation step is repeated. The installation step should only be repeated if the error was eliminated in the meantime (e.g. cancelation of a file lock) as otherwise the same error situation will arise.</p>
<code><tsn>.1</code>	<p>(=CANCEL)</p> <p>The activation procedure is canceled.</p> <p>The iactivation should be aborted if the error cannot be eliminated immediately. Once the error is eliminated, the activation can be restarted at the activation step that was interrupted. The user is prompted in a console message to select one of the following start options.</p> <ul style="list-style-type: none">• Restart an aborted processing step This start option is available only if the restart file (suffix RS) is still present. If it is not, a new start is performed automatically.• Restart from the first processing step Note that the activation steps that have already been carried out are not reconstructed, i.e. the initial status is not recreated for files that were changed.
<code><tsn>.2</code>	<p>(=IGNORE)</p> <p>The error is ignored and must be rectified manually. Processing is continued at the next activation step.</p> <p>The error should only be ignored in exceptional cases as during subsequent processing it will be assumed that the "defective" activation step was terminated without an error.</p> <p>This assumption can cause other errors under some circumstances (e.g. a subsystem that has not been stopped cannot be stopped in the following processing step and a further error occurs). A restart is not possible.</p>

The execution of an activation step with error handling is basically the same as the execution of an installation step as illustrated in [figure 80 \(The troubleshooting process\)](#).

The cause of the error is given in the execution log (SYSLST output of the batch job) `$SYSSAG.<prefix>.<time-stamp>.AL`. The error can be eliminated either in the system (e.g. by releasing a file lock) or directly by editing the generated activation procedure (e.g. by correcting the command that was generated with an incorrect operand). Corrections to the activation procedure do not take effect until the procedure is called again.

Important activation files

Not only the activation procedure but also log files are created during activation; these provide information on the activation process. The most important files are listed in the following table.

File	Filename
Activation procedure	<code>\$SYSSAG.<prefix>.<time-stamp>.DA</code>
Log of DSSM commands	<code>\$SYSSAG.<prefix>.<time-stamp>.RP</code>
Logs of the activation procedure started as a background procedure: <ul style="list-style-type: none"> • SYSLST log • SYSOUT log • console log 	<code>\$SYSSAG.<prefix>.<time-stamp>.AL</code> <code>\$SYSSAG.<prefix>.<time-stamp>.AO</code> <code>\$SYSSAG.<prefix>.<time-stamp>.AC</code>
Restart files: Information for restarting the activation procedure (RS file contains the number of the activation step last processed) Files are deleted if the activation procedure is completed successfully	<code>\$SYSSAG.<prefix>.<time-stamp>.RS</code> <code>\$SYSSAG.<prefix>.<time-stamp>.RE</code> <code>\$SYSSAG.<prefix>.<time-stamp>.TS</code>
Temporary POSIX processing files: Add and Remove actions are logged for the POSIX system	<code>\$\$SYSROOT.IMON.ACTION.ADD</code> <code>\$\$SYSROOT.IMON.ACTION.REM</code>
Temporary activation lock file: Used to serialize the <code>//ACTIVATE-UNITS</code> jobs with the package installation of POSIX products at POSIX startup.	<code>\$\$SYSROOT.IMON.ACTIVATE-UNIT.SYSLST99</code>

Explanation of the name components

<prefix> Prefix; default prefix is IMONACU

<time-stamp> Time stamp of the activation in the form <monat><tag><zeit><jahr>, mit

-
- <month> is the first three letters of the month name
 - <day> a one-letter or two-letter day of the month
 - <time> time at which the activation starts in the form hhmmss
 - <year> four-digit year

Restarting an aborted activation

The relevant number of the current activation step is stored in the restart file during the activation procedure. This means that if the procedure is aborted, the restart file contains the number of the activation step that was not fully executed. When the aborted procedure is restarted, the procedure run starts with the last activation step that was not concluded.

An activation that was not concluded correctly can be repeated in the following two cases:

1. If the activation was aborted with the response <tsn>.1 when an error occurred.
2. If the activation procedure was not checked before it was terminated (e.g. using CANCEL-JOB or SHUTDOWN).

The restart is activated by restarting the activation procedure with ENTER-PROCEDURE.

The message SOL0013 EXECUTE RESTART FUNCTION ? is output on the console.

This message must be answered with <tsn>.1 or <tsn>.2.

- Answer <tsn>.1
The activation starts at the interrupted installation step.
- Answer <tsn>.2
The entire activation is repeated.

Notes

- You must ensure that the logging files (AL, AO, and AC suffix) are closed before the activation is restarted.
- Changes to the activation procedure only become active if the changes were made to the file under the original name.

7.5 Troubleshooting at package installation of POSIX products (item types *PS, *NP)

Installation during POSIX startup:

Problems that occurred during the package installation of POSIX product by IMON during the POSIX startup are logged in the following file:

`$SYSROOT.IMON.POSIX-STARTUP.SYSOUT`

If an error is reported by the installation of POSIX products, the original `$SYSROOT.IMON.ACTIONS.ADD` and `*SYSROOT.IMON.ACTIONS.REM` files are saved to `$SYSROOT.IMON.ACTIONS.ADD.<timestamp>.SAV` and `$SYSROOT.IMON.ACTIONS.REM.<timestamp>.SAV`.

Installation during dynamic activation with //ACTIVATE-UNITS:

Problems that occurred during the package installation of POSIX product by IMON during the dynamic activation are logged in the following files:

`$SYSSAG.IMONACU.<timestamp>.A0`

`$SYSSAG.IMONACU.<timestamp>.AL`

If an error is reported by the installation of POSIX products, the original `$SYSROOT.IMON.ACTIONS.ADD` and `*SYSROOT.IMON.ACTIONS.REM` files are saved to `$SYSROOT.IMON.ACTIONS.ADD.<timestamp>.SAV` and `$SYSROOT.IMON.ACTIONS.REM.<timestamp>.SAV`.

Miscellaneous errors:

POSIX problems occurring during the generation of the POSIX configuration file are logged in the following file:

`$SYSROOT.IMON.POSIX-CONFUPD.SYSOUT`

8 Installing IMON

This chapter describes the hardware and software requirements for IMON, how IMON is supplied, and how you install it.

Hardware requirements

IMON V3.4 can be run on all servers which support BS2000 V21.0.

You need 9750, 9755, 9763, 8160, 3270 or equivalent data display terminals to output the FHS masks.

FHS masks are output in IMON

- on the menu interface of IMON-BAS and IMON-SIC
- for the SDF statements of IMON-BAS (dialog masks for requesting further input).

Software requirements

- BS2000 V21.0 or higher

Optionally required:

- SSCM (for dynamic activation)
- ARCHIVE (for MAREN free tape assignment during backup)
- EDT (for UNDO function and deinstallation)

IMON release units

The product IMON is made up of the three release units.

IMON-BAS, IMON-GPN, and IMON-SIC.

The IMON product is part of the basic configuration of BS2000.

Installation rules for IMON

The following installation IDs are possible for the IMON release units:

- IMON-GPN must be installed under the TSOS user ID.
- IMON-BAS can be installed under any user ID, the TSOS user ID is required for the release items SYSDAT.IMON-BAS.034.IDF and SYSPRC.IMON-BAS.034 only.
- IMON-SIC can be installed under any user ID.

Providing POSIX-IMON functions

The procedure file SYSPRC.IMON-BAS.034 must be made available under the name

`$TSOS.SYSPRC.IMON-BAS` with `USER-ACCESS=*ALL-USER`.

To ensure that the POSIX-IMON functions are available in further installation processes, IMON must be registered in the POSIX system as follows.

- Create an "in-file" with the following lines (the delivery includes a prepared in-file with the name `SYSDAT.IMON-BAS.034.POSIX`):

```
[PackageInstallation] IMON-BAS;Y;034;;
```

- Register IMON with the following command:
`/START-POSIX-INSTALLATION *FILE(FILE=<in-file>)`
- Stop and restart the POSIX subsystem

Installation variants for IMON

- Initial installation of IMON
 IMON-BAS, IMON-GPN and IMON-SIC are part of the scope of supply for the basic configuration and are installed along with BS2000.
- Correction installation of IMON
 IMON already available in the system; the IMON and IMON-GPN subsystems are started.
 All correction or delta release units of IMON can be installed using IMON and SOLIS2 as described in this manual.
- IMON-GPN V3.4A is required to run IMON V3.4A. IMON-GPN cannot be exchanged while BS2000 is running. After updating IMON-GPN, the BS2000 system must be reloaded before additional components are installed.

Supply components of the release unit IMON-BAS V3.4

File	Contents
SINLIB.IMON-BAS.034	Library for POSIX functions
SYSDAT.IMON-BAS.034.IDF	Exception file for specific functions (GENERATE-IDF)
SYSDAT.IMON-BAS.034.POSIX	In-file for registration in the POSIX system
SYSFHS.IMON-SIC.034.D	Library for IMON-BAS masks (German)
SYSFHS.IMON-SIC.034.E	Library for IMON-BAS masks (English)
SYSLIB.IMON-BAS.034	Library with the user macros for the various SHOW functions
SYSLNK.IMON-BAS.034.TU	Load library for non-privileged IMON-BAS programs
SYSLNK.IMON-BAS.034.TPR	Load library for privileged parts of IMON-BAS (for the /390 type)
SKMLNK.IMON-BAS.034.TPR	Load library IMON-BAS (for the x86 type)
SYSMES.IMON-BAS.034	Complete message file
SYSNRF.IMON-BAS.034	Non-referenced symbol file
SYSPAR.IMON-BAS.034	Sample of an IMON parameter file
SYSPRC.IMON-BAS.034	Procedure file for provision of POSIX processing
SYSREP.IMON-BAS.034	File for system corrections (dummy item)
SYSRMS.IMON-BAS.034	RMS supply set
SYSSDF.IMON-BAS.034	Syntax file

SYSSSC.IMON-BAS.034	Subsystem declarations
SYSSII.IMON-BAS.034	IMON installation information

