

English



Fujitsu Software BS2000

CMX (Unix)

Manual Changes

User Guide

Valid for:
CMX V6.0 (Unix)

Edition December 2018

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 © 2025 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.

Ergaenzungen zum Handbuch:

=====

Dieses Dokument verfolgt insgesamt 3 Zielsetzungen. Im ersten Abschnitt beschreibt es die grundlegenden Unterschiede zwischen den beiden Produkten PCMX und CMX, insbesondere im Protokollprofil RFC1006. Der zweite Abschnitt enthaelt Korrekturen und Klarstellungen zum Handbuch "CMX V6.0 (Unix) - Betrieb und Administration". Der dritte Abschnitt beschreibt Einschränkungen von PCMX gegenueber CMX und nimmt dabei Bezug auf das Handbuch "CMX - Anwendungen programmieren".

Unterschiede zwischen PCMX und CMX:

=====

CMX und PCMX stellen den Anwendungen die Transportdienste entsprechend der ISO-Norm 8072 zur Verfuegung und erlauben somit die Kommunikation zwischen Anwendungen in verschiedenen Rechnern. CMX und PCMX unterscheiden sich hinsichtlich

- * der unterstuetzten Unix-Plattformen,
- * der bereitgestellten Protokollprofile,
- * der Realisierung der angebotenen Dienste und somit auch
- * hinsichtlich des Funktionsumfangs der Kommunikation und der Administration.

Das Produkt CMX wird nur auf der Solaris-Sparc-Unix-Plattformen bereitgestellt, waehrend PCMX als reine Userland-Komponente saemtliche anderen marktgaengigen Unix-Plattformen unterstuetzt.

PCMX bietet den OSI-Transportdienst auf Basis von TCP/IP an, das Bestandteil aller Unix-Derivate ist, waehrend CMX auf solaris-Sparc den Transportdienst zusaetzlich ueber WAN- und ISDN-Subnetze zur Verfuegung stellt, auf die ueber eigene Controller zugegriffen wird.

PCMX ist als eine reine Bibliotheksloesung konzipiert, die auf TCP/IP ueber die in allen Unix-Systemen zur Verfuegung stehende Socket-Schnittstelle zugreift, waehrend CMX den Anwendungen das sog. API (application programming interface) als Bibliothek zur Verfuegung stellt. Die Protokollprofile sind in CMX als Komponenten des Betriebssystems realisiert. CMX stellt den Anwendungen die Kommunikationsfunktionen ueber zentrale, systeminhaerente Komponenten bereit. Architekturbedingt bietet CMX somit mannigfaltigere differenziertere Kommunikationsfunktion und Administrationsfunktionen wie die Routenwahl ueber bestimmte Controller oder die Ausgabe saemtlicher Transport-Verbindungen, die in dem als Bibliotheksloesung realisierten PCMX nicht moeglich sind.

Die wesentlichen Differenzen zwischen CMX und PCMX im Protokollprofil RFC1006:

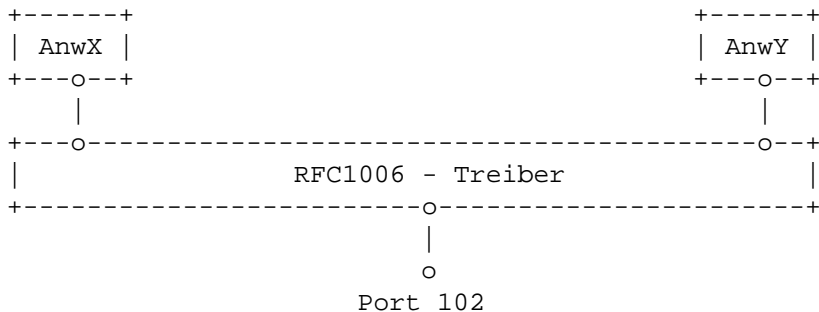
Diese Unterschiede resultieren aus der unterschiedlichen RFC1006-Implementierung von CMX und PCMX.

