

Deutsch



FUJITSU Software

# openUTM V6.5

Meldungen, Test und Diagnose auf BS2000-Systemen

Benutzerhandbuch

Ausgabe Dezember 2017

## **Kritik... Anregungen... Korrekturen...**

Die Redaktion ist interessiert an Ihren Kommentaren zu diesem Handbuch. Ihre Rückmeldungen helfen uns, die Dokumentation zu optimieren und auf Ihre Wünsche und Bedürfnisse abzustimmen.

Sie können uns Ihre Kommentare per E-Mail an [manuals@ts.fujitsu.com](mailto:manuals@ts.fujitsu.com) senden.

## **Zertifizierte Dokumentation nach DIN EN ISO 9001:2008**

Um eine gleichbleibend hohe Qualität und Anwenderfreundlichkeit zu gewährleisten, wurde diese Dokumentation nach den Vorgaben eines Qualitätsmanagementsystems erstellt, welches die Forderungen der DIN EN ISO 9001:2008 erfüllt.

cognitas. Gesellschaft für Technik-Dokumentation mbH  
[www.cognitas.de](http://www.cognitas.de)

## **Copyright und Handelsmarken**

Copyright © 2017 Fujitsu Technology Solutions GmbH.

Alle Rechte vorbehalten.

Liefermöglichkeiten und technische Änderungen vorbehalten.

Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und/oder Warenzeichen der jeweiligen Hersteller.

---

# Inhalt

<b>1</b>	<b>Einleitung</b> . . . . .	<b>7</b>
<b>1.1</b>	<b>Zielgruppe und Konzept des Handbuchs</b> . . . . .	<b>9</b>
<b>1.2</b>	<b>Wegweiser durch die Dokumentation zu openUTM</b> . . . . .	<b>10</b>
1.2.1	openUTM-Dokumentation . . . . .	10
1.2.2	Dokumentation zum openSEAS-Produktumfeld . . . . .	14
1.2.3	Readme-Dateien . . . . .	15
<b>1.3</b>	<b>Änderungen in openUTM V6.5</b> . . . . .	<b>16</b>
1.3.1	Neue Server-Funktionen . . . . .	16
1.3.2	Entfallene Server-Funktionen . . . . .	23
1.3.3	Neue Client-Funktionen . . . . .	24
1.3.4	Neue Funktionen für openUTM WinAdmin . . . . .	25
1.3.5	Neue Funktionen für openUTM WebAdmin . . . . .	25
<b>1.4</b>	<b>Darstellungsmittel</b> . . . . .	<b>26</b>
<b>2</b>	<b>Test und Fehlerdiagnose</b> . . . . .	<b>29</b>
<b>2.1</b>	<b>Testen von UTM-Anwendungen</b> . . . . .	<b>29</b>
2.1.1	Vorbereitungen für das Testen im Dialog . . . . .	30
2.1.2	Starten der Anwendung und Eingabe von Test-Kommandos . . . . .	32
2.1.3	STXIT-Verhalten beim Testen im Dialog . . . . .	34
2.1.4	Fehlerhafte Zugriffe auf Adresse NULL testen . . . . .	34
2.1.5	Beispiel . . . . .	35
<b>2.2</b>	<b>Fehlerdiagnose</b> . . . . .	<b>40</b>
2.2.1	Returncodes an der Programmschnittstelle . . . . .	40
2.2.2	UTM-Meldungen bei Programmfehlern . . . . .	41
2.2.3	Adressierungsfehler . . . . .	43
2.2.4	Diagnose-Dump bei festgelegten Meldungen/Ereignissen . . . . .	44
2.2.5	Fehlerunterlagen erstellen . . . . .	47

2.2.6	Traces . . . . .	48
2.2.6.1	Protokollierung von KDCS-Aufrufen aus C/C++-Teilprogrammen . . . . .	48
2.2.6.2	BCAM-Trace in openUTM . . . . .	48
2.2.6.3	ADMI-Trace . . . . .	54
<b>3</b>	<b>Der UTM-Dump . . . . .</b>	<b>55</b>
<b>3.1</b>	<b>Die Dateien des UTM-Dumps . . . . .</b>	<b>56</b>
3.1.1	Reduzierung der Dump-Information über Parameter DUMP-CONTENT . . . . .	57
<b>3.2</b>	<b>Das Tool KDCDUMP . . . . .</b>	<b>59</b>
3.2.1	KDCDUMP starten . . . . .	60
3.2.2	KDCDUMP-Anweisungen . . . . .	62
	! Systemkommando eingeben . . . . .	63
	!! Zuletzt ausgeführtes Systemkommando wiederholen . . . . .	64
	AFIND Adresse im Dump suchen . . . . .	64
	Blätteranweisungen für die Dialogauswertung . . . . .	64
	DUMP UTM-Dump in den Speicher einlesen . . . . .	66
	EDT Editor aufrufen . . . . .	67
	END KDCDUMP beenden . . . . .	67
	FGG Alle Dateien einer FGG aufbereiten . . . . .	68
	FILE Einzelne Dump-Datei aufbereiten . . . . .	70
	HELP Hilfe zu KDCDUMP . . . . .	72
	LIST Tabellenausschnitt aufbereiten . . . . .	73
	REPFIL Informieren über REP-Datei . . . . .	75
	SFIND String suchen . . . . .	76
	SH und SYS KDCDUMP unterbrechen . . . . .	77
	SYSLST Protokollierung ein-/ausschalten . . . . .	77
	TABLE Tabelle anzeigen . . . . .	78
3.2.3	Meldungen von KDCDUMP . . . . .	80
<b>3.3</b>	<b>Inhalt des UTM-Dumps . . . . .</b>	<b>81</b>
3.3.1	Die Tabelle CONS_ENTRIES . . . . .	82
3.3.2	Der KDCROOT-Bereich . . . . .	83
3.3.2.1	Program Table . . . . .	86
3.3.2.2	Load Module Table . . . . .	87
3.3.2.3	UTM Diagarea . . . . .	88
3.3.2.4	DB Diagarea . . . . .	100
3.3.2.5	Administration DIAGAREA . . . . .	104
3.3.2.6	Administration USERAREA . . . . .	108
3.3.2.7	Kommunikationsbereich KB . . . . .	109
3.3.3	Summary information . . . . .	111

<b>3.4</b>	<b>Dump-Beispiele</b> . . . . .	<b>112</b>
3.4.1	Dump-Beispiel K326 . . . . .	112
3.4.2	Dump-Beispiel STXIT . . . . .	115
<b>4</b>	<b>UTM-Meldungswesen</b> . . . . .	<b>117</b>
<b>4.1</b>	<b>Meldungsmodul, Meldungsdefinitionsdatei</b> . . . . .	<b>119</b>
<b>4.2</b>	<b>Meldungsziele</b> . . . . .	<b>120</b>
4.2.1	Ausgabebform der Meldungen . . . . .	121
4.2.2	UTM-Meldungen an die BS2000-Konsole . . . . .	122
4.2.3	UTM-Meldungen an eine TS-Anwendung . . . . .	122
4.2.4	UTM-Meldungen an Benutzer-spezifische Meldungsziele . . . . .	123
4.2.5	UTM-Meldungen an MSGTAC . . . . .	123
<b>4.3</b>	<b>Gestaltung der Meldungs Ausgabe durch den Anwender</b> . . . . .	<b>124</b>
4.3.1	Meldungen in anderen Sprachen - das Tool KDCMTXT . . . . .	125
4.3.1.1	Ein- und Ausgaben . . . . .	126
4.3.1.2	Aufruf von KDCMTXT . . . . .	127
4.3.1.3	KDCMTXT-Steueranweisungen . . . . .	127
4.3.1.4	Protokoll von KDCMTXT . . . . .	131
4.3.2	Erstellen eines eigenen Meldungsmoduls mit KDCMMOD . . . . .	132
4.3.2.1	Ein- und Ausgaben von KDCMMOD . . . . .	133
4.3.2.2	Starten und Beenden von KDCMMOD . . . . .	134
4.3.2.3	Steueranweisungen von KDCMMOD . . . . .	134
<b>4.4</b>	<b>UTM-Protokolldatei SYSLOG</b> . . . . .	<b>143</b>
4.4.1	SYSLOG-Datei auswerten . . . . .	143
4.4.1.1	Das Tool KDCCSYSL - SYSLOG-Datei in SAM-Format umwandeln . . . . .	144
4.4.1.2	Das Tool KDCPSYSL - Meldungstexte einfügen . . . . .	146
4.4.1.3	Meldungen von KDCCSYSL und KDCPSYSL . . . . .	147
<b>4.5</b>	<b>Aufbau der UTM-Systemmeldungen</b> . . . . .	<b>148</b>
<b>5</b>	<b>UTM-Meldungen</b> . . . . .	<b>149</b>
<b>5.1</b>	<b>Meldungen des Transaktionsmonitors</b> . . . . .	<b>149</b>
<b>5.2</b>	<b>Meldungen des XAP-TP-Providers</b> . . . . .	<b>295</b>
5.2.1	Allgemeine Inserts der XAP-TP-Meldungen . . . . .	314
<b>5.3</b>	<b>Meldungen des Generierungstools KDCDEF</b> . . . . .	<b>323</b>
<b>5.4</b>	<b>Meldungen der UTM-Tools KDCCSYSL / KDCPSYSL</b> . . . . .	<b>331</b>

<b>5.5</b>	<b>Meldungen der UTM-Tools KDCMMOD / KDCMTXT</b> . . . . .	<b>332</b>
<b>5.6</b>	<b>Meldungen des UTM-Tools KCDUMP</b> . . . . .	<b>334</b>
<b>5.7</b>	<b>Meldungen des UTM-Tools KDCUPD</b> . . . . .	<b>338</b>
<b>5.8</b>	<b>Standard-Meldungsdefinitionsdatei</b> . . . . .	<b>340</b>
5.8.1	Konstanten der Standard-Meldungsdefinitionsdatei . . . . .	340
5.8.2	Inserts in Meldungen . . . . .	341
5.8.3	Ziele der UTM-Meldungen . . . . .	353
<b>6</b>	<b>Anhang</b> . . . . .	<b>367</b>
<hr/>		
<b>6.1</b>	<b>KDCS-Returncodes in KCRCCC</b> . . . . .	<b>367</b>
<b>6.2</b>	<b>Interner Returncode KCRCDC</b> . . . . .	<b>371</b>
<b>6.3</b>	<b>STXIT-Ereignisse</b> . . . . .	<b>396</b>
<b>6.4</b>	<b>Codes der CDUMP-Aufrufe von openUTM</b> . . . . .	<b>398</b>
<b>6.5</b>	<b>Fehlercodes im UTM-Dump</b> . . . . .	<b>407</b>
	<b>Fachwörter</b> . . . . .	<b>455</b>
<hr/>		
	<b>Abkürzungen</b> . . . . .	<b>495</b>
<hr/>		
	<b>Literatur</b> . . . . .	<b>501</b>
<hr/>		
	<b>Stichwörter</b> . . . . .	<b>509</b>
<hr/>		

---

# 1 Einleitung

Zur Unterstützung unternehmerischer Entscheidungen werden heutzutage riesige Datenmengen genutzt und analysiert. Die sehr großen und von einer Vielzahl vernetzter Nutzer, Geräten und "Dingen" bereitgestellten Daten-Volumina stellen die IT Infrastruktur vor große Herausforderungen. Diese Datenmengen werden auch weiterhin wachsen und erfordern dementsprechend auch einen effizienten Transport, Verarbeitung und Speicherung. Zudem müssen auch zahlreiche neue Applikationen kurzfristig integrierbar sein, um das Potential der Digitalisierung ausschöpfen zu können.

Dementsprechend sind wesentliche Anforderungen an eine moderne IT Infrastruktur

- die Möglichkeit einer flexiblen, schrittweisen Skalierbarkeit - ohne Begrenzungen für die vorhersehbare Zukunft,
- die Bereitstellung von Server-, Storage- und Netzwerk-Ressourcen mit Self-service Portalen und automatisierten Lösungen,
- die Möglichkeit für flexible Verrechnungs- Modelle, wie beispielsweise eine verbrauchsabhängige Preisgestaltung,
- die Integrierbarkeit weiterer Optionen für eine Anpassung entsprechend den individuellen Kundenbedürfnissen - wie beispielsweise unterschiedliche Security Anforderungen etc.

Fujitsu bietet zur Bewältigung dieser Herausforderungen ein umfangreiches Portfolio innovativer Enterprise Hardware, Software und Support Services im Umfeld unserer Enterprise Mainframe Plattformen an und ist damit Ihr

- verlässlicher Service Provider, der Sie langfristig, flexibel und innovativ beim Betrieb der Mainframe-basierten Kernanwendungen Ihres Geschäftes unterstützt,
- optimaler Partner für die gemeinsame Abdeckung der Anforderungen einer Digitalen Transformation und
- langfristiger Partner aufgrund hoher Investitionen in die kontinuierliche Anpassung moderner Schnittstellen, die eine moderne IT Landschaft mit AI-Anwendungen, Robotics, IoT etc erfordern.

Mit openUTM stellt Ihnen Fujitsu eine vielfach erprobte und bewährte Lösung aus dem Middleware-Bereich zur Verfügung.

Die High-End-Plattform für Transaktionsverarbeitung openUTM bietet eine Ablaufumgebung, die all diesen Anforderungen moderner unternehmenskritischer Anwendungen gewachsen ist, denn openUTM verbindet alle Standards und Vorteile von transaktionsorientierten Middleware-Plattformen und Message Queuing Systemen:

- Konsistenz der Daten und der Verarbeitung
- Hohe Verfügbarkeit der Anwendungen (nicht nur der Hardware)
- Hohen Durchsatz auch bei großen Benutzerzahlen, d.h. höchste Skalierbarkeit
- Flexibilität bezüglich Änderungen und Anpassungen des IT-Systems

Eine UTM-Anwendung kann auf einem einzelnen Rechner als stand-alone UTM-Anwendung oder auf mehreren Rechnern gleichzeitig als UTM-Cluster-Anwendung betrieben werden.

openUTM ist Teil des umfassenden Angebots von **openSEAS**. Gemeinsam mit der Oracle Fusion Middleware bietet openSEAS die komplette Funktionalität für Anwendungsinnovation und moderne Anwendungsentwicklung. Im Rahmen des Produktangebots **openSEAS** nutzen innovative Produkte die ausgereifte Technologie von openUTM:

- BeanConnect ist ein Adapter gemäß der Java EE Connector Architecture (JCA) von Oracle/Sun und bietet den standardisierten Anschluss von UTM-Anwendungen an Java EE Application Server. Dadurch können bewährte Legacy-Anwendungen in neue Geschäftsprozesse integriert werden.
- Mit WebTransactions steht in openSEAS ein Produkt zur Verfügung, welches es ermöglicht, bewährte Host-Anwendungen flexibel in neuen Geschäftsprozessen und modernen Einsatzszenarien zu nutzen. Bestehende UTM-Anwendungen können unverändert ins Web übernommen werden.



## 1.1 Zielgruppe und Konzept des Handbuchs

Das openUTM-Handbuch „Meldungen, Test und Diagnose auf BS2000-Systemen“ richtet sich an Anwender, Administratoren und Programmierer von UTM-Anwendungen.

Es beschreibt das Testen einer UTM-Anwendung auf BS2000-Systemen, den Aufbau des UTM-Dumps, das Verhalten im Fehlerfall sowie alle von openUTM ausgegebenen Meldungen und Returncodes.

In Kapitel 4 ist außerdem das Meldungswesen von openUTM beschrieben mit den Möglichkeiten, Meldungen Benutzer-spezifisch in verschiedenen Sprachen auszugeben oder die von openUTM ausgelieferten Meldungen Anwendungs-spezifisch zu verändern oder an andere Meldungsziele auszugeben.

Voraussetzung für das Verständnis dieses Handbuchs sind Kenntnisse von openUTM und des BS2000-Betriebssystems.

Für die Unix- und Windows-Betriebssysteme steht Ihnen das openUTM-Handbuch „Meldungen, Test und Diagnose auf Unix-, Linux- und Windows-Systemen“ zur Verfügung.

## 1.2 Wegweiser durch die Dokumentation zu openUTM

In diesem Abschnitt erhalten Sie einen Überblick über die Handbücher zu openUTM und zum Produktumfeld von openUTM.

### 1.2.1 openUTM-Dokumentation

Die openUTM-Dokumentation besteht aus Handbüchern, den Online-Hilfen für den grafischen Administrationsarbeitsplatz openUTM WinAdmin und das grafische Administrations-tool WebAdmin sowie einer Freigabemitteilung für jede Plattform, auf der openUTM freigegeben wird.

Es gibt Handbücher, die für alle Plattformen gültig sind, sowie Handbücher, die jeweils für BS2000-Systeme bzw. für Unix-, Linux- und Windows-Systeme gelten.

Sämtliche Handbücher sind als PDF-Datei im Internet verfügbar unter der Adresse

<http://manuals.ts.fujitsu.com>

Geben Sie dort in das Feld **Produktsuche** den Suchbegriff "openUTM V6.5" ein, um sich alle openUTM-Handbücher der Version 6.5 anzeigen zu lassen.

Die folgenden Abschnitte geben einen Aufgaben-bezogenen Überblick über die Dokumentation zu openUTM V6.5. Eine vollständige Liste der Dokumentation zu openUTM finden Sie im Literaturverzeichnis.

#### **Einführung und Überblick**

Das Handbuch **Konzepte und Funktionen** gibt einen zusammenhängenden Überblick über die wesentlichen Funktionen, Leistungen und Einsatzmöglichkeiten von openUTM. Es enthält alle Informationen, die Sie zum Planen des UTM-Einsatzes und zum Design einer UTM-Anwendung benötigen. Sie erfahren, was openUTM ist, wie man mit openUTM arbeitet und wie openUTM in die BS2000-, Unix-, Linux- und Windows-Plattformen eingebettet ist.

## Programmieren

- Zum Erstellen von Server-Anwendungen über die KDCS-Schnittstelle benötigen Sie das Handbuch **Anwendungen programmieren mit KDCS für COBOL, C und C++**, in dem die KDCS-Schnittstelle in der für COBOL, C und C++ gültigen Form beschrieben ist. Diese Schnittstelle umfasst sowohl die Basisfunktionen des universellen Transaktionsmonitors als auch die Aufrufe für verteilte Verarbeitung. Es wird auch die Zusammenarbeit mit Datenbanken beschrieben.
- Wollen Sie die X/Open-Schnittstellen nutzen, benötigen Sie das Handbuch **Anwendungen erstellen mit X/Open-Schnittstellen**. Es enthält die openUTM-spezifischen Ergänzungen zu den X/Open-Programmschnittstellen TX, CPI-C und XATMI sowie Hinweise zu Konfiguration und Betrieb von UTM-Anwendungen, die X/Open-Schnittstellen nutzen. Ergänzend dazu benötigen Sie die X/Open-CAE-Spezifikation für die jeweilige X/Open-Schnittstelle.
- Wenn Sie Daten auf Basis von XML austauschen wollen, benötigen Sie das Dokument **XML für openUTM**. Darin werden die C- und COBOL-Aufrufe beschrieben, die zum Bearbeiten von XML-Dokumenten benötigt werden.
- Für BS2000-Systeme gibt es Ergänzungsbände für die Programmiersprachen Assembler, Fortran, Pascal-XT und PL/1.

## Konfigurieren

Zur Definition von Konfigurationen steht Ihnen das Handbuch **Anwendungen generieren** zur Verfügung. Darin ist beschrieben, wie Sie mit Hilfe des UTM-Tools KDCDEF sowohl für eine stand-alone UTM-Anwendung als auch für eine UTM-Cluster-Anwendung

- die Konfiguration definieren,
- die KDCFILE erzeugen,
- und im Falle einer UTM-Cluster-Anwendung die UTM-Cluster-Dateien erzeugen.

Zusätzlich wird gezeigt, wie Sie wichtige Verwaltungs- und Benutzerdaten mit Hilfe des Tools KDCUPD in eine neue KDCFILE übertragen, z.B. beim Umstieg auf eine neue Version von openUTM oder nach Änderungen in der Konfiguration. Für eine UTM-Cluster-Anwendung wird außerdem gezeigt, wie Sie diese Daten mit Hilfe des Tools KDCUPD in die neuen UTM-Cluster-Dateien übertragen.

## Binden, Starten und Einsetzen

Um UTM-Anwendungen einsetzen zu können, benötigen Sie für das betreffende Betriebssystem (BS2000- bzw. Unix-, Linux- oder Windows-Systeme) das Handbuch **Einsatz von UTM-Anwendungen**.

Dort ist beschrieben, wie man ein UTM-Anwendungsprogramm bindet und startet, wie man sich bei einer UTM-Anwendung an- und abmeldet und wie man Anwendungsprogramme strukturiert und im laufenden Betrieb austauscht. Außerdem enthält es die UTM-Kommandos, die dem Terminal-Benutzer zur Verfügung stehen. Zudem wird ausführlich auf die Punkte eingegangen, die beim Betrieb von UTM-Cluster-Anwendungen zu beachten sind.

## Administrieren und Konfiguration dynamisch ändern

- Für das Administrieren von Anwendungen finden Sie die Beschreibung der Programm-schnittstelle zur Administration und die UTM-Administrationskommandos im Handbuch **Anwendungen administrieren**. Es informiert über die Erstellung eigener Administrationsprogramme für den Betrieb einer stand-alone UTM-Anwendung oder einer UTM-Cluster-Anwendung sowie über die Möglichkeiten, mehrere UTM-Anwendungen zentral zu administrieren. Darüber hinaus beschreibt es, wie Sie Message Queues und Drucker mit Hilfe der KDCS-Aufrufe DADM und PADM administrieren können.
- Wenn Sie den grafischen Administrationsarbeitsplatz **openUTM WinAdmin** oder die funktional vergleichbare Web-Anwendung **openUTM WebAdmin** einsetzen, dann steht Ihnen folgende Dokumentation zur Verfügung:
  - Die **WinAdmin-Beschreibung** und die **WebAdmin-Beschreibung** bieten einen umfassenden Überblick über den Funktionsumfang und das Handling von WinAdmin/WebAdmin. Die Dokumente werden jeweils mit der Software ausgeliefert und sind zusätzlich auch online als PDF-Datei verfügbar.
  - Das jeweilige **Online-Hilfesystem** beschreibt kontextsensitiv alle Dialogfelder und die zugehörigen Parameter, die die grafische Oberfläche bietet. Außerdem wird dargestellt, wie man WinAdmin bzw. WebAdmin konfiguriert, um stand-alone UTM-Anwendungen und UTM-Cluster-Anwendungen administrieren zu können.



Details zur Integration von openUTM WebAdmin in den SE Manager des SE Servers finden Sie im SE Server Handbuch **Bedienen und Verwalten**.

## Testen und Fehler diagnostizieren

Für die o.g. Aufgaben benötigen Sie außerdem die Handbücher **Meldungen, Test und Diagnose** (jeweils ein Handbuch für Unix-, Linux- und Windows-Systeme und für BS2000-Systeme). Sie beschreiben das Testen einer UTM-Anwendung, den Inhalt und die

Auswertung eines UTM-Dumps, das Verhalten von openUTM im Fehlerfall, das Meldungsverhalten von openUTM, sowie alle von openUTM ausgegebenen Meldungen und Return-codes.

### **openUTM-Clients erstellen**

Wenn Sie Client-Anwendungen für die Kommunikation mit UTM-Anwendungen erstellen wollen, stehen Ihnen folgende Handbücher zur Verfügung:

- Das Handbuch **openUTM-Client für Trägersystem UPIC** beschreibt Erstellung und Einsatz von Client-Anwendungen, die auf UPIC basieren. Neben der Beschreibung der Schnittstellen CPI-C und XATMI erhalten Sie Informationen, wie Sie die C++-Klassen für die schnelle und einfache Programmerstellung nutzen können.
- Das Handbuch **openUTM-Client für Trägersystem OpenCPIC** beschreibt, wie man OpenCPIC installiert und konfiguriert. Es zeigt auf, was beim Programmieren einer CPI-C-Anwendung zu beachten ist und welche Einschränkungen es gegenüber der Programmschnittstelle X/Open CPI-C gibt.
- Für das mit **BeanConnect** ausgelieferte Produkt **openUTM-JConnect** wird die Dokumentation mit der Software ausgeliefert. Diese Dokumentation besteht aus Word- und PDF-Dateien, die die Einführung und die Installation beschreiben, sowie aus einer Java-Dokumentation mit der Beschreibung der Java-Klassen.
- Das Handbuch **BizXML2Cobol** beschreibt, wie Sie bestehende Cobol-Programme einer UTM-Anwendung so erweitern können, dass sie als Standard-Web-Service auf XML-Basis genutzt werden können. Die Arbeit mit der grafischen Bedienoberfläche ist in der zugehörigen **Online-Hilfe** beschrieben.
- Wenn Sie UTM-Services auf einfache Weise ins Web stellen möchten, benötigen Sie das Handbuch **Web-Services für openUTM**. Das Handbuch beschreibt, wie Sie mit dem Software-Produkt WS4UTM (WebServices for openUTM) Services von UTM-Anwendungen als Web Services verfügbar machen. Die Arbeit mit der grafischen Bedienoberfläche ist in der zugehörigen **Online-Hilfe** beschrieben.

### **Kopplung mit der IBM-Welt**

Wenn Sie aus Ihrer UTM-Anwendung mit Transaktionssystemen von IBM kommunizieren wollen, benötigen Sie außerdem das Handbuch **Verteilte Transaktionsverarbeitung zwischen openUTM und CICS-, IMS- und LU6.2-Anwendungen**. Es beschreibt die CICS-Kommandos, IMS-Makros und UTM-Aufrufe, die für die Kopplung von UTM-Anwendungen mit CICS- und IMS-Anwendungen benötigt werden. Die Kopplungsmöglichkeiten werden anhand ausführlicher Konfigurations- und Generierungsbeispiele erläutert. Außerdem beschreibt es die Kommunikation über openUTM-LU62, sowie dessen Installation, Generierung und Administration.

## Dokumentation zu PCMX

Mit openUTM auf Unix-, Linux- und Windows-Systemen wird die Kommunikationskomponente PCMX ausgeliefert. Die Funktionen von PCMX sind in folgenden Dokumenten beschrieben:

- Handbuch CMX (Unix-Systeme) "Betrieb und Administration" für Unix- und Linux-Systeme
- Online-Hilfe zu PCMX für Windows-Systeme

## 1.2.2 Dokumentation zum openSEAS-Produktumfeld

Die Verbindung von openUTM zum openSEAS-Produktumfeld wird im openUTM-Handbuch **Konzepte und Funktionen** kurz dargestellt. Die folgenden Abschnitte zeigen, welche der openSEAS-Dokumentationen für openUTM von Bedeutung sind.

### Integration von Java EE Application Servern und UTM-Anwendungen

Der Adapter BeanConnect gehört zur Produkt-Suite openSEAS. Der BeanConnect-Adapter realisiert die Verknüpfung zwischen klassischen Transaktionsmonitoren und Java EE Application Servern und ermöglicht damit die effiziente Integration von Legacy-Anwendungen in Java-Anwendungen.

- Das Handbuch **BeanConnect** beschreibt das Produkt BeanConnect, das einen JCA 1.5- und JCA 1.6-konformen Adapter bietet, der UTM-Anwendungen mit Anwendungen auf Basis von Java EE, z.B. mit dem Application Server von Oracle, verbindet.

Die Handbücher zum Application Server von Oracle sind bei Oracle beziehbar.

### Web-Anbindung und Anwendungsintegration

Zum Anschließen neuer und bestehender UTM-Anwendungen an das Web mit dem Produkt WebTransactions benötigen Sie die Handbücher zu **WebTransactions**.

Die Dokumentation wird durch JavaDocs ergänzt.

### 1.2.3 Readme-Dateien

Funktionelle Änderungen und Nachträge der aktuellen Produktversion zu diesem Handbuch entnehmen Sie bitte ggf. den Produkt-spezifischen Readme-Dateien.

Readme-Dateien stehen Ihnen online bei dem jeweiligen Produkt zusätzlich zu den Produkthandbüchern unter <http://manuals.ts.fujitsu.com> zur Verfügung. Für die Plattform BS2000 finden Sie Readme-Dateien auch auf der Softbook-DVD.

#### *Informationen auf BS2000-Systemen*

Wenn für eine Produktversion eine Readme-Datei existiert, finden Sie auf BS2000-Systemen die folgende Datei:

```
SYSRME.<product>.<version>.<lang>
```

Diese Datei enthält eine kurze Information zur Readme-Datei in deutscher oder englischer Sprache (<lang>=D/E). Die Information können Sie am Bildschirm mit dem Kommando /SHOW-FILE oder mit einem Editor ansehen.

Das Kommando /SHOW-INSTALLATION-PATH INSTALLATION-UNIT=<product> zeigt, unter welcher Benutzerkennung die Dateien des Produkts abgelegt sind.

#### *Readme-Datei auf Unix- und Linux-Systemen*

Die Readme-Datei und ggf. weitere Dateien wie z.B. eine Handbuchergänzungsdatei finden Sie im *utmpfad* unter */docs/sprache*.

#### *Readme-Datei auf Windows-Systemen*

Die Readme-Datei und ggf. weitere Dateien wie z.B. eine Handbuchergänzungsdatei finden Sie im *utmpfad* unter *\Docs\sprache*.

#### *Ergänzende Produkt-Informationen*

Aktuelle Informationen, Versions-, Hardware-Abhängigkeiten und Hinweise für Installation und Einsatz einer Produktversion enthält die zugehörige Freigabemitteilung. Solche Freigabemitteilungen finden Sie online unter <http://manuals.ts.fujitsu.com>.

## 1.3 Änderungen in openUTM V6.5

Die folgenden Abschnitte gehen näher auf die Änderungen in den einzelnen Funktionsbereichen ein.

### 1.3.1 Neue Server-Funktionen

#### Lange Rechnernamen bis 64 Zeichen

openUTM unterstützt Rechnernamen, die bis zu 64 Zeichen lang sein können. Damit ist es nicht mehr notwendig, den Namen aus der UTM-Generierung mittels Konfigurations- oder Mapping-Dateien auf einen realen langen Namen abzubilden.

Dazu wurden folgende Schnittstellen geändert:

- Generierung

*Alle Systeme:*

- KDCDEF-Anweisungen CON, OSI-CON, PTERM und TPOOL:  
Bei PRONAM= bzw. OSI-CON NETWORK-SELECTOR= kann ein bis zu 64 Zeichen langer Name für den Partner-Rechner angegeben werden.

*BS2000-Systeme:*

- KDCDEF-Anweisungen CON:  
Der Operand LISTENER-PORT= kann auch auf BS2000-Systemen angegeben werden.

*Unix-, Linux- und Windows-Systeme:*

- KDCDEF-Anweisung CLUSTER-NODE:  
Bei HOSTNAME= und VIRTUAL-HOST= kann ein bis zu 64 Zeichen langer Name angegeben werden.
- KDCDEF-Anweisung MAX:  
Bei HOSTNAME= kann ein bis zu 64 Zeichen langer Name angegeben werden.

- Programmierung

- KDCS-Aufruf INFO SI:
  - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 180 angegeben werden.
  - Neue Felder KCHSTNML und KCPRONML für die Rückgabe langer Rechnernamen.



- KDCS-Aufruf INIT PU:
  - Die Version der Schnittstelle wurde auf 6 erhöht.
  - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLI der Wert 356 angegeben werden.
  - Neue Felder KCHSTNML und KCPRONML für die Rückgabe der langen Rechnernamen.
- KDCS-Aufruf SIGN ST:
  - Die Version der Schnittstelle wurde auf 4 erhöht.
  - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 180 angegeben werden.
  - Der Knotenname im Feld KCCLNODE kann bis zu 64 Zeichen lang sein.
- Administrationsschnittstelle KDCADMI
  - Neue Struktur *kc\_long\_triple\_str* im Identifikationsbereich für die Unterstützung langer Rechnernamen. Diese Struktur muss für alle Objekte außer KC\_MUX verwendet werden. Daher wurde auch die Datenstrukturversion von KDCADMI geändert auf Version 10 (Feld *version\_data* im Parameterbereich).
  - Neue Felder für die langen Rechnernamen:

Datenstruktur	Feldname(n)
<i>kc_cluster_node_str</i>	<i>hostname_long</i> und <i>virtual_host_long</i>
<i>kc_con_str</i> <i>kc_lses_str</i> <i>kc_lterm_str</i> <i>kc_pterm_str</i> <i>kc_tpool_str</i>	<i>prname_long</i>
<i>kc_osi_con_str</i>	<i>network_selector_long</i>
<i>kc_max_par_str</i> <i>kc_system_par_str</i>	<i>hostname_long</i>

- Kommando-Schnittstelle KDCADM
  - Kommando KDCINF:
 

Wenn ein Objekt vom Typ CON, LSES, LTERM, OSI-CON, PTERM oder TPOOL einen langen Rechnernamen enthält, dann wird die Information in zwei Bildschirmzeilen ausgegeben.
  - Kommandos KDCLSES, KDCPOOL, KDCPTERM und KDCSWTCH:
 

Wenn ein Objekt einen langen Rechnernamen enthält, dann wird die Information in zwei Bildschirmzeilen ausgegeben.

- Meldungen  
Bei jeder Namensumsetzung zwischen langem Rechnernamen und kurzem lokalen Namen wird die Meldung K037 ausgegeben.
- Tool KDCNAMEINFO (Unix-, Linux- und Windows-Systeme)  
Für die Zuordnung IP-Adresse zu Rechnernamen steht das Tool KDCNAMEINFO zur Verfügung.

### Lademodul-Version \*HIGHEST-EXISTING (BS2000-Systeme)

Eine UTM-Anwendung auf einem BS2000-System kann so generiert werden, dass jeweils die höchste in der Bibliothek verfügbare Version des Lademoduls geladen wird, d.h. die Version muss nicht mehr explizit angegeben werden.

Dazu wurden folgende Schnittstellen geändert:

- Generierung  
KDCDEF-Anweisung LOAD-MODULE:  
Neue Operandenwerte VERSION = \*HIGHEST-EXISTING und \*UPPER-LIMIT.  
\*UPPER-LIMIT ist synonym für den bisher schon möglichen Wert VERSION=@.
- Administrationsschnittstelle KDCADMI  
Datenstruktur *kc\_load\_module\_str*:
  - Feld *version*: Neue Werte \*HIGHEST-EXISTING und \*UPPER-LIMIT. Diese können beim Modifizieren mit KC\_MODIFY\_OBJECT angegeben werden.
  - Neues Feld *version\_gen* für die generierte Version des Lademoduls bzw. Shared Objects.

**i**

 Das Feld *version\_gen* wird auch auf Unix-, Linux- und Windows-Systemen versorgt.
- Kommando-Schnittstelle KDCADM
  - Kommando KDCINF:  
Beim Objekt-Typ LOAD-MODULE wird in VERSION (GENERATED) die generierte Version des Lademoduls ausgegeben.
  - Kommando KDCPROG:  
Beim Parameter VERSION kann auch \*HIGHEST-EXISTING und \*UPPER-LIMIT angegeben werden,

## Passwortlänge bis 16 Zeichen

Das Passwort einer UTM-Benutzerkennung kann bis zu 16 Zeichen lang sein.

- Generierung
  - KDCDEF-Anweisung USER:  
Bei PASS= kann ein bis zu 16 Zeichen langes Passwort angegeben werden.  
Bei PROTECT-PW= wurde der Maximalwert für die Mindestlänge auf 16 erhöht.
- Programmierung
  - KDCS-Aufruf SIGN ON, CK:  
Im Feld KCLA muss der Wert 16 angegeben werden.
  - KDCS-Aufruf SIGN CP:  
Im Feld KCLA muss der Wert 32 angegeben werden.
  - KDCS-Aufruf SIGN ST:
    - Die Version der Schnittstelle wurde auf 4 erhöht.
    - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 120 angegeben werden.
    - Neues Feld KCRPASSL für die Rückgabe des Passworts aus dem UPIC-Protokoll.
- Administrationsschnittstelle KDCADMI
  - Datenstruktur *kc\_user\_str* und *kc\_user\_dyn2\_str*:  
Neue Felder *password16*, *protect\_pw16\_lth* und *pw\_encrypted*.
- Kommando-Schnittstelle KDCADM
  - Kommando KDCUSER  
Bei PASS= kann ein bis zu 16 Zeichen langes Passwort angegeben werden.
- Programm CALLUTM (BS2000-Systeme)
  - Anweisung CREATE-CONFIGURATION  
Bei PASSWORD= kann ein bis zu 16 Zeichen langes Passwort angegeben werden.

## Dead Letter Queue für LPAP und OSI-LPAP

Nachrichten an LU6.1- oder OSI-TP-Partner, die gelöscht werden, weil sie wegen eines permanenten Fehlers nicht zugestellt werden konnten, können jetzt in der Dead Letter Queue gesichert werden.

Dazu wurden die Schnittstellen wie folgt geändert:

- Generierung
  - KDCDEF-Anweisungen LPAP und OSI-LPAP:  
Der neue Operand DEAD-LETTER-Q= steuert, ob nicht zustellbare Nachrichten, die aus ihrer Message Queue gelöscht werden, in der Dead Letter Queue gesichert werden sollen.
- Programmierung
  - Beim KDCS-Aufruf DADM mit KCOM=MV oder MA (Verschieben von Nachrichten aus der Dead Letter Queue) kann auch ein LPAP- oder OSI-LPAP-Partner angegeben werden.
- Administrationsschnittstelle KDCADMI
  - Datenstrukturen *kc\_lpap\_str* und *kc\_osi\_lpap\_str*: Neues Feld *dead\_letter\_q*, um die Sicherung nicht zustellbarer Nachrichten in der Dead Letter Queue zu steuern.

### Code-Konvertierungstabellen

Für die Code-Konvertierung zwischen ASCII-kompatiblen Codes und EBCDIC-Codes stellt openUTM auf allen Plattformen Tabellen für vier Code-Konvertierungen bereit (zwei 8-Bit-Konvertierungen und zwei 7-Bit-Konvertierungen). Standardmäßig verwendet openUTM auf BS2000-, Unix- und Linux-Systemen eine 8-Bit-Konvertierung zwischen ISO8859-i und EBCDIC.DF.04-i und auf Windows-Systemen eine 8-Bit-Konvertierung zwischen Windows-1252 und EBCDIC.DF.04-F.