Supply components of the release unit IMON-GPN V3.4

File	Contents
SYSLIB.IMON-GPN.034	Library containing the user macros (program interface)
SYSLNK.IMON-GPN.034	Load library IMON-GPN (for the /390 type)
SKMLNK.IMON-GPN.034	Load library IMON-GPN (for the x86 type)
SYSMES.IMON-GPN.034	Complete message file
SYSRMS.IMON-GPN.034	RMS supply set
SYSSDF.IMON-GPN.034	Syntax file
SYSSII.IMON-GPN.034	IMON installation information
SYSSSC.IMON-GPN.034	Subsystem declarations

Supply components of the release unit for IMON-SIC V3.4

File	Contents
SYSFHS.IMON-SIC.034.D	Library for IMON-SIC masks (German)
SYSFHS.IMON-SIC.034.E	Library for IMON-SIC masks (English)
SYSLIB.IMON-SIC.034	Library containing the user macros (program interface)
SYSLNK.IMON-SIC.034	Load library for IMON-SIC
SYSMES.IMON-SIC.034	Complete message file
SYSNRF.IMON-SIC.034	Non-referenced symbol file
SYSREP.IMON-GPN.034	File for system corrections (dummy item)
SYSRMS.IMON-SIC.034	RMS supply set
SYSSDF.IMON-SIC.034	Syntax file
SYSSII.IMON-SIC.034	IMON installation information

IMON subsystems

IMON's tasks are carried out by three subsystems:

-
- IMON-GPN subsystem (is started at STARTUP) for IMON-GPN's tasks
 - IMON (TPR) subsystem for IMON-BAS' tasks (is started when IMON-BAS is called using the START-IMON command)
 - IMON-ACT (TU) subsystem for IMON-BAS' tasks (is started during the course of an installation procedure)

9 Appendix

The Appendix contains the following tables and overviews:

- the syntax description for SDF statements and SDF commands
- the conventions for command return codes
- the metasyntax for macro calls
- the sequence plans of the installation sequences (for the examples in *Installation sequences under BS2000*, which were created in BS2000)

9.1 SDF syntax representation

The following example shows the representation of the syntax of a command in a manual.

The command format consists of a field with the command name. All operands with their legal values are then listed. Operand values which introduce structures and the operands dependent on these operands are listed separately.

HELP-SDF Alias: HPSD

```
GUIDANCE-MODE = *NO / *YES
,SDF-COMMANDS = *NO / *YES
,ABBREVIATION-RULES = *NO / *YES
,GUIDED-DIALOG = *YES (...)
  *YES(...)
    | SCREEN-STEPS = *NO / *YES
    | ,SPECIAL-FUNCTIONS = *NO / *YES
    | ,FUNCTION-KEYS = *NO / *YES
    | ,NEXT-FIELD = *NO / *YES
,UNGUIDED-DIALOG = *YES (...) / *NO
  *YES(...)
    | SPECIAL-FUNCTIONS = *NO / *YES
    | ,FUNCTION-KEYS = *NO / *YES
```

This syntax description is valid for SDF V4.6A. The syntax of the SDF command/statement language is explained in the following three tables.

Table 1: Notational conventions

The meanings of the special characters and the notation used to describe command and statement formats are explained in Table 1.

Table 2: Data types

Variable operand values are represented in SDF by data types. Each data type represents a specific set of values. The number of data types is limited to those described in Table 2.

The description of the data types is valid for the entire set of commands/statements.

Therefore only deviations (if any) from the attributes described here are explained in the relevant operand descriptions.

Table 3: Suffixes for data types

Data type suffixes define additional rules for data type input. They contain a length or interval specification and can be used to limit the set of values (suffix begins with *without*), extend it (suffix begins with *with*), or declare a particular task mandatory (suffix begins with *mandatory*). The following short forms are used in this manual for data type suffixes:

cat-id	cat
completion	compl
correction-state	corr
generation	gen
lower-case	low
manual-release	man
odd-possible	odd
path-completion	path-compl
separators	sep
temporary-file	temp-file
underscore	under
user-id	user
version	vers
wildcard-constr	wild-constr
wildcards	wild

The description of the 'integer' data type in Table 3 contains a number of items in italics; the italics are not part of the syntax and are only used to make the table easier to read.

For special data types that are checked by the implementation, Table 3 contains suffixes printed in italics (see the *special* suffix) which are not part of the syntax.

The description of the data type suffixes is valid for the entire set of commands/statements. Therefore only deviations (if any) from the attributes described here are explained in the relevant operand descriptions.

Metasyntax

Representation	Meaning	Examples
UPPERCASE LETTERS		
	Uppercase letters denote keywords (command, statement or operand names, keyword values) and constant operand values. Keyword values begin with *.	
		HELP-SDF SCREEN-STEPS = *NO
UPPERCASE LETTERS in boldface=		
	Uppercase letters printed in boldface denote guaranteed or suggested abbreviations of keywords.	
		GUIDANCE-MODE = *YES

=	The equals sign connects an operand name with the associated operand values.
	GUIDANCE-MODE = *NO
< >	Angle brackets denote variables whose range of values is described by data types and suffixes (see Tables 2 and 3).
	SYNTAX-FILE = <filename 1..54>
<u>Underscoring</u>	
	Underscoring denotes the default value of an operand.
	GUIDANCE-MODE = *NO
/	A slash serves to separate alternative operand values.
	NEXT-FIELD = *NO / *YES
(...)	Parentheses denote operand values that initiate a structure.
	,UNGUIDED-DIALOG = *YES (...) / *NO
[]	Square brackets denote operand values which introduce a structure and are optional. The subsequent structure can be specified without the initiating operand value.
	SELECT = [*BY-ATTRIBUTES](...)
<u>Indentation</u>	
	Indentation indicates that the operand is dependent on a higherranking operand.
	,GUIDED-DIALOG = *YES (...) *YES(...) SCREEN-STEPS = *NO / *YES
	A vertical bar identifies related operands within a structure. Its length marks the beginning and end of a structure. A structure may contain further structures. The number of vertical bars preceding an operand corresponds to the depth of the structure.

		SUPPORT = *TAPE(...) *TAPE(...) VOLUME = *ANY(...) *ANY(...) ...
,		
	A comma precedes further operands at the same structure level.	
		GUIDANCE-MODE = *NO / *YES ,SDF-COMMANDS = *NO / *YES
list-poss(n):		
	The entry "list-poss" signifies that a list of operand values can be given at this point. If (n) is present, it means that the list must not have more than n elements. A list of more than one element must be enclosed in parentheses.	
		list-poss: *SAM / *ISAM list-poss(40): <structured-name 1..30> list-poss(256): *OMF / *SYSLST(...)/<filename1..54>
Alias:		
	The name that follows represents a guaranteed alias (abbreviation) for the command or statement name.	
		HELP-SDF Alias: HPSDF

Data types

Data type	Character set	Special rules
alphanum-name	A...Z 0...9 \$, #, @	
cat-id	A...Z 0...9	Not more than 4 characters; must not begin with the string PUB
command-rest	freely selectable	

composed-name	A...Z 0...9 \$, #, @ hyphen period catalog ID	Alphanumeric string that can be split into multiple substrings by means of a period or hyphen. If a file name can also be specified, the string may begin with a catalog ID in the form :cat: (see data type filename).
c-string	EBCDIC character	Must be enclosed within single quotes; the letter C may be prefixed; any single quotes occurring within the string must be entered twice.
date	0...9 Structure identifier: hyphen	Input format: yyyy-mm-dd jjj: year; optionally 2 or 4 digits mm: month tt: day
device	A...Z 0...9 hyphen	Character string, max. 8 characters in length, corresponding to a device available in the system. In guided dialog, SDF displays the valid operand values. For notes on possible devices, see the relevant operand description.
fixed	+, - 0...9 period	Input format: [sign][digits].[digits] [sign]: + oder - [digits]: 0...9 must contain at least one digit, but may contain up to 10 characters (0...9, period) apart from the sign.
filename	A...Z 0...9 \$, #, @ hyphen period	Input format: $[:cat:][\$user.] \left\{ \begin{array}{l} \text{file} \\ \text{file(no)} \\ \text{group} \\ \text{group} \left\{ \begin{array}{l} (*abs) \\ (+rel) \\ (-rel) \end{array} \right\} \end{array} \right\}$:cat: optional entry of the catalog identifier; character set limited to A...Z and 0...9; maximum of 4 characters; must be enclosed in colons; default value is the catalog identifier assigned to the user ID, as specified in the user catalog.

\$user.

optional entry of the user ID;
character set is A...Z, 0...9, \$, #, @;
maximum of 8 characters; first character cannot be a digit; \$ and period
are mandatory;
default value is the user's own ID.

\$. (special case)

system default ID

file

file or job variable name;
may be split into a number of partial names using a period as a delimiter:
name₁[.name₂[...]]
name₁ does not contain a period and must not begin or end with a hyphen;
file can have a maximum length of 41 characters; it must not begin with a
\$ and must include at least one character from the range A...Z.

#file (special case)

@file (special case)

or @ used as the first character indicates temporary files or job
variables, depending on system generation.

file(no)

tape file name
no: version number;
character set is A...Z, 0...9, \$, #, @.
Parentheses must be specified.

group

name of a file generation group
(character set: as for "file")

group { (*abs)
 (+rel)
 (-rel) }

(*abs)

absolute generation number (1-9999);
* and parentheses must be specified.

		(+rel) (-rel) relative generation number (0-99); sign and parentheses must be specified.
integer	0...9, +, -	+ or -, if specified, must be the first character.
name	A...Z 0...9 \$, #, @	Must not begin with 0...9.
partial-filename	A...Z 0...9 \$, #, @ hyphen period	Input format: [:cat:][user.][partname.] :cat: see filename \$user. see filename partname optional entry of the initial part of a name common to a number of files or file generation groups in the form: name ₁ .[name ₂ .[...]] name (see filename). The final character of "partname" must be a period. At least one of the parts :cat:, \$user. or partname must be specified.
posix-filename	A...Z 0...9 special characters	String with a length of up to 255 characters; consists of either one or two periods or of alphanumeric characters and special characters. The special characters must be escaped with a preceding \ (backslash); the / is not allowed. Must be enclosed within single quotes if alternative data types are permitted, separators are used, or the first character is a ?, ! or ^. A distinction is made between uppercase and lowercase.
posix-pathname	A...Z 0...9 special characters structure identifier: slash	Input format: [/]part ₁ /.../part _n where part is a posix-filename; i max. 1023 characters; must be enclosed within single quotes if alternative data types are permitted, separators are used, or the first character is a ?, ! or ^.