PCMX:

In der TCP/IP-Welt werden Anwendungen bzw. Dienste (services) ueber Ports eindeutig adressiert. In der PCMX-Architektur ist die RFC1006-Protokollimplementierung durch die Nutzung der Bibliotheksfunktionen implizit Bestandteil der Anwendung. Durch die Nutzung der Socket-Schnittstelle muss im PCMX jeder RFC1006-Anwendung genau ein TCP/IP-Port zugeordnet sein. Somit sind verschiedene Anwendungen auch nur ueber verschiedene Ports erreichbar.



Im CMX hingegen meldet sich der zentrale RFC1006-Protokolltreiber am privilegierten Port 102 von TCP/IP an und empfaengt ueber diesen Port alle ankommenden Verbindungsaufbauten. Die Anwendungen melden sich beim RFC1006-Treiber an und sind von extern ueber den Port 102 adressierbar. Um die Anwendungen unterschieden zu koennen, muessen sie sich mit verschiedenen T-Selektoren anmelden.



Die Implementierungsunterschiede bedingen auch geringfuegig abweichende Adressierungskonzepte. Sollen in einem PCMX-System mehrere lokale RFC1006-Anwendungen gleichzeitig betrieben werden, so muss die lokale Anwendung bei ihrer Anmeldung neben ihren RFC1006-T-Selektor zusaetzlich den Port angeben, an den sie sich bindet. Netzpartner, die RFC1006-Anwendungen in Systemen mit PCMX adressieren, haben neben der IP-Adresse des Zielrechners und dem RFC1006-T-Selektor zusaetzlich noch den Port anzugeben, an den sich die Anwendung gebunden hat.

Korrekturen und Ergaenzungen zum Handbuch CMX V6.0(UNIX):
 =====

Die Korrekturen und Ergaenzungen beziehen sich auf das Handbuch "CMX V6.0(UNIX) - Betrieb und Administration" mit der Bestellnummer U20871-J-Z145-2. PCMX wird in diesem Manual durchgaengig als CMX(Unix) bezeichnet.

5.2.8.4 Versionsangabe fuer Format und Syntax

 Fuer 'version' ist die Versionsnummer 5.0 anzugeben statt 6.0.

6.1 Transport Provider Selection-Selektion verwalten und pflegen

 In dem Beispiel wird empfohlen, den Eintrag fuer LOOPSBKA nie auszukomentieren. Wenn jedoch keine Transportverbindung ueber XTI/TLI konfiguriert wird, sollte der Eintrag fuer LOOPSBKA auskommentiert werden. Dies entspricht der Grundeinstellung in der aktuellen Version.

7.1.1 Konfigurationsdaten fuer lokale TS-Anwendungen

 Allgemein zur Konfiguration:

Jeder TCP/IP-Anwendung ist genau eine Portnummer zugeordnet. Da PCMX die RFC1006-Anwendungen 1:1 auf TCP/IP-Anwendungen abbildet, muss in PCMX im Gegensatz zum kernelbasierten CMX fuer lokale RFC1006-Anwendungen neben dem RFC1006-T-Selektor zusaetzlich ein Eintrag
 TSEL LANINET A'<Portnummer>
 mit einer Portnummer im LOKALEN NAMEN angegeben werden.

Um RFC1006-Anwendungen im PCMX adressieren zu koennen, muss im Eintrag fuer die TRANSPORTADRESSE zu einem PCMX-Partner die Portnummer mit angegeben werden, da sonst der Port 102 adressiert wird.

TA RFC1006 <IP-Adresse> PORT <Portnummer> A'T-Selector'

Seite 60, 4. Absatz

Die Aussagen des Abschnittes "Wenn Sie beide Adress-Formate angegeben haben ..." treffen nur fuer das CMX zu. PCMX enthaelt keinen RFC1006-Treiber und besitzt keinen zentralen TCP-Listener auf dem Port 102. Der Absatz lautet fuer PCMX richtigerweise:

Wenn Sie beide Adress-Formate angegeben haben, handelt es sich um eine sogenannte RFC1006-Anwendung. Wenn eine solche TS-Anwendung fuer passiven Verbindungsaufbau angemeldet ist, belegt sie den im lokalen Namen angegebenen Port, d.h. keine andere TS-Anwendung kann sich mit dieser Portnummer fuer den passiven Verbindungsaufbau anmelden.