Dazu wurden folgende Schnittstellen geändert:

- Generierung
  - KDCDEF-Anweisungen PTERM und TPOOL:  
Die Operandenwerte MAP=SYS1, SYS2, SYS3 und SYS4 können auch auf Unix-, Linux- und Windows-Systemen angegeben werden. Die Angabe ist nur für Transportsystem-Anwendungen erlaubt (Partner-Typ APPLI oder SOCKET).
  - KDCDEF-Anweisungen OSI-CON und SESCHA:  
Neue Operandenwerte MAP=SYS1, SYS2, SYS3 und SYS4 auf Unix-, Linux- und Windows-Systemen für die vier Konvertierungen.
- Programmierung
  - Die Sourcen mit den Konvertierungs-Tabellen werden auf allen Plattformen ausgeliefert (BS2000: Assembler Modul KDCEA; Unix, Linux- und Windows-Systeme: C-Source *kcsaeaa.c*) und können auf allen Plattformen angepasst oder durch eigene Tabellen ersetzt werden.

- Administrationsschnittstelle KDCADMI
  - Datenstrukturen *kc\_pterm\_str* und *kc\_tpool\_str*:  
Für das Feld *map* gibt es auf Unix, Linux- und Windows-Systemen die zusätzlichen Werte '1', '2', '3' und '4' (für die vier Konvertierungen). Die Angabe ist nur für Transportsystem-Anwendungen erlaubt (Partner-Typ APPLI oder SOCKET).
  - Datenstrukturen *kc\_lpap\_str* und *kc\_osi\_con\_str*:  
Für das Feld *map* gibt es auf Unix, Linux- und Windows-Systemen die zusätzlichen Werte '1', '2', '3' und '4' (für die vier Konvertierungen).

### Anwenderspezifische Fehlerbehandlung (Unix- und Linux-Systeme)

Auf Unix- und Linux-Systemen kann eine eigene User Signal Routine erstellt werden. Diese Routine wird aufgerufen, sobald ein Signal aufgetreten ist. Sie ersetzt die Standard-Fehlerbehandlung von openUTM (abnormales Vorgangsende + PENDER-Dump), d.h. ein Vorgang kann damit trotz Fehler fortgesetzt werden.

Dazu wurden folgende Schnittstellen geändert:

- Programmierung
  - Neue Funktionen *KCX\_REG\_SIGNAL\_HANDLER* und *KCX\_UN\_REG\_SIGNAL\_HANDLER* zum Registrieren und De-Registrieren der Signal Routine bei openUTM.
  - Neue Funktionen *KCX\_SET\_RELOAD\_FLAG* zum Austausch eines Workprozesses nach Transaktionsende und *KCX\_WRITE\_DUMP* zum Erstellen eines UTM-Dumps ohne Vorgangsabbruch.

Die Funktionen stehen in COBOL und C zur Verfügung.

### Zugangsdaten für den XA-Datenbank-Anschluss

- Die Zugangsdaten für den XA-Datenbank-Anschluss können (wieder) in den Startparametern angegeben werden.
- Der Benutzername und das Passwort für den XA-Datenbank-Anschluss können per Administration (KDCADMI) geändert werden:
  - Operationscode *KC\_MODIFY\_OBJECT*:
    - Neuer Objekt-Typ *KC\_DB\_INFO*
    - Neue Werte *KC\_IMMEDIATE* und *KC\_DELAY* im Feld *subopcode1*, um den Zeitpunkt für die Passwortänderung zu steuern.
  - Datenstruktur *kc\_db\_info\_str*: Neue Felder *db\_userid* und *db\_password*.

- Die Zugangsdaten für den XA-Datenbank-Anschluss können bei KDCUPD übertragen werden:
  - TRANSFER-Anweisung: Neuer Operand DB-CREDENTIALS=.

### KDCUPD bei geänderter Anzahl Datenbanken

KDCUPD überträgt alle Daten auch dann, wenn sich die Anzahl der Datenbanken in der neuen Generierung erhöht. Voraussetzung ist, dass sich die openUTM-Version nicht ändert und dass die Reihenfolge der bisherigen Datenbanken in der Generierung unverändert bleibt.

### Multi-DB auf BS2000-Systemen

Auf BS2000-Systemen können bis zu drei Datenbanken generiert werden.

### Umhängen wartender Asynchron-Nachrichten für LTERM-, LPAP- und OSI-LPAP-Bündel

Bei LTERM-, LPAP- und OSI-LPAP-Bündeln können wartende Asynchron-Nachrichten für Slave-LTERMS, Slave-LPAPs oder Slave-OSI-LPAPs, nach Ablauf der Wartezeit automatisch an einen Slave mit aufgebauter Verbindung umgehängt werden.

- Generierung, KDCDEF-Anweisung MAX: Neuer Operand MOVE-BUNDLE-MSGS=, um das automatische Umhängen zu steuern.
- Administrationsschnittstelle KDCADMI, Datenstruktur *kc\_max\_par\_str*: Neues Feld *move\_bundle\_msgs* für die Ausgabe des generierten Wertes.

### Sonstige Änderungen

- Ausgabe der Generierungszeit an der Administrationsschnittstelle:
  - KDCADMI-Datenstruktur *kc\_curr\_par\_str*: Neue Felder *gen\_date\_xxx* und *gen\_time\_xxx* zur Ausgabe von Datum und Zeitpunkt der Generierung.
  - Kommando KDCINF STAT: Zusätzliche Ausgabe von Datum und Zeitpunkt der Generierung.
- Meldung bei Verlust einer Nachricht an OSI TP- und LU6.1-Partner  
Wird eine Nachricht an einen OSI TP- und LU6.1-Partner gelöscht, weil sie wegen eines permanenten Fehlers nicht zugestellt werden konnte, dann wird die neue Meldung K239 ausgegeben.
- Verbesserte Performance bei der Kommunikation zwischen UPIC-Clients und UTM-Server-Anwendungen.

- Modifizieren von UTM-Meldungen auf Unix-, Linux- und Windows-Systemen
  - Meldungstexte von K- und P-Meldungen sind auch auf Windows modifizierbar.  
Dazu werden das UTM-Tool KDCMMOD sowie das neue UTM-Tool KDCWMSG verwendet.
  - Meldungstexte und -ziele von U-Meldungen sind auf Unix-, Linux- und Windows-Systemen modifizierbar. Damit können U-Meldungen auch in die System-Protokoll-datei SYSLOG ausgegeben werden.  
Das bisher für das Ändern von K- und P-Meldungen beschriebene Verfahren kann jetzt auch für U-Meldungen angewendet werden.
- Der Portnummernbereich für Unix-, Linux- und Windows-Systeme ist nicht mehr beschränkt, d.h. es können alle Portnummern von 1 bis 65535 angegeben werden.
- Bei CON-Anweisungen sowie bei PTERM- und TPOOL-Anweisungen mit PTYP=APPLI/UPIC-R/SOCKET ist für Unix-, Linux- und Windows-Systeme die Angabe des Parameters PRONAM= jetzt Pflicht. Das gleiche gilt für den Parameter NETWORK-SELECTOR der OSI-CON-Anweisung.
- Das UTM-Tool KDCEVAL wurde wie folgt erweitert:
  - Neuer Operand SHOW-TSN im KDCEVAL-Kommando OPTION, um die wiederholte Ausgabe der TSN bzw. der PID in der Liste TRACE2 zu aktivieren oder zu deaktivieren.
  - Zusätzliche Ausgaben im Listenkopf wie z.B. Appli-Mode oder Test-Mode.
- Verschlüsselung
  - BS2000-Systeme:  
Es wird die Verschlüsselungs-Funktionalität des BS2000-Systems verwendet.
  - Unix-, Linux- und Windows-Systeme:  
Es wird die Funktionalität der OpenSSL Bibliothek verwendet.

### 1.3.2 Entfallene Server-Funktionen

Mehrere Funktionen wurden gestrichen und können nicht mehr in KDCDEF generiert werden. Wenn sie dennoch angegeben werden, wird dies im KDCDEF-Lauf mit einem Syntaxfehler abgelehnt. Im Einzelnen wurde Folgendes gestrichen:

- *BS2000-Systeme*
  - KDCDEF-Anweisung USER:  
Operanden CERTIFICATE= und CERTIFICATE-AUTHORITY= für die Zuordnung von Zertifikaten.

- KDCDEF-Anweisung DATABASE:  
Operandenwert TYPE=PRISMA für das Datenbank-System PRISMA
- *Unix-, Linux- und Windows-Systeme*
  - Die Umwandlungsdatei zwischen kurzen und langen Rechnernamen (Mapping-Funktion) und die Umgebungsvariable UTM\_NET\_HOSTNAME werden nicht mehr unterstützt, da openUTM standardmäßig Rechnernamen bis zur Länge von 64 Zeichen unterstützt.
  - KDCDEF-Anweisung MAX:  
Operand NET-ACCESS= für die Einstellung der Netzanbindung (single-/multi-threaded). Es wird immer die Multi-threaded Netzanbindung verwendet.
  - KDCDEF-Anweisung FORMSYS für ein Formatierungssystem
  - KDCDEF-Anweisung OPTION:  
Operand CHECKTNS= für den Abgleich der UTM-Generierung mit der TNS-Generierung

### 1.3.3 Neue Client-Funktionen

#### UPIC-Routing

Im Programm oder per statischer Konfiguration in der `upicfile` kann eine Liste von Partner-Anwendungen angegeben werden. Damit lässt sich z.B. eine flexible Adressierung oder eine Lastverteilung realisieren. Bisher wurde eine solche Funktionalität nur für die statische Konfiguration einer UTM-Cluster-Anwendung angeboten.

Dazu wurden folgende Schnittstellen geändert:

- `upicfile`: Neues Kennzeichen ND, mit der eine Liste von stand-alone UTM-Anwendungen definiert werden kann.
- CPI-C-Schnittstelle:
  - Neuer Aufruf `Set_Partner_Index`, um den Index für einen Listen-Eintrag für eine Partner-Anwendung festzulegen. Die Partner-Anwendung muss anschließend mit `Set_Partner_xxx`-Aufrufen definiert werden. Danach können weitere Partner-Anwendungen in die Liste eingetragen werden.
  - Neuer Aufruf `Get_Max_Partner_Index`, um den maximalen Index der Liste von Partner-Anwendungen abzufragen.



## Sonstige Änderungen

- 64-Bit-Unterstützung von openUTM-Client für die Plattformen AIX und HP-Itanium.
- Unterstützung von langen Passwörtern für UTM-Benutzerkennungen  
Dazu wurden folgende Schnittstellen geändert:
  - CPI-C-Aufrufe *Set\_Conversation\_Security\_Password* und *Set\_Conversation\_Security\_New\_Password*:  
Sowohl das bestehende Passwort als auch das neue Passwort für die UTM-Benutzerkennung können bis zu 16 Zeichen lang sein.
  - XATMI-Aufruf *tpinit*:  
Das Passwort für die UTM-Benutzerkennung kann bis zu 16 Zeichen lang sein.
- Code-Konvertierungstabellen  
Für die Code-Konvertierung zwischen ASCII-kompatiblen Codes und EBCDIC-Codes auf Client-Seite stellt openUTM-Client auf allen Plattformen Tabellen für vier Code-Konvertierungen bereit (zwei 8-Bit-Konvertierungen und zwei 7-Bit-Konvertierungen). Standardmäßig wird auf BS2000-, Unix- und Linux-Systemen eine 8-Bit-Konvertierung zwischen ISO8859-i und EBCDIC.DF.04-i und auf Windows-Systemen eine 8-Bit-Konvertierung zwischen Windows-1252 und EBCDIC.DF.04-F verwendet.  
Die Tabellen können angepasst werden. Es kann in einer Client-Anwendung immer nur eine Tabelle verwendet werden.
- Verschlüsselung
  - BS2000-Systeme:  
Es wird die Verschlüsselungs-Funktionalität des BS2000-Systems verwendet.
  - Unix-, Linux- und Windows-Systeme:  
Es wird die Funktionalität der OpenSSL Bibliothek verwendet.

### 1.3.4 Neue Funktionen für openUTM WinAdmin

WinAdmin unterstützt alle Neuerungen der openUTM V6.5 bzgl. der Programmschnittstelle zur Administration sowie die Erweiterung des DADM-Aufrufs an der KDCS-Schnittstelle.

### 1.3.5 Neue Funktionen für openUTM WebAdmin

WebAdmin unterstützt alle Neuerungen der openUTM V6.5 bzgl. der Programmschnittstelle zur Administration sowie die Erweiterung des DADM-Aufrufs an der KDCS-Schnittstelle.

## 1.4 Darstellungsmittel

### Metasyntax

Die in diesem Handbuch verwendete Metasyntax können Sie der folgenden Tabelle entnehmen:

Formale Darstellung	Erläuterung	Beispiel
GROSSBUCHSTABEN	Großbuchstaben bezeichnen Konstanten (Namen von Aufrufen, Anweisungen, Feldnamen, Kommandos und Operanden etc.), die in dieser Form anzugeben sind.	LOAD-MODE=STARTUP
kleinbuchstaben	In Kleinbuchstaben sind in Syntaxdiagrammen und Operandenbeschreibung die Platzhalter für Operandenwerte dargestellt.	KDCFILE=filebase
<i>kleinbuchstaben</i>	Im Fließtext werden Variablen sowie Namen von Datenstrukturen und Feldern in kursiven Kleinbuchstaben dargestellt.	<i>utm-installationsverzeichnis</i> ist das UTM-Installationsverzeichnis
Schreibmaschinenschrift	In Schreibmaschinenschrift werden im Fließtext Kommandos, Dateinamen, Meldungen und Beispiele ausgezeichnet, die in genau dieser Form eingegeben werden müssen bzw. die genau diesen Namen oder diese Form besitzen.	Der Aufruf <code>tpcall</code>
{ } und	In geschweiften Klammern stehen alternative Angaben, von denen Sie eine auswählen müssen. Die zur Verfügung stehenden Alternativen werden jeweils durch einen Strich getrennt aufgelistet.	STATUS={ ON   OFF }
[ ]	In eckigen Klammern stehen wahlfreie Angaben, die entfallen können.	KDCFILE=( filebase [ , { SINGLE  DOUBLE} ] )
( )	Kann für einen Operanden eine Liste von Parametern angegeben werden, sind diese in runde Klammern einzuschließen und durch Kommata zu trennen. Wird nur ein Parameter angegeben, kann auf die Klammern verzichtet werden.	KEYS=(key1,key2,...keyn)
<u>Unterstreichen</u>	Unterstreichen kennzeichnet den Standardwert.	CONNECT= { YES   <u>NO</u> }
<b>Kurzform</b>	Die Standardkurzform für Anweisungen, Operanden und Operandenwerte wird „fett“ hervorgehoben. Die Kurzform kann alternativ angegeben werden.	TRANSPORT- <b>SELECTOR</b> =c‘C‘

Formale Darstellung	Erläuterung	Beispiel
...	Punkte zeigen die Wiederholbarkeit einer syntaktischen Einheit an. Außerdem kennzeichnen die Punkte Ausschnitte aus einem Programm, einer Syntaxbeschreibung o.ä.	KDCDEF starten ... OPTION DATA=statement_file ... END

### Sonstige Symbole



für Hinweistexte.



für Warnhinweise.



---

## 2 Test und Fehlerdiagnose

In diesem Kapitel erfahren Sie

- wie Sie eine UTM-Anwendung testen
- wie Sie Returncodes und Meldungen zur Diagnose verwenden
- welche Traces Sie zur Diagnose einsetzen können.

### 2.1 Testen von UTM-Anwendungen

Eine UTM-Anwendung können Sie interaktiv im Dialog (Teilnehmerbetrieb des BS2000-Systems, TIAM) testen. Dazu können Sie die Testhilfen des Betriebssystems und der jeweiligen Compiler nutzen, siehe insbesondere Sprach-spezifische AID-Benutzerhandbücher.

Eine komfortable Möglichkeit zum Testen Ihrer UTM-Anwendung bietet außerdem die FUJITSU Software BS2000 BS2IDE – Eclipse-based Integrated Development Environment for BS2000, siehe auch Webseite <https://bs2000.ts.fujitsu.com/bs2ide/>.

Das Testen im Dialog bietet Ihnen umfassende Möglichkeiten:

- alle UTM-Funktionsbereiche können getestet werden, insbesondere auch Funktionen zur verteilten Verarbeitung
- alle Partnerverbindungen können in den Test einbezogen werden
- der Test kann sowohl im Single-Task als auch im Multi-Task-Betrieb erfolgen
- Druckausgaben können getestet werden
- Dialog-Ein-/Ausgaben für die BS2000-Testhilfe und Ein-/Ausgaben für die UTM-Anwendung werden in unterschiedlichen Fenstern (bzw. Terminals) abgewickelt und stören sich nicht gegenseitig. Dadurch läuft der Benutzerdialog ab wie im Produktivbetrieb. Testanweisungen für die Anwendungstask werden in dem entsprechenden Fenster eingegeben und ausgewertet.
- Sie können Ihre Anwendung auf eine neue openUTM-Version umstellen und parallel zu der „alten“ Version testen. Im Parallelbetrieb müssen Sie lediglich dafür sorgen, dass alle Transportsystem-Endpunkte Ihrer Anwendungen (MAX APPLINAME, BCAMAPPL, T-SEL des ACCESS-POINT) unterschiedliche Namen und Portnummern haben.

- die ausgetestete Anwendung kann ohne Änderungen in den Produktivbetrieb übernommen werden.
- In einer Dialog-Task dürfen nur Anwendungen einer openUTM-Version getestet werden, d.h. es darf nicht zunächst eine UTM-Anwendung einer älteren openUTM-Version und anschließend eine Anwendung der Version V6.5 oder umgekehrt getestet werden.

## 2.1.1 Vorbereitungen für das Testen im Dialog

Im Dialog gestartete Anwendungen sind ausschließlich zum Testen vorgesehen. Unterschiede zu mit ENTER-JOB oder ENTER-PROCEDURE gestarteten Anwendungen bestehen darin, dass Tasks nicht automatisch nachgestartet werden und dass die UTM-STXIT-Behandlung ausgeschaltet werden kann.

### Symbolisch testen

Wenn Sie symbolisch testen wollen, müssen Sie die Programme so übersetzen, dass der Compiler eine „List for Symbolic Debugging“ (LSD) erstellt (in COBOL zum Beispiel durch die Angabe COMOPT SYMTEST=ALL).

Sie sollten die LSD-Information nicht fest einbinden, da Sie diese Informationen im Bedarfsfall aus Ihrer Modulbibliothek nachladen können. Die ausgetesteten Programme können Sie dann direkt in den Produktivbetrieb übernehmen.

### Austausch von Programmteilen

Die Angabe LOAD-MODE={ STARTUP | ONCALL} in der LOAD-MODULE-Anweisung kann vorteilhaft sein für das Testen Ihrer Teilprogramme und Event Exits. Nach einer Programmänderung und -Neuübersetzung kann das Binden der Anwendung entfallen. Sie müssen nur das LLM neu binden, welches das geänderte Teilprogramm enthält. Wenn das LLM nur aus diesem Teilprogramm besteht, entfällt auch dieser Schritt. Sie aktivieren das geänderte Teilprogramm, indem Sie entweder die Anwendung neu starten oder bei laufender Anwendung das LLM austauschen (Kommando KDCPROG LOAD-MODULE=..., VERS=...) oder über die Schnittstelle zur Programmadministration (opcode=KC\_MODIFY\_OBJECT) oder mit WinAdmin/WebAdmin.



Ist das geänderte Teilprogramm einem Lademodul zugeordnet, das mit LOAD-MODE=ONCALL generiert ist, dann muss das geänderte Teilprogramm eine andere Version haben als das bisher geladene Teilprogramm.

### Parallelbetrieb für Versionsumstieg

Wenn Sie den Parallelbetrieb nutzen, können Sie den Aufwand bei einem Umstieg auf eine neue openUTM-Version wie folgt minimieren:

- Installieren Sie die neue openUTM-Version zusätzlich zu Ihrer aktuellen Version.
- Binden Sie Ihr Anwendungsprogramm (mit einem anderen Namen) mit der neuen Version, siehe openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“.
- Testen Sie parallel zum Produktionsbetrieb.

### Anzahl der Tasks beim Testen

Im Normalfall reicht es zum Testen aus, die Anwendung mit nur einer Task zu starten. Bestimmte Funktionalitäten einer UTM-Anwendung, wie z.B. Programme die PWGT-Aufrufe nutzen, lassen sich jedoch nur testen, wenn die Anwendung mit mehr als einer Task gestartet wird.

Wollen Sie eine Anwendung im Dialog mit mehreren Tasks starten, so beachten Sie bitte folgende Hinweise:

- Es ist nicht möglich, Tasks im Dialog **und** im Batch-Betrieb gemischt zu starten.
- Folgetasks können Sie nicht per Administrationskommando starten, sondern nur „von Hand“ an einem Terminal bzw. in einem eigenen Fenster.
- die AID-Kommandos müssen für jede Task wiederholt werden, da eine Benutzeranforderung von einer nicht vorherbestimmbaren Task der Anwendung bearbeitet wird. Um den Aufwand der wiederholten Eingabe von Kommandos beim Test im Mehrtaskbetrieb zu vermeiden, können die erforderlichen AID-Kommandos schon in die Startprozedur eingefügt werden.
- Wenn Sie eine OSI TP-Anwendung in mehreren Dialog-Tasks testen und beenden, sollte die Anwendung in denselben Dialog-Tasks nicht noch einmal gestartet werden, da dies zu Startfehlern bzw. zu abnormalem Anwendungsende führen kann.

Eine Dialog-Task arbeitet nur so lange für die Anwendung, wie das Programm geladen bleibt. Nach einer Programmbeendigung können Sie mit der Task im BS2000-Teilnehmerbetrieb beliebig weiterarbeiten. Wenn Sie nur mit einer Task arbeiten, so führt jede Programmbeendigung zum Ende der Anwendung.

## 2.1.2 Starten der Anwendung und Eingabe von Test-Kommandos

Folgende Arbeitsschritte müssen Sie durchführen, wenn Sie eine Anwendung im Dialog zum Testen starten:

1. Sie öffnen eine \$DIALOG-Session und starten in der Dialogtask Ihre UTM-Anwendung. Dabei gehen Sie vor wie im Produktivbetrieb, starten aber die Anwendung nicht, sondern laden sie nur mit LOAD-EXECUTABLE-PROGRAM. Das zugehörige Fenster wird im folgenden *Taskfenster* genannt. Mit LOAD-EXECUTABLE-PROGRAM werden alle fest eingebundenen Programme geladen. Wenn Sie z.B. ein Teilprogramm für einen Start-Exit testen wollen, können Sie nun Testkommandos dafür eingeben.
2. Starten Sie das Programm mit %RESUME. openUTM lädt nun alle Programme, die in Lademodulen enthalten sind, welche mit LOAD-MODE=STARTUP generiert sind. Dazu gehören auch shareable Teile (LOAD-MODE=(POOL/POOL,STARTUP)).
3. Drücken Sie die Taste K2, damit das Programm unterbrochen wird und Sie Test-Kommandos eingeben können.  
Die K2-Taste wirkt dabei jedoch erst, wenn sich die UTM-Task im nicht-privilegierten Funktionszustand des BS2000 (TU) befindet. Das ist eventuell erst dann der Fall, nachdem Sie Schritt 4 durchgeführt haben.
4. Schließen Sie sich über einen UTM-Client an Ihre UTM-Anwendung an und melden Sie sich ggf. über eine Benutzerkennung bei der UTM-Anwendung an. Anschließend starten Sie ein Teilprogramm oder machen Ihre erste Benutzereingabe.  
Nun können Sie Testkommandos für das Anwendungsprogramm im *Taskfenster* eingeben. Beachten Sie bitte, dass mit LOAD-MODE=ONCALL generierte Lademodule erst dann geladen werden, nachdem sie einmal in einer Task aufgerufen wurden. Erst danach können Sie für diese Programme Testkommandos eingeben.
5. Wenn Sie symbolisch testen wollen und die LSD-Informationen nicht statisch eingebunden sind, müssen Sie mit %SYMLIB *symlib1*, *symlib2* . . . die Bibliotheken zuweisen, in denen sich die LSD-Informationen befinden.  
Falls in Ihren Programmen Variablennamen mit Groß-/Kleinschreibung verwendet werden (C/C++-Teilprogramme) und Sie sich auf diese Namen beziehen wollen, müssen Sie mit dem Kommando %AID LOW die Voreinstellung von AID ändern.  
Nun können Sie im *Taskfenster* Ihre Test-Kommandos eingeben, z. B. Haltepunkte setzen, siehe auch AID-Benutzerhandbücher.  
Mit RESUME-PROGRAM setzen Sie das UTM-Anwendungsprogramm fort.
6. Der UTM-Client erhält nach der Bearbeitung des Dialogschrittes die Antwort auf die Eingabe; genauso wie im Produktivbetrieb.

Wenn Sie eine weitere Dialogtask starten wollen, müssen Sie die Schritte 1 und 2 wiederholen, anschließend wieder etwas vom UTM-Client aus eingeben und dann die in Schritt 4 durchgeführten Aktionen wiederholen. Falls Sie bei den Startparametern „TASKS=1“



angegeben haben, müssen Sie zunächst administrativ die Anzahl der Tasks erhöhen, da ansonsten der Startfehler 31 auftritt. Beachten Sie bitte, dass beim Testen im Dialog die zweite Task ein UTM-System-Prozess ist und deshalb ggf. drei Tasks gestartet werden müssen. Für das Testen im Dialog empfiehlt es sich deshalb, die Anwendung ohne UTM-System-Prozesse zu generieren (MAX SYSTEM-TASKS=0).

Bitte beachten Sie, dass Sie **keine Batch-Tasks** nachstarten können.

Um den Aufwand der wiederholten Eingabe von Kommandos beim Test im Mehrtaskbetrieb zu vermeiden (Haltepunkte müssen für jede Task gesetzt werden), können die erforderlichen AID-Kommandos schon in die Startprozedur eingefügt werden.

Sie können jederzeit weitere UTM-Clients an Ihre UTM-Anwendung anschließen, unabhängig davon, mit wie vielen Tasks die Anwendung läuft.

## Hinweise

Beim Starten einer Anwendung im Dialog ergeben sich die folgenden Besonderheiten:

- Der KDCS-Aufruf PEND ER beendet das Programm nicht, das Programm wird somit auch nicht nachgeladen. Der Event-Exit SHUT wird **nicht** aufgerufen. Wollen Sie mit einem neu geladenen Programm weitertesten, so müssen Sie selbst das Programm beenden und neu starten. Wenn Sie mit einer Task arbeiten, so führt die Programmbeendigung gleichzeitig zum Anwendungsende.
- UTM-Dialog-Tasks werden durch das BS2000-System nicht gegenüber anderen Teilnehmer-Tasks priorisiert. Die Tasks laufen nicht als TP-Tasks, sondern als normale Dialog-Tasks. Der Cache-Speicher wird nicht resident eingerichtet (siehe openUTM-Handbuch „Anwendungen generieren“).
- Der Austausch des Anwendungsprogramms per Administration ist nicht möglich. Einzelne LLM's können jedoch ausgetauscht werden (siehe „[Austausch von Programmteilen](#)“ auf Seite 30).
- Der Startparameter TASKS= legt lediglich die Obergrenze der Anzahl der Tasks fest, bewirkt aber keinen Start der Folgetasks. Jede gestartete Task belegt eine Teilnehmer-Task im BS2000-System und somit im Normalfall auch eine BS2000-Session.
- Das Kommando KDCAPPL TASKS=n führt nur dazu, dass der Wert n als neuer Maximalwert für die Anzahl der Tasks eingetragen wird (wenn n größer ist als die Anzahl der gestarteten Tasks). Siehe hierzu auch openUTM-Handbuch „Anwendungen generieren“.
- Einen UTM-Dump sollten Sie vor dem nächsten Start der Anwendung auswerten oder sichern (umkatalogisieren), weil die Dump-Datei sonst überschrieben werden kann.

### 2.1.3 STXIT-Verhalten beim Testen im Dialog

Sie haben die Möglichkeit, mit dem Startparameter STXIT=OFF die UTM-STXIT-Behandlung beim Testen auszuschalten.

Wenn Sie STXIT=OFF angegeben haben, wird Ihr Programm im Fehlerfall an der Stelle unterbrochen, an der der Fehler auftritt, und Sie können mit entsprechenden Testanweisungen den Fehler diagnostizieren.



#### **ACHTUNG!**

Wenn Sie STXIT=OFF angeben, werden im Fehlerfall keine Transaktionen zurückgesetzt. Das kann zu Inkonsistenzen im Datenbestand führen. Sie sollten diesen Parameter daher nie beim Zugriff auf produktiv eingesetzte Datenbanken setzen!

### 2.1.4 Fehlerhafte Zugriffe auf Adresse NULL testen

openUTM bietet eine Funktion, mit der Sie testen können, ob Teilprogramme der Anwendung fehlerhaft auf Adresse Null zugreifen. Dies wird dadurch erreicht, dass openUTM einen tasklokalen Memorypool auf Adresse Null einrichtet. Daher ist sie nur in Anwendungen einsetzbar, die selbst nicht mit einem Memorypool auf Adresse 0 arbeiten, oder selbst auf Adresse 0 geladen sind (24-Bit-Anwendung).

#### **Testfunktion aktivieren/deaktivieren**

Die Testmethode wird aktiviert, indem Sie in der Startprozedur bzw. im ENTER-Job ein FILE-LINK auf den Namen KDCWRPR absetzen:

```
/SET-FILE-LINK LINK-NAME=KDCWRPR
```

Wenn beim Einrichten des Memorypools ein Fehler auftritt, z.B. weil in der Anwendung schon ein Memorypool auf der Adresse 0 arbeitet, dann gibt openUTM beim Starten die Meldung K078 aus:

```
K078 ENAMP xxxxxxxx. In KDCRTST.
```

In diesem Fall kann die Methode nicht eingesetzt werden.

Sie können die Testmethode wieder deaktivieren, indem Sie die Anwendung ohne das obengenannte SET-FILE-LINK-Kommando neu starten.

#### **Zugriff auf Adresse NULL**

Wenn im Anwendungsbetrieb auf die Adresse 0 zugegriffen wird, dann wird das Teilprogramm durch ein STXIT-Ereignis (STXIT mit IW = 48 oder 5C) abgebrochen.

Das betreffende Teilprogramm kann dann korrigiert werden.

## 2.1.5 Beispiel

Das folgende Ablaufprotokoll zeigt den Beginn eines Tests im Dialog.



- Dieses Beispiel, insbesondere der Start-Parameter FILEBASE, gilt nur für stand-alone UTM-Anwendungen. Eine Knoten-Anwendung einer UTM-Cluster-Anwendung kann ebenfalls im Dialog getestet werden.
- Abhängig von der Compiler- und AID-Version können sich die Syntax der AID-Kommandos und die AID-Ausgaben ändern.

Mit der Kommando-Prozedur START-APPLI wird die Anwendung im Dialog gestartet. Im Benutzerdialog wird der TAC DB4TP1 aufgerufen. Der TAC ist in dem C-Programm TA4DB realisiert.

Die symbolische Information zu diesem Programm ist in der PLAM-Bibliothek OML.TA4DB enthalten.

Der TAC DB4TP1 gibt die empfangene Nachricht zusammen mit allgemeinen Informationen wie z.B. Vorgangsstart oder Folge-TAC im Line-Mode aus. Enthält die Nachricht den String „@end“, dann wird der Vorgang beendet.

**Schritt 1: Starten der Anwendung im Taskfenster**

```

/CALL-PROC NAME=START-APPLI,LOGGING=*YES
 1 1 /SET-PROCEDURE-OPTIONS IMPLICIT-DECLARATION=*NO,-
      DATA-ESCAPE-CHAR=*STD,JV-REPLACEMENT=*AFTER-BUILTIN-FUNCTION
 7 1 /WRITE-TEXT TEXT='** Procedure starts openUTM - application DBAPP **'
** Procedure starts openUTM - application DBAPP **
 9 1 /ASSIGN-SYSDTA TO=*SYSCMD
11 1 /SET-FILE-LINK FILE=DBAPP.SYSLOG(+1),LINK=SYSLOG,SHARED-UPDATE=*YES
14 1 /RELEASE-SUBSYSTEM-SPACE
% ESM0254 COMMAND 'RELEASE-SUBSYSTEM-SPACE' COMPLETELY PROCESSED
16 1 /LOAD-EXECUTABLE-PROGRAM -
      FROM-FILE=*LIB-ELEM(LIB=plam-lib,ELEM=start-llm)
% BLS0523 ELEMENT 'DBAPPT', VERSION '' FROM LIBRARY
      ':40SQ:$userid.LIB.DBAPP.TD' IN PROCESSING
% BLS0524 LLM 'DBAPPT', VERSION '' OF '2017-01-11:13:21:55' LOADED
22 1 /%R
K044 Bitte Startparameter eingeben
.UTM START FILEBASE=DBAPP
.UTM START TABLIB=NEWV5,ROOTNAME=DBAPPT
.UTM START TASKS=1,ASYNTASKS=0,STXIT=ON
.UTM END
K136 (Erste) SYSLOG-Datei ist :40SQ:$userid.DBAPP.SYSLOG(*0001)
K044 Bitte Startparameter eingeben
.FHS MAPLIB=MAPLIB=$userid.SYSLIB.UTM.065.EXAMPLE
.FHS ISTD=RUNP
END
FC01 FHS Version 08.3B00 ready
K051 Kaltstart fuer Anwendung DBAPP mit UTM V06.5A00/BS2000 (390) /32 Bit
erfolgreich
K156 UTM-UPIC Verschluesselungskomponente verfuegbar: N
K052 Start beendet - Task 6HVK fuer Anwendung DBAPP aktiviert; System-Task: N

```

**Schritt 2: Benutzer meldet sich bei der UTM-Anwendung an (Benutzerdialog-Fenster)**

**Schritt 3: K2-Taste drücken** im Taskfenster*Benutzerdialog-Fenster*

```
K008 KDCSIGN akzeptiert – Bitte Eingabe
DB4TP1 ** message 1 **
```

**Schritt 4: Kommandos eingeben***Taskfenster*

```
% CMD0170 DO YOU WANT TO INSERT COMMANDS? REPLY (Y=YES; N=NO)?
Y
% SSM2014 TASK IN ESCAPE MODE AT LEVEL NUMBER '1'
%SYMLIB OML.TA4DB
%AID LOW
/ %INSERT S=N'TA4DB.C'.TA4DB
%R
STOPPED AT SRC_REF: 148, SOURCE: TA4DB@, PROC: TA4DB
%R
```

*Benutzerdialog-Fenster*

```
-----
Vorgangstart mit TAC = DB4TP1

** message 1 **

Folge-TAC = DB4TP1
-----
```

**Schritt 5: Stopp beim Haltepunkt***Benutzerdialog-Fenster*

```
** message 2 * @end **
```

*Taskfenster*

```

STOPPED AT SRC_REF: 148, SOURCE: TA4DB@, PROC: TA4DB
%t 10
148          EXT.PROC START    , BLOCK START, ASSIGN
156          ASSIGN
157          ASSIGN
159          ASSIGN
163          CALL
164          CALL
165          ASSIGN
166          ASSIGN
168          CALL
170          IF
STOPPED AT SRC_REF: 170 , SOURCE: TA4DB@ , PROC: TA4DB , END OF TRACE
%d ca->ca_head.kcpr_tac
ca_area.ca_head.kcpr_tac( 0: 7)
( 0) |D| ( 1) |B| ( 2) |4| ( 3) |T| ( 4) |P| ( 5) |1| ( 6) | | ( 7) |
|
%d ca->ca_head.kcpr_tac %x1
CURRENT PC: 0106E396 CSECT: TA4DB@ *****
V'0101B3E0' = ca_area.ca_head.kcpr_tac + #'00000000'
0101B3E0 (00000000) C4C2F4E3 D7F14040          DB4TP1
%R

```

**Schritt 6:** Testende*Benutzerdialog-Fenster*

```

-----
Vorgangsende mit TAC = DB4TP1
** message 2 * @end **
-----
KDCSHUT NORMAL

```

*Taskfenster*

```
K056 Die Task 6HVK wurde beendet  
K138 SYSLOG-Datei :40SQ:$userid.DBAPP.SYSLOG(*0001) geschlossen  
K057 Der Anwendungslauf wurde beendet  
K199 ROOT-Beendigung eingeleitet mit TERM-UNIT: STEP, Ursache: 40
```

## 2.2 Fehlerdiagnose

In diesem Abschnitt erfahren Sie,

- welche Returncodes die Programmschnittstelle liefert
- wie openUTM Fehler durch Meldungen signalisiert
- welche Unterlagen bei Fehlern erstellt werden müssen
- welche Traces Sie zur Diagnose verwenden können

Die Beschreibung und Auswertung eines UTM-Dumps finden Sie im [Kapitel „Der UTM-Dump“ auf Seite 55](#).

### 2.2.1 Returncodes an der Programmschnittstelle

Nach jedem KDCS-Aufruf (außer PEND) gibt openUTM im Rückgabebereich des Kommunikationsbereichs folgende Returncodes zurück:

- den KDCS-Returncode,
- den internen Returncode.

#### KDCS-Returncode im Feld KCRCCC

Beachten Sie bitte:

- Treten mehrere Fehler gleichzeitig auf, so werden die Fehler der jeweils höchsten Kategorie angezeigt. Innerhalb einer Kategorie wird meistens, aber – abweichend von DIN 66 265 – nicht immer der niedrigste Rückkehrcode angegeben.
- Die genaue Bedeutung der KDCS-Returncodes ist im openUTM-Handbuch „Anwendungen programmieren mit KDCS“ bei jedem KDCS-Aufruf beschrieben. Eine Übersicht aller KCRCCC-Returncodes finden Sie ab [Seite 367](#).