<p>product-version</p>	<p>A...Z 0...9 period single quote</p>	<p>Input format:</p> $[[C]][V][m]m.naso[']$ <p style="margin-left: 150px;"> correction status release status</p> <p>where m, n, s and o are all digits and a is a letter.</p> <p>Whether the release and/or correction status may/must be specified depends on the suffixes to the data type (see suffixes without-corr, without-man, mandatory-man and mandatorycorr in Table 3).</p> <p>product-version may be enclosed within single quotes (possibly with a preceding C).</p> <p>The specification of the version may begin with the letter V.</p>
<p>structured-name</p>	<p>A...Z 0...9 \$, #, @ hyphen</p>	<p>Alphanumeric string which may comprise a number of substrings separated by a hyphen.</p> <p>First character: A...Z or \$, #, @</p>
<p>text</p>	<p>freely selectable</p>	<p>For the input format, see the relevant operand descriptions.</p>
<p>time</p>	<p>0...9 structure identifier: colon</p>	<p>Time-of-day entry:</p> <p>Input format: $\left. \begin{array}{l} hh:mm:ss \\ hh:mm \\ hh \end{array} \right\}$</p> <p>hh: hours mm: minutes ss: seconds } Leading zeros may be omitted</p>
<p>vsn</p>	<p>1. A...Z 0...9 2. A...Z 0...9 \$, #, @</p>	<p>a) Input format: pvsid.sequence-no</p> <p>max. 6 characters</p> <p>pvsid: 2-4 characters; PUB must not be entered</p> <p>sequence-no: 1-3 characters</p> <p>b) Max. 6 characters;</p> <p>PUB may be prefixed, but must not be followed by \$, #, @.</p>

x-string	Hexadecimal: 00...FF	Must be enclosed in single quotes; must be prefixed by the letter X. There may be an odd number of characters.
x-text	Hexadecimal: 00...FF	Must not be enclosed in single quotes; the letter X must not be prefixed. There may be an odd number of characters.

Suffixes for data types

Suffix	Meaning										
x..y <i>unit</i>	<p>With data type "integer": interval specification</p> <p>x minimum value permitted for "integer". x is an (optionally signed) integer.</p> <p>y maximum value permitted for "integer". y is an (optionally signed) integer.</p> <p><i>unit</i> with "integer" only: additional units. The following units may be specified:</p> <table style="width: 100%; border: none;"> <tr> <td style="padding-left: 2em;"><i>days</i></td> <td style="padding-left: 2em;"><i>byte</i></td> </tr> <tr> <td style="padding-left: 2em;"><i>hours</i></td> <td style="padding-left: 2em;"><i>2Kbyte</i></td> </tr> <tr> <td style="padding-left: 2em;"><i>minutes</i></td> <td style="padding-left: 2em;"><i>4Kbyte</i></td> </tr> <tr> <td style="padding-left: 2em;"><i>seconds</i></td> <td style="padding-left: 2em;"><i>Mbyte</i></td> </tr> <tr> <td style="padding-left: 2em;">milliseconds</td> <td></td> </tr> </table>	<i>days</i>	<i>byte</i>	<i>hours</i>	<i>2Kbyte</i>	<i>minutes</i>	<i>4Kbyte</i>	<i>seconds</i>	<i>Mbyte</i>	milliseconds	
<i>days</i>	<i>byte</i>										
<i>hours</i>	<i>2Kbyte</i>										
<i>minutes</i>	<i>4Kbyte</i>										
<i>seconds</i>	<i>Mbyte</i>										
milliseconds											

<p><i>x..y special</i></p>	<p>With the other data types: length specification</p> <p>For data types catid, date, device, product-version, time and vsn the length specification is not displayed.</p> <p>x minimum length for the operand value; x is an integer.</p> <p>y maximum length for the operand value; y is an integer.</p> <p>x=y the length of the operand value must be precisely x.</p> <p><i>special</i> Specification of a suffix for describing a special data type that is checked by the implementation. "special" can be preceded by other suffixes. The following specifications are used:</p> <p><i>arithm-expr</i> arithmetic expression (SDF-P)</p> <p><i>bool-expr</i> logical expression (SDF-P)</p> <p><i>string-expr</i> string expression (SDF-P)</p> <p><i>expr</i> freely selectable expression (SDF-P)</p> <p><i>cond-expr</i> conditional expression (JV)</p> <p><i>symbol</i> CSECT or entry name (BLS)</p>
<p>with</p>	<p>Extends the specification options for a data type.</p>
<p>-compl</p>	<p>When specifying the data type "date", SDF expands two-digit year specifications in the form yy-mm-dd to:</p> <p>20jj-mm-tt if jj < 60</p> <p>19jj-mm-tt if jj >= 60</p>
<p>-low</p>	<p>Uppercase and lowercase letters are differentiated.</p>
<p>-path-compl</p>	<p>For specifications for the data type "filename", SDF adds the catalog and/or user ID if these have not been specified.</p>
<p>-under</p>	<p>Permits underscores (_) for the data type "name".</p>
<p>-wild(n)</p>	<p>Parts of names may be replaced by the following wildcards.</p> <p>n denotes the maximum input length when using wildcards.</p>

Due to the introduction of the data types `posix-filename` and `posixpathname`, SDF now accepts wildcards from the UNIX world (referred to below as POSIX wildcards) in addition to the usual BS2000 wildcards.

However, as not all commands support POSIX wildcards, their use for data types other than `posix-filename` and `posix-pathname` can lead to semantic errors.

Only POSIX wildcards or only BS2000 wildcards should be used within a search pattern. Only POSIX wildcards are allowed for the data types `posixfilename` and `posix-pathname`. If a pattern can be matched more than once in a string, the first match is used.

BS2000 wildcards	Meaning
*	Replaces an arbitrary (even empty) character string. If the string concerned starts with *, then the * must be entered twice in succession if it is followed by other characters and if the character string entered does not contain at least one other wildcard.
Terminating period	Partially-qualified entry of a name. Corresponds implicitly to the string <code>./*</code> , i.e. at least one other character follows the period.
/	Replaces any single character.
< $s_x:s_y$ >	Replaces a string that meets the following conditions: <ul style="list-style-type: none"> • It is at least as long as the shortest string (s_x or s_y) • It is not longer than the longest string (s_x or s_y) • It lies between s_x and s_y in the alphabetic collating sequence; numbers are sorted after letters (A... Z, 0...9) • s_x can also be an empty string (which is in the first position in the alphabetic collating sequence) • s_y can also be an empty string, which in this position stands for the string with the highest possible code (contains only the characters X'FF')
< s_1, \dots >	Replaces all strings that match any of the character combinations specified by s . s may also be an empty string. Any such string may also be a range specification " $s_x:s_y$ " (see above).

-s	<p>Replaces all strings that do not match the specified string s.</p> <p>The minus sign may only appear at the beginning of string s.</p> <p>Within the data types filename or partial-filename the negated string -s can be used exactly once, i.e. -s can replace one of the three name components: cat, user or file.</p>
<p>Wildcards are not permitted in generation and version specifications for file names.</p> <p>Only system administration may use wildcards in user IDs.</p> <p>Wildcards cannot be used to replace the delimiters in name components cat (colon) and user (\$ and period).</p>	
POSIX wildcards	Meaning
*	Replaces any single string (including an empty string). An * appearing at the first position must be duplicated if it is followed by other characters and if the entered string does not include at least one further wildcard.
?	Replaces any single character; not permitted as the first character outside single quotes.
[c _x -c _y]	Replaces any single character from the range defined by c _x and c _y , including the limits of the range. c _x and c _y must be normal characters.
[s]	<p>Replaces exactly one character from string s.</p> <p>The expressions [c_x-c_y] and [s] can be combined into [s₁c<sub>1</sub>c_x-c_ys<sub>y</sub>s₂].</p>
[!c _x -c _y]	Replaces exactly one character not in the range defined by c _x and c _y , including the limits of the range. c _x and c _y must be normal characters. The expressions [!c _x -c _y] and [!s] can be combined into [!s ₁ c _x -c _y s_ys ₂].
[!s]	Replaces exactly one character not contained in string s. The expressions [!s] and [!c _x -c _y] can be combined into [!s ₁ c₁c _x -c _y s_ys ₂].

-wild-
constr
(n)

Specification of a constructor (string) that defines how new names are to be constructed from a previously specified selector (i.e. a selection string with wildcards). See also with-wild. n denotes the maximum input length when using wildcards.

The constructor may consist of constant strings and patterns. A pattern (character) is replaced by the string that was selected by the corresponding pattern in the selector.

The following wildcards may be used in constructors:

Wildcard	Meaning
*	Corresponds to the string selected by the wildcard * in the selector.
Terminating period	Corresponds to the partially-qualified specification of a name in the selector; corresponds to the string selected by the terminating period in the selector.
/ or ?	Corresponds to the character selected by the / or ? wildcard in the selector.
<n>	Corresponds to the string selected by the n-th wildcard in the selector, where n is an integer.

Allocation of wildcards to corresponding wildcards in the selector:

All wildcards in the selector are numbered from left to right in ascending order (global index).

Identical wildcards in the selector are additionally numbered from left to right in ascending order (wildcard-specific index).

Wildcards can be specified in the constructor by one of two mutually exclusive methods:

1. Wildcards can be specified via the global index: <n>
2. The same wildcard may be specified as in the selector; substitution occurs on the basis of the wildcard-specific index. For example:
the second "/" corresponds to the string selected by the second "/" in the selector

The following rules must be observed when specifying a constructor:

- The constructor can only contain wildcards of the selector.
- If the string selected by the wildcard <...> or [...] is to be used in the constructor, the index notation must be selected.
- The index notation must be selected if the string identified by a wildcard in the selector is to be used more than once in the constructor. For example: if the selector "A/" is specified, the constructor "A<n><n>" must be specified instead of "A//".
- The wildcard * can also be an empty string. Note that if multiple asterisks appear in sequence (even with further wildcards), only the last asterisk can be a non-empty string, e.g. for "****" or "//**".
- Valid names must be produced by the constructor. This must be taken into account when specifying both the constructor and the selector.