Umgekehrt scheitert die Anmeldung, falls bereits ein TCP-Listener mit dieser Portnummer existiert.

Beim aktiven Verbindungsaufbau werden die angegebene Portnummer und der T-Selektor dem Partnersystem als Adressinformation ueber das RFC1006-Protokoll in Form des Parameters calling TSAP-ID mitgeteilt.

Der mit dem Adress-Format RFC1006 angegebene T-Selektor muss eindeutig sein und kann nur genau einer fuer den passiven Verbindungsaufbau angemeldeten TS-Anwendung angehoren.

Auf TCP/IP Ebene wird zur IP-Adresse + Port Nummer aufgebaut. Damit besteht bereits auf dieser Ebene eine Verbindung zur Applikation (die zuvor diesen Listening Port aufgebaut hat). Beim darueberliegenden RFC1006 Verbindungsaufbau wird auf die uebereinstimmung des T-Selektors mit dem bei der Anmeldung angegebenen geprueft.

7.1.2 Konfigurationsdaten fuer ferne TS-Anwendungen

Hinweis: Wenn sich die ferne TS-Anwendung auf einem System mit PCMX befindet, muss immer die Portnummer der Anwendung mitangegeben werden. Eine Verteilung ueber den Port 102 findet in PCMX nicht statt.

7.2.1 Aktiver Verbindungsaufbau

Die Beschreibung im 2-ten Absatz ist nicht zutreffend fuer PCMX und ist durch folgenden Absatz zu ersetzen:

Die CMX-Anwendung uebergibt die Partneradresse im Verbindungsaufbau t_conrq. Hierdurch wird zuerst eine TCP-Verbindung zu dem angegebenen Port im Partnersystem aufgebaut. Ueber diese TCP-Verbindung wird dann die CR TPDU des RFC1006-Protokolls mit calling TSAP-ID und called TSAP-ID an das Partnersystem gesendet. Der calling TSAP enthaelt den T-Selektor der rufenden Anwendung, der called TSAP den T-Selektor der fernen gerufenen Anwendung. Der weitere Ablauf ist im folgenden Abschnitt aus der Sicht des passiven Systems beschrieben.

Bild 7:

PCMX unterstuetzt den standardmaeßig dem RFC1006-Protokoll zugewiesenen Port 102 nicht. Es erfolgt auch

keine Verteilung der eingehenden Verbindungsanforderungen anhand des T-Selektors.

In dem Beispiel der TNS Eintraege muss beim Verbindungsaufbau zu einem System mit CMX(Unix) auch die Portnummer mit angegeben werden.

7.2.2 Passiver Verbindungsaufbau

Bild 8:

CMX(Unix) unterstuetzt den RFC1006 Standard-Port 102 nicht. Es erfolgt auch keine Verteilung der eingehenden Verbindungsanforderungen anhand des T-Selektors.

Dieser Abschnitt ersetzt den dem Bild 8 folgenden Abschnitt:

Ausgeloest durch die Initiative der Partneranwendung erhaelt die CMX-Anwendung ueber den obengenannten TCP-Listener eine Verbindungsanforderung mit der Quell-Adresse (Internet-Adresse und Portnummer) des fernen TCP-Verbindungsendpunkts und der Zieladresse. Nachdem die TCP-Verbindung etabliert ist, empfaengt die RFC1006-Implementierung ueber diese ein RFC1006-Protokoll-Element (CR TPDU). Dieses Protokoll-Element enthaelt in Form des Parameters calling TSAP-ID den T-Selektor der rufenden CMX-Anwendung und in Form des Parameters called TSAP-ID den T-Selektor der gerufenen CMX-Anwendung, naemlich der angemeldeten lokalen CMX-Anwendung. RFC1006 prueft intern, ob sich die lokale CMX-Anwendung mit der called TSAP-ID angemeldet hat.