#### Interner Returncode von openUTM in KCRCDC

Der interne Returncode enthält im Fehlerfall eine genauere Spezifikation des Fehlers als der KDCS-Returncode in KCRCCC. Meist wird dieser Returncode zusammen mit den KDCS-Returncodes 40Z oder 70Z (System- bzw. Generierungsfehler) gesetzt. Die genaue Bedeutung finden Sie ab [Seite 371](#).

Der interne UTM-Returncode ist nicht Bestandteil der genormten Schnittstelle KDCS.



## 2.2.2 UTM-Meldungen bei Programmfehlern

Programmfehler sind Formatierungsfehler, Fehler bei der Programmierung der KDCS-Schnittstelle (siehe KDCS-Returncode) oder Fehler, die zu STXIT-Ereignissen führen. openUTM erzeugt dann bestimmte Meldungen, die standardmäßig nach SYSOUT, SYSLST und in der Regel auch nach SYSLOG ausgegeben, siehe [Seite 353f](#).

### Formatierungsfehler

Bei Formatierungsfehlern gibt openUTM immer die Meldung K087 aus und - wenn das Formatierungssystem es anfordert - die Meldung K153. Bei Formatierungsfehlern bei der Ausgabe einer asynchronen Nachricht gibt openUTM zusätzlich die Meldung K063 aus. Bei Formatierungsfehlern im Dialog sendet openUTM die Meldung K015, wenn der Formatierungsfehler nicht zu einer abnormalen Vorgangsbeendigung führt.

Bei Fehlern im Event-Exit FORMAT trägt openUTM im Feld KCRCDC den Returncode FRxx ein. Die Bedeutung dieser Returncodes legt der FORMAT-Exit fest.

Bei schweren Formatierungsfehlern bei der Formatierung mit FHS meldet openUTM dies mit einem internen Returncode KCRCDC = FHxx, z.B. 'FH01'. Die Formatierung kann nicht durchgeführt werden. In diesem Fall enthält die Meldung den 4 Byte langen, abdruckbaren sekundären FHS-Returncode, der im Handbuch „FHS Formatierungssystem für openUTM, TIAM, DCAM“ beschrieben ist.

Diese Meldungen enthalten folgende Einträge:

- K015 Formatierungsfehler &RDC &RCF2A – Bitte Eingabe

enthält die Einträge

&RDC : interner Returncode im Feld KCRCDC  
&RCF2A : sekundärer FHS-Returncode oder VTSU-Returncode

- K063 Formatierungsfehler &RCF1A &RCF2A bei Format &FMTN fuer asynchrone Nachricht auf LTERM = &LTRM

enthält die Einträge

&RCF1A : interner Returncode im Feld KCRCDC  
&RCF2A : sekundärer FHS-Returncode  
&FMTN : Formatname  
&LTRM : Name des LTERMs, an das die Nachricht gerichtet war

- K087 Fehler in Zusammenarbeit mit dem Formatierungssystem:  
OPCODE = &OPCD3, ERRORCODE = &ERCD4 &ERCD5

enthält die Einträge

&OPCD3 : Operationscode an der Schnittstelle IUTMFORM  
&ERCD4 : interner Returncode FHxx im Feld KCRCDC  
&ERCD5 : sekundärer FHS-Returncode

*Beispiel*

**Die Meldung**

K015 Formatierungsfehler FH01 080C – Bitte Eingabe **besagt**:

Format nicht verfügbar (siehe auch DUMP-Beispiel, ab Seite 112).

Ein MSGTAC-Service, der auf einen Formatierungsfehler reagieren soll, kann so einfach auf den sekundären Rückkehrcode zugreifen.

### **Abnormale Beendigung eines Vorgangs**

Bei einer abnormalen Beendigung eines Dialog-Vorgangs sendet openUTM die Meldung K017, bei einer abnormalen Beendigung eines Asynchron-Vorgangs wird die Meldung K055 ausgegeben.

Diese Meldungen enthalten Returncodes, die weiteren Aufschluss über die Fehlerursache geben:

- K017 Vorgang &TCVG durch UTM beendet (&RCCC/&RCDC &RCF2A) – Bitte Eingabe

**Die Einträge haben folgende Bedeutung:**

&TCVG : TAC, mit dem der Vorgang gestartet wurde  
&RCCC : KDCS-Returncode im Feld KCRCCC  
&RCDC : interner Returncode im Feld KCRCDC  
&RCF2A : sekundärer FHS-Returncode

- K055 Asynchron-Vorgang &ATAC1 durch UTM beendet; KCRCCC= &RCCC ; KCRCDC= &RCDC ; Benutzer= &USER ; LTERM= &LTRM

**Die Einträge haben folgende Bedeutung**

&ATAC1 : TAC, mit dem der Asynchron-Vorgang gestartet wurde  
&RCCC : KDCS-Returncode im Feld KCRCCC  
&RCDC : interner Returncode im Feld KCRCDC  
&USER : Benutzerkennung, die den Asynchron-Vorgang erzeugt hat.  
&LTRM : LTERM-Partner, von dem Asynchron-Vorgang erzeugt wurde.

### Fehler im Input-Exit

- Bei Fehlern im INPUT-Exit sendet openUTM in der Regel die Meldung K098 an das Terminal. Die Erklärung der in K098 enthaltenen Fehlercodes finden Sie auf [Seite 206](#).

### STXIT-Ereignisse

- Bei STXIT-Ereignissen gibt openUTM die Meldungen K102 und K103 aus und beendet den Vorgang abnormal. Die Meldungen sind auf [Seite 207](#) beschrieben.

## 2.2.3 Adressierungsfehler

Zur Unterstützung bei der Suche nach Adressierungsfehlern in Anwenderprogrammen überschreibt openUTM die ersten 4 KB des Tabellenmoduls KDCROOT mit X'EF EF' und legt einen Schreibschutz auf die erste Seite von KDCROOT an. Damit werden unzulässige Schreibzugriffe auf das Tabellenmodul aufgedeckt (Adressfehler, PEND ER). Solche Zugriffe entstehen z.B. als Folge falscher Adressübergabe (KB, SPAB,...) oder als Folge falscher Versorgung von Registern.

### Fehlerhafte Zugriffe auf Adresse NULL

Wenn Sie die auf [Seite 34](#) beschriebene Testfunktion aktiviert haben, dann wird das Teilprogramm durch ein STXIT-Ereignis (STXIT mit IW = 48 oder 5C) abgebrochen, falls im Anwendungsbetrieb auf die Adresse 0 zugegriffen wird.

Das betreffende Teilprogramm kann dann korrigiert werden.

## 2.2.4 Diagnose-Dump bei festgelegten Meldungen/Ereignissen

Beim Eintreten eines bestimmten Ereignisses können Sie das Ziehen eines Diagnose-Dumps, genannt Message-Dump, veranlassen, dessen Kennzeichen abhängig vom Ereignistyp ist.

Ein Message-Dump wird von der Task erstellt, in der das Ereignis eintritt. Dabei wird die UTM-Anwendung nicht beendet. Für das Ziehen eines solchen Message-Dumps müssen Sie den Testmodus für die Anwendung einschalten und das Ereignis, bei dem der Message-Dump gezogen werden soll, setzen. Beides kann per Startparameter oder administrativ erfolgen.

Folgende Ereignisse können Sie angeben:

- die Ausgabe einer bestimmten K-Meldung
- das Auftreten eines bestimmten KDCS-Returncodes (CC oder DC) in einem Teilprogramm
- das Auftreten eines bestimmten SIGN-Status beim Anmelden eines Benutzers

### Einschalten des Testmodus

- über Startparameter

```
.UTM TESTMODE = ON
```

- über Administrationskommando

```
KDCDIAG TESTMODE = ON
```

- über die Programmschnittstelle zur Administration (z.B. über WinAdmin oder WebAdmin)

Im Objekt-Typ `KC_DIAG_AND_ACCOUNT` geben Sie an:

```
testmode='Y' (Datenstruktur kc_diag_and_account_par_str)
```

### Die Funktion Message-Dump einschalten und zurücksetzen

Sie können die Funktion Message-Dump per Startparameter oder administrativ einschalten und zurücksetzen.

Standardmäßig ist die Funktion beim Start der Anwendung ausgeschaltet.

### Einschalten der Funktion Message-Dump

- über Startparameter

`.UTM START DUMP-MESSAGE = (event-type,event)`

Damit wird ein Message-Dump gezogen, sobald das Ereignis eintritt.

Beim Einschalten über Startparameter können Sie nur ein Event angeben. Die Angabe von Inserts zu einer Meldung ist nicht möglich.

- über Administrationskommando

`KCDCIAG DUMP-MESSAGE = (event-type,event)`

Bis zu drei unterschiedliche Ereignisse können Sie in gleicher Weise über die Parameter `DUMP-MESSAGEx` (mit  $x = 1, 2, 3$ ) für das Ziehen des Message-Dumps angeben. Dabei ist `DUMP-MESSAGE` synonym zu `DUMP-MESSAGE1`.

Für das Ereignis „Ausgabe einer bestimmten K-Meldung“ können Sie als zusätzliche Bedingung bis zu drei Inserts angeben (Parameter `INSERTx` mit  $x = 1,2,3$ ). Näheres siehe `KCDCIAG` im openUTM-Handbuch „Anwendungen administrieren“.

- über die Programmschnittstelle zur Administration

Im Objekt-Typ `KC_DIAG_AND_ACCOUNT` geben Sie in den Feldern `event` und `event_type` das Ereignis und den Ereignis-Typ an (Datenstruktur `kc_diag_and_account_par_str` mit entsprechenden Unterstrukturen). Zusätzlich können Sie bis zu drei Inserts als Bedingung angeben. An der Programmschnittstelle können Sie pro Aufruf ein Ereignis angeben. Weitere Details siehe Beschreibung von `KC_DIAG_AND_ACCOUNT` im openUTM-Handbuch „Anwendungen administrieren“.

- über WinAdmin oder WebAdmin

Im Dialog *Anwendungseigenschaften*, Registerblatt *Message Dump* wählen Sie für das betreffende Dump-Ereignis den Ereignis-Typ und das Ereignis aus, siehe Online-Hilfe zu WinAdmin/WebAdmin.

Mit `event-type` geben Sie den Ereignistyp und mit `event` dazu ein bestimmtes Ereignis an, bei dem der Message-Dump gezogen werden soll. Folgende Ereignisse können Sie angeben:

- Ausgabe einer bestimmten K-Meldung (`event-type = MSG`)  
Als `event` geben Sie die UTM-Meldungsnummer `Knnn` an.  
Bei jedem Auftreten der Meldungsnummer wird ein Dump erzeugt, solange, bis Sie die Meldungsnummer zurücksetzen. Bei den Meldungsnummern K043, K061, K062 wird nur einmal ein Dump erzeugt, und dann die Meldungsnummer automatisch zurückgesetzt.
- Auftreten eines bestimmten kompatiblen KDCS-Returncodes (`event-type = RCCC`)

Als *event* geben Sie die Nummer des kompatiblen KDCS-Returncodes (KCRCCC), z.B. 14Z an. Tritt der Returncode bei einem KDCS-Aufruf auf, wird nur einmal ein Dump erzeugt und anschließend der Parameter DUMP-MESSAGE[x] auf \*NONE zurückgesetzt.

- Auftreten eines bestimmten inkompatiblen KDCS-Returncodes (*event-type* = RCDC)  
Als *event* geben Sie den inkompatiblen KDCS-Returncode (KCRCDC), z.B. KD10 an. Tritt der Returncode bei einem KDCS-Aufruf auf, wird nur einmal ein Dump erzeugt und anschließend der Parameter DUMP-MESSAGE[x] auf \*NONE zurückgesetzt.

#### *Hinweis*

Bei allen KDCS-Returncodes  $\geq 70Z$  und den zugehörigen inkompatiblen KDCS-Returncodes, bei denen grundsätzlich kein PENDER-Dump geschrieben wird (z.B. 70Z/K316), wird kein Message-Dump erzeugt.

- Auftreten eines bestimmten Anmeldestatus (*event-type* = SIGN)  
Als *event* geben Sie den SIGNON-Statuscode in der Form *xyy* (z.B. U05) an:
  - *x* entspricht dem Wert in KCRSIGN1, wobei die Werte U, I, A oder R möglich sind
  - *yy* entspricht dem Wert in KCRSIGN2
 Tritt der Statuscode bei einem SIGN-Aufruf auf, wird nur einmal ein Dump erzeugt und anschließend der Parameter DUMP-MESSAGE[x] auf \*NONE zurückgesetzt. Dies geschieht unabhängig davon, ob in der Anwendung ein Anmelde-Vorgang generiert ist oder nicht.

#### *Ausschalten der Funktion Message-Dump*

- über Administrationskommando

```
KDCDIAG DUMP-MESSAGE = *NONE
```

Bei Ereignissen, die mit den Parametern DUMP-MESSAGE<sub>x</sub> eingeschaltet wurden, ist der entsprechende Parameter auf \*NONE zu setzen.

- über die Programmschnittstelle zur Administration

Im Objekt-Typ KC\_DIAG\_AND\_ACCOUNT setzen Sie alle Ereignisse zurück, indem Sie für jedes Ereignis *event\_type=NONE* setzen.

- über WinAdmin oder WebAdmin

Im Dialog *Anwendungseigenschaften*, Registerblatt *Message Dump* wählen Sie für das betreffende Dump-Ereignis den Ereignis-Typ *<ohne>*.

## 2.2.5 Fehlerunterlagen erstellen

Dieser Abschnitt listet auf, welche Diagnoseunterlagen ein Anwender erstellen soll, wenn er einen Systemfehler an die Entwicklung melden möchte.

- Eine genaue Beschreibung der Fehlersituation und die Angabe, ob und wie der Fehler reproduzierbar ist.
- Die Versionsnummer des BS2000-Betriebssystems mit Korrekturstand.
- openUTM-spezifische Unterlagen:
  - UTM-Dumps, wobei darauf zu achten ist, dass *alle* Dumps von *allen* Tasks, die zum Zeitpunkt des Fehlers aktiv waren, mitgeliefert werden.
  - Alle zur Verfügung stehenden Traces. Beim Reproduzieren von Fehlern sollte TESTMODE=ON eingeschaltet werden, wobei TRACEREC mindestens auf 3000 eingestellt werden sollte (bei openUTM-D-Einsatz: mindestens 10000).
  - Bei Warmstartfehlern, bei Fehlern im KDCUPD und bei Anwendungsabbruch mit PMIO22 werden die Datei(en) der KDCFILE benötigt.
  - openUTM-Versionnummer mit Korrekturstand
  - Die Protokolldatei von KDCDEF
  - Binderlisting des Anwendungsprogramms
  - Übersetzungslistings aller betroffenen Teilprogramme
  - Die SYSLOG-Datei(en).
  - User-Dumps bei Fehlern wie z.B. XT48, XT58...
  - SYSLST- und SYSOUT-Protokoll.
- Bei Fehlern in UTM-Cluster-Anwendungen werden zusätzlich folgende Unterlagen benötigt:
  - Alle Cluster-globalen Dateien, Protokolle (und DUMPs) aller Knoten-Anwendungen
  - Die Cluster-Konfigurationsdatei und bei administrativen Problemen auch alle Dateien des Administrations-Journals mit Suffix JKAA, JRN1, JRN2.
  - Bei Problemen, die durch das Zusammenspiel der Knoten-Anwendungen verursacht wurden, die Protokolldateien von allen anderen Knoten-Anwendungen
  - Die Startprozedur und die bei der UTM-Generierung als EMERGENCY-CMD und FAILURE-CMD angegebenen Prozeduren
  - Bei Problemen zu Benutzern (z.B. Anmeldeprobleme) auch die Cluster-User-Datei (d.h. die Datei mit dem Suffix UTM-C.USER)
- Bei Fehlern in Zusammenhang mit FHS zusätzlich:
  - Angabe der eingesetzten FHS-Version mit Korrekturstand
  - Formatausdruck mit IFG
  - Einsatzbereites Formatmodul in Dateiform
  - Benutzerdumps und UTM-Dump, wenn vorhanden

- Bei Fehlern in Zusammenarbeit mit Datenbanken: Bitte Freigabemitteilungen der jeweiligen Datenbanksysteme beachten.

## 2.2.6 Traces

Für openUTM können Sie außer den Traces im UTM-Dump noch folgende Traces zur Diagnose heranziehen:

- Protokollierung von KDCS-Aufrufen aus C/C++-Teilprogrammen
- BCAM-Trace in openUTM
- ADMI-Trace, d.h. Trace der Programmschnittstelle zur Administration (KDCADMI)

### 2.2.6.1 Protokollierung von KDCS-Aufrufen aus C/C++-Teilprogrammen

Mit dieser Funktion können alle KDCS-Aufrufe aus C-Teilprogrammen protokolliert werden. Das C-Teilprogramm muss dazu die Makros aus dem Headerfile `kcmac.h` verwenden.

*Protokollierung einschalten*

Setzen Sie den Job-Variablen-Link `*KDCSCDB`.

Der Job-Variablen-Link wird bei jedem Start einer UTM-Task ausgewertet. Die Protokollierung wird standardmäßig nach SYSOUT ausgegeben.

Falls die Protokollierung in eine andere Datei geschrieben werden soll, müssen Sie die Präprozessor-Variablen `KDCS_DEBUG_FP` in jedem Teilprogramm, in dem Sie Traces in diese Datei schreiben wollen, vor dem Inkudieren von `kcmac.h` auf den Namen einer selbstdefinierten, globalen FILE-Variablen setzen und diese FILE-Variablen als extern definieren. Diese FILE-Variablen definieren Sie im Startexit und versorgen ihren Wert.

Voreinstellung: Die Aufrufe werden nicht protokolliert.

### 2.2.6.2 BCAM-Trace in openUTM

Mit der Funktion BCAM-Trace von openUTM können alle Verbindungs-bezogenen Aktivitäten innerhalb einer UTM-Anwendung protokolliert werden.

#### Inhalt des BCAM-Trace

Es werden folgende Typen von Trace-Records geschrieben:

- Parameterblock : Die BCAM-Parameterblöcke der Aufrufe REQCON, ACCON, REJCON, DISCON sowie die BCAM-Parameterblöcke der Aufrufe RECLET und SENDLET, die einen Returncode lieferten, oder die RSO-Parameterblöcke der Aufrufe der RSO-Schnittstelle.
- Announcement: Alle Verbindungs-bezogenen Announcements



- Connection-Letter
- Kommunikation mit TS-Anwendung vom Typ Socket: Parameterblöcke der Funktionen `connection request`, `connection response` und `disconnect request` der Socket-Schnittstelle
- Nachricht: Alle Ein-/Ausgabe-Nachrichten

Jeder Trace-Record enthält folgende Einträge in der angegebenen Reihenfolge:

1. Zeitstempel
2. BCAMAPPL-Name
3. PTERM/MUX/CON-Name
4. PROCESSOR-Name
5. LTERM/LPAP-Name
6. USER-Name
7. Typ des Trace-Records (s.o.):
  - Announcement
  - Parameterblock
  - Connection-Letter
  - TCP/IP-Record
  - Nachricht (Ein-/Ausgabe-Nachricht)
8. Bis zu 32624 Byte Daten (abhängig vom Recordtyp und vom Wert des Operanden *length* beim Startparameter BTRACE). Bei längeren Datenelementen (z.B. Ein-/Ausgabe-Nachrichten) ist das Verhalten abhängig von der eingestellten maximalen Länge der Daten, die mitgeschnitten werden:
  - Ist ein Wert kleiner als das Maximum 32624 eingestellt, enthält der Trace-Record die ersten  $length/2$  Bytes und die letzten  $length/2$  Bytes des Datenelements.
  - Ist der Maximalwert 32624 eingestellt, wird versucht, die Daten komprimiert zu schreiben. Falls die Daten nicht auf eine Länge von 32624 Bytes komprimiert werden können, wird der Trace-Record nur mit den ersten 16312 Bytes und den letzten 16312 Bytes der Daten geschrieben.

### Vorbereitende Tätigkeiten

In der UTM-Startprozedur muss für jede Task eine Trace-Datei eingerichtet und dieser mit dem Kommando SET-FILE-LINK der LINK-Name KDCBTRC zugewiesen werden.

Jede Task einer UTM-Anwendung schreibt die Trace-Sätze in eine eigene Datei. Daher müssen die Trace-Dateien in der Startprozedur so eingerichtet werden, dass sie unterschiedliche Dateinamen erhalten. Zu diesem Zweck kann z.B. die TSN der Task an den Dateinamen angehängt werden.

Mit SDF-P kann dies auf folgende Weise geschehen:

```
/CREATE-FILE FILE-NAME = &(FILEBASE//'.BTRC.'//TSN()) -
/ , SUPPORT=*PUBLIC-DISK(SPACE=*RELATIVE -
/ (PRIMARY-ALLOCATION=<integer_1..2147483647> -
/ ,SECONDARY-ALLOCATION=<*STD or integer_0..32767>))

/ADD-FILE-LINK FILE-NAME = &(FILEBASE//'.BTRC.'//TSN()) -
/ ,LINK-NAME = KDCBTRC [ , BUFFER-LEN = *STD(SIZE=<1..16>) ]
```

Dabei sollte der Prozedurparameter FILEBASE den <filebase>-Namen der UTM-Anwendung enthalten.

Ab einer maximalen Länge der Trace-Sätze von 1960 müssen Sie zusätzlich den Parameter BUFFER-LENGTH mit der passenden Anzahl Blöcke angeben. Berücksichtigen Sie dabei, dass die Primärzuweisung mindestens doppelt so groß sein muss wie die Blockgröße, und dass die Sekundärzuweisung Null oder mindestens so groß sein muss wie die Blockgröße.

Maximale Trace-Länge	BUFFER-LENGTH=
> 1960	STD(2)
> 4008	STD(4)
> 8104	STD(8)
> 16296	STD(16)

Steht SDF-P nicht zur Verfügung, dann kann dasselbe mittels Job-Variablen oder EDT-Prozeduren erreicht werden.

### Ein-/Ausschalten BCAM-Trace

Der BCAM-Trace kann per Startparameter oder per Administrationskommando ein- und ausgeschaltet werden. Standardmäßig ist die Funktion beim Start der Anwendung ausgeschaltet.

- Ein-/Ausschalten über Startparameter

$$.UTM \text{ START,BTRACE} = \left\{ \begin{array}{l} \text{ON/OFF} \\ \text{( ON / OFF, length )} \end{array} \right\}$$

Damit wird die Trace Funktion beim Start der Anwendung eingeschaltet (ON) bzw. bleibt ausgeschaltet (OFF).

Außerdem können Sie die maximale Länge der Daten, die aufgezeichnet werden, angeben.

Minimum: 32

Maximum: 32680

Standardwert: 256

Diese Maximallänge kann nur über Startparameter festgelegt werden.

Wenn Sie den BCAM-Trace für die Funktion UPIC Capture einsetzen (siehe openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“), dann wird empfohlen, den Maximalwert zu verwenden.

- Ein-/Ausschalten über Administrationskommando

```
KDCDIAG BTRACE=ON/OFF [ ,LTERM=lterm-name / LPAP=lpap-name /
                        MUX=(mux-name,prozessor-name,bcamapp1) / USER=user-name]
```

Damit wird die Trace Funktion im Betrieb der Anwendung ein- bzw. ausgeschaltet. Bei Angabe eines LTERM-, LPAP- oder MUX-Namens werden nur die zu dieser Verbindung gehörenden Ereignisse aufgezeichnet.

Bei Angabe eines USER-Namens werden nur die zu dieser Benutzerkennung gehörenden Ereignisse aufgezeichnet.

- Ein-/Ausschalten über Programmschnittstelle zur Administration (z.B. über WinAdmin oder WebAdmin)

Im Objekt-Typ KC\_DIAG\_AND\_ACCOUNT geben Sie an:

```
bcam_trace='Y' bzw. 'N' (Datenstruktur kc_diag_and_account_par_str)
```

Nach dem Einschalten der BCAM-Trace Funktion werden die durch die SET-FILE-LINK-Kommandos zugewiesenen Trace-Dateien (SAM-Datei) mit EXTEND geöffnet.

Wird der BCAM-Trace mit dem Kommando KDCDIAG eingeschaltet, ohne dass Trace-Dateien mit SET-FILE-LINK-Verknüpfung zugewiesen sind, wird das Kommando zwar erfolgreich ausgeführt, aber die Anwendungstask läuft beim Schreiben der Trace-Informationen auf einen DMS-Fehler und der BCAM-Trace wird daraufhin wieder ausgeschaltet.

Nach dem Ausschalten der BCAM-Trace-Funktion werden die Trace-Dateien geschlossen und können danach ausgewertet werden. Da das Schließen der Dateien von allen Tasks der Anwendung ausgeführt werden muss, kann es einige Zeit erfordern, bis alle Trace-Dateien geschlossen sind. Mit Hilfe des Kommandos SHOW-FILE-ATTRIBUTES können die eröffneten Dateien abgefragt werden.

Tritt für eine der Trace-Dateien ein DMS-Fehler auf, beendet openUTM die Trace-Funktion.

### Auswerten des BCAM-Trace

Die Trace-Dateien können nach Ausschalten der BCAM-Trace Funktion bzw. dem Beenden der UTM-Anwendung ausgewertet werden. Dazu steht das Tool KDCBTRC zur Verfügung. Mit dem Tool KDCBTRC können nur Trace-Dateien der gleichen openUTM-Version ausgewertet werden.

Vor dem Aufruf des Tools KDCBTRC müssen die Trace-Dateien der einzelnen Tasks in eine Datei gemischt werden.

Es gibt eine Beispielprozedur BTRACE für das Mischen der einzelnen Trace-Dateien und das Auswerten der gemischten Datei mit dem Dienstprogramm KDCBTRC. Die SDF-P-Prozedur BTRACE wird in der Bibliothek SYSPRC.UTM.065 mit ausgeliefert.

Im Allgemeinen wertet KDCBTRC die Datei mit dem Linknamen KDCBTRC aus. Das Ergebnis wird in die Datei mit dem Linknamen LISTFILE geschrieben. Die Steueranweisung wird von SYSDTA gelesen.

Die Steueranweisung im Format

[Argument-1, ..., Argument-n]

erlaubt die Eingabe von optionalen Argumenten zur Steuerung der Aufbereitung. Die Argumente sind im folgenden Abschnitt beschrieben. Wenn Sie eine „leere“ Steueranweisung (Leerzeichen) eingeben, wird die gesamte Trace-Datei aufbereitet.

Argumente des Aufbereitungsprogramms

LT=ltn / (ltn1, ...,ltn5)

Mit diesem Operanden können ein bis fünf LTERM-Namen oder LPAP-Namen angegeben werden. Damit werden nur die Trace-Records aufbereitet, die einen der angegebenen LTERM/LPAP-Namen enthalten.

Voreinstellung: alle Trace-Records werden aufbereitet.

PT=ptn / (ptn1, ...,ptn5)

Mit diesem Operanden können ein bis fünf PTERM-Namen, CON-Namen oder MUX-Namen angegeben werden. Damit werden nur die Trace-Records aufbereitet, die einen der angegebenen PTERM/MUX/CON-Namen enthalten.

Voreinstellung: alle Trace-Records werden aufbereitet.

BC=bcn / (bnc1, ...,bnc5)

Mit diesem Operanden können ein bis fünf BCAMAPPL-Namen oder ACCESS-POINT-Namen angegeben werden. Damit werden nur die Trace-Records aufbereitet, die einen der angegebenen BCAMAPPL/ACCESS-POINT-Namen enthalten.

Voreinstellung: alle Trace-Records werden aufbereitet.

PR=prn / (prn1, ...,prn5)

Mit diesem Operanden können ein bis fünf Prozessornamen angegeben werden. Damit nur die Trace-Records aufbereitet, die einen der angegebenen PROCESSOR-Namen enthalten.

Voreinstellung: alle Trace-Records werden aufbereitet.

- US=usr / (usr1, ...,usr5)**  
Mit diesem Operanden können ein bis fünf Benutzerkennungen (USER) angegeben werden. Damit werden nur die Trace-Records aufbereitet, die den Namen einer der angegebenen Benutzerkennungen enthalten.  
Voreinstellung: alle Trace-Records werden aufbereitet.
- AN=Y/N**  
Bei Angabe von AN=Y werden die Trace-Records aufbereitet, die An-  
nouncements enthalten. Bei AN=N wird die Aufbereitung unterdrückt.  
Voreinstellung: AN=N
- PB=Y/N**  
Bei Angabe von PB=Y werden die Trace-Records aufbereitet, die BCAM-  
oder RSO-Parameterblöcke enthalten. Bei PB=N wird die Aufbereitung un-  
terdrückt.  
Voreinstellung: PB=N
- CL=Y/N**  
Bei Angabe von CL=Y werden die Trace-Records aufbereitet, die Connec-  
tion-Letters enthalten. Bei CL=N wird die Aufbereitung unterdrückt.  
Voreinstellung: CL=Y
- IN=Y/N**  
Bei Angabe von IN=Y werden die Trace-Records aufbereitet, die Eingabe-  
Nachrichten enthalten. Bei IN=N wird die Aufbereitung unterdrückt.  
Voreinstellung: IN=Y
- OUT=Y/N**  
Bei Angabe von OUT=Y werden die Trace-Records aufbereitet, die Ausga-  
be-Nachrichten enthalten. Bei OUT=N wird die Aufbereitung unterdrückt.  
Voreinstellung: OUT=Y
- DVC=T/P/B**  
Mit dem DVC-Operanden wird das Format der Ausgabedatei bestimmt.  
DVC=T: Zeilenbreite max. 79 Zeichen, keine Druckersteuerzeichen  
DVC=B: Zeilenbreite max. 79 Zeichen, mit Druckersteuerzeichen  
DVC=P: Zeilenbreite max. 132 Zeichen, mit Druckersteuerzeichen  
Voreinstellung: DVC=T
- SOCKET=Y/N**  
Bei Angabe von SOCKET=Y werden die Trace-Records, die Traces für  
Funktionen zur Kommunikation mit dem Protokoll TCP/IP enthalten, aufbe-  
reitet. Bei SOCKET=N wird die Aufbereitung unterdrückt.  
Voreinstellung: SOCKET=Y

### 2.2.6.3 ADMI-Trace

Der ADMI-Trace protokolliert alle Aufrufe der Programmschnittstelle KDCADMI.

Vor dem Aufruf werden folgende Daten geschrieben:

- Inhalt des Datenbereichs

Nach dem Aufruf werden folgende Daten geschrieben:

- Adressen von Parameterbereich, Identifikationsbereich, Selektionsbereich und Datenbereich
- Inhalt der Felder des Parameterbereichs, z.B. Operationscode (opcode), Objekttyp, Länge des Datenbereichs, Returncode
- Inhalt des Datenbereichs

Der ADMI-Trace kann über den Startparameter ADMI-TRACE eingeschaltet und über WinAdmin, WebAdmin oder die Programmschnittstelle zur Administration (KDCADMI) ein- oder ausgeschaltet werden.

Der Trace wird standardmäßig in eine Datei geschrieben:

- Für stand-alone-Anwendungen hat diese Datei den Namen `KDC.TRC.ADMI.appliname.tsn`.
- Für UTM-Cluster-Anwendungen hat diese Datei den Namen `KDC.TRC.ADMI.appliname.nodename.tsn`.

Dabei ist *appliname* der Name der UTM-Anwendung (MAX APPLINAME), *nodename* der Name des Cluster-Knotens, auf dem die Knoten-Anwendung läuft, und *tsn* die Nummer der Task. Sie können auch in der UTM-Startprozedur für jede Task eine andere Trace-Datei einrichten und mit dem Kommando SET-FILE-LINK den LINK-Namen KDCADMI zuweisen.

---

## 3 Der UTM-Dump

Bei schwerwiegenden Fehlern erzeugt openUTM einen Speicherabzug (UTM-Dump) von allen relevanten Daten. Ein UTM-Dump kann folgende Ursachen haben:

- ein Teilprogramm hat einen PEND ER-Aufruf abgesetzt
- im Anwendungsprogramm trat ein KDCS-Returncode  $KCRCCC \geq 70Z$  auf wegen eines schweren Fehlers bei einem KDCS-Aufruf oder bei einem Fehler in Zusammenarbeit mit einer Datenbank oder einem Formatierungssystem oder weil ein STXIT-Ereignis eingetreten ist
- ein Diagnosedump wurde angefordert (z.B. KDCDIAG)
- ein Fehler im UTM-Systemcode oder in anderen Softwarekomponenten wie z.B. BS2000-System, FHS, eigenen Formatierungsroutinen oder VTSU verursachte einen Abbruch des Anwendungslaufs.

Im letzten Fall werden Dumps von allen Tasks der Anwendung geschrieben; in den anderen Fällen nur der Dump der betroffenen Task. Grundsätzlich werden alle UTM-Dumps komprimiert geschrieben.

### 3.1 Die Dateien des UTM-Dumps

openUTM schreibt die Speicherabzüge in eine Dateigenerationsgruppe (FGG, File Generation Group) oder eine normale BS2000-Datei (Dump-Datei). Standardmäßig sind die Namen der Dump-Dateien bzw. der FGG nach folgendem Schema aufgebaut:

```
DUMP.UTM.rrrrrr.ttttff.aaaaaaaa
```

Wenn der Dump während des Anwendungslaufs geschrieben wurde und der Start-Parameter DUMP-PREFIX angegeben ist, sind die Namen der Dump-Dateien bzw. der FGG nach folgendem Schema aufgebaut:

```
dump-prefix.rrrrrr.ttttff
```

Die Buchstaben bedeuten:

dump-prefix	Präfix, der im Startparameter DUMP-PREFIX angegeben wurde.
rrrrrr	Kennzeichen für die Ursache des Speicherabzugs.
tttt	die TSN der Task, die den Dump erzeugt hat. Liegt ein Anwendungsabbruch vor, so ist <code>tttt</code> die TSN-Nummer der Task, die den Abbruch einleitete.  Wenn der Dump in einem UTM-Dienstprogramm (KDCDEF oder KDCUPD) erzeugt wird, ist <code>tttt</code> = UTIL.
ff	bedeutet eine laufende Nummer für die Dumps, die eine Task einer Anwendung erzeugt, wenn die Dump-Dateien nicht als Dateigenerationsgruppe (FGG) eingerichtet sind; oder hexadezimaler Wert des Zählers für die Anzahl Kaltstarts der Anwendung, wenn die Dump-Dateien als FGG-Dateien eingerichtet sind.
aaaaaaaa	Name der Anwendung, zu der der Dump gehört.

Die Dump-Dateien sind entweder normale BS2000-Dateien oder Dateien einer Dateigenerationsgruppe (FGG).

Eine Dateigenerationsgruppe wird erzeugt, wenn die Dumps auf Grund eines Anwendungsabbruchs (term application) entstehen. Der Name der FGG wird bestimmt durch die Task, die den Anwendungsabbruch einleitet. Für die anderen Tasks schreibt openUTM die Dump-Information in weitere Dateien der FGG.

In einigen Fällen ist das nicht möglich. Die Dump-Information von Folgetasks wird dann in Einzeldateien geschrieben, und der Zähler `ff` zählt Task-spezifisch. Außerdem wird an diese Einzeldateien das Suffix `.iii` angehängt (`iii` = Nummer der Dateigeneration, die die Task im Normalfall geschrieben hätte). Ist diese Dateigenerationsnummer nicht verfügbar, z.B. bei Start oder Ende einer Task, wird `.000` angehängt.



Es kann vorkommen, dass openUTM für tttt=0000, für ff=00 und für aaaaaaaa=NONAME einsetzt, weil diese Daten nicht verfügbar sind, z.B bei Start oder Beendigung einer Task.

Mit dem Startparameter DUMP-USERID ist steuerbar, auf welche Kennung openUTM einen UTM-Dump schreibt, siehe openUTM-Handbuch „Einsatz von UTM-Anwendungen“. Syntax des Startparameters:

```
.UTM START DUMP-USERID={ STANDARD | SYSUSER }
```

Bei Angabe STANDARD (Voreinstellung) werden die Dump-Dateien auf die eigene Kennung geschrieben (Kennung, unter der die UTM-Anwendung läuft), bei Angabe SYSUSER auf die Kennung SYSUSER.

### Hinweis

Wenn mehrere Tasks für die Anwendung arbeiten, dann enthält bei einem Anwendungsabbruch der zeitlich erste Speicherauszug den Grund für den Abbruch (REASON). Die übrigen Speicherauszüge enthalten einen Code, der aussagt, dass es sich um einen Folgedump handelt.

Wurde die UTM-Anwendung mit TESTMODE=OFF gestartet, und tritt ein PEND ER mit einem der nachfolgend aufgeführten KCRCDC-Codes auf, so wird der UTM-Dump unterdrückt:

```
FH01, FH02,
FR01, FR02,
K301, K302, K345, K601, K602, K603, K608,
KM01, KM02, KM03, KM04, KM05, KM07, KM08,
KR01, KR02,
KT01, KT02, KT04,
KU14,
XT80
```

In diesen Fällen wird auch das Nachladen unterdrückt. Ausnahme: Für den aktuellen TAC sind PGWT-Aufrufe erlaubt und der Aufruf, bei dem der Fehler auftrat, war kein PEND-Aufruf.

Beim KCRCDC-Code K316 wird unabhängig vom Test-Modus kein UTM-Dump geschrieben. Wenn für den aktuellen TAC PGWT-Aufrufe erlaubt sind und wenn der Aufruf, bei dem der Fehler auftrat, kein PEND-Aufruf war, dann wird das Programm nachgeladen.

## 3.1.1 Reduzierung der Dump-Information über Parameter DUMP-CONTENT

Über den Startparameter DUMP-CONTENT können Sie einstellen, ob openUTM die Dump-Information reduzieren soll oder nicht. Reduzieren heißt, dass Task-übergreifende KAA-Speicherbereiche (Common Memory Pools) nur im Dump der Task enthalten sind, die den

Anwendungsabbruch verursacht hat. Durch die Reduzierung der Dump-Information benötigen die Diagnoseunterlagen bei einem Anwendungsabbruch erheblich weniger Platz. Die Reduzierung der Dump-Information ist als Standard eingestellt. Sie kann durch den Startparameter DUMP-CONTENT bei Bedarf ausgeschaltet und wieder eingeschaltet werden.