- Depending on the constructor, identical names may be constructed from different names selected by the selector. For example:
 “A/*” selects the names “A1” and “A2”; the constructor “B*” generates the same new name “B” in both cases.
 To prevent this from occurring, all wildcards of the selector should be used at least once in the constructor.
- If the constructor ends with a period, the selector must also end with a period. The string selected by the period at the end of the selector cannot be specified by the global index in the constructor specification.

Examples:

Selector	Selection	Constructor	New name
A/*	AB1	D<3><2>	D1
	AB2		D2
	A.B.C		D.CB
C.<A:C>/<D,F>	C.AAD	G.<1>.<3>.XY<2>	G.A.D.XYA
	C.ABD		G.A.D.XYB
	C.BAF		G.B.F.XYA
	C.BBF		G.B.F.XYB
C.<A:C>/<D,F>	C.AAD	G.<1>.<2>.XY<2>	G.A.A.XYA
	C.ABD		G.A.B.XYB
	C.BAF		G.B.A.XYA
	C.BBF		G.B.B.XYB
A/B	ACDB	G/XY/	GCXYD
	ACEB		GCXYE
	AC.B		GCXY. ¹
	A.CB		G.XYC

¹The period at the end of the name may violate naming conventions (e.g. for fully-qualified file names).

without	Restricts the specification options for a data type.
-cat	Specification of a catalog ID is not permitted.
-corr	Input format: [[C]][V][m]m.na['] Specifications for the data type product-version must not include the correction status.
-gen	Input format: [[C]][V][m]m.n[']

-man	Specifications for the data type product-version must not include either release or correction status.
-odd	The data type x-text permits only an even number of characters.
-sep	With the data type "text", specification of the following separators is not permitted: ; = () < > 'BLANK' (i.e. semicolon, equals sign, left and right parentheses, greater than, less than, and blank).
- tempfile	Specification of a temporary file is not permitted (see #file or @file under filename).
-user	Specification of a user ID is not permitted.
-vers	Specification of the version (see "file(no)") is not permitted for tape files.
-wild	The file types posix-filename and posix-pathname must not contain a pattern (character).
mandatory	Certain specifications are necessary for a data type.
-corr	Input format: [[C]'][V][m]m.naso['] Specifications for the data type product-version must include the correction status and therefore also the release status.
-man	Input format: [[C]'][V][m]m.na[so]['] Specifications for the data type product-version must include the release status. Specification of the correction status is optional if this is not prohibited by the use of the suffix without-corr.
-quotes	Specifications for the data types posix-filename and posix-pathname must be enclosed in single quotes.

9.2 SDF command return codes

SDF returns information relating to the analysis of the command input and command execution in a command return code. This command return code can be compared with the return code at program level and allows you to react in a specific manner to certain error situations.

The command return code consists of three parts:

- The main code that corresponds to a message key. The HELP-MSG-INFORMATION command can be used with this main code to query detailed information.
- The subcode1 that arranges that error situations that arose into an error class. The severity of the error can then be derived from this class.
- The subcode2 that can contain additional information (value not equal to zero).

Command return codes are represented in tabular format in the order subcode2, subcode1, maincode, and explanatory text.

Command return codes that can return a BS2000 command when executed are part of the relevant command description and are listed in tabular form in this manual.

For more detailed information on command return codes, please refer to the manual "Introductory Guide to the SDF Dialog Interface" [[2 \(Related publications\)](#)].

IMON-GPN command return codes

Many utilities can be called via separate START command. Internal IMON-GPN functions are used in these calls. These functions can return the following additional command return codes if an error occurs:

(SC2)	SC1	Maincode	Meaning
	32	IMO9101	Command terminated abnormally; internal error in IMON-GPN
	64	IMO9100	Command not executed; installation unit, version or logical name not found, etc.; the cause can be found in the SYSOUT message.

9.3 Macro syntax

The macro operands are divided into two groups:

- Format operands that define the form and the generation of the macro
- Function operands that define the contents of the parameter range in an interfacespecific manner

Syntax of the format operands

The syntax of the format operands corresponds to the BS2000 rules (see the manual “Executive Macros” [6]).

MF controls generation of the code (“macroform”)

PREFIX controls generation of the names (first letter)

MACID controls generation of the names (second through fourth letters)

PARAM controls addressing of the parameter range

XPAND controls expansion of the individual data structures;

If XPAND is contained in the macro syntax, it must be assigned a value as it does not have any default (mandatory operand).

Macro forms

The MF operand defines the form of the macro. It can accept the following values:

MF=C / D / L / E / M / S

MF=C

The layout of the data structure (generally the parameter range) is generated, whereby each field and each equation are named. This data structure becomes part of the current program section (CSECT/DSECT). The function operands of the macro are not evaluated.

XPAND

If this is required for the interface, the expansion of individual data structure can be selected using control operands – for instance, if different output areas can be selected in an interface (e.g. XPAND=PARAM / out1).

PREFIX

The PREFIX operand is used to generate the names to be created.

PREFIX, which is exactly one letter, is used as the first letter of all names.

The default value is the code letter of the function unit to which the macro belongs. To avoid similarities in names, PREFIX must be used if the same data structure is used a number of times within a module.

MACID

The MACID operand is used to generate the names to be created and specifies the second through fourth characters of the name. Default value: two characters as a development group code and one character as a macro-specific code. The default value guarantees that no names are the same within the component group.

MF=D

Generates the layout of the data structure as in MF=C. A DSECT statement is also generated. The MACID operand is ignored, the default value is assumed.

MF=L

Generates an instance of the parameter range by evaluating the function operands. This macro form does not generate any field names. The label specification is used to name the generated constants.

MF=E

Generates the commands needed to call the function. The function operands are ignored. Addressing of the parameter range must be checked using the PARAM control operand:

PARAM

The PARAM operand controls addressing of the parameter range.

=<adr>

Address of the parameter range as a name

=(<reg>)

Address of the parameter range is contained in the register with the name <reg>

MF=M

Modifies a parameter range previously initialized by copying an MF=L form taking into account an evaluation of the specified function operands.

Operands that are not specified retain their original state.

The consistency of the parameter range is the responsibility of the person calling the macro.

MF=M requires that the MF=D form or MF=C form was called with the same values of the PREFIX and MACID operands and that a USING statement was issued before the DSECT (MF=D form) was addressed.

MF=S

Generates an instance of the parameter range taking into account an evaluation of the function operands, as with MF=L. This also generates commands that – if necessary – skip the parameter range instance and call the function.

Elements of the macro syntax

Identifier	Meaning	Example
UPPERCASE LETTERS		

	Uppercase letters denote keywords or constants and must be entered by the user exactly as shown. Keywords must begin with * if both keywords and names of constants or variables can be specified as alternatives.
	DELETE=*NO / *YES
lowercase letters	
	Lowercase letters denote data types of values or variables which can be specified by the user.
	COUNT=<integer 1..8>
=	
	The equals sign links the operand name to the operand values associated with it.
	FILE=<c-string>
< >	
	Angle brackets denote variables whose allowed values are described by the data types.
	REPEAT=<integer>
<u>Underscoring</u>	
	Underscoring denotes the default value of an operand. If an operand has no default value, specification of an operand is mandatory.!
	ACTION=* <u>ADD</u> / *MOD
/	
	The slash separates alternative operand values.
	CHECK=*NO / *YES
list-poss(n)	
	A list can be formed from the operand values following list-poss. n specifies the maximum number of list elements (for the type Keylist). The list must be enclosed in round brackets if more than one element is specified.
	OUTPUT=list-poss(2): *SYSOUT / *SYSLST Specification: OUTPUT=*SYSOUT OUTPUT=(*SYSOUT, *SYSLST)

Table 4: Macro syntax: Elements

An operand is assigned an operand value from a defined range of values by means of an equals sign.

This range of values is specified by a data type. The following table contains the data types of the operand values.

Data types of the operand values

Data type	Character range	Remarks
c-string	EBCDIC character	Must be enclosed in single quotes
integer	[+-] 0..2147483647	Is a decimal number
var:	Introduces a variable specification. The variable type follows a colon (see table 6)	<var:var-type>
reg:	Register 0..15	Specification: (<reg:var-type>)

Table 5: Macro syntax: Data types of operand values

Suffixes for data types

Suffix	Meaning
n..m	For the integer data type, n..m represents an interval; n: minimum value m: maximum value
	For the c-string data type, n..m represents a length in bytes; n: minimum length m: maximum length with $n < m$
n	For the c-string data type, n represents a length in bytes; n must be adhered to exactly.

Table 6: Macro syntax: Suffixes for data types

The operand values can be entered directly as a character string or integer number (see c-string and integer data types) or identified indirectly by means of a variable (see var: data type). The following table contains the data types that are possible for variables.

Data types of the variables

Data type	Meaning	Definition in the program
char:n	This variable is a character string of n characters. If no length is specified, n=1 is assumed.	CLn
int:n	This variable is an integer number that occupies n bytes. If no length is specified, n=1 is assumed. Condition: n 4	FLn
enum-of E:n	The variable is the enumeration E that occupies n bytes. If no length is specified, n=1 is assumed. (n 4)	XLn
pointer	The variable is an address or an address value.	A

Table 7: Macro syntax: Data types of the variables

Standard headers

All macros use the standard header to identify their interface.

The standard header is a field 8 bytes in length at the start of the parameter range containing the (standardized) name of the interface and 4 bytes for recording a return code.

The standard header is generated and initialized by the relevant macro. It is assigned the relevant values for UNIT, FUNCTION, and VERSION.