Der TNS Eintrag der lokalen Anwendung muss auch einen LANINET Eintrag fuer die Portnummer aufweisen.

Kapitel 3: CMX Anwendungen Programmieren

PCMX und CMX stellen die gleiche Programmierschnittstelle fuer die Kommunikationsfunktionen zur Verfuegung.

PCMX hat keine Kernel Komponenten, alle Pruefungen erfolgen daher nur auf Prozess-Basis. Die hier beschriebenen, sich daraus ergebenden, Einschraenkungen, beziehen sich auf Verbindungen ueber TCP/IP (RFC1006, LANINET).

3.4.1 TS-Anwendungen und Prozesse

Hinweis zu Bild 4: Eine TS-Anwendung - mehrere Prozesse
Wenn eine TS-Anwendung aus mehreren Prozessen besteht, die auch Verbindungen entgegennehmen sollen (passive Verbindungsannahme), so erhaelt nur der erste Prozess dieser TS-Anwendung, der sich bei PCMX anmeldet, diese Verbindungen zugestellt.

5.1 Anmelden bei CMX

Seite 44, Anmeldung zum passiven Verbindungsaufbau
Nur der erste Prozess einer TS-Anwendung erhaelt ankommende Verbindungswuensche zugestellt.

8.1 uebersicht der Programmschnittstelle

Seite 90, Signalisierung bei asynchroner Ereignisverarbeitung
PCMX unterstuetzt die mit CMXINIT aktivierte Signalisierung nur wenn sie vom verwendeten Transportsystem angeboten werden. In allen bisher untersuchten Faellen war das nicht der Fall. Die Implementierung sollte auf die Verwendung von t_callback() umgestellt werden.

Seite 92, Verbindungsaufbau, -abbau und -umlenkung
Eine Verbindungsumlenkung kann nur zu solchen Prozessen erfolgen, die nach abgeschlossenem Verbindungsaufbau von dem Prozess gestartet wurden.

8.6.1 t_attach - Anmelden eines Prozesses bei CMX (attach process)

Seite 110:
Haben sich mehrere Prozesse fuer eine TS-Anwendung mit T_PASSIVE angemeldet, so erhaelt nur der erste Prozess Verbindungsanzeigen zugestellt. Eine Verteilung auf alle Prozesse einer TS-Anwendung wie in CMX erfolgt nicht.

Seite 111:
PCMX belegt keine Dateikennzahl.

Seite 116, Rueckgabewert:
T_NOTFIRST bedeutet bei Kommunikation ueber TCP/IP, dass der Port bereits belegt ist. Dem Prozess werden keine Verbindungsanzeigen zugestellt.

8.6.2 t_callback - Rueckrufroutine registrieren (callback)

Seite 121:
Die Rueckrufroutine steht in allen Unix oder Unix-aehnlichen Betriebssystemen zur Verfuegung und kann in PCMX ohne Einschraenkungen verwendet werden.

8.6.5 t_conrq - Verbindung anfordern (connection request)

t_optc1/t_optc3 t_timeout: Die Inaktiv-Zeit wird formal geprueft aber ignoriert.

8.6.6 t_conrs - Verbindungswunsch beantworten (connection response)

t_optc1/t_optc3 t_timeout: Die Inaktiv-Zeit wird formal geprueft aber ignoriert.

8.6.24 t_redin - Umgelenkte Verbindung annehmen (redirection indication)

In PCMX kann eine unerwuenschte Verbindung nicht dem urspruenglichen Prozess zurueckgegeben werden sondern muss mit t_disrq() abgebaut werden.

8.6.33 t_xdatgo - Vorrangdatenanzeige freigeben (expedited data go)

Nicht unterstuetzt.

8.6.34 t_xdatin - Vorrangdaten empfangen (expedited data indication)

Nicht unterstuetzt.

8.6.35 t_xdatrq - Vorrangdaten senden (expedited data request)

Nicht unterstuetzt.

8.6.36 t_xdatstop - Vorrangdatenanzeige sperren (expedited data stop)

Nicht unterstuetzt.