**Syntax des Startparameters:**

```
.UTM START DUMP-CONTENT={ STANDARD | EXTENDED }
```

- STANDARD** Wenn openUTM eine Dump-Dateigeneration erzeugt, dann sind Task-übergreifende Speicherbereiche nur im Dump der ersten Task (Verursacher) enthalten. Für die Diagnose ist das normalerweise ausreichend. STANDARD ist voreingestellt.
- EXTENDED** Die Task-übergreifenden Speicherbereiche sind in allen Dumps einer Dump-Dateigeneration enthalten. Diesen Wert sollten Sie nur auf besondere Anforderung des System-Service einstellen.

## 3.2 Das Tool KDCDUMP

Die von openUTM erzeugten Dump-Dateien haben PAM-Format. Diese Dateien müssen Sie mit Hilfe des Tools KDCDUMP aufbereiten. KDCDUMP bietet zwei Möglichkeiten der Aufbereitung:

- **Auswertung im Dialog:**  
KDCDUMP lädt einen Dump oder einen Teil davon in den Speicher, so dass Sie ihn am Terminal im Dialog bearbeiten können. Im Dialog können Sie z.B. in einem umfangreichen Dump gezielt bestimmte Tabellen aufbereiten und durchsuchen.
- **Aufbereitung ganzer Dump-Dateien in Listenform:**  
KDCDUMP erzeugt komplette aufbereitete Dump-Dateien oder Dateigenerationsgruppen (FGGs), die Sie auf einen Drucker ausgeben oder am Terminal z.B. mit /SHOW-FILE anschauen können.

Die Aufbereitung des Dumps steuern Sie über Anweisungen, die Sie nach dem Start von KDCDUMP eingeben. Sie können das Programm KDCDUMP auch über die Angabe von Linknamen steuern.

KDCDUMP wird mit der Steueranweisung END beendet.

Wenn KDCDUMP während der Auswertung auf Fehler läuft, wird der Linkname KDCDUMP nicht automatisch freigegeben. In diesem Fall sollten Sie den Linknamen vor dem nächsten KDCDUMP-Start manuell freigeben und die Aufbereitung ohne Summary-Information versuchen (INFO=DUMP).

Wenn KDCDUMP und die vorliegende Dump-Datei nicht zur selben Version von openUTM gehören, wird der Dump nicht aufbereitet. KDCDUMP gibt dabei die Meldung K719 aus, aus der man erkennen kann, zu welcher Version KDCDUMP und der UTM-Dump jeweils gehören.

### 3.2.1 KDCDUMP starten

Das Programm KDCDUMP kann für die interaktive Auswertung eines Dumps im Dialog gestartet werden oder zur Aufbereitung von Dump-Dateien in Listenform. Die folgenden Abschnitte beschreiben, wie Sie KDCDUMP per START-EXECUTABLE-PROGRAM ohne Linknamen und mit Linknamen starten.

Alternativ können Sie KDCDUMP auch über das SDF-Kommando START-KDCDUMP aufrufen. Dieses Kommando ist im SDF-Anwendungsbereich UTM abgelegt. Weitere Informationen finden Sie im openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Abschnitt „UTM-Tools aufrufen“.

#### Start ohne Linknamen zur Auswertung im Dialog

```
/START-EXECUTABLE-PROGRAM FROM-FILE= -
/          *LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL,ELEM=KDCDUMP)[,TEST-OPT=*AID]
```

Die TEST-OPTION=\*AID muss nur angegeben werden, wenn eine Tabelle symbolisch aufbereitet werden soll, siehe KDCDUMP-Anweisung TABLE. Nach dem Start werden die KDCDUMP-Anweisungen von SYSDTA gelesen.

Wenn Sie Dumps im Dialog aufbereiten möchten, darf der Linkname KDCDUMP keiner Datei zugeordnet sein.

#### Start mit Linknamen zur Aufbereitung ganzer Dump-Dateien in Listenform

Bei Verwendung von Linknamen kann KDCDUMP wie folgt gestartet werden:

```
/SET-FILE-LINK LINK-NAME=KDCDUMP,FILE-NAME=dumpfile
/SET-FILE-LINK LINK-NAME=LISTFILE,FILE-NAME=listfile
/START-EXECUTABLE-PROGRAM FROM-FILE= -
/          *LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL,ELEM=KDCDUMP)
```

dumpfile    Name der Dump-Datei

listfile    Name der aufbereiteten Datei

Als Dump-Datei kann im Kommando /SET-FILE-LINK auch eine Datei einer FGG angegeben werden.

Nach dem Start führt KDCDUMP folgende Anweisungen aus:

```
FILE *KDCDUMP
END
```

Die Anweisung FILE ist auf [Seite 70](#) beschrieben.

Ist der LINK-Name LISTFILE keiner Datei zugeordnet, so erzeugt KDCDUMP standardmäßig eine List-Datei mit folgendem Namen (Bedeutung der Buchstaben siehe [Seite 56](#)):

```
LST.UTMDMP.rrrrrr.ttttff.aaaaaaaa.
```

Anschließend können Sie die erzeugte Datei ausdrucken:

```
/PRINT-DOCUMENT FILE-NAME=listfile,LINE-SPACING=*BY-EBCDIC-CONTROL
```

### 3.2.2 KDCDUMP-Anweisungen

Die folgende Tabelle zeigt eine Übersicht über alle Anweisungen von KDCDUMP sowie deren Bedeutung:

Anweisung	Bedeutung
!	KDCDUMP unterbrechen und Systemkommando ausführen
!!	Zuletzt ausgeführtes Systemkommando nochmals ausführen
AFIND	Adresse im UTM-Dump suchen
Blätter-Anweisungen	Cursor im Arbeitsbereich positionieren
DUMP	Gesamten UTM-Dump oder Teil davon in den Speicher einlesen
EDT	Editor aufrufen
END	KDCDUMP beenden
FGG	Alle Dateien einer FGG (Dateigenerationsgruppe) aufbereiten
FILE	Einzelne Dump-Datei aufbereiten
HELP	Hilfe zu KDCDUMP-Anweisungen ausgeben
LIST	Tabellenausschnitt aufbereiten
REPFIL	Informieren über REP-Datei
SFIND	Suchen nach einem Muster im Dump
SH   SYS	KDCDUMP unterbrechen und auf Betriebssystemebene gehen
SYSLST	Protokollierung einschalten oder ausschalten
TABLE	Tabelle anzeigen

Tritt während des Ablaufs von KDCDUMP ein Fehler auf, wie z.B. Eingabefehler oder Fehler im Programm selber, dann wird eine entsprechende Fehlermeldung ausgegeben und zusätzlich der Auftragsschalter 3 auf „ON“ gesetzt, sofern es sich nicht um einen Adressfehler handelt oder der Dump unvollständig ist. Die Liste der KDCDUMP-Meldungen finden Sie im [Abschnitt „Meldungen des UTM-Tools KDCDUMP“ auf Seite 334](#).

Der Auftragsschalter 3 wird von KDCDUMP zu Beginn des Programmlaufs auf „ON“ gesetzt. Nur wenn während des Programmlaufs kein Fehler auftritt, wird der Schalter auf „OFF“ gesetzt. Der Schalter 3 darf also während des Programmlaufs nicht durch den Anwender manipuliert werden, insbesondere darf er nicht mit der !-Anweisung oder nach der SH- bzw. SYS-Anweisung ausgeschaltet werden.

## Eingabe von KDCDUMP-Anweisungen

Alle KDCDUMP-Anweisungen werden von SYSDTA gelesen. Eine Anweisung darf maximal aus 256 Zeichen bestehen, längere Eingaben führen zu der Fehlermeldung K759.

Sofern Anweisungsnamen abgekürzt werden können, ist die Abkürzung durch Fettdruck gekennzeichnet. Beispielsweise bedeutet **AFIND**, dass Sie den Anweisungsnamen durch AF abkürzen können.

Bei einigen Operanden muss eines der drei Eingabeformate „C-String“, „X-String“ oder „Dezimal“ eingehalten werden:

Bezeichnung	Format der Eingabe / Beispiel
C-String	[C]'Dies ist ein C-String'
X-String (hexadezimal)	X'AAAF' oder X'aaaf' oder X'AaAf'
Dezimal	12345

Nur bei C-Strings wird zwischen Groß- und Kleinbuchstaben unterschieden, bei den anderen Eingabeformaten ist die Groß- Kleinschreibung nicht relevant.

Werden Indizes oder Displacements/Offsets angegeben, dann werden diese immer positiv interpretiert.

Die folgenden Abschnitte beschreiben die KDCDUMP-Anweisungen in alphabetischer Reihenfolge. Es werden nur die Anweisungen und Operanden beschrieben, die zur Auswertung des KDCROOT-Bereichs (siehe [Seite 83](#)) benötigt werden.

## ! Systemkommando eingeben

Mit dieser Anweisung kann während einer KDCDUMP-Sitzung ein BS2000-Kommando ausgeführt werden.

---

```
!_ command
```

---

**command** BS2000-Kommando. Beachten Sie, dass einige BS2000-Kommandos dazu führen, dass das Programm KDCDUMP beendet wird.

## !! Zuletzt ausgeführtes Systemkommando wiederholen

Mit dieser Anweisung kann während einer KDCDUMP-Sitzung das zuletzt (mit ! oder !!) ausgeführte BS2000-Kommando nochmals ausgeführt werden.

---

!!

---

## AFIND Adresse im Dump suchen

Mit dieser Anweisung kann eine Speicheradresse im UTM-Dump gesucht werden, die während des Speicherabzuges eine gültige Adresse für einen UTM-Tabellenbereich war.

Als Ausgabe auf dem Terminal wird der zugehörige Tabelleneintrag entweder ab Anfang des Eintrages oder ab der gesuchten Stelle angezeigt.

Ist die Kommandosyntax fehlerhaft oder kein Dump im Speicher geladen, wird eine entsprechende Fehlermeldung ausgegeben (K710 bzw. K711). Sollte die angegebene Adresse keine Speicheradresse darstellen oder aber nicht genau einem Tabelleneintrag zuzuordnen sein, wird dies ebenfalls gemeldet (K712 bzw. K713).

---

AFIND<sub>L</sub> address

---

address Gesuchte Adresse im Speicher. Die Adresse muss hexadezimal angegeben werden.

### Beispiel

AF X'01018000'

## Blätteranweisungen für die Dialogauswertung

Falls die gewünschte Information nicht auf einem Bildschirm darstellbar ist, kann mit folgenden Anweisungen geblättert werden:

Anweisung	Bedeutung
+ - nur Return (leere Eing.)	Vorwärts blättern um eine Bildschirmseite.



Anweisung	Bedeutung
+n	<ul style="list-style-type: none"> <li>– Bei Tabellenelementen, die nicht komplett auf einer Bildschirmseite darstellbar sind, wird n zur Anfangsadresse der momentan dargestellten Information addiert. Das Ergebnis dieser Addition ist die Adresse, ab der die nächste Information angezeigt wird.</li> <li>– Bei Tabellen, von denen ein Element vollständig auf einer Bildschirmseite darstellbar ist, wird n zum Index des ersten momentan angezeigten Tabellenelementes addiert. Das Ergebnis dieser Addition ist der Index des Elementes, mit dem die Fortsetzung der Tabellenanzeige beginnt. Bei Überschreiten des Tabellenendes wird das letzte Tabellenelement angezeigt, bei Trace-Tabellen wird automatisch umbrochen. n kann dezimal oder hexadezimal angegeben werden. n wird für die Anweisungen HELP TABLE-NAMES und REPPFILE ignoriert. Es wird normal weitergeblättert.</li> </ul>
++	Es wird das Ende der Tabelle oder des Tabelleneintrages angezeigt. Ausnahmen bilden die Trace-Tabellen, bei denen zum zeitlichen Ende geblättert wird. Bei den Anweisungen HELP TABLE-NAMES und REPPFILE wird eine normale „+“-Anweisung durchgeführt.
- -n --	Für das Rückwärtsblättern gilt sinngemäß - mit umgekehrtem Vorzeichen - das Gleiche wie beim Vorwärtsblättern. Es wird höchstens bis zum Tabellenanfang zurückgeblättert. Beim Anzeigen der REP-File kann nicht rückwärts geblättert werden.
<<	Bei SFIND-Kommando kann mit „<<“ zum Anfang der Trefferliste geblättert werden.
>>	Bei SFIND-Kommando kann mit „>>“ zum Ende der Trefferliste geblättert werden.
X	Die Tabellenanzeige wird abgebrochen. Alternativ dazu kann z.B. auch das Kommando END eingegeben werden.

Zur Unterstützung wird auf dem Bildschirm angezeigt, welche Anweisungen jeweils erlaubt sind, z.B.:

- „+/X“ bei der Anweisung REPPFILE,
- „+/-/X“ für alle weiteren Tabellen des KDCROOT-Bereichs.

## DUMP UTM-Dump in den Speicher einlesen

Mit dieser Anweisung können Sie eine UTM-Dump-Datei (keine FGG) dekomprimieren und vollständig oder bereichsweise in den Arbeitsspeicher einlesen. Die UTM-Dump-Datei können Sie mittels ihres Dateinamens oder Linknamens angeben.

Die Anweisung müssen Sie absetzen, bevor Sie einen UTM-Dump interaktiv am Terminal bearbeiten können (z.B. mit den Anweisungen TABLE, LIST).

Besonders bei großen UTM-Dumps, bei denen der zur Verfügung stehende Arbeitsspeicher oder virtuelle Adressraum nicht ausreicht, um den gesamten dekomprimierten Dump aufzunehmen, ist es sinnvoll, mit dem Operanden DOMAIN=ROOT nur den KDCROOT-Bereich einzulesen.

Bei jedem DUMP-Kommando wird der zuvor in den Arbeitsspeicher eingelesene Dump bzw. ROOT-Bereich entfernt und der neue Dump oder ROOT-Bereich eingelesen.

---

```
DUMP_      { FILE = { filename | *SAME } | LINK = linkname }
           [ ,DOMAIN = { ALL | ROOT } ]
```

---

### FILE=

filename Name der UTM-Dump-Datei

\*SAME Geben Sie statt des Datei- bzw. Linknamens FILE=\*SAME an, wenn Sie einen neuen Bereich des zuvor mit DUMP dekomprimierten UTM-Dumps in den Arbeitsspeicher einlesen wollen (siehe Operand DOMAIN).

### LINK=linkname

Linkname einer UTM-Dump-Datei. Der Linkname darf nur aus alphanumerischen Zeichen bestehen.

DOMAIN= Dieser Operand legt fest, ob der gesamte Dump oder nur der KDCROOT-Bereich des UTM-Dumps in den Arbeitsspeicher eingelesen werden soll.

ALL Der gesamte UTM-Dump wird in den Speicher eingelesen.

Standardbelegung

ROOT KDCROOT-Bereiche mit den DIAGAREAs (UTM Diagarea, DB Diagarea,...) werden in den Speicher eingelesen.

## EDT Editor aufrufen

Der Editor EDT kann mittels dieser Anweisung als Unterprogramm aufgerufen werden, d.h. man kann damit eine Datei in den Arbeitsbereich des EDT einlesen.

---

```
EDT_      [ filename ]
```

---

filename Name der Datei, die eingelesen werden soll.

## END KDCDUMP beenden

Mit dieser Anweisung wird KDCDUMP ordnungsgemäß beendet.

---

```
END
```

---

## FGG Alle Dateien einer FGG aufbereiten

Mit der Anweisung FGG können alle Dateien einer FGG gemeinsam in einer Ausgabedatei aufbereitet werden. Der Name der Ausgabedatei wird mit dem Operanden OUTFILE festgelegt.

Im Kommandobetrieb können beliebig viele FGG-Kommandos angegeben werden.

---

```
FGG_      { fgg-name | *linkprefix }
          [,EDIT={ PRINTER | TERMINAL }]
          [,INFO= { LONG | DUMP | SHORT }]
          [,OUTFILE = filename ]
```

---

**fgg-name** Name einer Dateigenerationsgruppe (FGG), die den UTM-Dump enthält. Für jede Task der Anwendung erzeugt openUTM eine eigene Dateigeneration.

**\*linkprefix** Linknamenpräfix der Dateien, die aus einer Dump-FGG stammen, aber nur als Einzeldateien vorliegen. Das Präfix darf nur aus maximal drei alphanumerischen Zeichen bestehen. Der eigentliche Linkname hat die Form

linkprefix.zahl

wobei *zahl* eine maximal 4-stellige Dezimalzahl ohne führende Nullen ist. Begonnen wird mit '1', '2',... usw.

**EDIT=** Dieser Operand steuert die Aufbereitung:

Bei EDIT=PRINTER (Standard) wird eine Ausgabedatei erzeugt, die zum Ausdruck bestimmt ist. Die Ausgabe enthält Vorschubsteuerzeichen, Seitenköpfe und maximal 132 Zeichen pro Zeile. Das Inhaltsverzeichnis am Ende der Ausgabe bezieht sich auf Druckseiten und Zeilennummern (für den nicht ausgedruckten Dump). Die Datei lässt sich ausdrucken mit dem Kommando:

```
/PRINT-DOCUMENT FILE-NAME=listfile,
  LINE-SPACING=*BY-EBCDIC-CONTROL
```

Bei EDIT=TERMINAL wird die Ausgabedatei so aufbereitet, dass man sie mit einem Editor am Bildschirm auswerten kann. Die Zeilen enthalten maximal 80 Zeichen. Das Inhaltsverzeichnis am Ende der Ausgabe bezieht sich auf Zeilennummern.

- INFO=** Dieser Operand steuert die Ausgabe der Summary Information. Sie ist ein Extrakt der gesamten Dump-Information und enthält die Daten, die bei der Diagnose häufig benötigt werden.
- INFO=LONG: Aufbereitung mit Summary Information (Standard).  
 INFO=DUMP: Aufbereitung ohne Summary Information.  
 INFO=SHORT: KDCDUMP gibt nur Summary Information aus.
- OUTFILE=** Mit diesem Operanden können Sie festlegen, dass die Ausgabe in eine Datei mit dem Namen *filename* geschrieben werden soll.
- Ist OUTFILE nicht angegeben, prüft KDCDUMP, ob der Linkname LISTFILE vergeben ist bzw. vergibt den Standardnamen  
 LST.UTMDMP.rrrrrr.ttttff.aaaaaaaa (Erläuterung siehe [Seite 56](#)).

## Hinweise

- Bei der Eingabe der Steueranweisung FGG wird eine UTM-Dump-Datei, die mit dem Kommando DUMP eingelesen wurde, vollständig aus dem Speicher entfernt. Damit kann der vor dem FGG-Kommando untersuchte UTM-Dump nicht weiter diagnostiziert werden.
- Die einzelnen Dateigenerationen der FGG werden nacheinander gelesen und nach dem Bearbeiten wieder aus dem Speicher entfernt, so dass nach Ausführung des Kommandos keine Datei mehr im Speicher ist.
- Werden die Dump-Dateien mittels Linknamen eingelesen (Angabe von \*linkprefix), so wird nach jeder erfolgreichen Aufbereitung einer Dump-Datei der zugehörige Linkname aus der TFT entfernt.

## Beispiele

- Sie möchten alle Dateien einer Dump-FGG durch das KDCDUMP-Kommando FGG aufbereiten:

```
FGG DUMP.UTM.LKLC64.885301.ANW1,EDIT=T
```

- Sie möchten Dateien einer ehemaligen Dump-FGG aufbereiten, die nur noch als vier einzelne Dateien vorliegen.

Dazu weisen Sie den vier Dateien jeweils einen Linknamen mit Präfix UD1 zu:

```
SET-FILE-LINK LINK-NAME=UD1.1,FILE-NAME=DUMP.UTM.PCMM01.1
SET-FILE-LINK LINK-NAME=UD1.2,FILE-NAME=DUMP.UTM.WAITT2.2
SET-FILE-LINK LINK-NAME=UD1.3,FILE-NAME=DUMP.UTM.WAITT2.3
SET-FILE-LINK LINK-NAME=UD1.4,FILE-NAME=DUMP.UTM.WAITT2.4
```

Aufbereitet werden diese vier Dateien durch ein einziges KDCDUMP-Kommando:

```
FGG *UD1
```

## FILE Einzelne Dump-Datei aufbereiten

Mit dieser Steueranweisung wird eine einzelne Dump-Datei oder eine einzelne Generation einer UTM-Dump-FGG aufbereitet. Das Ergebnis der Aufbereitung wird in eine Ausgabe-datei geschrieben. Der Name der Ausgabedatei wird mit dem Operanden OUTFILE bzw. durch ein SET-FILE-LINK-Kommando des Anwenders mit dem Linknamen LISTFILE festgelegt. Bei fehlender Angabe erzeugt KDCDUMP eine Listdatei mit einem Standard-namen.

---

```
FILE_      [ { dumpfile | *linkname } ]
           [ ,EDIT={ PRINTER | TERMINAL } ]
           [ ,INFO= { LONG | DUMP | SHORT } ]
           [ ,OUTFILE = filename ]
```

---

dumpfile	Name der UTM-Dump-Datei. Diese Datei kann auch einer FGG angehören. Fehlt der Stellungsoperand, so wird vorausgesetzt, dass eine UTM-Dump-Datei mit der DUMP-Anweisung schon eingelesen worden ist. Auf diese wird dann die FILE-Anweisung angewandt. Die UTM-Dump-Datei wird dabei nicht aus dem Speicher entfernt.
*linkname	Linkname der UTM-Dump-Datei. Der Linkname darf nur aus maximal drei alphanumerischen Zeichen bestehen.
EDIT=	Dieser Operand steuert die Aufbereitung:  Bei EDIT=PRINTER (Standard) wird eine Ausgabedatei erzeugt, die zum Ausdruck bestimmt ist. Die Ausgabe enthält Vorschubsteuerzeichen, Seitenköpfe und maximal 132 Zeichen pro Zeile. Das Inhaltsverzeichnis am Ende der Ausgabe bezieht sich auf Druckseiten und Zeilennummern (für den nicht ausgedruckten Dump). Die Datei lässt sich ausdrucken mit dem Kommando:  <pre>/PRINT-DOCUMENT FILE-NAME=listfile,   LINE-SPACING=*BY-EBCDIC-CONTROL</pre> Bei EDIT=TERMINAL wird die Ausgabedatei so aufbereitet, dass man sie mit einem Editor am Bildschirm auswerten kann. Die Zeilen enthalten maximal 80 Zeichen. Das Inhaltsverzeichnis am Ende der Ausgabe bezieht sich auf Zeilennummern.
INFO=	Dieser Operand steuert die Ausgabe der SUMMARY-Information. Sie ist ein Extrakt der gesamten Dump-Information und enthält die Daten, die bei der Diagnose häufig benötigt werden.

INFO=LONG: Aufbereitung mit Summary-Information (Standard).

INFO=DUMP: Aufbereitung ohne Summary-Information.

INFO=SHORT: KDCDUMP gibt nur Summary-Information aus.

OUTFILE= Mit diesem Operanden können Sie festlegen, dass die Ausgabe in eine Datei mit dem Namen *filename* geschrieben werden soll.

Ist OUTFILE nicht angegeben, prüft KDCDUMP, ob der Linkname LISTFILE vergeben ist. Falls nicht, vergibt KDCDUMP den Standardnamen LST.UTMDMP.rrrrrr.ttttff.aaaaaaaa (Erläuterung siehe [Seite 56](#)).

### Hinweise

- Existiert beim Start von KDCDUMP ein Linkname KDCDUMP, so nimmt der KDCDUMP an, dass die zugehörige Datei ein UTM-Dump ist. Dieser wird dann aufbereitet. Danach beendet sich der KDCDUMP automatisch.
- Es ist zu beachten, dass eine UTM-Dump-Datei, die mit dem Kommando DUMP eingelesen worden ist, vollständig aus dem Speicher gelöscht wird, wenn die FILE-Steuerung mit dem Stellungsoperanden genutzt wird.
- Wird die Dump-Datei mittels Linknamen eingelesen (Angabe von \*linkname), so wird nach der erfolgreichen Aufbereitung der Dump-Datei der Linkname aus der TFT entfernt.

## HELP Hilfe zu KDCDUMP

Das Kommando HELP liefert Informationen zur Bedienung des Programms KDCDUMP. HELP gibt auch Informationen aus, die nur für den System-Service relevant sind und in diesem Kapitel nicht beschrieben werden.

Diese Informationen werden auf SYSOUT ausgegeben.

---

```
HELP_      [ { ALL | command-name | TABLE-NAMES } ]
```

---

HELP ohne Operanden gibt nur eine Liste aller KDCDUMP-Anweisungen aus. Die Operanden haben folgende Bedeutung:

**ALL**            Gibt eine Übersicht über die KDCDUMP-Anweisungen aus. Standardwert.

**command-name**

Name einer KDCDUMP-Anweisung, zu der eine Kurzinformation ausgegeben wird.

**TABLE-NAMES**

gibt alle gültigen Tabellennamen aus sowie eine zusätzliche Kurzinformation, ob die jeweilige Tabelle symbolisch aufbereitet werden kann.

Ist die symbolische Aufbereitung einer Tabelle möglich, so erscheint ein „D“ (sonst „N“).

Tabellengruppen werden mit einem dem Namen vorangestellten Stern (\*) gekennzeichnet. Die einzelnen Tabellen der Tabellengruppe werden nach dem Namen der Tabellengruppe aufgelistet und sind mit einem vorangestellten Plus (+) gekennzeichnet.

Befindet sich ein Dump im Speicher, dann wird die Anzahl der Tabelleneinträge ausgegeben (in hexadezimaler Form). Auf Grund von negativen Ergebnissen bei der Adressvalidierung beim Erzeugen eines Dumps kann es vorkommen, dass UTM-Bereiche oder Tabellen nicht im Dump enthalten sind. Da in solchen Fällen keine Tabelleneinträge vorhanden sind, werden die Tabellen wie folgt gekennzeichnet:

UA\_ERROR: Der UTM-Bereich, in dem die Tabelle liegt, konnte nicht abgezogen werden.

TA\_ERROR: Die Tabelle konnte nicht abgezogen werden.

REPFILe wird ebenfalls als Tabelle aufgelistet. Die REP-Datei muss jedoch über die Anweisung REPFILe ausgegeben werden (nicht über TABLE). Hinter REPFILe kommt die Anzahl der unterschiedlichen Problemnummern.



## LIST Tabellenausschnitt aufbereiten

Mit dieser Steueranweisung kann man einen Tabellenausschnitt auf eine SAM-Datei oder auf SYSLST schreiben lassen.

---

```
LIST_      { table-name | REPFIL } , { listfile | *SYSLST }
           [, START-INDEX = { FIRST | start } ]
           [, END-INDEX = { LAST | end } ]
           [, EDIT = { TERMINAL | PRINTER } ]
```

---

**table-name** Name der Tabelle, die ganz oder teilweise ausgegeben werden soll. Die gültigen Namen lassen sich mit HELP TABLE-NAMES erfragen. Als Tabellennamen können Sie auch einen Tabellengruppenamen angeben (bei der Ausgabe von HELP TABLE-NAMES mit einem Stern gekennzeichnet). In diesem Fall werden alle zu der Gruppe gehörenden Tabellen ausgegeben (gekennzeichnet durch ein vorangestelltes Plus (+)).

**REPFIL** Es soll die REP-Datei ausgegeben werden. Die Angabe weiterer Operanden ist dann wirkungslos.

**listfile** Der Tabellenausschnitt wird auf die Datei „listfile“ ausgegeben. Eine bestehende Datei wird ohne Warnung überschrieben.

**\*SYSLST** Der Tabellenausschnitt wird auf SYSLST geschrieben  
*listfile* bzw. SYSLST können mit folgendem Kommando ausgedruckt werden:

```
/PRINT-DOCUMENT <file>,LINE-SPACING=*BY-EBCDIC-CONTROL
```

**START-INDEX=**

Eintrag der Tabelle, mit dem die Ausgabe beginnen soll:

**FIRST** Erster Eintrag der Tabelle, Standardwert.

**start** Tabelleneintragsindex , bei dem die Ausgabe beginnen soll. Dieser Wert kann sowohl dezimal als auch hexadezimal eingegeben werden.

**END-INDEX=** Letzter Tabelleneintrag, der ausgegeben werden soll:

**LAST** Es wird bis Tabellenende ausgegeben, Standardwert.

**end** Tabelleneintragsindex, mit dem die Ausgabe beendet werden soll. Dieser Wert kann sowohl dezimal als auch hexadezimal eingegeben werden

**EDIT=** Aufbereitung für Drucker oder Terminal, siehe Beschreibung der Anweisung FGG auf [Seite 68](#).

**Hinweis**

Sollen mehrere Tabellenbereiche in eine Datei geschrieben werden, so bietet sich für den zweiten Stellungsoperanden der Operandenwert \*SYSLST an, denn mit folgendem BS2000-Kommando werden die Tabellenbereiche hintereinander in eine Ausgabedatei geschrieben:

```
ASSIGN-SYSLST TO-FILE=< output-file >, OUTPUT=EXTEND
```

## REPROFILE Informieren über REP-Datei

Mit dieser Anweisung können Informationen über die vom UTM-Dump-Erzeuger eingelesene REP-Datei auf dem Terminal ausgegeben werden.

---

```
REPROFILE_ [ { *INFO | *ALL | *ONLYPMNO | pm-no [ , E ] } ]
```

---

**\*INFO** Wenn die UTM-REP-Datei fehlerfrei eingelesen wurde, erhält man als Ausgabe den Namen der REP-Datei und die Anzahl der aktiven Korrekturen. Anderenfalls wird zusätzlich zum Namen der UTM-REP-Datei, die beim Start des Subsystems openUTM als REP-Datei eingebracht wurde, der DMS-Returncode ausgegeben, der beim Lesen dieser Datei erzeugt wurde.

Standardwert.

**\*ALL** Es wird die komplette REP-Datei angezeigt.

**\*ONLYPMNO**

Es werden alle Problemmeldungsnummern der UTM-REP-Datei in der Form ppppppppsjjj ausgegeben. Dabei ist:

pppppppp Nummer der Problemmeldung

s Zustand („-“ heißt aktiv, „\*“ heißt deaktiviert)

jjj julianisches REP-Datum

**pm-no** Acht Zeichen lange Problemmeldungsnummer (wie pppppppp) in Form eines C-Strings. KDCDUMP gibt die zugehörigen REP-Sätze der REP-Datei auf das Terminal aus (falls vorhanden).

**E** Damit werden alle REP-Sätze ab dem zu „pm-no“ gehörenden REP-Satz ausgegeben.

Dieser Operand muss immer zusammen mit einer Problemmeldungsnummer angegeben werden, andernfalls wird er ignoriert.

## SFIND String suchen

Mit dieser Anweisung können Sie im UTM-Dump einen String suchen und diesen auf dem Terminal ausgeben lassen.

Sie können aber nur Strings suchen, die zu den Nutzinformatoren im Dump zählen. Informationen, die vom Dump-Erzeuger KCSDUMP zur Aufbereitung des Dumps angelegt worden sind, werden nicht mit dem Suchstring verglichen.

Bei einigen Tabellen (Program Table, ...) werden die gefundenen Informationen aufbereitet ausgegeben, d.h. wenn der gesuchte String enthalten ist, dann wird die entsprechende Tabelle speziell aufbereitet angezeigt. Daher ist der gefundene String nicht immer unmittelbar zu sehen.



Bei großen Dumps können auch Treffer angezeigt werden, die in einem Bereich liegen, der hier nicht beschrieben ist und die daher nur für den Service interessant sind.

---

SFIND_	search [, ALIGN = { <u>1</u>   2   4   8}] [, HITS = { <u>ALL</u>   nr_max_hits}] [, DOMAIN = { <u>ALL</u>   ROOT } ]
--------	--

---

search	Der Stellungsoperand steht für die gesuchte Zeichenkette, die maximal 190 Bytes umfassen darf. Die Angabe kann als C- oder X-String erfolgen.
ALIGN =	Hiermit kann die Ausrichtungsgrenze des Suchstrings angegeben werden.
1	Byte-Ausrichtung. Standardbelegung
2	2-Byte-Ausrichtung
4	4-Byte-Ausrichtung
8	8-Byte-Ausrichtung
HITS =	Mit HITS kann angegeben werden, nach wieviel Treffern bei der Suche die Funktion beendet wird.
ALL	Der Dump wird vollständig durchsucht. Standardbelegung
nr_max_hits	Die Suche wird nach <i>nr_max_hits</i> Treffern beendet. <i>nr_max_hits</i> kann einen Wert zwischen 0 und 32767 annehmen. 0 wird wie ALL behandelt.
DOMAIN	Damit kann angegeben werden, ob der gesamte Dump oder nur der KDCROOT-Bereich des UTM-Dumps durchsucht werden sollen.

ALL	Gesamten Dump durchsuchen Standardbelegung
ROOT	KDC <b>ROOT</b> -Bereich mit den DIAGAREAs (UTM Diagarea, DB Diagarea, ...) durchsuchen

## SH und SYS KDCDUMP unterbrechen

Die Anweisung SH bzw. SYS unterbricht das Programm KDCDUMP und erlaubt die Eingabe von BS2000-Kommandos auf Betriebssystem-Ebene.

---

```
{ SH_ | SYS_ }
```

---

Mit dem BS2000-Kommando RESUME-PROGRAM können Sie zum Programm KDCDUMP zurückkehren.

Bitte beachten Sie, dass einige BS2000-Kommandos ein geladenes Programm beenden, so dass eine Rückkehr zu KDCDUMP nicht mehr möglich ist.

## SYSLST Protokollierung ein-/ausschalten

Diese Anweisung dient dazu, die Ergebnisse der Anweisungen AFIND, FIND, HELP TABLE-NAMES, REPFIL, TABLE auf SYSLST auszugeben

Die Länge einer Ausgabezeile beträgt maximal 80 Zeichen. Es werden keine Meldungen auf SYSLST geschrieben.

---

```
SYSLST_      { ON | OFF }
```

---

ON	Die Ausgabe wird auf SYSLST gegeben.
OFF	Keine Ausgabe auf SYSLST. Beim Start von KDCDUMP gilt „SYSLST OFF“.

Die Ausgabe auf das Terminal wird in keinem Fall unterdrückt.

## TABLE Tabelle anzeigen

Mit der Anweisung TABLE kann eine Tabelle des gerade zu bearbeitenden UTM-Dump teilweise oder ganz ausgegeben werden. Die Ausgabe kann im Dump-Format oder symbolisch erfolgen.

---

```
TABLE_      table-name
           [, start-index
           [, { END-INDEX = { SAME | LAST }      |
           DISPL = displacement                    |
           SDI = { *YES | struct-par }
           [, OUTFILE = { filename | *SYSLST } ]
           [, DB = db-no ]
           }
           ] ]
```

---

**table-name** Name der Tabelle, die ausgegeben werden soll.

Mit dem Kommando HELP TABLE-NAMES kann man sich eine Liste aller Tabellennamen anzeigen lassen.

Die Tabellennamen dürfen abgekürzt werden, solange die Eindeutigkeit gewahrt bleibt. Dabei gilt folgende Regel:

- Am Anfang und nach einem Bindestrich muss mindestens ein Zeichen vorhanden sein. Dieses Zeichen muss das erste Zeichen im ausgeschriebenen Namen sein.
- Die einzelnen Namensteile, die mit einem Bindestrich beginnen und durch den nächsten Bindestrich oder das Ende begrenzt werden, können entfallen, wenn der nachfolgende Namensteil - so weit einer vorhanden war - auch entfernt wird.

### *Beispiel*

Der Tabellename HLL-USER-ARE kann abgekürzt werden als HL-U oder HL-US oder H-U-A etc.

**start-index** Dieser Stellungsoperand gibt die Nummer des Eintrages an, ab dem die Ausgabe erfolgen soll.

Standardwert ist 0.

Erhält der Operand den Wert 0, so wird die ganze Tabelle ausgegeben, alle weiteren Operanden sind wirkungslos.

Die Eingabe kann hexadezimal oder dezimal erfolgen.

### Hinweise

- Bei der Angabe Null wird bei den Trace-Tabellen der aktuelle Eintrag als Letztes auf der Bildschirmseite angezeigt.
- Für Bit-Tabellen und Tabellen, deren Einträge immer zwei Bytes lang sind, kann i.a. nicht genau auf den gewünschten Eintrag positioniert werden. Der Eintrag befindet sich dann aber in der ersten ausgegebenen Zeile mit Tabelleninformationen.

END-INDEX= Mit diesem Operanden kann angegeben werden, ob außer dem bei „start-index“ angegebenen Eintrag noch weitere Tabelleneinträge angezeigt werden sollen.

SAME Es wird nur der Eintrag angezeigt, der mit „start-index“ angegeben wurde.

LAST Es wird die ganze Tabelle ab dem Eintrag „start-index“ angezeigt.

DISPL=displacement

Diese Möglichkeit kann man nutzen, um den im zweiten Stellungsoperanden angegebenen Tabelleneintrag mit einem Displacement ab Tabelleneintragsanfang auszugeben. Diese Möglichkeit wird nur für Tabellen im normalen Dump-Format, d.h. bei hexadezimaler Darstellung, unterstützt.

Das Eingabeformat ist dezimal oder hexadezimal (siehe Beispiel).

SDI = Der Operand SDI (**S**ymbolic **D**ump **I**nformation) dient zur Ausgabe der Tabelleneinträge, die mit Hilfe von AID aufbereitet wurden.

Es können *nicht* alle Tabellen symbolisch aufbereitet werden. Die Tabellen, die symbolisch aufbereitet werden können, werden bei der Ausgabe des Kommandos HELP TABLE-NAMES durch ein dem Tabellennamen folgendes „D“ gekennzeichnet. Ist eine symbolische Aufbereitung nicht möglich, so wird dies durch ein „N“ gekennzeichnet.

Weiterhin muss das Programm KDCDUMP mit dem Operanden TEST-OPTION=AID gestartet worden sein.