Structure of the standard header

Byte	Field contents and meaning
0 - 1	Name of the function unit (UNIT) with the demanded function
2	Name of the function (FUNCTION) within the function unit
3	Name of the modification status (VERSION) of the function
4	Subvalue 2 of the return code (subcode2)
5	Subvalue 1 of the return code (subcode1)
6 - 7	Main value of the return code (maincode)

The following return code values apply for all macros:

(SC2)	SC1	Maincode	Meaning
-------	-----	----------	---------

X'00'	X'00'	X'0000'	Function was executed successfully. There is no additional information for the maincode.
X'01'	X'00'	X'0000'	Function was executed successfully. No further actions are required.
X'00'	X'01'	X'FFFF'	The required function is not supported (incorrect specification for UNIT or FUNCTION in the default header). Error that cannot be resolved.
X'00'	X'02'	X'FFFF'	The required function is not available. Error that cannot be resolved.
X'00'	X'03'	X'FFFF'	The specified version of the interface is not supported (incorrect version specification in the default header). Error that cannot be resolved.
X'00'	X'04'	X'FFFF'	Parameter block not in line with the word limit.
X'00'	X'41'	X'FFFF'	The subsystem is not available; it must be generated explicitly.
X'00'	X'42'	X'FFFF'	The calling task is not connected with this interface; it must be connected explicitly.
X'00'	X'81'	X'FFFF'	The subsystem is not currently available.
X'00'	X'82'	X'FFFF'	The subsystem is in the status DELETE or HOLD.

Maincode identifies the result of executing the function. Subcode1 is used to classify the main value. Subcode2 is used to further divide the errors in error classes or contains additional diagnostic information.

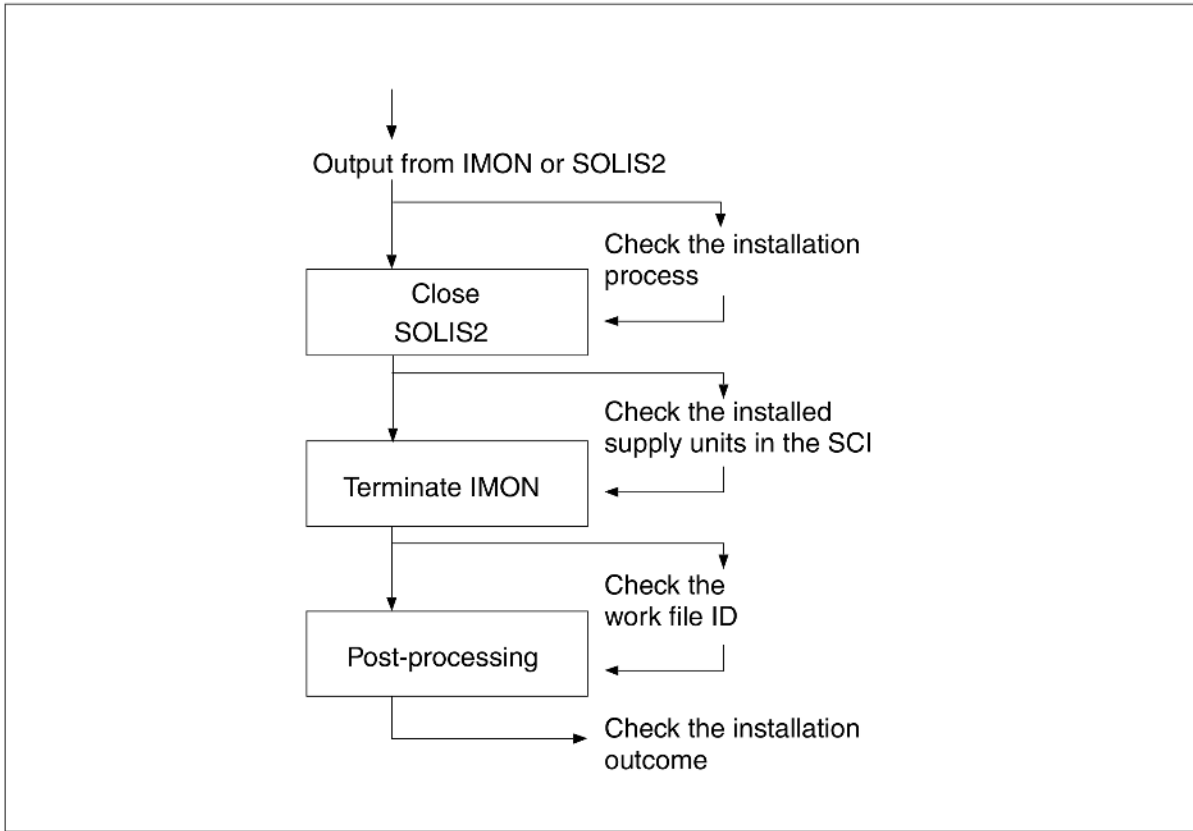
9.4 Installation sequence plans

- Default installation on the home pubset (imon_en_250305, #160) 1
- Default installation on an imported pubset (imon_en_250305, #161) 1
- Customer-specific installation on the home pubset with customizing (imon_en_250305, #162) 1
- Customer-specific installation with prior parking (imon_en_250305, #163) 1
- Sequence of a customer-specific multiple installation of a parked delivery (imon_en_250305, #164) 1
- Sequence of a customer-specific multiple installation of a previously installed delivery (imon_en_250305, #165) 1

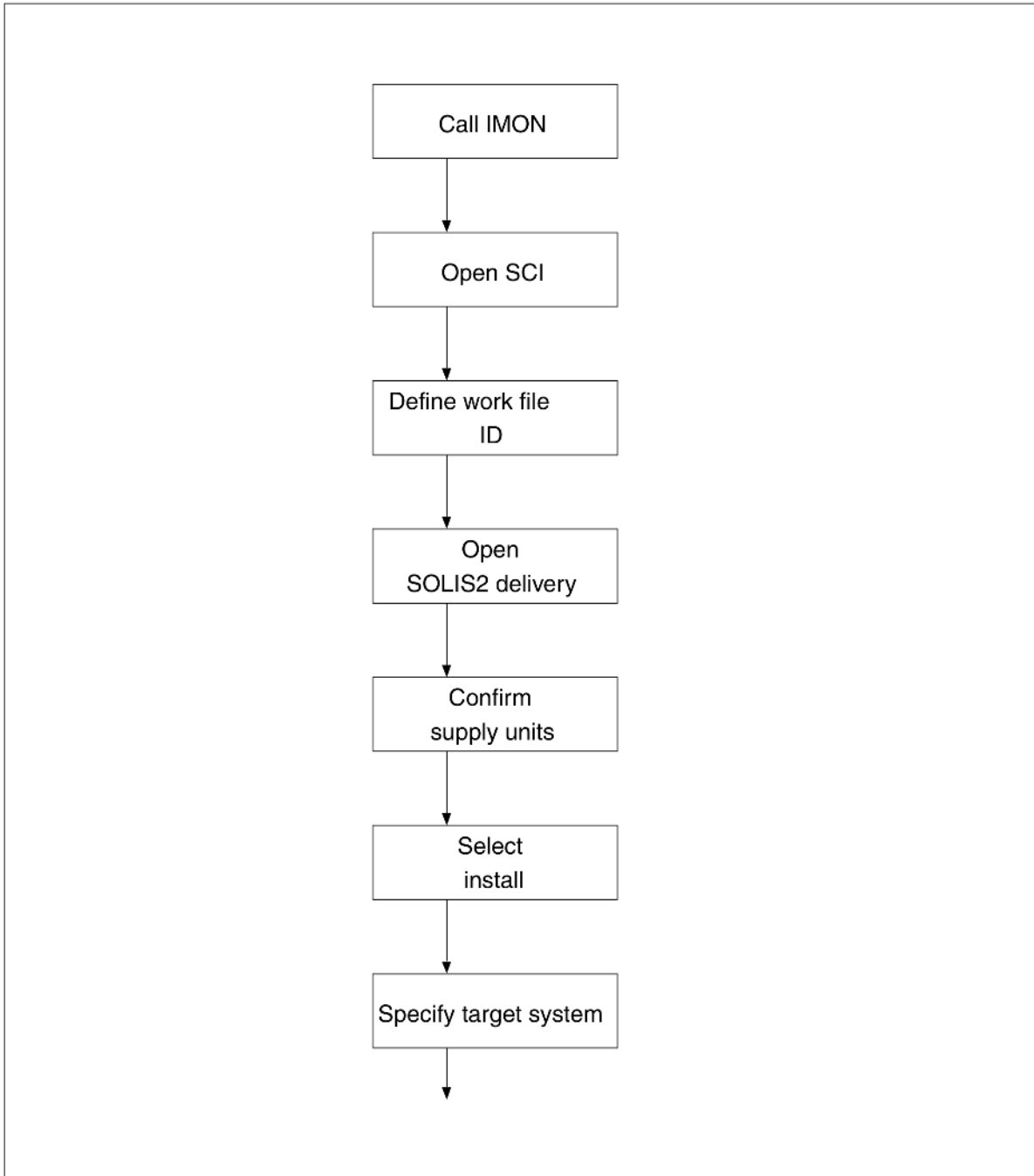
9.4.1 Default installation on the home pubset



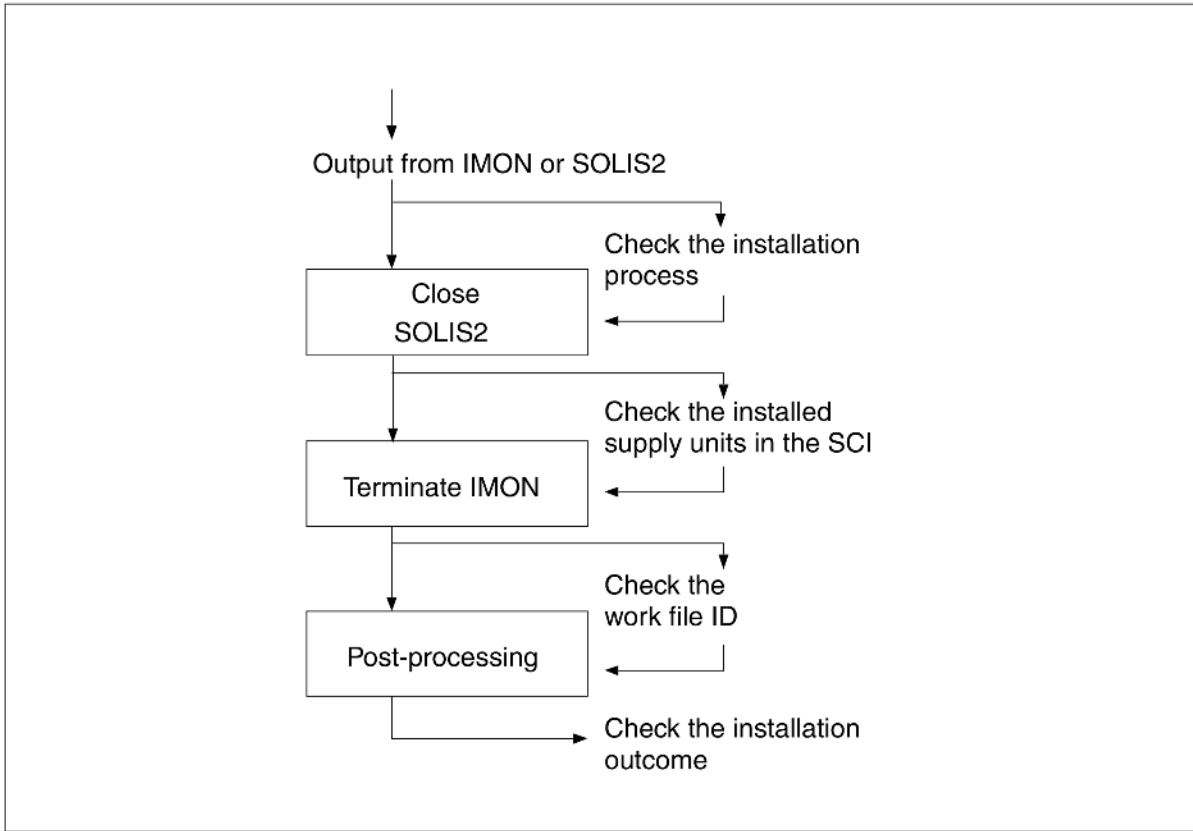
continued



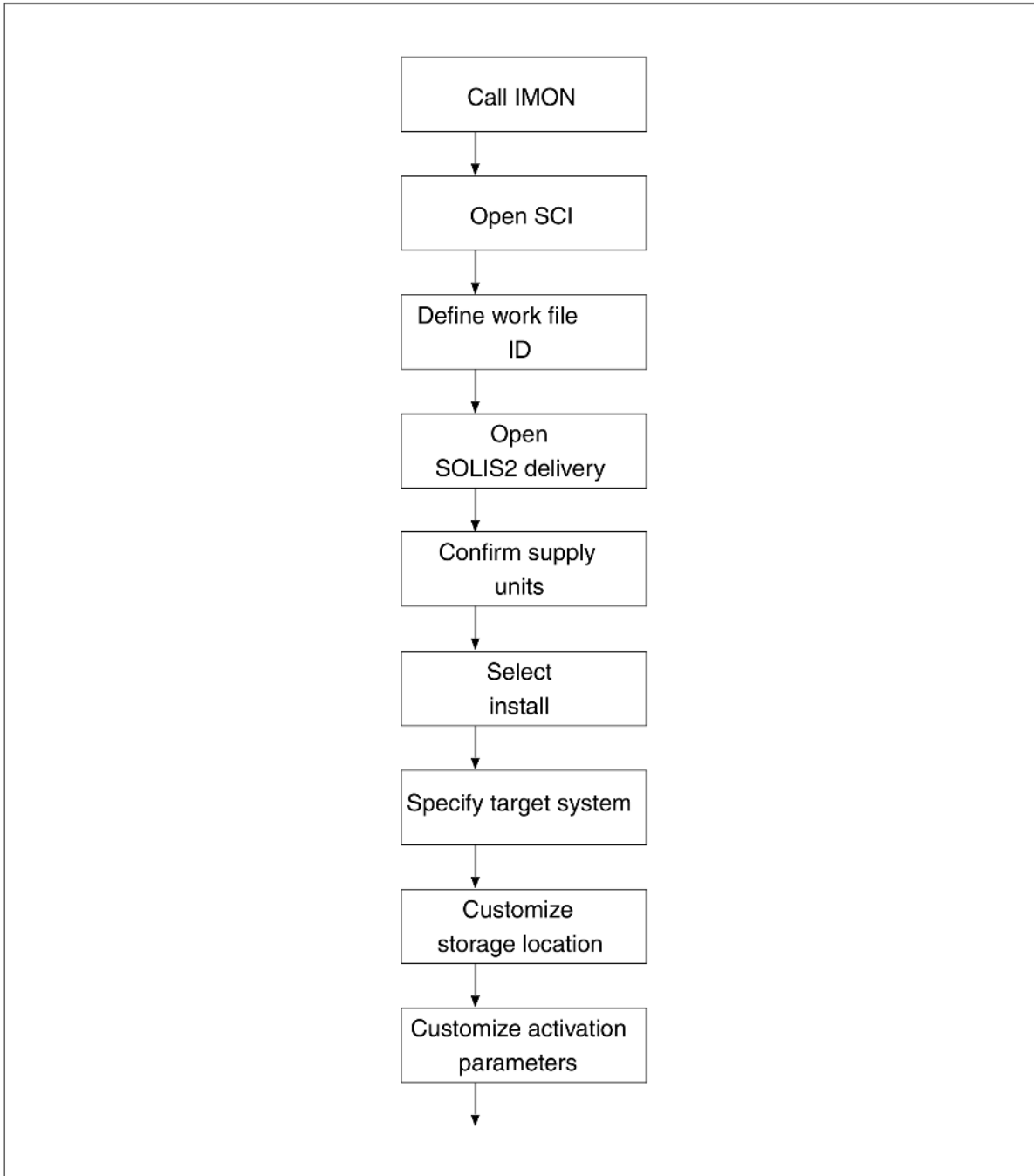
9.4.2 Default installation on an imported pubset