\*YES Das Tabellenelement wird symbolisch aufbereitet ausgegeben.

struct-par Vollqualifizierter Name eines Strukturelementes ohne Strukturnamen (siehe Beispiele [Seite 80](#)), das symbolisch aufbereitet werden soll.

Die maximale Länge von „structure-parameter“ beträgt 121 Zeichen. Der Name muss als C-String angegeben werden.

OUTFILE= Bei der symbolischen Aufbereitung kann mit diesem Operanden die Ausgabe auf eine Datei umgelenkt werden.

- filename Name der Datei, in die die aufbereitete Information geschrieben werden soll. Hierbei wird der AID-Linkname F0 des AID-Kommandos %OUTFILE zerstört.
- Existiert die angegebene Datei schon, so wird sie um die Ausgabe erweitert.
- \*SYSLST Die Ausgabe wird auf SYSLST gelenkt.
- DB=db-no Dieser Operand wird nur bei den ROOT-Tabellen TAM und TSKM ausgewertet. Er zeigt bei einer UTM-Anwendung, bei der mehrere Datenbanken generiert sind, den jeweiligen Bereich der oben aufgeführten Tabellen. Der Wert 0 steht für den allgemeinen Bereich.
- Standard: 0
- Für die optionalen Operanden END-INDEX, DISPL und SDI gilt der Standardwert END-INDEX = SAME.

### Hinweise

- Sollte die ausgesuchte Tabelleninformation nicht auf eine Bildschirmseite passen, so können mit Hilfe der Blätter-Anweisungen (siehe dort) weitere Nutzdaten sichtbar gemacht werden (gilt nicht für die symbolische Aufbereitung).
- Bei einer fehlerhaften Angabe eines Unterstrukturelementes für die symbolische Aufbereitung wird durch AID eine Fehlermeldung ausgegeben. In diesem Fall wird weder die Meldung K746 ausgegeben noch der Auftragschalter 3 gesetzt.

### Beispiele

1. T KB,1,S=\*YES  
Ausgabe des KB-Kopfs und KB-Rückgabebereichs.
2. T KB,1,S='KCRFELD'  
Ausgabe der Unterstruktur KB.KCRFELD.
3. T KB,1,D=116  
Ausgabe des KB-Programmbereichs ab Displacement 116.
4. T UTM-D,X'2EB'  
Ausgabe des Trace-Eintrags 2EB der UTM Diagarea.

## 3.2.3 Meldungen von KDCDUMP

Das Dienstprogramm KDCDUMP gibt Meldungen K7nn aus, die Meldungstexte und Zusatzinformationen zu den Meldungen findet man in weiteren Kapiteln dieses Handbuchs.



### 3.3 Inhalt des UTM-Dumps

Ein mit KDCDUMP aufbereiteter UTM-Dump enthält in der Regel folgende Informationen:

- Den KDCROOT-Bereich mit den KDCROOT-Tabellen und den DIAGAREAs (UTM Diagarea, DB Diagarea,...).
- Eine Zusammenfassung (Summary Information).
- Ein Inhaltsverzeichnis

In einigen Fällen kann der Dump noch zusätzliche Bereiche enthalten, die nur durch den System-Service ausgewertet werden können und daher in diesem Handbuch nicht beschrieben sind (Ausnahme CONS\_ENTRIES). Falls erforderlich sollten diese Bereiche dem System-Service zur Verfügung gestellt werden, siehe [Abschnitt „Fehlerunterlagen erstellen“ auf Seite 47](#).

Wird der Dump auf Grund eines PEND ER erzeugt (entweder vom Anwender programmiert oder intern erzeugt nach  $KCRCCC \geq 70Z$ ), enthält der Dump nur den KDCROOT-Bereich.

Zum leichteren Auffinden der Tabellen enthält der aufbereitete Dump am Ende eine Liste aller Tabellen mit den Seitennummern und Zeilennummern. Jede dieser aufbereiteten Tabellen hat eine Überschrift, welche die aufbereitete Tabelle identifiziert.

Die Überschrift beginnt jeweils mit dem abgezogenen Speicherbereich (z.B. KDCROOT) sowie dem Tabellennamen.

Im Folgenden sind die Tabellen beschrieben, die Sie in einem PENDER-Dump finden. Zusätzlich ist die Tabelle CONS\_ENTRIES aufgeführt, die bei Dumps mit anderen Ursachen erstellt wird und nützliche Informationen für den Anwender enthält.

### 3.3.1 Die Tabelle CONS\_ENTRIES

Die Tabelle CONS\_ENTRIES zeigt die Historie der KDCFILE an. Sie enthält maximal 10 Einträge. Die Einträge sind nach absteigender Erzeugungszeit sortiert, d.h. der jüngste Eintrag steht vorne. Falls für die KDCFILE noch keine 10 Einträge erstellt wurden, dann sind die ersten Einträge dieser Tabelle leer.

Einträge in die Tabelle werden bei folgenden Ereignissen geschrieben:

- bei der KDCDEF-Generierung
- bei einem Aufruf von KDCUPD
- bei jedem Start der UTM-Anwendung

Ein Tabelleneintrag ist folgendermaßen aufgebaut:

typ	r1	r2	f1	i1	i2	i3	f2	yymmdd	hhmmss	tttt	Byte
4	1	1	2	1	1	1	1	6	6	4	

typ 'DEF' für KDCDEF-Lauf  
'UPD' für KDCUPD-Lauf  
'STRT' für Anwendungsstart.

r1 und r2 bezeichnen den Korrekturzustand von KDCDEF, KDCUPD und UTM-Systemcode (z.B. '00' für V6.5A00).

r1,r2 'nm' zwei Ziffern als Source-Korrekturzeichen  
' ' Leerzeichen für die Erstfreigabe einer Version.

f1, f2 Platzhalter (filler).

Für *typ* 'STRT' und 'DEF' enthalten i1,i2, i3 und ttt zusätzliche Informationen.

Für *typ* = STRT

i1 'C' für UTM-Kaltstart, 'W' für UTM-Warmstart

i2 'P' für Produktion

i3 'B' für Batch-Task, 'D' für Dialog-Task

tttt TSN der UTM-Task bei *typ* = STRT.

Für *typ* = DEF

i2 '-' wenn die KDCFILE erfolgreich erzeugt wurde  
'W' wenn die KDCFILE fehlerhaft erzeugt wurde.

yymmdd  
Datum (Jahr, Monat, Tag).

hhmmss  
Uhrzeit (Stunde, Minute, Sekunden).

### 3.3.2 Der KDCROOT-Bereich

Die Tabellen und Bereiche des KDCROOT können in vielen Fällen zur Diagnose von Anwenderfehlern herangezogen werden. Bei PENDER-Dumps werden nur diese Daten zur Verfügung gestellt.

Der aufbereitete Dump enthält im Einzelnen folgende KDCROOT-Bereiche, in Klammern steht der Name, der bei der KDCDUMP-Anweisung TABLE angegeben werden muss:

#### Context Area (CONTEXT-AREA)

Hier wird der zum Zeitpunkt des STXIT aktuelle Registersatz ausgegeben, mit PC=Program-Counter, IW=Interrupt-Weight und den Registerinhalten von Register R0 bis R15, ILC , PM und CC.

#### Program Table (PROGRAM-TABL)

Programmtabelle. Ein Eintrag enthält u.A. den Programmnamen und die Startadresse eines Teilprogramms.

#### Load Module Table (LOAD-MODULE)

Bereich mit Information zu den Lademodulen.

#### Load Context Table (LOAD-CONTEXT)

Ladekontext-Tabelle.

#### Area Table (AREA)

Bereich mit Informationen zu den generierten Areas

#### Exit Table (EXIT)

Bereich mit Informationen zu den Exits

#### Library Table (LIB)

LMS-Bibliothekstabelle

#### Memory Pool Table (MEMORY-POOL)

Bereich mit Informationen zu den Memory Pools. Im Falle einer UTM-Generierung ohne Lademodule steht hier nur die Information aus der UTM-Generierung, weitere Information (z.B. die Adresse des Memory Pools) ist im Bereich User Root zu finden.

#### Message Mod Table (MSG-MODULE)

Bereich mit Informationen zu den Meldungsmodulen

#### UTM Diagarea (UTM-DIAGAREA)

Bereich mit Diagnose-Informationen zu allen KDCS- und FHS-Aufrufen

#### KB (KB)

Kommunikationsbereich, bestehend aus KB-Kopf, KB-Rückgabeinformation und KBPROG in der generierten Länge

- SPAB (SPAB)
  - Standard-Primärer Arbeitsbereich
- MPUT Buffer (MPUT-BUFFER)
  - Zwischenspeicher für MPUT-Nachrichten
- FORMUSER Buffer (FORMUSER-BUF)
  - Pufferbereich mit logischer Ein-/Ausgabe-Nachricht
- RESTART Buffer (RESTART-BUFF)
  - Wiederanlaufbereich für die Bildschirmformatierung
- IO Buffer (IO-BUFFER)
  - Pufferbereich mit physikalischer Ein-/Ausgabe-Nachricht
- ROOTDATA (ROOTDATA)
  - Verständigungsbereiche zwischen KDCROOT und den UTM-Systemmodulen
- ROOT Trace (ROOT-TRACE)
  - Bereich mit Trace-Records zum Root-Ablauf
- FORM USER Area (FORM-USER-AR)
  - Ausgabe der IUTMFORM-Schnittstellen-Parameter
- HLL User Area (HLL-USER-ARE)
  - Parameterliste der IUTMHLL
- VGM Area (VGM-AREA)
  - Bereich für das Vorgangsmemory einer angeschlossenen Datenbank
- Root gen by user (USER-ROOT)
  - Bereich enthält Daten aus dem benutzereigenen Root-Modul
- XA Area (XA-AREA)
  - Datenbankbereich
- TCB-ENTRY (COB1) (TCB-ENTRY)
  - Bereich für TCB-Entries, falls generiert
- Table Descriptors (TABDESC-AREA)
  - Bereich mit Daten zu den aufbereiteten Root-Tabellen (enthält Name, Adresse, Anzahl der Entries und Länge)
- Administration DIAGAREA (ADMI-DIAGAREA)
  - Bereich mit Trace-Records für alle Aufrufe der Programmschnittstelle der Administration aus den Teilprogrammen
- Administration USERAREA (ADMI-USERAREA)
  - Bereich mit einem Trace-Record für die an der Programmschnittstelle der Administration vom Teilprogramm übergebenen Daten

**LOG-EXIT-MESSAGE-BUFFER (LOGEXITBUFFER)**

Bereich für Logging-Exit

**ETPNDS-TU-LIST (ETPNDS-TU-LIST)**

Teil-Liste der im KDCROOT eingebundenen Komponenten mit ihren ETPNDs

**ENTER-PROC (ENTER-PROC)**

Die aufgesammelten Eingaben aus den UTM-Startparametern

ENTER-PROC-INPUT

**STRT-PAR-AREA (STRT-PAR)**

Bereich mit den zum Start der Anwendung angegebenen Start-Parametern

**Accounting Area (ACCOUNTING-A)**

Bereich für Abrechnungsdaten

**TAM (TAM)**

Transaktionsspeicher für eine angeschlossene Datenbank

**TSKM (TSKM)**

Task-spezifischer Speicher für die Datenbankkommunikation

**DB Diagarea (DB-DIAGAREA)**

Bereich mit Diagnose-Informationen zu allen Datenbank-Aufrufen (nur wenn Datenbank generiert ist)

**DB User Area (DB-USER-AREA)**

Parameterliste der IUTMDB (nur falls DB generiert)

**DB Info Program Unit (DB-INF-PROG)**

Bereich enthält Daten zum aktuellen Teilprogramm

**DB Info Application (DB-INF-APPL)**

Bereich enthält Daten zur Anwendung

**DB Summary Info (DB-SUMMARY)**

Bereich mit allgemeinen Informationen zu den generierten Datenbanken (z.B. Anzahl etc.)

**DB Table (DB-TABLE)**

Tabelle der generierten Datenbanken

Ferner wird der Inhalt der REP-Datei  $\$userid.SYSREP.UTM.065$  ausgegeben, so weit sie beim Ablauf der Anwendung vorhanden und zugreifbar war.

### 3.3.2.1 Program Table

Die Einträge haben folgenden Aufbau und Bedeutung:

Stelle	Byte (hexadez.)	Bedeutung
1	-	Index des Eintrages in der ROOT-Programm-Tabelle
2	24	Name des Programms
3	49	Sprachtyp des Programms wie in der Anweisung PROGRAM... COMP= angegeben. Mögliche Werte <sup>1</sup> X'00' = ASSEMB            X'06' = C X'01' = COB1            X'08' = FOR1 X'02' = SPL4            X'0B' = ILCS X'04' = PLI/1 X'05' = Pascal-XT
4	48	Lademodus des Programms wie in der Anweisung PROGRAM .. LOAD= angegeben. Falls die Anwendung mit Lademodulen generiert ist, entspricht diese Ausgabe dem in der Anweisung LOAD-MODULE (Parameter LOAD-MODE) angegebenen Wert. Bedeutung der Werte <sup>1</sup> : STATIC = X'00': Programm statisch zur Anwendung gebunden STARTUP = X'01': Programm wird beim Start der Anwendung nachgeladen ONCALL = X'02': Programm wird beim ersten Ansprung (Aufruf durch TAC) geladen POOL = X'03': Programm wird in einen Memory Pool geladen.
5	4C	Austauschmodus des Programms abhängig vom Lademodus. Bedeutung der Werte <sup>1</sup> : NOTCH = X'00': Programm nicht austauschbar LLMSI = X'01': Programm in einem austauschbaren Lademodul LLMGR = X'02': (derzeit nicht verwendet) APPLI = X'03': Programm in einem Lademodul, der in einen lokalen Pool geladen wird.
6	44	Hexadezimaler Index des Lademoduls (LMOD), in den dieses Programm eingebunden ist laut UTM-Generierung.
7	4E	Hexadezimaler Index des nächsten Programms im gleichen Lademodul
8	46	Hexadezimaler Index der Bibliothek, aus der das Programm bzw. das Lademodul nachgeladen wurde.
9	4D	Ladestatus des Programms. Bedeutung der Werte <sup>1</sup> : NOT_LOADABLE = X'00': Programm kann nicht geladen werden LOADED = X'01': Programm ist geladen LOAD_ONCALL = X'02': Programm wird bei Ansprung geladen RELOAD_ONCALL = X'03': (derzeit nicht verwendet)
10	58	Adresse des Programms

<sup>1</sup> Die Hexadezimalwerte sind bei einem CDUMP relevant.

### 3.3.2.2 Load Module Table

Die Einträge haben folgende Bedeutung:

Ab Byte (hexadez.)	Bedeutung
0-23	(nicht relevant)
24-43	Name des Lademoduls
44	Lademodus des Lademoduls, mögliche Werte: X'00' = Lademodul statisch zur Anwendung gebunden X'01' = Lademodul wird beim Start der Anwendung nachgeladen X'02' = Lademodul wird beim ersten Anspung geladen X'03' = Lademodul wird in einen Memory Pool geladen.
45	Autolink-Funktion des Lademoduls (ALTLIB), mögliche Werte: X'00' = Kein Suchen in ALTLIB zur Befriedigung von Externverweisen X'01' = Suchen in ALTLIB zur Befriedigung von Externverweisen
46	Austauschmodus des Lademoduls, mögliche Werte: X'00' = Lademodul nicht austauschbar X'01' = Lademodul einzeln austauschbar X'02' = (derzeit nicht verwendet) X'03' = Lademodul nur mit gesamter Anwendung austauschbar, da in lokalen Pool geladen.
47-5E	generierte Version des Lademoduls
5F-77	Version des zu ladenden Lademoduls (bei Austausch)
78-79	Index des Contexts, zu dem dieses Lademodul gehört
7A-7B	Index des Memorypools (MPOOL), in den dieses Lademodul geladen wird.
7C-7D	Index der Bibliothek (LIB), aus der dieses Lademodul geladen wurde
84-85	Index des ersten Programms dieses Lademoduls
88-89	Index des nächsten Lademoduls im gleichen Context.
8A-8B	Index der ersten AREA in diesem Lademodul
8C-A3	Version des aktuellen (=zuletzt geladenen) Lademoduls
A4	Ladestatus des Lademoduls: X'00' = Lademodul nicht geladen X'01' = Lademodul ist geladen X'02' = Lademodul wird bei Anspung eines Programms geladen, das in diesen Lademodul gebunden ist. X'03' = (derzeit nicht verwendet)

Ist die generierte Version = Version des zu ladenden Lademoduls, so bedeutet dies, dass seit dem letzten KDCDEF-Lauf kein Austausch dieses Lademoduls stattgefunden hat.

### 3.3.2.3 UTM Diagarea

Die UTM Diagarea ist ein Task-spezifischer Trace-Bereich, in den alle Ereignisse protokolliert werden. Dieser Bereich enthält somit auch alle Ereignisse, die unmittelbar vor dem Abbruch eines Vorgangs oder einer Anwendung aufgetreten sind.

Die UTM Diagarea wird zyklisch beschrieben. Zwei Zyklen werden durch eine Trennlinie bestehend aus '='-Zeichen und Leerzeichen getrennt. Oberhalb der Trennlinie steht der jüngste Eintrag und unterhalb davon der älteste Eintrag. Jeder Eintrag ist 136 Bytes lang. Wieviele Einträge insgesamt in die UTM Diagarea passen, ist abhängig vom UTM-Generierungsparameter `MAX TRACEREC`.

In die UTM Diagarea werden folgende Arten von Einträgen geschrieben:

- UTM-Records (Typ `KDCS`)

UTM-Records werden bei folgenden Ereignissen geschrieben:

- bei einem `KDCS`-Aufruf in einem Teilprogramm oder
- bei einem internen Aufruf an den UTM-Systemcode oder
- bei einem internen `PEND ER`-Aufruf durch `openUTM` (System-`PEND ER`) auf Grund eines schwer wiegenden Fehlers oder

- UTM-Records beim Starten und Beenden von benutzerspezifischen Event-Exits:

- `FORMAT`-Exit-Programm (Typ `FOXS` und `FOX`)
- `INPUT`-Exit-Programm (Typ `INXS` und `INX`)
- `LOGGING`-Exit-Programm (Typ `LGXS` und `LGX`)
- `START`-Exit-Programm (Typ `STXS` und `STX`)
- `VORGANG`-Exit-Programm (Typ `VGXS` und `VGX`)

Im Falle eines System `PEND ER` enthält der Eintrag in den Bytes 22 - 57 einen Fehlertext.

Für Aufrufe der Administrationschnittstelle (`KDCS-Opcode=ADMI`) werden zusätzliche Trace-Informationen geschrieben ([Abschnitt „Administration DIAGAREA“ auf Seite 104](#)).

- UTM-Records zur Vorgangsidentifizierung (Typ `VGID`)

Ein `VGID`-Record wird bei jedem Start eines Teilprogramms und bei Rückkehr eines `PGWT`-Aufrufes in die UTM Diagarea geschrieben.

- FORM-Records (Typ `FHCL`)

Ein `FORM`-Record wird bei einem Aufruf an ein Formatierungssystem geschrieben.



## Header der Records in der UTM Diagarea

Jeder Record beginnt mit einem Header, der folgende Informationen enthält:

Byte		Bedeutung
dez.	hexadez.	
0-1	00-01	Zähler des aktuellen Eintrages in den DIAGAREAs (UTM und DB)
2-5	02-05	Typ-Identifizierung (KDCS, VGID, FOXS, FOXE, INXS, INXE, LGXS, LGXE, STXS, STXE, VGXS, VGXE, FHCL, ITRC)
6-7	06-07	Derzeit nicht genutzt (mit '=' vorbelegt)
8-15	08-0F	Zeitstempel

Aufbau des Headers

Der Inhalt der Records ab Byte 16 ist abhängig vom Typ des Records.

## UTM Diagarea bei KDCS-Aufruf aus einem Teilprogramm (Typ KDCS)

Byte		Feldname und Bedeutung	
dez.	hexadez.		
16-19	10-13	KCOP <sup>1</sup>	USER-Operationscode: INIT, MGET, MPUT, usw.
		:	interner Operationscode: siehe Tabelle <a href="#">Seite 91</a>
20-21	14-15	KCOM: Operationsmodifikation	
22-23	16-17	KCLA: Bereichslänge oder Queue Level (bei QCRE) oder KCLKBPRG: Länge KB-Programmbereich beim INIT	
24-25	18-19	KCLM: Nachrichtenlänge oder KCLPAB: Länge des Standard Primären Arbeitsbereichs beim INIT. KCWTIME: Wartezeit in Sekunden beim DGET	
26-33	1A-21	KCRN: Bezugsname	
			MCOM-Aufruf
34-41	22-29	KCMF: Formatname oder Editprofilname KCLT: LTERM-Name des LTERM-Partners oder KCUS: Benutzerkennung oder KCPA: Name der Partner-Anwendung (beim APRO-Aufruf)	KCPOS: Ziel einer positiven Folgennachricht

Aufbau eines Eintrags der UTM Diagarea bei einem KDCS-Aufruf (Typ KDCS)

Byte		Feldname und Bedeutung					KCFELD	KCNEG:
dez.	hexadez.	DPUT/DADM	DGET	QCRE	PADM	APRO		
42-43	2A-2B	KCDF: Bildschirmfunktion, siehe Tabelle auf <a href="#">Seite 91</a>					KCNEG: Ziel einer negativen Folgenach- richt	
44	2C	KCMOD: Modus ("A"/"R"/"_"")	KCQTYP: Zieltyp ("U"/"Q"/"T")	KCQMODE: Modus ("S"/ "W"/binär null)	KCACT: Aktion (ON/ OFF/CON/DI S)	KCPI: Vorgangs- identi- fikation		
45-47	2D-2F	KCTAG: Tag	leer	leer	KCADRLT: Neuer LTERM- Name des Druckers	KCOF: OSI TP Funktio- n	KCCOMID: Komplex- Identifikatio- n	
48-49	30-31	KCSTD: Stunde						
50-51	32-33	KCMIN: Minute						
52-53	34-35	KCSEC: Sekunde						
54	36	KCQTYP: Zieltyp ("U"/"Q"/binär null)						
55-57	37-39	leer	leer	leer	leer	leer	leer	
58-89	3A-59	KCRFELD in KCKBC: KB-Rückgabebereich						
92-95	5C-5F	Rücksprungadresse zum Teilprogramm (Die Adresse zeigt hinter KDCS-Aufruf im Teilprogramm)						
96-99	60-63	Adresse des Benutzerdatenbereichs (2. Parameter bei KDCS-Aufruf).						
100-103	64-67	Vorgangs-Index						
104-111	68-6F	KCLOGTER in KCKBC: LTERM - Name						
112-119	70-77	KCBENID in KCKBC: Name der aktuellen Benutzererkennung						

Aufbau eines Eintrags der UTM Diagarea bei einem KDCS-Aufruf (Typ KDCS)

<sup>1</sup> Bei KCOP=INFO (Byte 16-19) und Operationsmodifikation KCOM=CK (Byte 20-21) wird ohne Erhöhung des Zählers der Nachrichtenbereich (NB) des protokollierten Aufrufs in der Länge KCPAC in den nächsten Eintrag der UTM Diagarea geschrieben. Diese Information ist für die Diagnose interessant, da sie den zu prüfenden Aufruf protokolliert.

Im Feld KCDF (Byte 42-43; Bildschirmfunktion) sind folgende Einträge möglich:

Inhalt KCDF, hexadezimal	symbolischer Name	Bedeutung
0000	-----	keine Bildschirmfunktion
0001	KCREPL	Bildschirm löschen und Format ausgeben
0001	KCRESTRT	Bildschirmwiederanlauf nach PEND RS
0002	KCERAS	Löschen ungeschützter Felder
0004	KCALARM	BEL Funktion
0008	KCREPR	Lokale Druckerausgabe
2000	KCEXTEND	Extended Linemode (erweiterter Zeilenmodus)
4000	KCCARD	Nächste Eingabe vom Ausweisleser

Bildschirmfunktionen

### Opcode bei internem Aufruf an openUTM

Inhalt KCOP	Situation, in der dieser Record geschrieben wird	Inhalt der Felder nach KCOP
STRT	Start des UTM-Anwendungsprogramms, Beginn Start-Behandlung im UTM-Systemcode	- - -
WAIT	UTM-Task wartet im Systemcode auf den nächsten Auftrag	- - -
CONT	Fortsetzung im UTM System Code nach Ausführung der Formatierung oder einer DB-Aktion über KDCROOT oder Aufruf des INPUT-Exits	KCRCCC, KCRCKZ,KCRCDC; beim INPUT-Exit die Parameter, siehe <a href="#">Seite 93</a>
NOOP	Pufferbereich der MESSAREA muss geleert werden (nur möglich bei eingeschaltetem Mess-Monitor)	- - -
ADMI	UTM-Administrationsaktion	UTM-interne Schnittstelle

Operationscodes bei einem internen UTM-Aufruf

### Aufbau von KCRFELD

Byte		Feldname und Bedeutung		
dez.	hexadez.			
58-59	3A-3B	KCRDF: Rückgabe Bildschirmfunktion KCRWVG: Anzahl wartender Vorgänge beim DGET		
60-61	3C-3D	KCRMLM: tatsächliche Länge der Nachricht		
		INFO CK-Aufruf	MGET-Aufruf	SIGN ON-Aufruf
62	3E	KCRINFCC: KDCS-Returncode des geprüften KDCS-Aufrufs	KCVGST: Vorgangstatus	KCRSIGN1: Primärkode
63	3F		KCTAST: Transaktionsstatus	KCRSIGN2: Sekundärkode
64	40		leer	
65	41	leer	KCRMGT: Nachrichtentyp	
66-68	42-44	KCRCCC: KCDS-Returncode		
69	45	KCRCKZ: Kennzeichen: P (Produktivanwendung)		
70-73	46-49	KCRCDC: interner Returncode		
74-81	4A-51	KCRMF: Rückgabe Formatkennzeichen, Editprofil oder abstrakte Syntax		
82-89	52-59	KCRPI: Rückgabe Vorgangs-Identifikation KCRUS: Benutzerkennung beim SIGN ST oder Erzeuger der Nachricht beim DGET KCRQN: Von openUTM vergebener Name der Temporären Queue bei QCRE NN		

### UTM Diagarea nach einem Aufruf des INPUT-Exits

Nach einem Aufruf des INPUT-Exit trägt openUTM in den KDCS-Record die wichtigsten Parameter ein, die der Exit bekommt bzw. setzt (KCPAC):

Byte		Feldname und Bedeutung
dez.	hexadez.	
16-19	10-13	KCOP: Hier wird "CONT" eingetragen
20-25	14-19	leer
26-33	1A-21	KCIFCH: Die ersten 8 Zeichen der Eingabe
34-35	22-23	KCICVST: Vorgangstatus: "ES"/"ET"/"RS"/"EC"
36-37	24-25	KCIFKEY: Wert der F-Taste: 1,...,24
38-39	26-27	KCIKKEY: Wert der K-Taste: 1,...,14
40-41	28-29	KCICFINF: Information über Steuerfelder: "UN"/"NO"/"ON"/"MO"
42-49	2A-31	KCINTAC/KCINCMD: Nächster zu startender TAC bzw. Benutzer-Kommando
50-51	32-33	KCICCD: Code für die Wirkung der Eingabe: "ER"/"CC"/"SC"/"ST"/"CD"
52	34	KCICUT: TAC abschneiden: "Y"/"N"
53	35	- - -
54-57	36-39	KCIERRCD: Fehlerinfo für Datensichtstation
58-...	3A-...	Folgendes ist für die Diagnose nicht relevant

Aufbau eines Eintrags der UTM Diagarea nach einem Aufruf des INPUT-Exits

*Fehlertexte in der UTM Diagarea bei SYSTEM-PEND ER*

Zur schnelleren Diagnose bei einem SYSTEM-PEND ER ist in der entsprechenden Zeile der UTM Diagarea in Byte 22-57 ein abdruckbarer Fehlertext abgelegt.

In der folgenden Tabelle sind alle möglichen Fehlertexte aufgeführt, dazu jeweils die Fehlerursache sowie mögliche Maßnahmen und die Fehlerart (Systemfehler oder Anwenderfehler).

<b>Fehlertext</b>	<b>Ursache</b>	<b>Fehlerart</b>
APPL. PROGRAM DOES NOT EXIST	Beim Start eines Teilprogramms war kein Indikator (KCRARA) für ein Teilprogramm vorhanden (70Z mit KR01).	Systemfehler
APPL. PROGRAM WITHOUT PEND	Der Teilprogrammlauf wurde nicht mit PEND beendet. (84Z)	Anwenderfehler
ASYNC. PROGRAM NOT FOUND	Asynchron-Programm ist nicht mehr verfügbar (evtl. Programmaustausch). PENDER-Dump mit 70Z und KR02 folgt.	Anwenderfehler
ERROR IN "START-TP" OF LGCON	Beim Start eines Teilprogramms liefert das Language-Connection-Modul einen fehlerhaften Returncode.	Systemfehler
KB END LABEL OVERWRITTEN	Das Teilprogramm verwendet den KB in einer größeren Länge als in der UTM-Generierung festgelegt (70Z mit KR04) Maßnahme: Parameter MAX KB in der UTM-Generierung ändern oder Teilprogramm anpassen.	Anwenderfehler
SPAB END LABEL OVERWRITTEN	Das Teilprogramm verwendet den SPAB in einer größeren Länge als in der UTM-Generierung festgelegt (70Z mit KR05). Maßnahme: Parameter MAX SPAB in der UTM-Generierung ändern oder Teilprogramm anpassen.	Anwenderfehler
KCRCCC >= 70Z AFTER SVC UTM	Nach SVC UTM wurde der Returncode KCRCCC im KB-Kopf auf $\geq 70Z$ gesetzt	Anwender-/Systemfehler
ERR IN STXIT-ROUTINE OF LGCON	Bei der STXIT-Behandlung wurde vom Language-Connection-Modul ein fehlerhafter Returncode geliefert	Systemfehler
STXIT WITH XT.. ENTERED	Es ist ein STXIT mit dem Gewicht .. aufgetreten.	Anwender-/Systemfehler

Fehlertexte in der UTM Diagarea bei SYSTEM-PEND ER

Fehlertext	Ursache	Fehlerart
DB ERRORCODE = TA_CHAIN_RSET	Das Datenbank-Verbindungsmodul lieferte den Returncode TA_CHAIN_RSET. Wurde dieser Returncode durch eine K210, K211, K216 mit XAER_DUPID ausgelöst, ist es möglich, dass ein abnormales Ende des vorherigen Anwendungslaufs diese Transaktion in der Datenbank hinterließ und in diesem Zustand eine neue KDCFILE generiert wurde. Mögliche Maßnahme: Herunterfahren und Neustarten der Datenbank	Systemfehler
KDCS CALL IN VORGANG EXIT	Im VORGANG-EXIT-Programm wurde ein unerlaubter KDCS-Aufruf gemacht.	Anwenderfehler
VORGANG-EXIT-PROGRAM NOT LOADED	Das Programm für den Vorgang-Exit ist nicht geladen.	Anwenderfehler
KDCS CALL IN FORMAT EXIT	Im FORMAT-EXIT-Programm wurde ein unerlaubter KDCS-Aufruf gemacht.	Anwenderfehler
DATABASE DOWN AT USER DB CALL	Bei einem Aufruf des Teilprogramms an das DB-System ist das DB-System nicht mehr konnektiert.	Anwender-/ Systemfehler
ILLEGAL RET CODE FROM DBCON	Das Datenbank-Verbindungsmodul setzt einen unerlaubten Returncode.	Systemfehler
NO DB CALL ALLOWED IN SIGN-ON	In einem Anmeldevogang wurde ein unerlaubter DB-USER-CALL-Aufruf gemacht.	Anwenderfehler
DYNAM. PROGRAM LOADING FAILED	Beim Nachladen eines Programmes trat ein Fehler auf. PENDER-Dump mit 70Z und KR09	Anwender-/ Systemfehler

Fehlertexte in der UTM Diagarea bei SYSTEM-PEND ER

**Eintrag zur Vorgangs-Identifikation (Typ VGID)**

Byte		Bedeutung
dez.	hexadez.	
16	10	Vorgangskennzeichen
17	11	Sessionzähler
18-19	12-13	Transaktionszähler im Vorgang
20-23	14-17	Vorgangszähler
24-27	18-1B	Summe von USED und ERROR für den aktuellen TAC
28-31	1C-1F	Länge des Global Transaction Identifier (GTRID) der XID
32-35	20-23	Länge des Branch-Qualifier (BQUAL) der XID
36-115	24-73	die ersten 80 Bytes der XID-Daten *)
116-117	74-75	Index zur Programm-Tabelle
118-119	76-77	Index des Vorgangsexit in Programm-Tabelle
120-127	78-7F	Name des Transaktionscodes, der den Vorgang gestartet hat
128-135	80-87	Name des aktuellen Transaktionscodes

Aufbau des Eintrags zur Vorgangs-Identifikation

\*) Details zu XID, GTRID und BQUAL siehe betreffende XA-Spezifikation



**Eintrag zum VORGANG-Exit (Typ VGXS und VGXE)**

Byte		Bedeutung
dez.	hexadez.	
16	10	Vorgangskennzeichen
17	11	Sessionzähler
18-19	12-13	Transaktionszähler im Vorgang
20-23	14-17	Vorgangszähler
24-27	18-1B	Summe von USED und ERROR für den aktuellen TAC
28-31	1C-1F	Länge des Global Transaction Identifier (GTRID) der XID
32-35	20-23	Länge des Branch-Qualifier (BQUAL) der XID
36-115	24-73	die ersten 80 Bytes der XID-Daten *)
116-117	74-75	Index zur Programm-Tabelle
118-119	76-77	Index des Vorgangsexit in Programm-Tabelle
120-127	78-7F	Name des Transaktionscodes, der den Vorgang gestartet hat
128-135	80-87	Name des VORGANG-Exit-Programmes

Aufbau des Eintrags zum VORGANG-Exit

\*) Details zu XID, GTRID und BQUAL siehe betreffende XA-Spezifikation

**Eintrag zum FORMAT-Exit (Typ FOXS und FOXE)**

Byte		Bedeutung
dez.	hexadez.	
16-31	10-1F	String FORMATEXIT
32-39	20-27	Programm-Name des FORMAT-Exits
40-47	28-2F	Format-Name
48-63	30-3F	Typ der Formatierung als Zeichenkette (z.B. OUTPUTFORMATTING)

Aufbau des Eintrags zum FORMAT-Exit

**Eintrag zum INPUT-Exit (Typ INXS und INXE)**

Byte		Bedeutung
dez.	hexadez.	
16-31	10-1F	String INPUT-EXIT-FORM oder INPUT-EXIT-USER oder INPUT-EXIT-LINE
32-39	20-27	Programm-Name des INPUT-Exits
40-47	28-2F	Erste 8 Zeichen Input-Message
48-55	30-37	Format-Name
56-63	38-3F	TAC-Name
64-65	40-41	Status
66-67	42-43	F-Taste
68-69	44-45	K-Taste
70-71	46-47	Kontroll-Feld
72-79	48-4F	Lterm-Name
80-87	50-57	User-Name
88-119	58-77	(ungenutzt)
120-127	78-7F	Nächster Tac oder Kommando
128-129	80-81	Fortsetzungs-Code
130	82	CUT TAC
131	83	(ungenutzt)
132-135	84-87	Fehler-Code

Aufbau des Eintrags zum INPUT-Exit

**Eintrag zum START-Exit (Typ STXS und STXE)**

Byte		Bedeutung
dez.	hexadez.	
16-31	10-1F	String START-EXIT-# (# = 1 - 8 steht für die Nummer des START-Exits)
32-39	20-27	Programm-Name des START-Exits
40-47	28-2F	String STARTUP (fixer Name des TAC)
48-63	30-3F	Kennzeichen, ob erste Task oder Folge-Task als String

Aufbau des Eintrags zum START-Exit

### FORM-Record in der UTM Diagarea

Bei jedem Aufruf an das Formatierungssystem über die Schnittstelle IUTMFORM wird ein Eintrag mit folgendem Inhalt in die UTM Diagarea geschrieben.

Byte		Bedeutung
dez.	hexadez.	
16	10	Operationscode für IUTMFORM
17	11	Returncode der IUTMFORM
18-21	12-15	Info-Returncode der IUTMFORM
22-24	16-18	Statusinformation nach dem Aufruf der IUTMFORM
25	19	Format-Typ
26-27	1A-1B	Leerzeichen (reserviert für Erweiterungen)
28-29	1C-1D	Device Feature vor Aufruf der IUTMFORM (siehe KCDF)
30-31	1E-1F	Drucker-Rückmeldebytes
32-39	20-27	Terminal-Charakteristika (siehe Makro TSTAT)
40-41	28-29	maximale Länge der (letzten) übergebenen Nachricht
42-43	2A-2B	reale Länge der (letzten) gelieferten Nachricht
44-47	2C-2F	Sprachkennzeichen und Territorialkennzeichen
48-55	30-37	Name des Zeichensatzes des USER oder des LTERM
56-63	38-3F	Name des Zeichensatzes des Formats
64-71	40-47	Name des (letzten) übergebenen Formats
72-79	48-4F	Name des (letzten) gelieferten Formats
80-83	50-53	Adresse des (letzten) übergebenen Nachrichtenbereichs
84-87	54-57	Adresse des (letzten) übergebenen Cursors

Aufbau des FORM-Records in der UTM Diagarea

### 3.3.2.4 DB Diagarea

Bei jedem USER-CALL an das DB-System wird ein DB-Record in die DB Diagarea geschrieben. Der Bereich wird, ebenso wie die UTM Diagarea, zyklisch mit Trace-Records beschrieben.

Zwei Zyklen werden durch eine Trennlinie bestehend aus '='-Zeichen getrennt. Der Record oberhalb der Trennlinie ist der jüngste, der Record unterhalb der älteste.