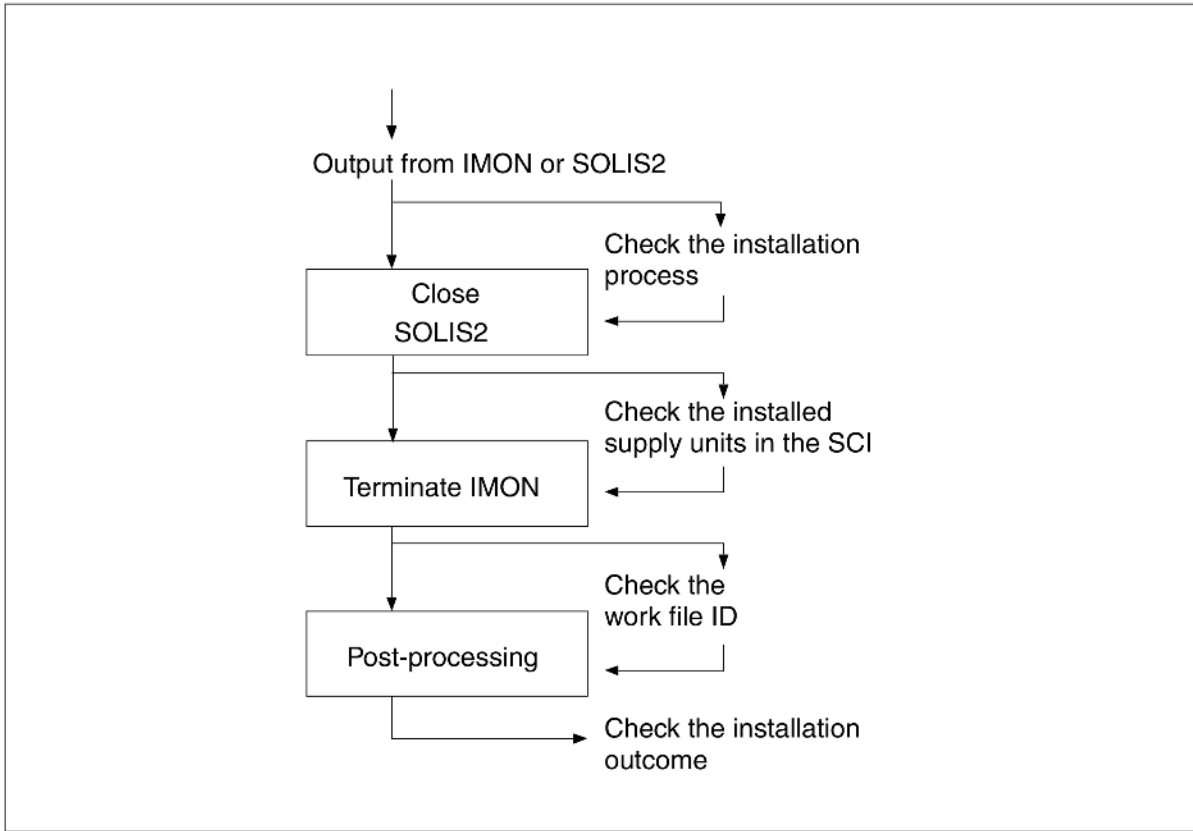
continued



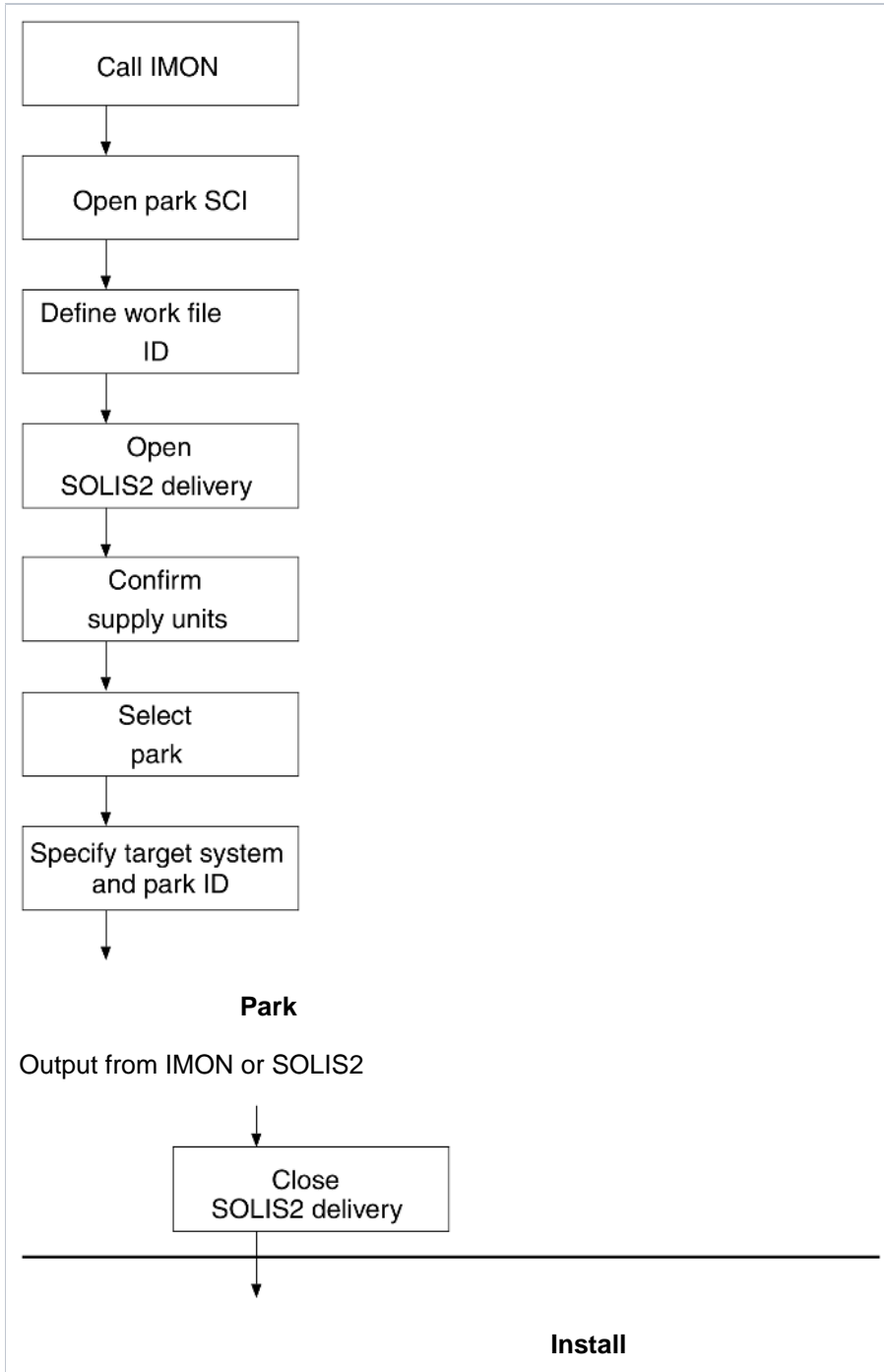
9.4.3 Customer-specific installation on the home pubset with customizing



continued

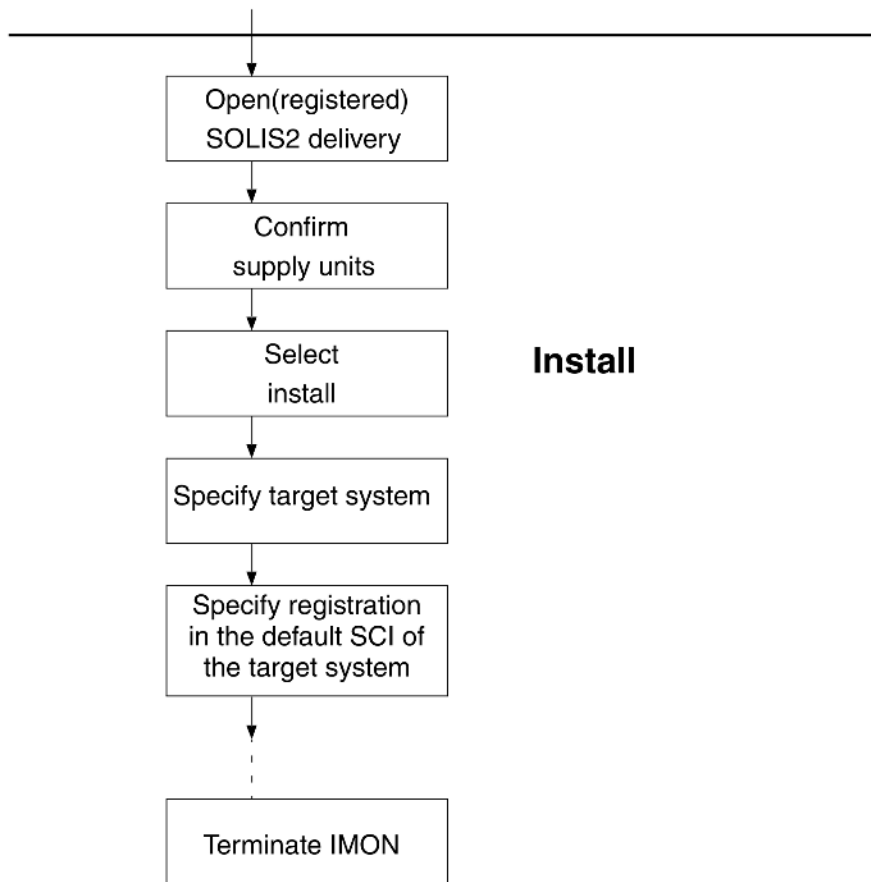


9.4.4 Customer-specific installation with prior parking



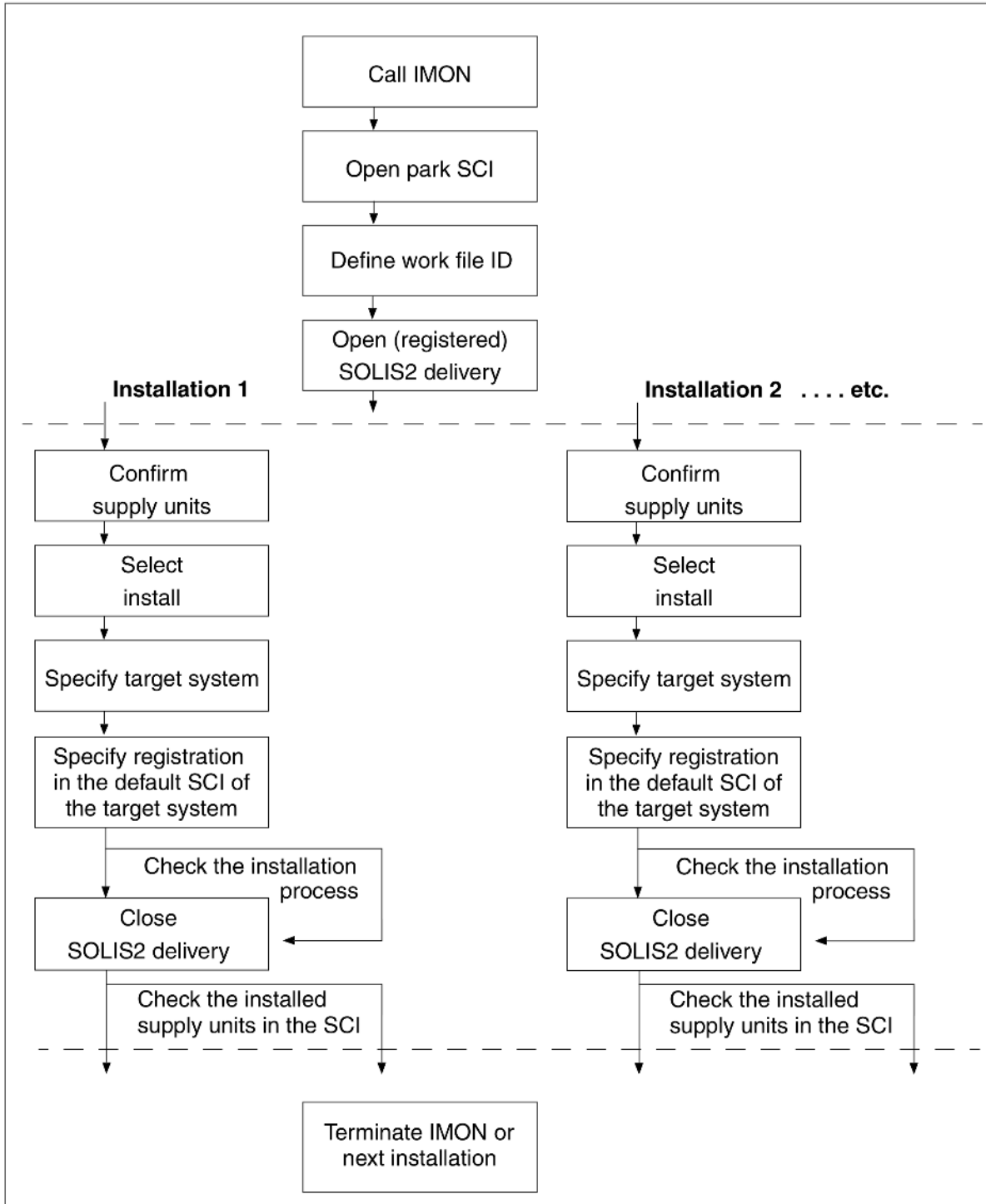
continued

Park

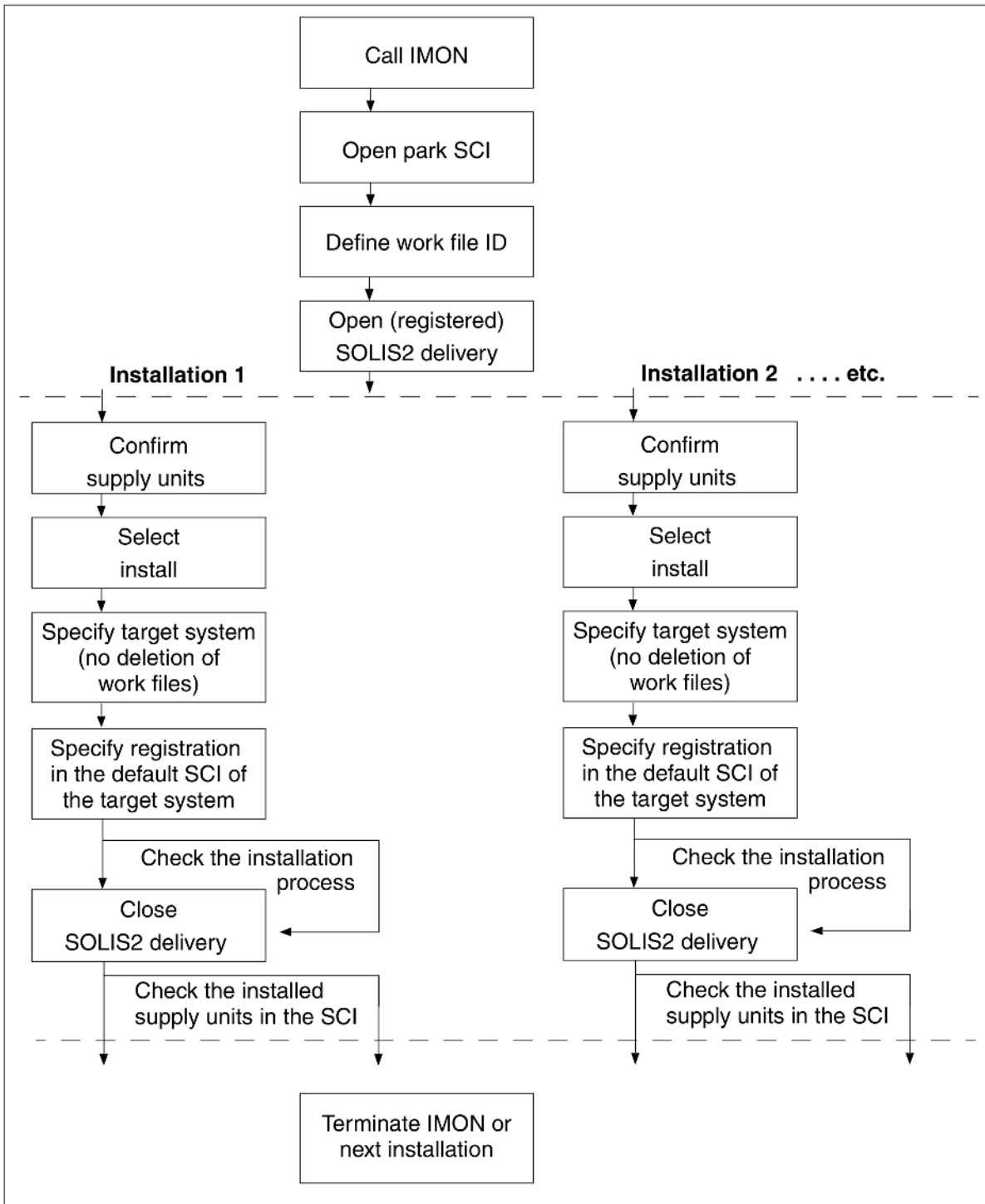


Install

9.4.5 Sequence of a customer-specific multiple installation of a parked delivery



9.4.6 Sequence of a customer-specific multiple installation of a previously installed delivery



10 Glossary

abbreviated command

Key or keystroke combination with a special key assignment. An abbreviated command can be used to call a function in an application without having to select the function from a *menu*.

activation

Way of preparing special installation items for use in the customer system. SDF syntax files, message files, the DSSM subsystem catalog, the RMS depot and rep loader can already be prepared during the installation.

body

The active area for a use in an FHS mask. This area contains information and/or prompts for a user selection.

correction version

That *version* of a *product* which contains corrections and which, identified by the correction state, supersedes the last valid correction state.

customizing

Specifications for adapting the installation to the customer's system. Customization of the system environment has an effect on the *placement* and *activation*.

deinstallation

Procedure in which installed software that is not needed any more is removed. The original state before installation can be restored with the *Undo function*.

delivery

A delivery consists of one or more *supply units* and the related *supply information*. It is the largest delivered unit, which can be installed as a whole with IMON.

delivery information

Information required in order to install the software supplied.

dialog box

Framed area on the screen enabling the user to enter additional information or make selections within a mask. Functions are triggered, parameters set and information output in dialog boxes.

dialog expansion for the Format Handling System (FHS-DE)

FHS-DE enables formats compatible with the alpha styleguide to be displayed on a terminal.

Format Handling System (FHS)

Program for format control which supports the exchange of formatted messages between application programs and data display terminals.

help mask

Dialog box with help information output by FHS.

help system

Part of an application program with which the user can quickly obtain information on the structure, purpose and method of operation of the application.

IDF (Installation Definition File)

File containing information on private software or software that is already installed - but not with IMON/SOLIS2 - and which is to be registered with IMON.

input field

Single-line or multiple-line field of predetermined length. Input fields accept the user's text and data inputs to the system.

installation

The process of making software available on a data processing system.

installation information

Information on software already installed on the system. IMON registers installation information in the *Software Configuration Inventory (SCI)*.

installation item (II)

A *release item* that has been installed. The smallest installed element (file) that can be administrated by IMON. An element of an *installation unit*.

installation unit (IU)

A *release unit* that has been installed. A group of *installation items*.

instruction area

That part of an FHS mask in which the user can issue FHS command statements.

local node

The machine on which a user works in a network. In a local area network a distinction is drawn between local and *remote nodes*. All nodes on which the user is working directly are local to this user. All other nodes are remote. If a user connects to a remote node, it becomes the local node for this user.

logical identifier

The logical ID of an item describes the item unambiguously within a unit, irrespective of the name, version and correction state of the unit. The logical ID is also independent of the path of the file in BS2000.

Unit: *release unit* or *installation unit*.

Item: *release item* or *installation item*.

main version

A new or *version* of a *product*, or one introducing significant changes in functional scope. The main version precedes the period in the version designation.

mask

Form displayed on the screen and used for the input and output of data. A mask is divided into a number of areas: *Status area*, *body*, *instruction area* and *message area*.

menu

Set of functions grouped together under an associating term. The functions are represented by the *menu items* and the associating term by the *menu title*.

menu bar

Single-line area located above the *body* and containing the *menu titles* of the available menus.

menu item

Represents a function in a *menu*. Selecting the menu item activates the function.

menu title

Associating term used to group the items in a *menu*. All menu titles appear in the *menu bar*.

message area

The part of a mask in which messages are output.

package

Generally a set of *units offered*, grouped for reasons relating to sales and marketing.

path name

Name of a BS2000 file complete with catalog ID and user ID.

placement

Process by which release items are read from the distribution medium and stored under the installation ID as installation items (files). This includes copying from the distribution medium (using ARCHIVE or LMS/LMSCONV), definition of the file attributes or the backing up of existing files.

product

Sales and marketing term for a *software product*.

product movement file (PBD)

File which describes the delivery. Each delivery contains at least one file which serves as *delivery information*.

product structure

Structure of a *software product* in sales-related and technical units.

Sales: *bid packages*, *bid units*.

Technical: *supply units*, *supply groups (release units)*, *supply components (release items)*.

release item (RI)

An element (file) of a *release unit* and the lowest technically addressable level of the *product structure* (synonym: *supply component*).

release unit (RU)

Element of a *supply unit*.

Group of multiple *release items* (synonym: *supply group*).

remote node

In a local area network a distinction is drawn between local and remote nodes.

All nodes on which the user is not working directly are remote to this user. The user can communicate with all remote nodes in the network. If a user connects to a remote node, it becomes the *local node* for this user.

revision version

The *version* of a *product* which, although resulting from functional changes, does not constitute a new *main version* for reasons associated with sales and marketing. An addition version is represented by the code which follows the period in the version.

Software Configuration Inventory (SCI)

Central database containing the information about the objects registered by IMON.

software product

Software for which the user purchases the right of usage.

SOLIS2 (software supply and information system)

Decentralized process for support in the technical task of product and correction-program updating for the BS2000 operating system.

status area

That part of a *mask* which provides information on the mask's contents.

supply component (SC)

Element (file) of a *supply group* and the lowest technically addressable level of the *product structure* (synonym: *release item*).

supply group (SG)

A group comprising multiple *supply components*.

Element in a *supply unit* (synonym: *release unit*).

supply unit (SU)

Technical equivalent of the *bid unit*. Each *bid unit* is assigned one and only one supply unit.

Comprises multiple *supply groups*.

SYSSII file (SYStem Structure and Installation Information file)

File supplied along with a *release unit*. The SYSSII file contains the information required for automatic *installation* with IMON/SOLIS2. The definition of the *logical ID* is contained in the SYSSII file.

target system

BS2000 system for which the installation is carried out. It is identified by the version and the associated pubset on which the installation items are stored.

Undo function

Procedure in which the installation of software is undone and the original state before installation is restored (special case of a *deinstallation*).

unit offered

Designation used in Sales & Marketing for a *supply unit* as per product file / Sales Handbook (SHB).

version

Status of a software product having a defined functional scope.

Format: <mm.naso> where

mm main version (01..99)

n addition version (0..9)

aso a correction state of user interface (A..Z)

so correction state of source and object (00..99)

11 Abbreviations

CAP	Customer Approved
DSSM	Dynamic SubSystem Management
FHS	Format Handling System
FHS-DE	FHS Dialog Expansion
IDF	Installation Definition File
II	Installation Item
IMON	Installation MONitor
IMON-BAS	Installation MONitor (BASic functionality)
IMON-GPN	Installation Path Manager (Get PathName)
IMON-SIC	IMON Structure Installation Creation
IU	Installation Unit
PBD	Product Movement File (German abbreviation)
OU	Offer Unit
OP	Offer Package
RI	Release Item
RU	Release Unit
SC	Supply Component
SG	Supply Group
SII	Structure and Installation Information
SU	Supply Unit
SCI	Software Configuration Inventory
SDF	System Dialog Facility
SG	Supply Group
SIR	System Install and Restore
SOLIS2	Software Supply and Information System (German abbreviation)
SU	Supply Unit
SYSSII	System Structure and Installation Information

12 Related publications

You will find the manuals on the internet at <https://bs2manuals.ts.fujitsu.com/index>.

- [1] **BS2000 OS DX**
Introduction to System Administration
User Guide
- [2] **SDF (BS2000)**
SDF Dialog Interface
User Guide
- [3] **BS2000 OS DX**
Commands
User Guide
- [4] **BS2000 OS DX**
System Installation
User Guide
- [5] **BS2000 OS DX**
DMS Introduction
User Guide
- [6] **BS2000 OS DX**
Executive Macros
User Guide
- [7] **FHS (BS2000)**
Dialog Extension for TIAM and SDF-P
User Guide