Byte		Bedeutung
dez.	hexadez.	
0-1	00-01	Zähler der aktuellen Einträge in den DIAGAREAs (UTM und DB)
2-5	02-05	"DBCL"= Kennzeichen für DB-Record
6-7	06-07	Derzeit nicht genutzt (mit '=' vorbelegt)
8-15	08-0F	Zeitstempel
16-19	10-13	Status DB-Transaktion vor dem DB Aufruf <sup>1</sup> , siehe Tabelle auf <a href="#">Seite 102</a>
20-23	14-17	Status DB-Transaktion nach dem DB Aufruf <sup>1</sup> , siehe Tabelle auf <a href="#">Seite 102</a>
24	18	DB Operations-Code, siehe Tabelle <a href="#">Seite 102</a>
25	19	DB Secondary-Opcode
26	1A	DB Error Code, siehe Tabelle <a href="#">Seite 103</a>
27	1B	Kennzeichen für DB-System (hexadezimal): 01=UDS, 02=SESAM, 03=LEASY, 06=CIS, 07=generisches DB-System (Typ=DB), 09=XA im BS2000
28-31	1C-1F	DB Trace Information (Inhalt und Bedeutung siehe DB-System) Bei XA im BS2000: DB Trace Information des letzten XA-Aufrufs: 28=RM-Nummer, 29-30=XA-Funktionscode, 31=XA-Returncode)
32-63	20-3F	Secondary DB Trace Information (siehe DB-System)
64-67	40-43	Kombinierte Statusinfo TAM (weitere Diagnoseinformationen, nur relevant bei Einsatz einer Multi-DB-Funktion)
68-71	44-47	Kombinierte Statusinfo VGM (weitere Diagnoseinformationen, nur relevant bei Einsatz einer Multi-DB-Funktion)
72-73	48-49	Zähler der Transaktionen innerhalb des Vorgangs
74	4A	Nummer des UTM-Anwendungslaufs (beginnt nach der KDCDEF-Generierung mit '1')
75	4B	Kennzeichen für Tabellenindex, =' T '
76-77	4C-4D	UTM-Tabellen-Index, der zum DB-Aufruf führte
78-79	4E-4F	Aktionsindex innerhalb dieser Tabelle

Aufbau der DB Diagarea

Byte		Bedeutung
dez.	hexadez.	
80-83	50-53	Vorgangszähler (eindeutig innerhalb des UTM-Anwendungslaufs)
84-87	54-57	intern benutzte Adresse
88-91	58-5B	Rücksprungadresse im Teilprogramm. Die Adresse zeigt hinter den Aufruf CALL DB-System.
92	5C	* als „eye catch“ zum Erkennen des Endes des Diagnose-Satzes

Aufbau der DB Diagarea

<sup>1</sup> Bei Operationscodes, die sich auf DB-Vorgänge beziehen, wird hier der Status des Vorgangs-Memorys angezeigt.

Ist das DB-System über die XA-Schnittstelle an openUTM gekoppelt und ein sogenannter „dynamischer xa-switch“ eingebunden (z.B. bei Oracle das Modul „oraswd“), dann wird beim An - bzw. Abmelden einer gemeinsamen Transaktion der folgende Diagnose-Satz geschrieben:

Byte		Bedeutung
dez.	hexadez.	
16-23	10-17	abdruckbar „ax_reg“ beim Anmelden der Transaktion abdruckbar „ax_unreg“ beim Abmelden der Transaktion
24-27	18-1B	Nummer der RM-Instanz
28-31	1C-1F	unbenutzt
32-47	20-2F	abdruckbar der Returncode des Aufrufs
48	30	interner Status der Transaktion
49	31	interner Status der Anmeldung bzw. Abmeldung
50-51	32-33	unbenutzt
52-54	34-36	befindet sich der Aufruf innerhalb einer openUTM-Transaktion (abdruckbar „yes“ oder „no“)
55	37	unbenutzt
56	38	* als „eye catch“ zum Erkennen des Endes des Diagnose-Satzes

Diagnose-Satz

Status der DB-Transaktion (Inhalt von Byte 0 bis 3 bzw. 4 bis 7 des DB-Records)

Inhalt (hexadez.)	Bedeutung des DB-Status
04	Die DB-Transaktion wurde in das vorläufige Transaktionsende (PTC) versetzt.
08	Innerhalb einer DB-Transaktion wurden Updates vorgenommen.
10	Die DB-Transaktion ist zurückgesetzt worden.
20	Die DB-Transaktion wurde geschlossen.
40	Der Teilprogrammmlauf hat einen Aufruf CLOSE DB-Transaktion gegeben, die DB-Transaktion ist aus Sicht des Teilprogrammmlaufs beendet.
80	Die DB-Transaktion ist offen.

Status der DB-Transaktion. Auch Kombinationen dieser Werte treten auf, z.B. „88“.

DB-Operationscodes (Byte 8 des DB-Records)

Inhalt (hexadez.)	DB-Op.-codes	Bedeutung
00	STPA	DB Connection Modul soll DB-spezifische Startparameter prüfen. Aufruf erfolgt beim Start eines UTM-Anwendungsprogramms.
04	CONC	Aufbau der Verbindung zwischen UTM-Task und Verbindungsmodul des DB-Systems. Aufruf erfolgt beim Start eines UTM- Anwendungsprogramms.
08	DCON	Abbau der Verbindung zwischen UTM-Task und dem DB-System. Aufruf erfolgt in der Endbehandlung eines UTM- Anwendungsprogramms.
10	USRC	DB-Aufruf des Teilprogramms ausführen (User Call).
14	FITA	DB-Transaktion normal beenden. Der Aufruf wird von openUTM am Ende einer gemeinsamen DB/DC Transaktion gegeben.
18	CATA	DB-Transaktion abnormal beenden. D.h. die DB-Transaktion wird auf den letzten Sicherungspunkt zurückgesetzt.
1C	BKTA	Der Aufruf wird bei PEND KP oder PEND PA/PR mit TASK-Wechsel (wegen TAC-Klassen) verwendet, die UTM-Task löst sich von der Bearbeitung einer DB/DC Transaktion.
20	COTA	Der Aufruf erfolgt dann, wenn eine Mehrschritt-Transaktion fortgesetzt wird, also nach vorherigem PEND KP oder PEND PA/PR mit TASK-Wechsel (wegen TAC-Klassen).
24	STAT	Status einer DB-Transaktion anzeigen oder Auftrag zum Löschen aller DB-Stati. Das DB-System führt Statusinformationen über DB-Transaktionen als Mittel zum koordinierten Wiederanlauf in openUTM und dem DB-System.
28	PETA	Vorläufiges Ende einer DB-Transaktion anfordern

DB-Operationscodes

Inhalt (hexadez.)	DB-Op.-codes	Bedeutung
2C	EDVG	Ende eines UTM-Vorgangs bei offenem DB-Vorgang.
30	BKVG	Unterbrechung eines offenen DB-Vorgangs.
34	COVG	Fortsetzung eines unterbrochenen DB-Vorgangs nach BKVG.
38	RSVG	openUTM zeigt dem DB-System einen UTM-Vorgangswiederanlauf an, falls auch ein DB-Vorgang offen ist.
3C	CNFPTC	Das DB-Verbindungsmodul bestätigt dem DB-System über die XA-Schnittstelle den Zustand der precommitted Transaktionen und löscht diese aus der internen Liste.
40	STRT	Das DB-Verbindungsmodul übermittelt dem DB-System über die XA-Schnittstelle den Beginn einer DB-Transaktion.
44	PEND	Das DB-Verbindungsmodul übermittelt dem DB-System über die XA-Schnittstelle das vorläufige Transaktionsende und leitet die erste Commitment-Phase ein.

DB-Operationscodes

#### *DB-Error-Codes*

Inhalt (hexadez.)	Bedeutung des DB-Error-Codes
00	Auftrag ausgeführt.
04	Die DB-Transaktion musste zurückgesetzt werden. openUTM setzt dann auch die UTM-Transaktion zurück. Bei einem DB-Aufruf (User Call) bekommt das Anwendungsteilprogramm wieder die Kontrolle.
08	Bewirkt bei openUTM Abbruch des Vorgangs mit PEND ER.
0C	Das DB-System ist nicht verfügbar, nicht hochgefahren.
10	Das DB-System ist aus einem anderen Grund nicht verfügbar.
14	Auftrag wurde nicht ausgeführt, später wieder versuchen!
18	Ein evtl. behebbarer Systemfehler liegt vor. openUTM erzeugt einen Task-Dump und versucht die offene DB-Transaktion zu beenden.
1C	Ein nicht behebbarer Fehler liegt vor. Die Zusammenarbeit mit dem DB-System in dieser DB-Session erscheint nicht mehr sinnvoll.
20	Vom DB-System wurde ein Benutzerfehler entdeckt, z.B. bei der Prüfung der DB-Startparameter.
24	Vom DB-System wurde ein Schnittstellenfehler entdeckt.

DB-Error-Codes

### 3.3.2.5 Administration DIAGAREA

Die Administration DIAGAREA ist ein Task-spezifischer Trace-Bereich in KDCROOT. Dieser wird, ebenso wie die UTM Diagarea, zyklisch mit Trace-Records beschrieben. Bei jedem Aufruf der Programmschnittstelle der Administration wird ein Record in diesen Bereich geschrieben.

In den Bereich passen 71 Records, ein Record ist 112 Bytes lang. Der Bereich wird zyklisch beschrieben. Zwei Zyklen werden durch eine Trennlinie bestehend aus '='-Zeichen getrennt. Der Record oberhalb der Trennlinie ist der jüngste, der Record unterhalb der älteste.

Ein Record hat folgende Struktur:

Byte		Bedeutung
dez.	hexadez.	
0-1	00-01	Zähler des aktuellen Eintrags in der Administration DIAGAREA
2-3	02-03	Zähler des zugehörigen Eintrages (Opcode „ADMI“) in der UTM Diagarea
4-5	04-05	abdruckbares Kürzel für den Opcode (siehe Tabelle unten)
6-7	06-07	abdruckbares Kürzel für den Objekttyp bzw. den Subopcode1, je nach angegebenem Opcode (siehe Tabelle unten)
8-11	08-0B	Adresse parameter area
12-15	0C-0F	Adresse identification area
16-19	10-13	Adresse selection area
20-23	14-17	Adresse data area
24-79	18-4F	parameter area
80-111	50-6F	Name des administrierten Objektes in der Objekttyp-spezifischen Länge (2-32) aus der identification area bzw. aus der data area

Aufbau der Administration DIAGAREA

Dabei werden der Eintragszähler und die Adressen der Bereiche vor Aufruf des UTM-Systemcodes mitprotokolliert. Die restlichen Daten werden nach der Rückkehr aus dem UTM-Systemcode, vor dem Rücksprung in das Teilprogramm mitprotokolliert. Daher sind in der parameter area auch die Rückgabewerte - unter anderem der Returncode - enthalten. Der Inhalt der identification area wird nur dann mitprotokolliert, wenn der Bereich bei dem Administrationsaufruf verwendet wurde. Der Name aus der data area wird nur bei KC\_CREATE\_OBJECT mitprotokolliert.

Wird der Administrationsaufruf mit PEND ER beendet, weil die Adresse der parameter area nicht zugreifbar oder nicht auf Wortgrenze ausgerichtet ist, dann kann man diesem Protokoll entnehmen, welche Adresse angegeben wurde.



Es werden folgende Kürzel verwendet:

Kürzel Opcode	Opcode	Kürzel Sub- code1 /Objektyp	Subopcode1 bzw. Objekttyp
CA	KC_CHANGE_APPLICATION	N S O	KC_NEW KC_SAME KC_OLD
CD	KC_CREATE_DUMP		
CS	KC_CREATE_STATEMENTS		
EN	KC_ENCRYPT	V C D A N	KC_ACTIVATE_KEY KC_CREATE_KEY KC_DELETE_KEY KC_READ_ACTIV_PUBLIC_KEY KC_READ_NEW_PUBLIC_KEY
OI	KC_ONLINE_IMPORT	A	KC_ALL
PE	KC_PTC_TA	R	KC_ROLLBACK
LO	KC_LOCK_MGMT	UF US UA AB AA AP	KC_UNLOCK_USF KC_SIGNOFF_SINGLE KC_SIGNOFF_ALL KC_ABORT_BOUND_SERVICE KC_ABORT_ALL_BOUND_SERVICE KC_ABORT_PTC_SERVICE
UP	KC_UPDATE_IPADDR	A P	KC_ALL KC_PARTNER
CR DL GT MD	KC_CREATE_OBJECT KC_DELETE_OBJECT KC_GET_OBJECT KC_MODIFY_OBJECT	AB AC AP BC CL CN CC CO CP CU DA DI DP EC	KC_ABSTRACT_SYNTAX KC_ACCESS_POINT KC_APPLICATION_CONTEXT KC_BCAMAPPL KC_TACCLASS KC_CLUSTER_NODE KC_CLUSTER_PAR KC_CON KC_CURR_PAR KC_CLUSTER_CURR_PAR KC_DIAG_AND_ACCOUNT_PAR KC_DB_INFO KC_DYN_PAR KC_ENCRYPTION

Kürzel für Opcode und Subcode1 bzw. Objekttyp

Kürzel Opcode	Opcode	Kürzel Subcode1 /Objektyp	Subopcode1 bzw. Objektyp
(Forts.)	(Forts.)		
CR	KC_CREATE_OBJECT	ED	KC_EDIT
DL	KC_DELETE_OBJECT	GB	KC_GSSB
GT	KC_GET_OBJECT	KS	KC_KSET
MD	KC_MODIFY_OBJECT	LC	KC_LTAC
		LM	KC_LOAD_MODULE
		LP	KC_LPAP
		LS	KC_LSES
		LT	KC_LTERM
		MS	KC_MSG_DEST_PAR
		MM	KC_MESSAGE_MODULE
		MP	KC_MAX_PAR
		MX	KC_MUX
		OA	KC_OSI_ASSOCIATION
		OC	KC_OSI_CON
		OL	KC_OSI_LPAP
		PE	KC_PTC
		PO	KC_TPOOL
		PP	KC_PAGEPOOL
		PR	KC_PROGRAM
		PT	KC_PTERM
		QP	KC_QUEUE_PAR
		QU	KC_QUEUE
		SI	KC_SIGNON
		SF	KC_SFUNC
		SP	KC_SYSTEM_PAR
		SN	KC_SUBNET
		TA	KC_TASKS_PAR
		TC	KC_TAC
		TI	KC_TIMER_PAR
		TR	KC_TRANSFER-SYNTAX
		UP	KC_UTMD_PAR
		US	KC_USER
		UF	KC_USER_FIX
		U1	KC_USER_DYN1
		U2	KC_USER_DYN2
SH	KC_SHUTDOWN	K	KC_KILL
		N	KC_NORMAL
		W	KC_WARN
		G	KC_GRACEFUL

Kürzel für Opcode und Subcode1 bzw. Objektyp

Kürzel Opcode	Opcode	Kürzel Subop- code1 /Objektyp	Subopcode1 bzw. Objekttyp
SL	KC_SYSLOG	I CS SC SW WB	KC_INFO KC_CHANGE_SIZE KC_SWITCH_AND_CHANGE KC_SWITCH KC_WRITE_BUFFER
SM	KC_SEND_MESSAGE		
SP	KC_SPOOLOUT		
UL	KC_USLOG	SW	KC_SWITCH

Kürzel für Opcode und Subcode1 bzw. Objekttyp

### 3.3.2.6 Administration USERAREA

Die Administration USERAREA ist ein Task-spezifischer Trace-Bereich in KDCROOT. Dieser Bereich dient dazu, die an der Programmschnittstelle vom Teilprogramm an openUTM übergebenen Daten mitzuprotokollieren.

Da die übergebenen Daten sehr umfangreich sein können, werden nur die Daten eines Aufrufs in dem Bereich abgelegt. Der Bereich besteht also nur aus einem Record und fasst 4140 Bytes. Der Bereich wird nur für die Aufrufe beschrieben, bei denen Daten an openUTM übergeben werden. Es wird der Inhalt der `data area` oder der Inhalt der `selection area` protokolliert, je nachdem, welcher Bereich bei dem Aufruf verwendet wurde. Wird der Bereich für die Diagnose benötigt, so muss darauf geachtet werden, dass der entsprechende Aufruf der letzte Aufruf ist, für den Daten mitprotokolliert werden, damit der Bereich nicht durch einen nachfolgenden Aufruf überschrieben wird.

Wird in der `data area` für einen Benutzer ein Passwort übergeben, dann wird dieses nicht mitprotokolliert, sondern mit binär null überschrieben.

Ein Record hat folgende Struktur:

Byte		Bedeutung
dez.	hexadez.	
0-1	00-01	Zähler des zugehörigen Eintrages in der Administration DIAGAREA
2-3	02-03	irrelevant
4-59	04-3B	parameter area
60-4139	3C-102B	Inhalt data area bzw. selection area in der übergebenen Länge

Aufbau der Administration USERAREA

Der Inhalt der `parameter area` und die übergebenen Daten werden im UTM-Systemcode mitprotokolliert. Die `parameter area` wird so protokolliert, wie sie vom Teilprogramm übergeben wird, also ohne besetzte Rückgabewerte. Der Subreturncode ist auf null gesetzt.

Der Eintragszähler wird nach der Rückkehr aus dem UTM-Systemcode vor dem Rücksprung in das Teilprogramm mitprotokolliert.

### 3.3.2.7 Kommunikationsbereich KB

Der Kommunikationsbereich besteht aus dem KB-Kopf, dem KB-Rückgabebereich und dem KB-Programmbereich in der generierten Länge.

Die Feldnamen sind für COBOL angegeben. Die Feldnamen für C / C++ finden Sie im Anhang des openUTM-Handbuch „Anwendungen programmieren mit KDCS“.

Byte		Feldnamen und Inhalt	
dez.	hexadez.		
0-7	00-07	KCBENID:	UTM-Benutzerkennung
8-15	08-0F	KCTACVG:	TAC, der verwendet wurde, um diesen Vorgang zu starten.
16-17	10-11	KCTAGVG:	Tag
18-19	12-13	KCMONVG:	Monat
20-21	14-15	KCJHRVG:	Jahr
22-24	16-18	KCTJHVG:	Industrietag
25-26	19-1A	KCSTDVG:	Stunde
27-28	1B-1C	KCMINVG:	Minute
29-30	1D-1E	KCSEKVG:	Sekunde
31	1F	KCKNZVG:	Vorgangskennzeichen
32-39	20-27	KCTACAL:	TAC, mit dem das Programm adressiert wurde
40-41	28-29	KCSTDAL:	Stunde
42-43	2A-2B	KCMINAL:	Minute
44-45	2C-2D	KCSEKAL:	Sekunde
46	2E	KCAUSWEIS:	Ausweis-Kennzeichen: A (Ausweis steckt) oder Leerzeichen.
47	2F	KCTAIND:	Transaktionskennzeichen: F (erste) oder N (Folge-Transakt.)
48-55	30-37	KCLOGTER:	Name des LPAP-Partners (Sender)
56-57	38-39	KCTERMN:	Gerätetyp des Terminals oder Druckers, siehe Tabelle bei PTERM-Anweisung
58-59	3A-3B	KCLKBPB:	Maximallänge des KB-Programmbereichs gemäß UTM-Generierung
60-61	3C-3D	KCHSTA:	Anzahl der gekellerten Vorgänge aus der Sicht des aktuellen Vorgangs.
62	3E	KCDSTA:	Veränderung der Anzahl der gekellerten Vorgänge
63	3F		leer
64	40	KCPRIND:	Programmkennzeichen. "A" = Teilprogrammmlauf in einem Asynchron-Vorgang "D" = Teilprogrammmlauf in einem Dialog-Vorgang

KDCS-Kommunikationsbereich

Byte		Feldnamen und Inhalt		
dez.	hexadez.			
65	41	KCOF1: erlaubte OSI TP Funktionen "B" = Basisfunktionen "H" = Basisfunktionen + Handshake-Funktionen "C" = Basis- und Commit-Funktionen mit Chained Transactions "O" = Other combination Leerzeichen, wenn Vorgang nicht über OSI TP gestartet wurde		
66	42	KCCP: UTM Client Protocol "0" = Asynchron-Vorgang "1" = LU6.1 "2" = OSI TP "3" = UPIC "4" = TIAM "5" = APPLI "6" = SOCKET		
67	43	KCTARB: TA ist mit Rollback gekennzeichnet.		
68-71	44-47	KCYEARVG: 4-stellige Jahresangabe des Vorgangbeginns		
72-83	48-53	leer (FILLER)		
84-115	54-73	KCRFELD KB-Rückgabebereich		
84-85	54-55	KCRDF: Rückgabe Bildschirmfunktion KCRWVG: Anzahl wartender Vorgänge beim DGET		
86-87	56-57	KCRMLM: Rückgabe tatsächliche Länge der Nachricht		
		INFO CK-Aufruf	MGET-Aufruf	SIGN ON-Aufruf
88	58	KCRINFCC: KDCS-Returncode des geprüften KDCS-Aufrufs	KCVGST: Vorgangstatus	KCRSIGN1: Primärkode
89	59		KCTAST: Transaktionsstatus	KCRSIGN2: Sekundärkode
90	5A		leer	
91	5B	leer	KCRMGT: Nachrichtentyp	
92-94	5C-5E	KCRCCC: KDCS-Returncode		
95	5F	KCRCKZ: Kennzeichen: P (Produktivanwendung)		
96-99	60-63	KCRCDC: interner Returncode		
100-107	64-6B	KCRMF: Rückgabe Formatkennzeichen, Editprofi oder abstrakte Syntax		

KDCS-Kommunikationsbereich

Byte		Feldnamen und Inhalt
dez.	hexadez.	
108-115	6C-73	KCRPI: Rückgabe Vorgangs-Identifikation KCRUS: Benutzererkennung beim SIGN ST oder Erzeuger der Nachricht beim DGET KCRQN: Von openUTM vergebener Name der Temporären Queue bei QCRE NN
116- ...	74- ...	KB-Programmbereich

KDCS-Kommunikationsbereich

### 3.3.3 Summary information

Am Schluss des Dumps finden Sie eine Zusammenfassung (Bereich Summary information), ein Inhaltsverzeichnis und einen Meldungsabschnitt, der die Meldungen enthält, die während der Aufbereitung des UTM-Dumps ausgegeben wurden.

Die Summary Information ist ein Extrakt der gesamten Dump-Information. Dabei sind solche Daten ausgewählt, die bei der Diagnose häufig benötigt werden. Das erspart Sucharbeit bei der Bearbeitung eines UTM-Dumps, vor allem bei der Vordiagnose und Duplikat-Erkennung.

Die Summary Information enthält wichtige Generierungsparameter der Anwendung, die Version und den Korrekturstand des Betriebssystems und von openUTM, Startparameter der Anwendung, die wichtigsten aktuellen Tabelleneinträge und die letzten Sätze in der UTM Diagarea.

Es gibt UTM-Dumps, die nicht alle Tabellen enthalten, z.B. der PENDER-Dump. Bei der Auswertung solcher Dumps kann das Summary deshalb einige Daten nicht zeigen.

Die Ausgabe der Summary Information und des Dumps kann man bei der Aufbereitung über den Operanden INFO bei den Anweisungen FILE und FGG steuern.

Wurde der Dump von KDCUPD erzeugt, dann wird nur die erste Seite der Summary Information geschrieben.

### 3.4 Dump-Beispiele

Nachfolgend finden Sie zwei Beispiele für einen UTM-Dump.

Allgemeines Vorgehen: Suchen Sie im Dump die Werte der Returncodes KCRCCC und KCRCDC heraus und informieren Sie sich im Abschnitt „KDCS-Returncodes in KCRCCC“ auf Seite 367 und Abschnitt „Interner Returncode KCRCDC“ auf Seite 371 über deren Bedeutung.

#### 3.4.1 Dump-Beispiel K326

1. UTM Diagarea aufsuchen --> Letzter Aufruf: PEND ER

Die Rückkehrcodes finden Sie im davor liegenden Eintrag der UTM Diagarea oder im KB

2. KB aufsuchen -->            Inhalt KCRCCC (Byte 92-94):            70Z  
                                  Inhalt KCRCDC (Byte 96-99):            K326

#### UTM Diagarea

```

KDCR00T :   UTM Diagarea

0001 0240F460 0000 01FDC9D5 E7E27E7E D2EE54BE 728FA100 C9D5D7E4 E360C5E7 C9E360D3 C9D5C540 .)INXS==K..... INPUT-EXIT-LINE
0240F480 0020 C9D5D7D3 C9D5C540 C7C5E3C7 E2E2C240 40404040 40404040 40404040 40404040 INPLINE GETGSSB
0240F4A0 0040 C5C30000 000005D6 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 EC....NOANY97631EURO
0240F4C0 0060 40404040 40404040 40404040 40404040 40404040 40404040 40404040
0240F4E0 0080 C5D9D540 40404040
0002 0240F4E8 0000 01FEC9D5 E7C57E7E D2EE54BE 72929C00 C9D5D7E4 E360C5E7 C9E360D3 C9D5C540 .)INXE==K....k.. INPUT-EXIT-LINE
0240F508 0020 C9D5D7D3 C9D5C540 C7C5E3C7 E2E2C240 40404040 40404040 40404040 40404040 INPLINE GETGSSB
0240F528 0040 C5C30000 000005D6 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 EC....NOANY97631EURO
0240F548 0060 40404040 40404040 40404040 40404040 40404040 40404040 40404040 GETGSSB
0240F568 0080 E2C3E840 40404040
0003 0240F570 0000 01FFD2C4 C3E27E7E D2EE54BE 728BE000 C3D6D5E3 0B010000 0000C7C5 E3C7E2E2 SCY
0240F590 0020 C240C5C3 00000000 D5D6C7C5 E3C7E2E2 C240E2C3 E8004040 40404040 00004040 .-KDCS==K.....CONT.....GETGSS
0240F5B0 0040 4040F0F0 F0E3F0F0 F0F04040 40404040 40404040 40404040 40404040 B EC....NOGETGSSB SCY.
0240F5D0 0060 00000000 00000012 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 000T0000 ==a.L.
0240F5F0 0080 7E7E7E7E 7E7E7E7E D2EE54BE 72840E00 C6020001 00000012 00000001 00000028 .....ANY97631EURO =====
0004 0240F5F8 0000 0200E5C7 C9C47E7E D2EE54BE 72840E00 00000000 00000000 00000000 ..VGID==K.....F.....
0240F618 0020 00000028 C4F0F1F6 E9C5F0F9 D9C1C9D5 C5D94040 OA012022 F0F8F1F7 F4F6000D ....D016ZE09RAINER .....081746...
0240F638 0040 04280001 C4F0F1F6 E9C5F0F9 D9C1C9D5 C5D94040 OA012022 F0F8F1F7 F4F6000D ....D016ZE09RAINER .....081746...
0240F658 0060 04280001 00000012 00000071 00000000 00000000 00140000 C7C5E3C7 E2E2C240 .....GETGSSB
0240F678 0080 C7C5E3C7 E2E2C240
0005 0240F680 0000 020D2D2C4 C3E27E7E D2EE54C1 D984B700 C9D5C9E3 00000100 01000000 00000000 .KDCS==K..ARd..INIT.....
0240F6A0 0020 00000000 00000000 00000000 00000000 00000000 00000000 00004040 00004040 .....
0240F6C0 0040 4040F0F0 F0E3F0F0 F0F04040 40404040 40404040 40404040 40407E7E 810E70DA 000T0000 ==a...
0240F6E0 0060 02D51010 00000112 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 .N.....ANY97631EURO =====
0240F700 0080 7E7E7E7E 7E7E7E7E
0006 0240F708 0000 0203D2C4 C3E27E7E D2EE54C3 D16CC001 D4C7C5E3 D5E37FFF 00004040 40404040 .KDCS==K..CJ%.MGETNT"-...
0240F728 0020 40404040 40404040 40400000 00000000 00000000 00000000 00000000 0007D6C3 .....OC
0240F748 0040 40D4F0F0 F0E3F0F0 F0F04040 40404040 40404040 40404040 40407E7E 810E70DA M00T0000 ==a...
0240F768 0060 02D51010 00000112 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 .N.....ANY97631EURO =====
0240F788 0080 7E7E7E7E 7E7E7E7E
0007 0240F790 0000 0204D2C4 C3E27E7E D2EE54D4 5F15C101 E2C7C5E3 C7C20050 0000C7E2 E2C2E3C5 .KDCS==K..M..A.SGETGB.&..GSSBTE
0240F7B0 0020 E2E34040 40404040 40400000 00000000 00000000 00000000 00000000 0000D6C3 ST
0240F7D0 0040 40D4F1F4 E9E3F0F0 F0F04040 40404040 40404040 40404040 40407E7E 810E70DA M14ZT0000 ==a...
0240F7F0 0060 02D51010 00000112 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 .N.....ANY97631EURO =====
0240F810 0080 7E7E7E7E 7E7E7E7E
0008 0240F818 0000 0205D2C4 C3E27E7E D2EE54F1 8A63C500 D4D7E4E3 D5E30000 00154040 40404040 .KDCS==K..I..E.MPUTNT....
0240F838 0020 40404040 40404040 40400000 00000000 00000000 00000000 00000000 0000D6C3 .....OC
0240F858 0040 40D4F0F0 F0E3F0F0 F0F04040 40404040 40404040 40404040 40407E7E 810E70DA M00T0000 ==a...
0240F878 0060 02D51010 00000112 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 .N.....ANY97631EURO =====
0240F898 0080 7E7E7E7E 7E7E7E7E
0009 0240F8A0 0000 0206D2C4 C3E27E7E D2EE54FB A656DE00 D7C5D5C4 D2D70000 0000C1D7 C5D5C4C3 .KDCS==K..{w...PENDKP...APENDC
0240F8C0 0020 D6D40000 00000000 00000000 00000000 00000000 00000000 00000000 0000D6C3 OM.....OC
0240F8E0 0040 40D4F7F4 E9E3D2F3 F2F64040 40404040 40404040 40404040 40407E7E 810E70DA M74ZTK326 ==a...
0240F900 0060 02D51010 00000112 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 40404040 40404040 .N.....ANY97631EURO =====
0240F920 0080 7E7E7E7E 7E7E7E7E
000A 0240F928 0000 0207D2C4 C3E27E7E D2EE54FB A6F9B000 D7C5D5C4 C5D97A40 4040D2C3 D9C3C3C3 .KDCS==K..{w9..PENDER: KCRCCC
0240F948 0020 406E7E40 F7F0E940 C1C6E3C5 D940E4E3 D440E2E5 C340405A 5A404040 40404040 >= 70Z AFTER UTM SVC !!

```



```

0240F968 0040 40404040 40404040 40404040 40404040 40404040 40404040 40407E7E 810E70DA ==a...
0240F988 0060 00000000 00000012 C1D5E8F9 F7F6F3F1 C5E4D9D6 40404040 7E7E7E7E 7E7E7E7E .....ANY97631EURO =====
0240F9A8 0080 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
= 000B 0240F9B0 0000 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
= 0240FA10 0060 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
0240FA30 0080 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
AUTM Dump Utility (V06.5A00) - BS2000 Application = RAINER TSN = 8P6K ITN = FFFFDDC6 Dump Nr. 1 Page 52
REASON = PENDER CREATED 17-08-03 08:18:51 PRINTED 17-08-03 08:20:00

000C 0240FA38 0000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....
= 0240FA98 0060 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....
0240FAB8 0080 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....
=7D00 02826778 0000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....
= 028267D8 0060 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....
028267F8 0080 00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000 .....

```

### Ausschnitt aus der Summary information

Summary information

KDCDEF related information

UTM-mode: S  
Coldstart and first task: YES  
Number of asynchron tasks: 10  
Accounting generated: YES  
Filebase name: RAIT650

UTM-D: YES

Maximum values generated

KBPROG: 32767  
SPAB: 32767  
MPUT length: 32764  
CPU-time: 120000 msec

FORMIO length: 65532  
FORMAREA length: 32764  
Restart buffer length: 65536

System information

BS2000 version: 2180  
Hardware type: XS31

Hostname: D016ZE09

Settings at start time

KDCMON: OFF  
Version number of KDCDATA: V06.5A00  
Number of tasks: 10  
Formatting system: FHS

Number of asynchron tasks: 10

Database related information

- 1. Type: XA (BS2) Entry: TPXASWD
- 2. Type: UDS Entry: DML
- 3. Type: UDS Entry: \$UNIBASE

Information on REP's

```

KDCRTST:
AUTM Dump Utility (V06.5A00) - BS2000 Application = RAINER TSN = 8P6K ITN = FFFFDDC6 Dump Nr. 1 Page 79
REASON = PENDER CREATED 17-08-03 08:18:51 PRINTED 17-08-03 08:20:00

```

Information on exits

Input exit: YES  
USERFORM input exit: YES  
Start exit: YES  
Shut exit: YES  
MSGTAC: YES  
Logging exit: YES  
Active exits: NO EXIT ACTIVE

Format input exit: YES  
Line input exit: YES  
Format exit: YES  
SIGNON conversation: YES  
BADTAC: YES

Current state

Current TAC: GETGSSB  
FGG number of program: 0  
Terminal type: APPLI  
Programming language of current TAC: ILCS  
Actual length of KBPROG: 256

Program name: KCSTPRG  
Kind of terminal: DIALOG  
Actual length of SPAB: 256

Diagnostic information

Dump reason: PENDER  
KCRCC: 74Z  
FHS secondary returncode: 0000  
Last ROOT action: CALL\_WAIT

Dump created: 17-08-03 08:18:51  
KCRDC: K326  
Formatting mode: INPUT FORMATTING  
Action index: 2

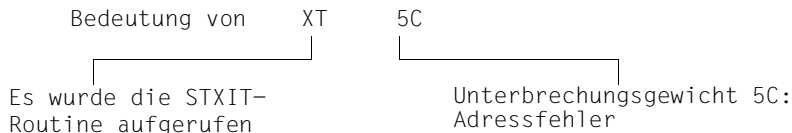
### 3.4.2 Dump-Beispiel STXIT

1. UTM Diagarea aufsuchen --> letzter Aufruf: PEND ER

Die Rückkehrcodes finden Sie im Klartext des PENDER-Eintrags oder im KB.

2. KB aufsuchen

Transaktionscode:	(Byte 32-39):	STXITPI
Inhalt KCRCCC	(Byte 92-94):	70Z
Inhalt KCRCDC	(Byte 96-99):	XT5C



3. Analyse der Register (Context Area)

Das Problem tritt im Modul STXITPI auf Distanz x'4A2' auf. Zur weiteren Diagnose müssen der User-Dump und das Modul STXITPI analysiert werden.

#### Beispiel

```

KDCR00T :   UTM Diagarea

003B  01BC2ED0 0000 003DE5C7 C9C47E7E D0D3102E 33BE4D00 ..VGID==.L....(
      01BC2EE0 0010 C6040001 00000003 00000000 00000020 F.....
      01BC2EF0 0020 00000030 C4F0F1F6 E9C5F0F9 E2E3D9C2 ...D016ZE09STRB
      01BC2F00 0030 D3E24040 0A010204 00010000 00000003 LS .....
      01BC2F10 0040 00000002 C4F0F1F6 E9C5F0F9 E2E3D9C2 ...D016ZE09STRB
      01BC2F20 0050 D3E24040 0A010204 00010000 00000003 LS .....
      01BC2F30 0060 00000002 00000000 0000F1F0 F2F0F5F5 .....102055
      01BC2F40 0070 000249E6 00190000 E2E3E7C9 E3D7C940 ...W...STXITPI
      01BC2F50 0080 E2E3E7C9 E3D7C940 STXITPI
003C  01BC2F58 0000 003ED2C4 C3E27E7E D0D3102E 33C3DE00 ..KDCS==.L...C..
      01BC2F68 0010 C9D5C9E3 00000074 00000000 00000000 INIT.....
      01BC2F78 0020 00000000 00000000 00000000 00000000 .....
      01BC2F88 0030 00000000 00000000 00004040 00004040 .....
      01BC2F98 0040 4040F0F0 F0E3F0F0 F0F04040 40404040 000P0000
      01BC2FA8 0050 40404040 40404040 40407E7E 81592158 ==a...
      01BC2FB8 0060 40404040 00000003 E3D7F6F0 F0F0F0F1 ...TP600001
      01BC2FC8 0070 D2D3C1E4 E2404040 7E7E7E7E 7E7E7E7E KLAUS =====
      01BC2FD8 0080 7E7E7E7E 7E7E7E7E =====
003D  01BC2FE0 0000 003FD2C4 C3E27E7E D0D3102E 33C5A600 ..KDCS==.L...Ew.
      01BC2FF0 0010 D4C7C5E3 00000050 00000000 00000000 MGET...&.....
      01BC3000 0020 00004040 40404040 40400000 00000000 ..
      01BC3010 0030 00000000 00000000 00000000 0004D6C3 .....OC
      01BC3020 0040 40D4F0F0 F0E3F0F0 F0F04040 40404040 M000P0000
      01BC3030 0050 40404040 40404040 40407E7E 815921DC ==a...
      01BC3040 0060 01592D32 00000003 E3D7F6F0 F0F0F0F1 ...TP600001
      01BC3050 0070 D2D3C1E4 E2404040 7E7E7E7E 7E7E7E7E KLAUS =====
      01BC3060 0080 7E7E7E7E 7E7E7E7E =====
    
```

```

003E 01BC3068 0000 0040D2C4 C3E27E7E D0D31036 6605FE00 . KDCS==.L.....
      01BC3078 0010 D7C5D5C4 C5D97A40 404040E2 E3E7C9E3 PENDER: STXIT
      01BC3088 0020 40E6C9E3 C840E7E3 F5C340C5 D5E3C5D9 WITH XT5C ENTER
      01BC3098 0030 C5C44040 4040405A 5A404040 40404040 ED !!
      01BC30A8 0040 40404040 40404040 40404040 40404040
      01BC30B8 0050 40404040 40404040 40407E7E 815921DC
      01BC30C8 0060 00000000 00000003 E3D7F6F0 F0F0F0F1 ==a...
      01BC30D8 0070 D2D3C1E4 E2404040 7E7E7E7E 7E7E7E7E .....TP600001
      01BC30E8 0080 7E7E7E7E 7E7E7E7E KLAUS =====
      003F 01BC30F0 0000 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
      = 01BC3160 0070 7E7E7E7E 7E7E7E7E 7E7E7E7E 7E7E7E7E =====
      01BC3170 0080 7E7E7E7E 7E7E7E7E =====

```

KDCROOT : KB

```

      01009E40 0000 D2D3C1E4 E2404040 E2E3E7C9 E3D7C940 KLAUS STXITPI
      01009E50 0010 F3F1F0F5 F1F6F1F5 F2F1F0F2 F0F5F5C6 310516152102055F
      01009E60 0020 E2E3E7C9 E3D7C940 F1F0F2F0 F5F540C6 STXITPI 102055 F
      01009E70 0030 E3D7F6F0 F0F0F0F1 C6C57FFF 0000F000 TP600001FE"~..0.
      01009E80 0040 C440F440 F2F0F1F6 00004040 40404040 D 4 2016..
      01009E90 0050 40404040 00000000 D6C340D4 F7F0E9E3 ....OC M70ZP
      01009EA0 0060 E7E3F5C3 40404040 40404040 40404040 XT5C
      01009EB0 0070 40404040 FFFFFFFF FFFFFFFF FFFFFFFF ~~~~~
      01009EC0 0080 FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF ~~~~~
      = 01011EA0 8060 FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF ~~~~~
      01011EB0 8070 FFFFFFFF ~~~

```

KDCROOT : Context Area

```

PC=01592532 IW=5C ILC=40 PM=0C CC=00 AMODE=31 STXITPI + 000004A2
R00 00000002 R01 01B0D4E0 R02 01592790 R03 010241C0
R04 01592790 R05 01B5948C R06 01016000 R07 01015000
R08 01BF2708 R09 00000050 R10 01592090 R11 0107BF80
R12 01AD8B08 R13 01B0D480 R14 815921DC R15 0102D000

```

---

## 4 UTM-Meldungswesen

openUTM erzeugt beim Ablauf einer UTM-Anwendung Meldungen, die über bestimmte Ereignisse informieren.

Eine **UTM-Meldung** besteht aus einer Meldungsnummer, einem festen **Meldungstext** und variablen Teilen, so genannten **Inserts**. Diese **Inserts** werden dynamisch bei Ausgabe der jeweiligen Meldung mit den aktuellen Werten versorgt. Bei den Inserts handelt es sich z.B. um den Namen der Anwendung oder des Kommunikationspartners oder um Zähler, Returncodes oder Ähnliches.

Jede Meldung wird über ihre **Meldungsnummer** identifiziert. UTM-Meldungsnummern beginnen immer mit dem Buchstaben K oder P gefolgt von einer 3-stelligen Nummer, z.B. K008.

Die UTM-Meldungen haben unterschiedliche Aufgaben und können an verschiedene Empfänger (**Meldungsziele**) gerichtet werden, wobei Sie in einem gewissen Rahmen die Meldungsziele selbst Anwendungs-spezifisch festlegen können.

Nachfolgend einige Erläuterungen und Beispiele für UTM-Meldungen.

- Ein Benutzer kann am Terminal durch eine Meldung über ein bestimmtes Ereignis informiert und ggf. zu einer Eingabe aufgefordert werden.

*Beispiel*

Bei der Berechtigungsprüfung war das eingegebene Passwort ungültig. openUTM fordert den Benutzer am Terminal auf, die KDCSIGN Eingabe zu wiederholen.

- Ein Ereignis innerhalb der UTM-Anwendung wird durch eine Meldung in der UTM-Protokolldatei SYSLOG protokolliert. So werden Daten zur Überwachung des Anwendungslaufes und für Diagnosezwecke gesammelt.

*Beispiel*

Eine Meldung informiert darüber, dass sich ein Benutzer an die Anwendung angemeldet hat.

- Werden bestimmte UTM-Meldungen erzeugt, für die Sie das Meldungsziel MSGTAC festgelegt haben, dann ruft openUTM ein ereignisgesteuertes Teilprogramm der Anwendung auf (siehe Event-Service MSGTAC im openUTM-Handbuch „Anwendungen programmieren mit KDCS“). Dieser Event-Service kann u.a. per FPUT asynchrone Aufrufe an die Administration absetzen. Man kann somit auf Ereignisse, bei denen Meldungen erzeugt werden, mit programmierter Administration reagieren.

### *Beispiel*

Beim Ereignis „Vorgangsabbruch“ wird die Meldung K017 bzw. K055 erzeugt. Der MSGTAC-Service kann darauf reagieren, indem er z.B. den TAC sperrt und eine Nachricht an den Administrator schickt.

Zur Ausgabe einer Meldung greift openUTM auf die UTM-eigenen Meldungsmodule zu. Diese enthalten die Eigenschaften und Texte aller UTM-Meldungen.

Sie können die Ausgabe von UTM-Meldungen in einem gewissen Rahmen Anwendungsspezifisch gestalten, siehe auch [Abschnitt „Gestaltung der Meldungsabgabe durch den Anwender“ auf Seite 124](#).

## 4.1 Meldungsmodul, Meldungsdefinitionsdatei

Bei der Ausgabe einer Meldung greift openUTM auf die UTM-Meldungsmodulare zu. Diese enthalten für jede Meldung u.a.

- die Meldungsnummer Knnn oder Pnnn
- den Meldungstext
- die Meldungsziele
- die Platzhalter für die Inserts

Mit openUTM werden die Meldungsdefinitionsdatei SYSMSH.UTM.065.MSGFILE und die beiden Systemmeldungsmodulare

KCSMSGs (deutsche Meldungstexte)

KCSMSGSE (englische Meldungstexte)

ausgeliefert. Die Systemmeldungsmodulare enthalten Standardeinstellungen für die Meldungsziele und deutsche bzw. englische Meldungstexte. Die Meldungsdefinitionsdatei dient als Basis zum Ändern der UTM-Meldungen durch den Anwender. Sie enthält die Meldungstexte in deutscher und englischer Sprache.



### **VORSICHT!**

Sie dürfen die Meldungsdefinitionsdatei nur mit den openUTM-Tools KDCMtxt und KDCMMOD bearbeiten! Andere Schreibzugriffe - z.B. mit einem Editor - **zerstören** diese Datei!

Die Meldungsdefinitionsdatei kann mit Meldungstexten in weiteren Sprachen angereichert werden (siehe Seite [125ff](#)). Aus der Meldungsdefinitionsdatei können Sie mit dem Tool KDCMMOD (siehe Seite [132ff](#)) eigene Meldungsmodulare erstellen.

In den Standardmeldungsmodularen ist eine bestimmte Art der Meldungsangabe vorgegeben. Wenn Sie diese Vorgaben ändern wollen, müssen Sie einen (oder mehrere) eigene Meldungsmodulare erzeugen.

## 4.2 Meldungsziele

Jede Meldung, die openUTM während einer laufenden Anwendung erzeugt, kann an eines oder mehrere der folgenden Meldungsziele ausgegeben werden:

SYSLST	Ausgabe auf SYSLST.
SYSOUT	Ausgabe auf SYSOUT.
STATION	Ausgabe an einen Client, der über ein PTERM oder einen TPOOL mit PTY-PE≠APPLI, SOCKET oder UPIC angeschlossen ist.
SYSLINE	Ausgabe in die Systemzeile des Terminals, dabei bleiben Benutzerformate am Bildschirm erhalten. Soll eine Meldung in der Systemzeile erscheinen, so sind SYSLINE und STATION als Meldungsziel anzugeben.
CONSOLE	Ausgabe auf die Konsole des Systemoperators, der Anwendungsname wird mit ausgegeben.
PARTNER	Ausgabe an einen Client, der über ein PTERM oder einen TPOOL mit PTY-PE=APPLI oder SOCKET angeschlossen ist.
SYSLOG	Ausgabe in die System-Protokolldatei SYSLOG (siehe <a href="#">Abschnitt „UTM-Protokolldatei SYSLOG“ auf Seite 143</a> ).
MSGTAC	Ausgabe an das MSGTAC-Teilprogramm (siehe Steueranweisung TAC im openUTM-Handbuch „Anwendungen generieren“ und Event-Service MSGTAC im openUTM-Handbuch „Anwendungen programmieren mit KDCS“).
USER-DEST-1 bis USER-DEST-4	<p>Ausgabe an Benutzer-spezifische Meldungsziele, denen Sie bei der UTM-Generierung als konkrete Meldungsziele eine USER-Queue, eine TAC-Queue, einen Asynchron-TAC oder einen LTERM-Partner zuordnen können.</p> <p>WinAdmin und WebAdmin können aus diesen UTM-Queues, zyklisch oder auf Anforderung des Anwenders, Meldungen abholen, diese in Listen ausgeben und gegebenenfalls in der Konfigurationsdatenbank abspeichern. Siehe hierzu die Online-Hilfe zu WinAdmin/WebAdmin, Stichwort "Meldungskollektoren".</p> <p>Die Zuordnung zwischen USER-DEST-1..4 und dem konkreten Ziel wird über die KDCDEF-Anweisung MSG-DEST definiert.</p>



## 4.2.1 Ausgabeform der Meldungen

Abhängig vom Meldungsziel gibt openUTM die Meldungen in unterschiedlicher Form aus:

- an SYSLOG und MSGTAC:  
den Meldungskopf ohne den Meldungstext und die aktuellen Werte aller Inserts, wie in [Abschnitt „Aufbau der UTM-Systemmeldungen“ auf Seite 148](#) beschrieben,
- an CONSOLE:  
den Anwendungsnamen, die Meldungsnummer und den Meldungstext mit den im Text enthaltenen Inserts,
- an USER-DEST-1...USER-DEST-4:
  - den Meldungskopf ohne den Meldungstext und die aktuellen Werte aller Inserts, wie in [Abschnitt „Aufbau der UTM-Systemmeldungen“ auf Seite 148](#) beschrieben, falls USER-DEST-*n* mit MSG-FORMAT=FILE generiert ist,
  - Datum/Uhrzeit gefolgt von Meldungsnummer, Meldungstext und allen Inserts, falls USER-DEST-*n* mit MSG-FORMAT=PRINT generiert ist,
- an alle anderen Ziele:  
die Meldungsnummer und den Meldungstext mit den im Text enthaltenen Inserts. In bestimmten Fehlersituationen im Lauf der Anwendung (abnormale Beendigung der Anwendung, abnormale Programm- oder Prozessbeendigung) ist es möglich, dass openUTM eine Meldung nur an die Ziele SYSOUT, SYSLST oder CONSOLE ausgibt, auch wenn die Meldung für weitere Ziele (z.B. SYSLOG) bestimmt ist. Dies geschieht, um eventuelle Folgefehler zu verhindern.  
Aus dem gleichen Grund entnimmt openUTM in bestimmten Fehlersituationen den Meldungstext auch nicht dem Meldungsmodul, das der Anwender verändern kann, sondern dem Systemmeldungsmodul.

### Zeitangaben-Header für Meldungen auf SYSLST / SYSOUT

Bei der Ausgabe von Meldungen an die Meldungsziele SYSLST und SYSOUT stellt openUTM den Meldungen standardmäßig einen Header mit Datum und Uhrzeit voran. Damit lassen sich z.B. Fehlersituationen leichter diagnostizieren.

Dieser Header hat folgende Form:

jjjj-mm-tt hh:mm:ss

Die Meldungen K038 und K044 werden immer ohne diesen Header ausgegeben. Mit dem Startparameter UTM-MSG-DATE=NO lässt sich die Ausgabe dieses Headers auch für alle anderen Meldungen unterdrücken.

## 4.2.2 UTM-Meldungen an die BS2000-Konsole

Soll openUTM Meldungen auf die BS2000-Konsole ausgeben (Meldungsziel = CONSOLE), muss die BS2000-Meldungsdatei die UTM-Meldung UTM0100 enthalten. Diese Meldung wird von openUTM nur als „Behälter“ für die jeweilige aktuelle UTM-Meldung benutzt. In Meldungen, die auf die BS2000-Konsole ausgegeben werden, trägt openUTM zusätzlich zum Meldungstext noch den Anwendungsnamen ein.

## 4.2.3 UTM-Meldungen an eine TS-Anwendung

Sind UTM-Meldungen für eine Transportsystem-Anwendung vom Typ PTYPE=APPLI oder SOCKET bestimmt (Meldungsziel = PARTNER), so ist darauf zu achten, dass diese Anwendung die Meldungen erkennt und sinnvoll darauf reagiert. Geschieht dies nicht, kann es z.B. bei der Kopplung zweier UTM-Anwendungen als TS-Anwendungen dazu kommen, dass die Anwendungen sich gegenseitig endlos Meldungen übermitteln wie

```
K009 Der Transaktionscode K009 ist ungueltig.
```

In diesem Fall könnte die UTM-Anwendung beispielsweise einen TAC K009 definieren und diesem ein Programm zuordnen, das auf diese Meldung reagieren soll.

### USP-Header bei UTM-Meldungen an eine Socket-Anwendung

Für UTM-Meldungen an eine Socket-Anwendung (PTYPE=SOCKET, Meldungsziel = PARTNER) können Sie bei der UTM-Generierung festlegen, ob openUTM der Meldung einen USP-Header (openUTM-**S**ocket-**P**rotokoll-Header) voranstellen soll. Der USP-Header dient u.a. dazu, dem Socket-Partner die Länge der empfangenen Nachrichten mitzuteilen.

Dazu legen Sie bei der KDCDEF-Generierung im Operanden USP-HDR= der PTERM- oder TPOOL-Anweisung folgendes fest:

```
USP-HDR = MSG oder USP-HDR = ALL
```

Falls Sie USP-HDR = NO (Standardeinstellung) generieren, so wird kein USP-Header erzeugt.

Eine Beschreibung des USP-Headers finden Sie im openUTM-Handbuch „Anwendungen programmieren mit KDCS“.

#### 4.2.4 UTM-Meldungen an Benutzer-spezifische Meldungsziele

Bei Auftreten einer Meldung, für die als Meldungsziel USER-DEST-1 ...USER-DEST-4 vereinbart wurde, erzeugt UTM intern einen Asynchron-Auftrag an dieses Meldungsziel. Diesem Asynchron-Auftrag wird als Verursacher der Benutzer KDCMSGUS und der LTERM-Partner KDCMSGLT zugeordnet. Wird der Asynchron-Auftrag zurückgewiesen, z.B. weil das Meldungsziel gesperrt ist (STATUS=OFF), geht die Meldung für das Meldungsziel verloren. Wird wieder eine Meldung für dieses Meldungsziel erzeugt, versucht UTM erneut, einen Asynchron-Auftrag zu erzeugen.

Ist als Meldungsziel ein Asynchron-TAC generiert, dann startet UTM das dem TAC zugeordnete Programm jedes Mal, wenn die betreffende Meldung erzeugt wurde (d.h. in einem Programmlauf kann anders als im MSGTAC-Programm immer nur eine Meldung mit FGET gelesen werden).

#### 4.2.5 UTM-Meldungen an MSGTAC

MSGTAC ist ein spezielles Asynchron-Teilprogramm, das der Anwender selbst programmieren kann, siehe openUTM-Handbuch „Anwendungen programmieren mit KDCS“.

Ist ein MSGTAC-Programm vorhanden und tritt eine Meldung auf, für die das Meldungsziel MSGTAC vereinbart wurde, dann wird der Asynchron-Vorgang MSGTAC gestartet. Das MSGTAC-Programm kann in einem Teilprogrammlauf mehrere Meldungen lesen.

Das MSGTAC-Teilprogramm läuft unter der internen UTM-Benutzerkennung KDCMSGUS mit KSET=MASTER und PERMIT=ADMIN. Es muss in einer TAC-Anweisung definiert werden mit `TAC KDCMSGTC,PROGRAM=...`

## 4.3 Gestaltung der Meldungsausgabe durch den Anwender

In der Meldungsdefinitionsdatei ist festgelegt, wie openUTM standardmäßig Meldungen ausgibt, d.h. mit welchen Texten, an welche Ziele, usw. sowie die Einschränkungen für deren Änderbarkeit. Wenn Sie diese Standardeigenschaften ändern wollen, müssen Sie ein eigenes Meldungsmodul erzeugen und zu der UTM-Anwendung binden. Folgende Änderungen sind möglich:

- Hinzufügen oder Weglassen von Meldungszielen (mit KDCMMOD) wie z.B. Ausgabe weiterer Meldungen nach SYSOUT oder Eintragen des Ziels MSGTAC als Voraussetzung für den Einsatz eines MSGTAC-Programms.
- Meldungstexte in eine andere Sprache übersetzen und mit dem Tool KDCMTXT in die Meldungsdefinitionsdatei einbringen ([Seite 125](#)).
- Ändern von Meldungstexten mit dem Tool KDCMMOD ([Seite 132](#)) wie z.B. Hinzufügen oder Weglassen von (für die Meldung definierten) Inserts in einem Meldungstext, Texte in Klein-/Groß-Buchstaben umsetzen usw.
- Definieren von Textkonstanten. Diese können dann in Meldungstexten verwendet werden. Dies können auch Steuerzeichen sein.

### *Beispiel*

Sie können bestimmte Meldungen akustisch ankündigen lassen.



Die individuelle Gestaltung der Meldungsausgabe gilt jeweils nur für die UTM-Anwendungen, die mit dem geänderten Meldungsmodul gebunden werden. Sie hat keine Auswirkung auf andere UTM-Anwendungen auf dem selben Rechner und erfordert keine Änderungen im BS2000-System.

Sie können mehrere eigene Meldungsmodule erstellen und in die Anwendung einbinden. Damit haben Sie die Möglichkeit, Meldungstexte Benutzer-spezifisch auszugeben, z.B. in unterschiedlichen Sprachen.

Die Meldungsziele werden aus einem Anwendungs-spezifischen Standardmeldungsmodul entnommen (siehe Abschnitt „[Erstellen eines eigenen Meldungsmoduls mit KDCMMOD](#)“ auf [Seite 132](#)).

Die Modifikationen der Meldungsdefinitionsdatei dürfen Sie nur mit Hilfe der Tools KDCMTXT durchführen.

Bei Übersetzung aller Meldungstexte in eine andere Sprache muss das Tool KDCMTXT verwendet werden ([Seite 125](#)). Wenn nur wenige Texte für eine in der Meldungsdefinitionsdatei vorhandenen Sprache geändert werden sollen, kann das Tool KDCMMOD verwendet werden.

### Ausnahmen bei der Meldungsabgabe

openUTM nimmt bei der Ausgabe von UTM-Meldungen den Meldungstext und die aktuellen Meldungsziele aus dem Systemmeldungsmodul bzw. die Meldungsziele aus dem Standardmeldungsmodul und die Meldungstexte aus dem Standardmeldungsmodul oder einem Benutzermeldungsmodul, falls einer oder mehrere eigene Meldungsmodulare generiert sind. Davon gibt es jedoch folgende Ausnahmen:

- Für einige Meldungen, die von ROOT-Modulen ausgegeben werden, sind englische Meldungstexte einprogrammiert, z.B. für K078. Für diese Meldungen kann die Ausgabe nicht verändert werden.
- In der Start- und Ende-Behandlung der Tasks kann openUTM aus technischen Gründen nicht auf das/die Meldungsmodul(e) der Anwendung zugreifen. openUTM nimmt in diesen Fällen den Meldungstext und die Meldungsziele aus dem Meldungsmodul KCSMSGSE, der deutsche Meldungstexte enthält. KCSMSGSE wird beim Start des Subsystems UTM als Teil des UTM-Systemcodes geladen.

Wenn Sie in einer englischsprachigen Anwendung das englische Systemmeldungsmodul einsetzen wollen, können Sie so vorgehen:

1. Löschen Sie vor dem Start des Subsystems UTM das Meldungsmodul KCSMSGSE in der Bibliothek SYSLNK.UTM.065.TPR.
2. Kopieren Sie an seiner Stelle das Modul KCSMSGSE aus SYSLNK.UTM.065 und benennen Sie den Entry KCSMSGSE in KCSMSGSE um.

### 4.3.1 Meldungen in anderen Sprachen - das Tool KDCMTXT

Die mit openUTM ausgelieferte Meldungsdefinitionsdatei SYSMH.UTM.065.MSGFILE enthält die Texte der UTM-Meldungen in deutscher und englischer Sprache. Wenn openUTM Meldungen in einer anderen Sprache ausgeben soll, dann können Sie die UTM-Meldungstexte in diese Sprache übersetzen und mit Hilfe des Tools KDCMTXT in die Meldungsdefinitionsdatei einfügen.



#### **VORSICHT!**

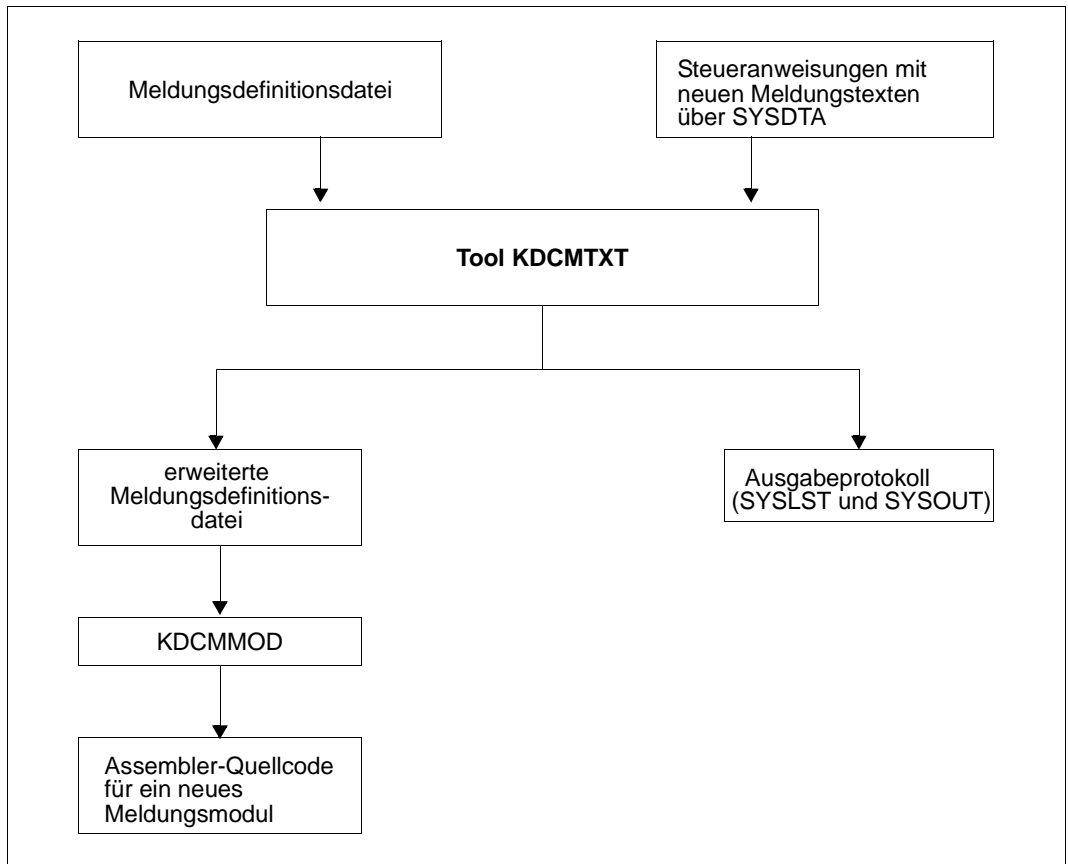
Änderungen und Erweiterungen der Meldungsdefinitionsdatei dürfen Sie nur mit Hilfe des Tools KDCMTXT vornehmen. Andere Schreibzugriffe - z.B. mit einem Editor - **zerstören** die Datei!

Die Texte müssen Sie in Form spezieller Steueranweisungen an KDCMTXT übergeben. Aus der so erweiterten Meldungsdefinitionsdatei können Sie mit KDCMMOD das Assembler-Quellprogramm für ein neues Meldungsmodul erzeugen, das die Meldungen in der von Ihnen gewünschten Sprache enthält. KDCMMOD ist ab [Seite 132](#) beschrieben. Nach dem Assemblieren können Sie dieses Meldungsmodul an Stelle des Systemmeldungsmoduls in Ihre Anwendung einbinden.

## Hinweis

Bei der Definition von Meldungstexten darf ein für die jeweilige Meldung zugelassenes Insert nur einmal verwendet werden.

### 4.3.1.1 Ein- und Ausgaben



Meldungen in anderen Sprachen erstellen mit KDCMTXT

### 4.3.1.2 Aufruf von KDCMTXT

Das Tool KDCMTXT wird in der Bibliothek SYSLNK.UTM.065.UTIL ausgeliefert. Es wird mit /START-EXECUTABLE-PROGRAM gestartet, z.B.

```
/START-EXECUTABLE-PROGRAM FROM-FILE=*LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL
, ELEM=KDCMTXT)
```

KDCMTXT liest die Steueranweisungen von SYSDTA.



Sie können KDCMTXT auch per SDF-Kommando START-KDCMTXT starten, siehe auch openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Abschnitt „UTM-Tools über eigene SDF-Kommandos starten“.

### 4.3.1.3 KDCMTXT-Steueranweisungen

KDCMTXT kennt folgende Steueranweisungen:

OPTION	Namen der Meldungsdefinitionsdatei angeben
FU	Funktionseinheit und Landessprache definieren.
MSGBASE	Auswahl der Meldungsgruppe bei der FU SYS: UTM (K-Meldungen) oder XAPTP (P-Meldungen)
STDTXT	Anfang des Standardmeldungstextes definieren
ENDTXT	Ende des Standardmeldungstextes definieren
END	Ende der Eingabe von Steueranweisungen definieren

Bei den Eingaben der Anweisungen gelten folgende Regeln:

- Kommentarzeilen werden durch einen Stern (\*) in der 1. Spalte gekennzeichnet
- Endet eine Zeile mit einem Komma, dann interpretiert KDCMTXT die folgende Zeile als Fortsetzungszeile der Anweisung.

Zweckmäßigerweise schreiben Sie die Steueranweisungen für KDCMTXT in eine Datei.

Beim Erstellen dieser KDCMTXT-Steueranweisungsdatei kann Ihnen die Datei SYSDAT.UTM.065.MTXTIN helfen, die mit openUTM ausgeliefert wird. Diese Datei enthält die deutschen und englischen Standardmeldungstexte in der Syntax der KDCMTXT-Steueranweisungen. Sie können diese Datei als Vorlage für Ihre eigene Eingabedatei benutzen.

### Anweisung OPTION

Mit OPTION können Sie den Namen der Meldungsdefinitionsdatei angeben, die durch KDCMTXT geändert werden soll.

Die Anweisung OPTION muss, wenn sie angegeben wird, die erste Steueranweisung sein, andernfalls erhalten Sie einen Syntaxfehler. Wenn Sie keine OPTION-Anweisung angeben, verwendet KDCMTXT die standardmäßig mit openUTM ausgelieferte Meldungsdefinitionsdatei SYSMSH.UTM.065.MSGFILE.

Aus Sicherheitsgründen sollten Sie die Meldungsdefinitionsdatei kopieren und mit KDCMTXT die Kopie bearbeiten.

Operation	Operanden
OPTION	MSGFILE=filename

filename Name der Meldungsdefinitionsdatei, die erweitert werden soll.

Standard: SYSMSH.UTM.065.MSGFILE

### Anweisung FU

Mit dieser Anweisung können Sie die Funktionseinheit und die Landessprache definieren.

Operation	Operanden
FU	[funktionseinheit][,][LANG=sprachkennzeichen]

funktionseinheit

Funktionseinheit, für die die neue Landessprache eingetragen werden soll. Zurzeit ist nur der Wert SYS erlaubt. SYS steht für den Transaktionsmonitor.

SYS: K-Meldungen (K001-K399) bzw. P-Meldungen (P001 - P049)

Standard: SYS

LANG=sprachkennzeichen

Das Sprachkennzeichen ist maximal 3 Zeichen lang und kennzeichnet die Landessprache. Das Sprachkennzeichen kann frei vergeben werden, es muss jedoch innerhalb einer Meldungsdefinitionsdatei eindeutig sein.



Als Sprachkennzeichen können z.B. die Kfz-Länderkennzeichen oder die in ISO IS/R639 festgelegte Sprachkennzeichen verwendet werden.

KFZ-Kennz.	ISO IS/R639	Sprache
DK	Da	Dänisch
D	De	Deutsch
GB	En	Englisch
E	Es	Spanisch
FI	Fi	Finnisch
F	Fr	Französisch
GR	Gr	Griechisch
I	It	Italienisch
NL	Nl	Niederländisch
N	No	Norwegisch
P	Pt	Portugiesisch
S	Sv	Schwedisch

Standard: GB

Das Komma muss nur dann angegeben werden, wenn sowohl *funktionseinheit* als auch LANG=sprachkennzeichen angegeben werden., z.B. FU SYS, LANG=DK.

Die Angabe LANG= dient nur der Kennzeichnung der Texte in der Meldungsdefinitionsdatei und hat keinen Bezug zu dem mit dem KDCDEF-Parameter LOCALE generierten Sprachkennzeichen eines Benutzers!

Für eine Funktionseinheit müssen alle Meldungstexte für eine Landessprache definiert werden, damit die Meldungstexte für diese Sprache und Funktionseinheit in die Meldungsdefinitionsdatei übernommen werden.

Nach Abarbeitung der Anweisung FU wird implizit die Anweisung MSGBASE UTM ausgeführt.



### **VORSICHT!**

Die ausgelieferte Meldungsdefinitionsdatei enthält bereits die deutschen und die englischen Meldungstexte mit Sprachkennzeichen D bzw. GB! Geben Sie eines dieser Kennzeichen an, überschreibt KDCMTXT die entsprechenden Standardmeldungstexte in der Meldungsdefinitionsdatei mit den neuen Meldungstexten.

### Anweisung MSGBASE (nur bei FU SYS)

openUTM unterscheidet innerhalb der FU SYS zwei Meldungsgruppen: Meldungen der Gruppe UTM, die mit dem Buchstaben K beginnen, und Meldungen der Gruppe XAPTP, die mit dem Buchstaben P beginnen.

Mit der Anweisung MSGBASE können Sie die Meldungsgruppe auswählen, auf die sich die nachfolgenden STDTXT-Anweisungen bis zur nächsten MSGBASE- bzw. FU-Anweisung beziehen.

Operation	Operanden
MSGBASE	{ UTM   XAPTP }

UTM           Auswahl der Meldungsgruppe UTM (K-Meldungen).

XAPTP        Auswahl der Meldungsgruppe XAPTP (P-Meldungen).

### Anweisungen STDTXT und ENDTXT

Die Steueranweisung STDTXT leitet die Definition des Standardtextes für eine Meldung ein. Die nachfolgenden Eingabezeilen beschreiben den Standardtext.

Die ENDTXT-Anweisung schließt die Definition des Meldungstextes ab. Der übersetzte Meldungstext wird wie folgt an KDCMTXT übergeben:

Operation	Operanden
STDTXT	msg-nr text
ENDTXT	

msg-nr       Meldungsnummer, für die der Standardtext bestimmt ist.

msg-nr gibt an, welche K- bzw. P-Meldung modifiziert werden soll. Die Meldungsnummer muss im Meldungsnummernbereich der spezifizierten Funktionseinheit liegen. Die Meldungsnummer ist dreistellig und **ohne** das vorangestellte **K** bzw. **P** anzugeben.

Pflichtoperand.

text           Hier ist der Meldungstext anzugeben, die Syntax ist dieselbe wie beim Programm KDCMMOD (siehe Seite 139); ebenso gelten die gleichen Längenbeschränkungen.

Die Steueranweisungen STDTXT und ENDTXT sowie der Meldungstext müssen jeweils in einer neuen Zeile beginnen.

Die STDTXT / ENDTXT-Anweisungen für eine Funktionseinheit und eine Sprache müssen alle hinter der FU-Anweisung für diese Funktionseinheit stehen und vor der nächsten FU-Anweisung.

KDCMTXT übernimmt die Meldungstexte für eine weitere Sprache nur dann in die Meldungsdefinitionsdatei, wenn die Eingabedaten Meldungstexte für alle Meldungen dieser Sprache enthalten, und wenn alle Eingaben fehlerfrei sind. Sind die Meldungstexte einer Sprache bereits in der Meldungsdefinitionsdatei vorhanden, dann übernimmt KDCMTXT auch einzelne Meldungen. Bereits vorhandene Meldungstexte mit derselben Meldungsnummer und demselben Sprachkennzeichen werden dann überschrieben.

### **Anweisung END**

Diese Anweisung beendet die Eingabe von Steueranweisungen an KDCMTXT. END hat keine Operanden.

#### **4.3.1.4 Protokoll von KDCMTXT**

Nachdem das Tool KDCMTXT alle Änderungen bearbeitet und eine neue Meldungsdefinitionsdatei erstellt hat, gibt es als Protokoll eine aufbereitete Liste aller Meldungen nach SYSLST und SYSOUT aus (Ausgabeprotokoll). In der Liste werden die Meldungstexte in der Form aufbereitet, in der sie an eine Datensichtstation ausgegeben würden. Dabei werden die Inserts je nach Typ in der entsprechenden Länge wie folgt gefüllt:

CHAR	mit	@-Zeichen,
INT	mit	#-Zeichen,
HEXA	mit	X-Zeichen.

### 4.3.2 Erstellen eines eigenen Meldungsmoduls mit KDCMMOD

Zum Erzeugen eines eigenen Meldungsmoduls steht das Tool KDCMMOD zur Verfügung. Es erstellt aus der Meldungsdefinitionsdatei und einer vom Anwender verfassten Änderungsbeschreibung ein Assembler-Quellprogramm, aus dem nach Übersetzung das Anwendungs-spezifische Meldungsmodul entsteht. Dieses wird mit den Teilprogrammen einer Anwendung zusammen gebunden.

Ein Anwendungs-spezifisches Modul wird in der Startphase von openUTM auf Gültigkeit überprüft und dann zusätzlich zum Systemmeldungsmodul benutzt.

In Fällen, in denen nicht auf das Anwendungs-spezifische Meldungsmodul zugegriffen werden kann, wird auf das Systemmeldungsmodul zurückgegriffen (z.B. bei Beendigung einer UTM-Task). Das Systemmeldungsmodul ist im UTM-Systemcode enthalten.

Der Anwender kann mehrere eigene Meldungsmodulare erstellen und in die Anwendung einbinden, aber nur ein Meldungsmodul kann das Anwendungs-spezifische Meldungsmodul sein. Welches Meldungsmodul das Anwendungs-spezifische Meldungsmodul ist wird bei der UTM-Generierung über den Parameter LOCALE der KDCDEF-Anweisung MESSAGE festgelegt. Dieses Anwendungs-spezifische Meldungsmodul wird auch Standardmeldungsmodul genannt. Alle anderen Meldungsmodulare sind Benutzer-spezifische Meldungsmodulare.

Die Meldungsziele werden immer aus dem Standardmeldungsmodul genommen.

Weitere Informationen zu diesem Thema, z.B. wie eigene Meldungsmodulare zum Anwendungsprogramm gebunden werden, sind im openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“ enthalten.

Das Tool KDCMMOD legt das erzeugte Assembler-Quellprogramm ab in der Datei SRC.ASSEMB.object-modul-name, wobei object-modul-name der in der GEN-Anweisung festgelegte Name des Meldungsmoduls ist.

Sie können Folgendes angeben bzw. ändern:

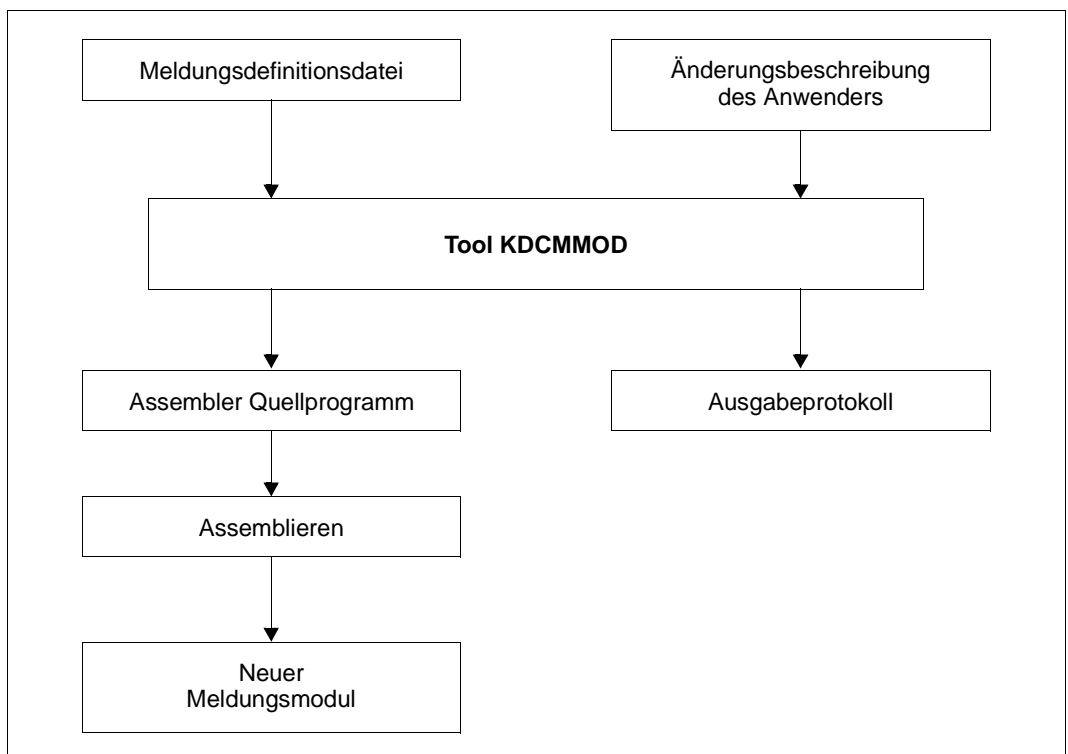
- die Landessprache, in der die Meldungen generiert werden sollen
- die Meldungstexte
- die Meldungsziele
- Meldungsattribute
- Steuerzeichen

Die Meldungstexte können in einem weit gesteckten Rahmen verändert werden, wobei die Längenbeschränkungen zu beachten sind, siehe [Seite 140](#).

Es ist möglich, Inserts, die im Standardtext enthalten sind, aus dem Meldungstext zu entfernen oder Inserts einer Meldung, die im Standardtext nicht enthalten sind, in den Text aufzunehmen. Außerdem ist es möglich, die Reihenfolge der Inserts innerhalb des Textes zu verändern. Jedes Insert darf jedoch höchstens einmal in der Meldung vorkommen. Auch die Meldungsnummer kann an eine andere Stelle im Meldungstext verschoben oder sogar ganz aus diesem entfernt werden. Dies sollte aber nur in besonderen Ausnahmefällen gemacht werden, da dies eine Diagnose anhand der Meldungen erschweren kann.

Die Meldungsziele können innerhalb gewisser Grenzen geändert werden (siehe dazu [Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353](#)). Beachten Sie dabei, dass für jede Meldung unterschiedliche Ziele vorgeschrieben bzw. verboten sein können.

#### 4.3.2.1 Ein- und Ausgaben von KDCMMOD



Ablauf beim Erstellen eines eigenen Meldungsmoduls

Nachdem das Tool KDCMMOD alle Änderungen bearbeitet hat, gibt es als Protokoll auf SYSOUT eine aufbereitete Liste aller Meldungen aus. Außerdem erstellt das Programm eine Übersicht der Meldungsdefinitionen (Insertliste, Ziele, Compression), entstanden aus den Rahmendefinitionen und Modifikationen.

In der Ausgabeliste werden die definierten Standardtexte in der Form aufbereitet, in der sie an eine Datensichtstation ausgegeben würden. Dabei werden die Inserts je nach Typ in der entsprechenden Länge wie folgt gefüllt:

CHAR mit @-Zeichen,  
 INT mit #-Zeichen,  
 HEXA mit X-Zeichen.

Die Meldungen von KDCMMOD finden Sie ab [Seite 332](#).

#### 4.3.2.2 Starten und Beenden von KDCMMOD

Das Tool KDCMMOD wird wie folgt gestartet:

```
START-EXECUTABLE-PROGRAM FROM-FILE=*LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL
, ELEM=KDCMMOD)
```



Sie können KDCMMOD auch per SDF-Kommando START-KDCMMOD starten, siehe auch openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Abschnitt „UTM-Tools über eigene SDF-Kommandos starten“.

KDCMMOD liest die Anweisungen der Änderungsbeschreibung von der Systemdatei SYSDTA. Es werden einzelne Zeilen gelesen, wobei

- eine Kommentarzeile durch das Zeichen „\*“ (Stern) in Spalte 1 gekennzeichnet wird,
- für Zeilen mit einem Komma als letztem Zeichen die nächste Zeile als Fortsetzungszeile interpretiert wird.

Alle anderen Zeilen werden vom Programm einzeln analysiert.

Beendet wird der Programmablauf von KDCMMOD mit der Steueranweisung END.

#### 4.3.2.3 Steueranweisungen von KDCMMOD

KDCMMOD kennt folgende Steueranweisungen:

OPTION	Namen der Meldungsdefinitionsdatei festlegen
CONSTANT	Definieren von Konstanten
GEN	Generieren von Meldungen für eine Funktionseinheit
MSGBASE	Auswahl der Meldungsgruppe: UTM (K-Meldungen) oder XAPTP (P-Meldungen)
MODMSG	Ändern von Meldungen
ENDMSG	Ende der Meldungsdefinition
END	Ende der Eingabe der Steueranweisungen

Bei der Reihenfolge der Steueranweisungen von KDCMMOD ist zu beachten:

1. Die OPTION-Anweisung kann nur als erste Steueranweisung angegeben werden.
2. Die CONSTANT-Anweisung muss in der Eingabedatei stehen, bevor eine Textdefinition Bezug auf sie nimmt.
3. Die GEN-Anweisung muss in der Eingabedatei vor allen MODMSG/ENDMSG-Steueranweisungen stehen.
4. Die MSGBASE-Anweisung wirkt auf alle nachfolgenden MODMSG-Anweisungen.
5. Die END-Anweisung muss am Ende der Eingabedatei stehen.

Treten bei einer Anweisung für KDCMMOD Fehler auf, so wird die falsche Anweisung nach Ausgabe einer entsprechenden Fehlermeldung (K6xx) ignoriert. Der Programmlauf wird dadurch nicht beendet.

### Anweisung OPTION

Mit der OPTION-Anweisung legt man den Namen der Meldungsdefinitionsdatei fest, mit der das Tool KDCMMOD arbeiten soll.

Operation	Operanden
OPTION	MSGFILE=dateiname

dateiname      Name der zu bearbeitenden Meldungsdefinitionsdatei

Wird keine OPTION-Anweisung angegeben, dann wird als Standard die Datei SYSMSH.UTM.065.MSGFILE verwendet.

### Anweisung CONSTANT

Die CONSTANT-Anweisung definiert eine Konstante, die man anschließend in den Textdefinitionen der Meldungen verwenden kann.

Operation	Operanden
CONSTANT	constant-name, constant-wert

constant-name      legt den maximal 8 Zeichen langen Namen einer Konstanten fest. Falls eine Konstante mit dem gleichen Namen bereits existiert, wird die CONSTANT-Anweisung als fehlerhaft abgewiesen.

Pflichtoperand.

constant-wert ordnet dem constant-name einen Wert zu. Der Wert ist entweder als hexadezimale Größe (X'.....') oder abdruckbar in der Form '...' anzugeben. Die maximale Länge beträgt 50 Zeichen.

Pflichtoperand.

In der Meldungsdefinitionsdatei sind einige Konstanten bereits enthalten (siehe auch [Seite 340](#)). Weitere logische Steuerzeichen finden Sie im Macro VTCSET.

### Anweisung GEN

Die GEN-Anweisung gibt an, für welche Funktionseinheit und in welcher Landessprache die Meldungen generiert werden sollen. In einem Programmlauf darf sie nur **einmal** gegeben werden und muss vor allen MODMSG-Anweisungen stehen.

Operation	Operanden
GEN	[funktionseinheit] [, LANG=sprach-kennzeichen] ,MODULE=objekt-module-name

funktionseinheit

Name der Funktionseinheit, für die Meldungen generiert werden sollen. Zurzeit ist nur die Funktionseinheit SYS möglich, d.h. es werden die Meldungen des Transaktionsmonitors erzeugt.

Standardwert: SYS

LANG=sprach-kennzeichen

bezeichnet die Landessprache, für welche die Meldungstexte generiert werden sollen. Es muss ein Sprachkennzeichen angegeben werden, für das Meldungstexte in der Meldungsdatei vorhanden sind.

Standardwert: GB

MODULE=objekt-modul-name

bezeichnet den Namen des Meldungsmoduls. Er entspricht dem MODULE-Namen in der MESSAGE-Anweisung des Tools KDCDEF (siehe openUTM-Handbuch „Anwendungen generieren“).

Pflichtoperand.



### Anweisung MSGBASE (für FU SYS)

openUTM unterscheidet innerhalb der FU SYS zwei Meldungsgruppen: Meldungen der Gruppe UTM, die mit dem Buchstaben K beginnen, und Meldungen der Gruppe XAPTP, die mit dem Buchstaben P beginnen.

Mit der Anweisung MSGBASE können Sie die Meldungsgruppe auswählen, auf die sich die nachfolgenden MODMSG-Anweisungen innerhalb der Funktionseinheit SYS beziehen sollen.

Operation	Operanden
MSGBASE	{ <u>UTM</u>   XAPTP }

UTM           Voreinstellung. Auswahl der Meldungsgruppe UTM (K-Meldungen).

XAPTP        Auswahl der Meldungsgruppe XAPTP (P-Meldungen).

### Anweisungen MODMSG und ENDMSG

Mit der MODMSG-Anweisung ändern Sie eine Meldung der Funktionseinheit, die in der GEN-Anweisung angegeben wurde. Die Änderungen können den Text, die Inserts und die Meldungsziele betreffen. Es sind nur die Änderungen möglich, die durch die Rahmendefinitionen erlaubt sind (siehe [Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353](#)).

Mit der ENDMSG-Anweisung wird eine Meldungstextdefinition abgeschlossen. Sie muss immer in einer eigenen Zeile stehen und hat keine Operanden.

Operation	Operanden
MODMSG	<p>msg-nr</p> <p>[,BEL= {YES   NO}]</p> <p>[,COMPRESSION= {YES   NO}]</p> <p>[,EXTEND= {YES   NO}]</p> <p>[,CONSOLE= {YES   NO}]</p> <p>[,MSGTAC= {YES   NO}]</p> <p>[,PARTNER= {YES   NO}]</p> <p>[,STATION= {YES   NO}]</p> <p>[,SYSLINE= {YES   NO}]</p> <p>[,SYSLOG= {YES   NO}]</p> <p>[,SYSLST= {YES   NO}]</p> <p>[,SYSOUT= {YES   NO}]</p> <p>[,USER-DEST-1= {YES   NO}]</p> <p>[,USER-DEST-2= {YES   NO}]</p> <p>[,USER-DEST-3= {YES   NO}]</p> <p>[,USER-DEST-4= {YES   NO}]</p> <p>[text]</p>
ENDMSG	



- Die Operanden der MODMSG-Anweisung müssen durch ein Komma getrennt werden. Bei Fortsetzungszeilen in einer MODMSG-Anweisung muss das Komma immer als letztes Zeichen in der vorangehenden Zeile stehen (als Fortsetzungszeichen).
- Die Zeile vor [text] darf nicht mit einem Komma abgeschlossen werden.

**msg-nr** gibt an, welche K- bzw. P-Meldung modifiziert werden soll. Die Meldungsnummer muss im Meldungsnummernbereich der im GEN-Kommando spezifizierten Funktionseinheit liegen. Die Meldungsnummer ist **ohne** das vorangestellte **K** bzw. **P** anzugeben.

Pflichtoperand.

**BEL** gibt an, ob mit der Ausgabe der Nachricht an das Meldungsziel STATION oder SYSLINE ein akustisches Signal ausgelöst wird.

**YES** Ein akustisches Signal wird ausgelöst. YES ist nur sinnvoll für Meldungen, denen das Meldungsziel STATION oder SYSLINE zugeordnet ist.

NO Es wird kein akustisches Signal ausgelöst.  
Standard für alle Meldungen: NO

#### COMPRESSION=

YES Überflüssige Leerzeichen werden aus der Meldung entfernt. YES wirkt nur für Meldungsausgaben, die mit Meldungstext erfolgen.

NO Überflüssige Leerzeichen verbleiben in der Meldung.  
Der Standardwert ist für die einzelnen Meldungen unterschiedlich. Welcher Wert für welche Meldung voreingestellt ist, entnehmen Sie dem [Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353](#).

#### EXTEND=

YES Der Meldungstext soll im Extended Line Mode ausgegeben werden. YES ist nur sinnvoll für Meldungen, denen das Meldungsziel STATION oder SYSLINE zugeordnet ist.

NO Der Meldungstext soll nicht im Extended Line Mode ausgegeben werden  
Standard: Rahmendefinition in SYSMSH.UTM.065.MSGFILE.  
Der Standardwert ist für die einzelnen Meldungen unterschiedlich. Welcher Wert für welche Meldung voreingestellt ist, entnehmen Sie dem [Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353](#).

#### Meldungsziele CONSOLE, MSGTAC usw.

bezeichnet die Meldungsziele, an welche die Meldung ausgegeben (YES) bzw. nicht ausgegeben (NO) werden soll. Es dürfen nur die Meldungsziele angegeben werden, die in den Rahmendefinitionen mit 'erlaubt' (+) oder 'Default' (D) angegeben sind (siehe Tabelle ab Seite [353](#)). Für alle Meldungsziele, die nicht im MODMSG-Kommando angegeben werden, gelten die Rahmendefinitionen.

Die Zuordnung der Benutzer-spezifischen Meldungsziele USER-DEST-1,..., USER-DEST-4 zum konkreten Meldungsziel müssen Sie über die KDCDEF-Anweisung MSG-DEST vornehmen.

Die einzelnen Meldungsziele sind auf [Seite 120](#) näher beschrieben.

text Im Anschluss an das MODMSG-Kommando kann ein neuer Meldungstext definiert werden. Wird kein neuer Text definiert, dann gilt der Text, wie er in der Meldungsdefinitionsdatei vorgegeben ist.

Ein neuer Meldungstext wird in einer oder mehreren Zeilen definiert. Er besteht aus einer Folge von Textelementen, die entweder durch Komma oder durch Zeilenende voneinander getrennt sind.

Text = Textelement, Textelement,.....  
Textelement,.....

Das erste Textelement muss immer in einer neuen Zeile stehen.

### Längenbeschränkungen

Der Meldungstext darf inklusive Meldungsnummer und eventuell vorhandener Inserts nicht länger als 512 Zeichen sein. Die Längen der Inserts entnehmen Sie bitte der Tabelle auf [Seite 341](#). Ist der Text länger als 512 Zeichen, dann lehnt KDCMMOD ihn mit der Meldung K686 ab.

Meldungstexte mit dem Meldungsziel SYSLINE (=Systemzeile) dürfen nicht länger als 80 Zeichen sein. Ist der Text länger als 80 Zeichen, so warnt KDCMMOD mit der Meldung K687. Bei der Ausgabe werden längere Meldungen auf 80 Zeichen begrenzt, der Rest wird abgeschnitten.

Folgende Textelemente können verwendet werden:

- |                   |   |
|-------------------|---|
| Literal           | Ein Literal ist eine Zeichenkette, die in Hochkommata eingeschlossen ist. Doppelte Hochkommata in einem Literal ergeben das Zeichen Hochkomma im Literal.   |
| Insert            | Name eines Feldes (Insert) im Meldungstext, dem das Zeichen „&“ vorangestellt wird. Im Text dürfen nur die Feldnamen (Inserts) verwendet werden, die für die jeweilige Meldung zulässig sind, siehe <a href="#">Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353</a> . Inserts sind Platzhalter in Meldungstexten, in die vor Ausgabe der Meldung aktuelle Werte eingetragen werden. |
| Konstante         | Name einer Konstanten, dem das Zeichen „#“ vorangestellt ist.   |
| Built-in-Funktion | Funktion, die an der betreffenden Stelle ausgeführt wird. Es gibt die Built-in-Funktion:<br><br>MSGID<br>Zur Erleichterung der Diagnose sollte jeder Meldungstext mit MSGID beginnen.<br><br>Die Funktion liefert für die Meldungsnummer die Zeichenkette für die Meldungs-Identifikation.  |

Jede MODMSG-Anweisung muss mit einer **ENDMSG**-Anweisung abgeschlossen werden.

## Anweisung END

Mit der END-Anweisung werden die Anweisungen für KDCMMOD beendet. END muss als letzte Anweisung eingegeben werden.

Operation	Operanden
END	

Ohne Operanden.

## Beispiel 1

Der Anwender möchte für die Meldung K006 das Meldungsziel MSGTAC einschalten und ansonsten die unveränderten Meldungen mit deutschen Standardtexten verwenden.

Die folgende Prozedur liefert die Assembler-Quelldatei für das gewünschte eigene Meldungsmodul, wenn die Meldungsdefinitionsdatei SYSMH.UTM.065.MSGFILE heißt und das Programm KDCMMOD in der Bibliothek SYSLNK.UTM.065.UTIL steht:

```
/BEGIN-PROCEDURE LOGGING=N
/ASSIGN-SYSDTA TO=*SYSCMD
/ASSIGN-SYSLST TO=LIST.KDCMMOD
START-EXECUTABLE-PROGRAM FROM-FILE=*LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL
, ELEM=KDCMMOD)
OPTION MSGFILE=SYSMH.UTM.065.MSGFILE
GEN SYS, LANG=D, MODULE=KCSMSGSD
MODMSG 6, MSGTAC=YES
ENDMSG
END
/ASSIGN-SYSDTA TO=*PRIMARY
/ASSIGN-SYSLST TO=*PRIMARY
/END-PROCEDURE
```

Als Ergebnis der Prozedur erzeugt das Tool KDCMMOD die Assembler-Quell-Datei SRC.ASSEMB.KCSMSGSD. Sie ist zu assemblieren und als Ergebnis dieser Assemblierung erhält man ein Objektmodul mit dem Namen KCSMSGSD.

Um dieses Meldungsmodul benutzen zu können, muss der Anwender in die KDCDEF-Generierung die Anweisung MESSAGE MODULE=KCSMSGSD, LIB=... aufnehmen.

**Beispiel 2**

In folgendem Beispiel werden das Meldungsziel, eine Meldungseigenschaft und der Meldungstext geändert.

```
GEN SYS,LANG=D,MODULE=KCSMSGSD
MODMSG 6, MSGTAC=YES
ENDMSG
MODMSG 070,
COMPRESSION = NO,
SYSOUT = YES
MSGID, ' Benutzer abgemeldet: ', &USER, ', CPU-time: ', &CPTM
ENDMSG
END
```

## 4.4 UTM-Protokolldatei SYSLOG

openUTM führt für jede UTM-Anwendung eine eigene Protokolldatei, die SYSLOG-Datei. In diese Datei protokolliert openUTM alle Meldungen, die das Meldungsziel SYSLOG haben. Welche Meldungen das sind, können Sie in gewissen Grenzen selbst bestimmen (siehe dazu auch den [Abschnitt „Ziele der UTM-Meldungen“ auf Seite 353](#)). Die SYSLOG-Datei muss vom Anwender vor dem Start der UTM-Anwendung angelegt werden.

Die System-Protokolldatei SYSLOG können Sie anlegen als:

- einfache Datei mit dem Namen *filebase*.SLOG bzw. Datei mit dem Linknamen SYSLOG.  
*filebase* ist der Basisname der KDCFILE aus der MAX-Anweisung.
- Dateigenerationsgruppe SYSLOG-FGG (**F**ile **G**eneration **G**roup).  
Es genügt, wenn Sie die Dateigenerationsgruppe anlegen; die einzelnen Generationen der FGG legt openUTM für Sie an.  
Wenn Sie eine automatische Größenüberwachung der SYSLOG-Datei generiert haben, muss die SYSLOG als FGG angelegt werden.

Im openUTM-Handbuch „Einsatz von UTM-Anwendungen“ ist ausführlich beschrieben, wie Sie die System-Protokolldatei SYSLOG anlegen.

In der SYSLOG-Datei werden Ereignisse aus dem Lauf der Anwendung in Form von UTM-Meldungen protokolliert, die für die laufende Überwachung oder für spätere Kontrollen nützlich sein können (z.B. Meldungen K033, K070). Die SYSLOG-Datei enthält insbesondere wichtige Informationen für die Diagnose von Problemen der Anwendung.

Bei jedem Folgestart der Anwendung wird die SYSLOG-Datei von openUTM überschrieben. Die Protokollinformation aus dem vorherigen Anwendungslauf geht verloren. Nach dem Ende eines Anwendungslaufs sollten Sie deshalb, falls nötig, den Inhalt auswerten oder die Datei kopieren.

### 4.4.1 SYSLOG-Datei auswerten

openUTM erstellt die SYSLOG-Datei im PAM-Format. Um sie lesbar ausdrucken oder mit einem Programm bearbeiten zu können, muss man die Datei aufbereiten. openUTM bietet dazu die Aufbereitungstools:

- **KDCCSYSL**  
wandelt die PAM-Datei in eine SAM-Datei um. KDCCSYSL wird in der Bibliothek SYSLNK.UTM.065.UTIL zur Verfügung gestellt.

- KDCPSYSL

erzeugt aus der mit KDCCSYSL erstellten SAM-Datei eine weitere SAM-Datei, die auch die Meldungstexte enthält. KDCPSYSL wird in der Bibliothek SYSLNK.UTM.065.UTIL zur Verfügung gestellt.

Um die Auswertung der SYSLOG-Datei zu erleichtern, werden mit openUTM die folgenden Beispielprozeduren ausgeliefert:

- SYSPRC.UTM.065(SYSLOG)

Mit dieser Prozedur können Sie eine einzelne SYSLOG-Datei bzw. eine einzelne Dateigeneration einer SYSLOG-FGG auswerten. Als Eingabedatei wird <filebase>.SYSLOG erwartet.

- SYSPRC.UTM.065(SLOG-FGG)

Mit dieser Prozedur können Sie eine einzelne oder alle Dateigenerationen einer SYSLOG-FGG auswerten. Die Prozedur ruft in einer Schleife die beiden Tools KDCCSYSL und KDCPSYSL für die einzelnen Dateigenerationen auf. Die Auswertung kann auch bei laufender Anwendung erfolgen.

#### 4.4.1.1 Das Tool KDCCSYSL - SYSLOG-Datei in SAM-Format umwandeln

Das Tool KDCCSYSL (Convert SYSLOG) wandelt die von openUTM als PAM-Datei erzeugte System-Protokolldatei in eine SAM-Datei um. Für jede Meldung erzeugt KDCCSYSL einen Satz variabler Länge. Der Aufbau wird durch Datenstrukturen KCMMSGx beschrieben, die für alle Programmiersprachen bereitstehen, die openUTM unterstützt, beispielsweise KCMMSGC für COBOL.

#### Aufruf von KDCCSYSL

```
/SET-FILE-LINK FILE-NAME=file_in, LINK-NAME=SLOGPAM -
    [, SUPPORT=DISC(SHARED-UPDATE=YES)]
/CREATE-FILE FILE-NAME=file_out,
/SET-FILE-LINK FILE-NAME=file_out, LINK-NAME=SLOGSAM, ACCESS-METHOD=SAM
/START-EXECUTABLE-PROGRAM FROM-FILE=*LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL
    , ELEM=KDCCSYSL)
```



Sie können KDCCSYSL auch per SDF-Kommando START-KDCCSYSL starten, siehe auch openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Abschnitt „UTM-Tools über eigene SDF-Kommandos starten“.

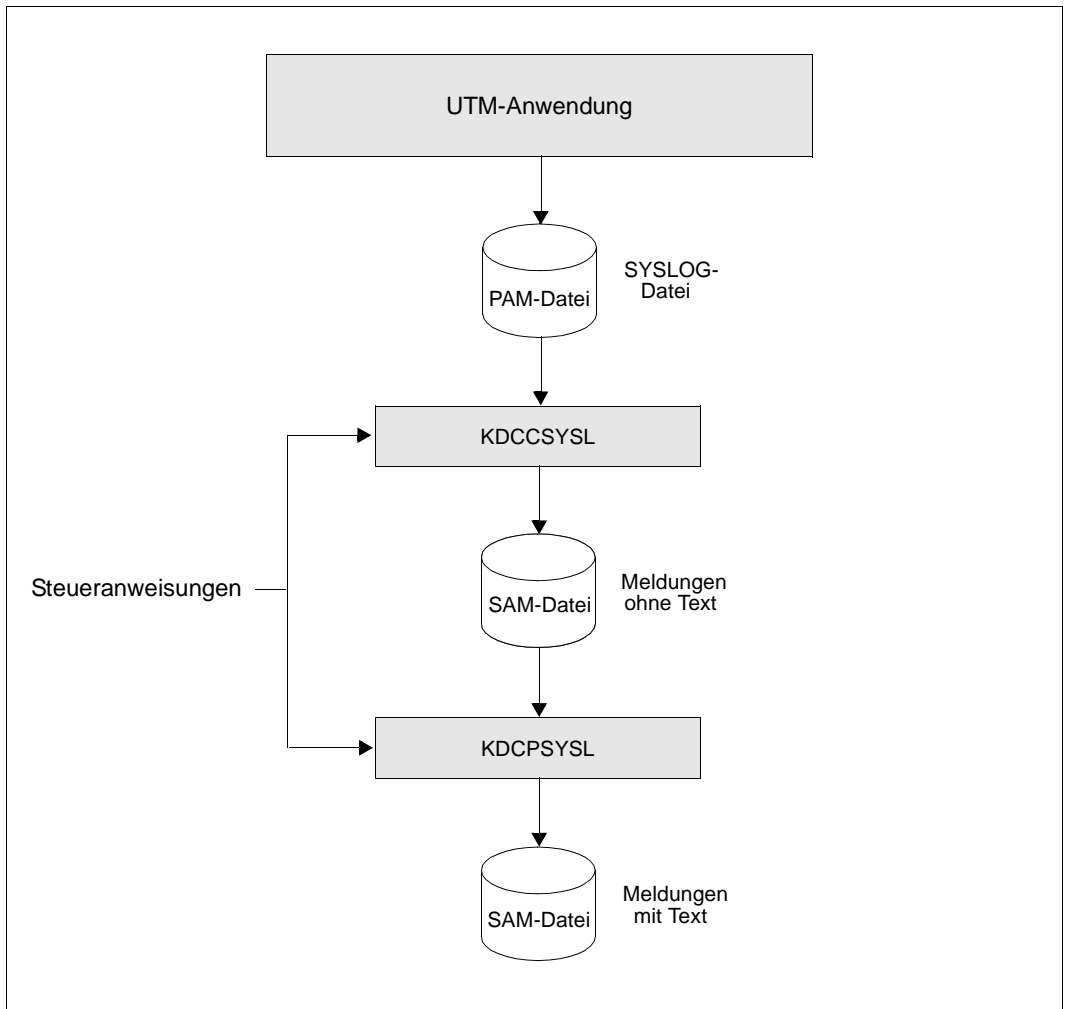
Bedeutung der Parameter:

file_in	Name der SYSLOG-Datei
file_out	Name, den die SAM-Datei bekommen soll



SHARED-UPDATE=YES ist im SET-FILE-LINK-Kommando anzugeben, wenn die SYSLOG-Datei aufbereitet werden soll, während die Anwendung läuft. Beachten Sie dabei aber, dass openUTM SYSLOG-Sätze zwischenspeichert, also nicht sofort in die PAM-Datei schreibt.

Soll die SYSLOG-Datei bei laufender Anwendung ausgewertet werden, dann empfiehlt es sich, vorher das Administrationskommando KDCCSYSL einzugeben. Damit erreicht man, dass openUTM SYSLOG-Meldungen, die noch im virtuellen Speicher gepuffert sind, auf die SYSLOG-Datei schreibt.



Aufbereiten der SYSLOG-Datei

#### 4.4.1.2 Das Tool KDCPSYSL - Meldungstexte einfügen

Das Aufbereitungstool KDCPSYSL liest die mit KDCPSYSL erzeugte SAM-Datei und erzeugt daraus eine SAM-Datei, die zu jeder Meldung die Meldungsnummer und den Meldungstext mit den aktuellen Inserts enthält.

Datum und Uhrzeit werden am Anfang der Meldung eingefügt. Die Inserts, die nicht im Meldungstext enthalten sind, werden am Ende hinzugefügt. Inserts, die hexadezimale Information liefern, werden abdruckbar aufbereitet (z.B. BCAM-Infoword in der K065-Meldung).

Sie können die SYSLOG-Datei auch mit Meldungstexten aus einem eigenen Meldungsmodul aufbereiten lassen. Dazu geben Sie nach der Meldung

K608 Bitte Namen des Benutzermeldungsmoduls, Leerzeichen oder /EOF eingeben.

den Namen des eigenen Meldungsmoduls an. Wenn zuvor mit /SET-TASKLIB LIBRARY= eine benutzereigene Modulbibliothek zugewiesen wurde, sucht KDCPSYSL das angegebene Modul in dieser Bibliothek.

Wird auf die Meldung ein Leerzeichen (blank) oder „/EOF“ eingegeben oder wird das Modul in der TASKLIB nicht gefunden, so bereitet KDCPSYSL die Meldungen mit dem Standard-Meldungsmodul von openUTM auf.

Die mit KDCPSYSL erzeugte Datei ist eine SAM-Datei, die pro Meldung eine oder mehrere Druckzeilen enthält. Diese Datei können Sie mit dem PRINT-FILE-Kommando ausdrucken:

```
/PRINT-FILE FILE-NAME=file_out, LAYOUT-CONTROL=PARAMETERS(
CONTROL-CHARACTERS=EBCDIC)
```

#### KDCPSYSL aufrufen

```
/SET-FILE-LINK FILE-NAME=file_in, LINK-NAME=SLOGSAM
/CREATE-FILE FILE-NAME=file_out,
/SET-FILE-LINK FILE-NAME=file_out, LINK-NAME=SLOGEDIT, ACCESS-METHOD=SAM
[/SET-TASKLIB LIBRARY=omlname]
/START-EXECUTABLE-PROGRAM FROM-FILE=*LIB-ELEM(LIB=SYSLNK.UTM.065.UTIL
, ELEM=KDCPSYSL)
```



Sie können KDCPSYSL auch per SDF-Kommando START-KDCPSYSL starten, siehe auch openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Abschnitt „UTM-Tools über eigene SDF-Kommandos starten“.

KDCPSYSL antwortet mit der Meldung:

K608 Bitte Namen des Benutzermeldungsmoduls, Leerzeichen oder /EOF eingeben.



## 4.5 Aufbau der UTM-Systemmeldungen

Aufbau der Meldungen, wie sie an SYSLOG oder MSGTAC geschickt werden:

Meldungskopf	Daten der Meldung, abhängig von der Meldungsnummer
1	28 29

Aufbau der Meldungen:

Byte	Format	Bedeutung
1	X'40'	Leerzeichen
2-5	C'Kxxx'	Meldungsnummer
6	X'40'	Leerzeichen
7-17	C'mm/tt/jjddd'	Datum mit mm=Monat, tt=Tag im Monat, jj=Jahr, ddd=Tag im Jahr
18	X'40'	Leerzeichen
19-24	C'hhmmss'	Uhrzeit mit hh=Stunde, mm=Minute, ss=Sekunde
25-28	C'yyyy'	4-stellige Jahresangabe
ab 29		Daten der Meldung

Dem Aufbau des Meldungskopfes entspricht im COBOL-COPY-Element KCMSGC die folgende Datenstruktur:

```

05      MSGKOPF.
*
*          07      FILLER-1      PIC X.          MESSAGE HEADER      *
*          07      MSGNR        PIC X(4).        FILLER                *
*          07      FILLER-2      PIC X.          MESSAGE NUMBER      *
*          07      FILLER-3      PIC X.          FILLER                *
*          07      MSGDATE       PIC X(11).       DATE OF ORIGIN        *
*          07      FILLER-4      PIC X.          FILLER                *
*          07      MSGTIME       PIC X(6).        TIME OF ORIGIN        *
*          07      MSGYEAR       PIC X(4).        YEAR OF ORIGIN        *
*****
*          INSERTS OF MESSAGES
*****
05      KXXX      PIC X(152).
.
.
    
```

---

## 5 UTM-Meldungen

Die folgenden Aufstellungen enthalten alle Meldungen, die von openUTM ausgegeben werden können. Die Meldungen der UTM-Tools sind in diese Liste integriert. Die Beschreibungen wurden um „Zusatzinformationen“ erweitert, die Ihnen die Reaktion auf die betreffende Meldung erklären sollen.

Vor dem Namen eines Inserts steht ein „&“. UTM-Meldungen können mehr Inserts beinhalten als in den Standard-Meldungstexten enthalten sind. Die Bedeutung der Inserts und ihre Zuordnung zu den einzelnen Meldungen finden Sie in den Tabellen „[Inserts in Meldungen](#)“ auf Seite 341ff und „[Ziele der UTM-Meldungen](#)“ auf Seite 353ff.

Wird in einer UTM-Anwendung mit einem Anmelde-Vorgang gearbeitet, dann werden die Meldungen K001, K002, K004, K005, K006, K007, K008, K030, K031, K097, K109, K110, K120, K123, K125 und K145 nicht erzeugt.

### 5.1 Meldungen des Transaktionsmonitors

**K001** Verbunden mit Anwendung &APPL - Bitte Eingabe

**K002** Verbunden mit Anwendung &APPL - Bitte KDCSIGN

**K003** Das Kommando &CMD ist in dieser Situation nicht erlaubt

**K004** Anmeldung fehlgeschlagen - Bitte KDCSIGN

Diese Meldung besitzt zusätzlich folgendes Insert:


&REA7            Ursache warum die Anmeldung fehlgeschlagen ist. Die Werte in &REA7 sind bei der Meldung K147 auf [Seite 236](#) beschrieben.

**K005** Die Benutzerkennung &USER ist gesperrt - Bitte KDCSIGN

**K006** Anmeldung fehlgeschlagen - Bitte KDCSIGN



Bei der Ausgabe an ein Terminal enthält der Text dieser Meldung die Meldungsnummer K004 anstatt K006. Die Meldungsnummer K006 wird nur bei Ausgabe dieser Meldung an andere Meldungsziele sichtbar.

- K007** Der Benutzer &USER ist bereits angeschlossen - Bitte KDCSIGN  
Diese Meldung besitzt zusätzlich folgendes Insert:  
&REA7 Ursache warum die Anmeldung fehlgeschlagen ist. Die Werte in &REA7 sind bei der Meldung K147 auf [Seite 236](#) beschrieben.
- K008** KDCSIGN akzeptiert.  
Letzte erfolgreiche Anmeldung am &SGNT - Bitte Eingabe
-  Beim ersten Anmelden eines Benutzers nach einem KDCDEF-Lauf kann das Insert &SGNT Fragezeichen enthalten, da noch kein Zeitpunkt eines letzten Anmeldens verfügbar ist.
- K009** Der Transaktionscode &TAC ist ungueltig (&RCDC) - Bitte Eingabe  
Das Insert &RCDC enthält den inkompatiblen Returncode KCRCDC, siehe [Abschnitt „Interner Returncode KCRCDC“ auf Seite 371](#).
- K010** Der Transaktionscode &TAC ist gesperrt - Bitte Eingabe
- K011** Die Transaktion &ATAC1 wurde akzeptiert - Bitte Eingabe
- K012** &NUMMSGs asynchrone Nachricht(en) vorhanden
- K013** Fehlerhafte Eingabe im Kommando &CMD - Bitte Eingabe
- K014** Es ist kein Basisformat vorhanden - Bitte Eingabe
- K015** Formatierungsfehler &RCDC &RCF2A - Bitte Eingabe
- K016** Die Anwendung wird beendet - Bitte mit KDCOFF beenden
- K017** Vorgang &TCVG durch UTM beendet (&RCCC/&RCDC &RCF2A) - Bitte Eingabe  
Mit dieser Meldung wird ein Benutzer am Terminal über das abnormale Ende des von ihm gestarteten Vorgangs informiert.  
Die vollständige Meldung K017 in der SYSLOG-Datei besitzt folgende Inserts:
- &PTRM Name des PTERMs, von dem aus der abgebrochene Vorgang gestartet wurde.
- &PRNM Name des Prozessors, an dem das Terminal angeschlossen ist.
- &BCAP Name der BCAM-Anwendung, an die sich der Benutzer angemeldet hat.
- &LTRM Name des LTERMs, von dem aus der abgebrochene Vorgang gestartet wurde.
- &USER Name des Benutzers, der den Vorgang gestartet hat.
- &TCVG Name des Vorgangs-TACs des abgebrochenen Vorgangs.
- &RCCC Wert des kompatiblen Returncodes KCRCCC.

&RCDC Wert des inkompatiblen Returncodes KCRDC.  
&RCF2A Sekundärer FHS- oder VTSU-Returncode (wenn vorhanden).  
&TAC Nur bei Returncodes RCDC=KMxx:  
Name des TACs, der zu dem Vorgangsabbruch geführt hat.  
Sonst enthält dieses Insert Leerzeichen.

**K018** KDCOFF von Anwendung &APPL akzeptiert - Bitte KDCSIGN

**K019** KDCOFF von Anwendung &APPL akzeptiert

**K020** Es ist keine Nachricht vorhanden

**K021** Eine Eingabe ist nicht in der vorgegebenen Zeit erfolgt

**K022** Die folgende Nachricht von &BCAP wurde eventuell bereits gesendet

**K023** &OMSG2

Die Meldung wird durch das Administrationskommando KDCSEND oder KDCSHUT WARN bzw. die entsprechenden Aufrufe an der Administrator-Schnittstelle erzeugt.

**K024** Die Eingabenachricht ging verloren - Bitte wiederholen

**K025** Die Ausgabenachricht der Anwendung &BCAP an LTERM &LTRM wurde verkürzt  
Eine Nachricht an einen LTERM-Partner konnte nicht vollständig gesendet werden, die Nachricht wurde verkürzt ausgegeben.

Mögliche Maßnahmen zur Behebung des Fehlers:

- Vergrößern der Längenangabe im Operanden TRMSGLTH in der KDCDEF-Generierung, falls diese Länge kleiner als die max. Paketlänge bei BCAM ist.
- Vergrößern der maximalen Paketlänge bei BCAM, falls diese kleiner als TRMSGLTH generiert ist.
- Einfügen von Steuerzeichen „new line“ (X'15') bei Ausgabe im Zeilenmodus (um dem VTSU das Fragmentieren zu ermöglichen).

**K026** Mitteilung an &LTRM akzeptiert - Bitte Eingabe

**K027** LTERM-Partner &LTRM gesperrt - Administrator verstaendigen oder KDCOFF eingeben

**K029** Bitte Karte einlegen

**K030** Es ist ein Kartenleser erforderlich - Bitte KDCSIGN

**K031** Falsche Karte - Bitte KDCSIGN

**K032** UTM-D Verbindungsmeldung : &CON/&PRNM/&BCAP/&LPAP/&USER; Grund1: &RCF1B; Grund2: &RCF2B

Bei der folgenden Beschreibung der Returncodes &RCF1B und &RCF2B ist PLU (Primary Logical Unit) die Anwendung, in der SESCHA PLU=NO generiert wurde, und SLU (Secondary Logical Unit) die Anwendung, in der PLU=YES angegeben wurde.

Das Insert &USER enthält den UTM-Sessionnamen der Verbindung (LSES-Name).

&RCF1B	Bedeutung	
C01	BIND_REJECTED	BIND-Request oder BIND-Response wurde abgelehnt. Eine Session kann nicht zustandekommen.
C02	BIND_CHANGED	BIND-Parameter wurden modifiziert. BIND-Request darf von SLU modifiziert werden, PLU kann die BIND-Response nicht verändern, nur ablehnen. Kann die BIND-Response von PLU nicht akzeptiert werden, kommt keine Session zustande.
C03	CONNECTION / SESSION FAILED	Eine Verbindung bzw. Session kann nicht aufgebaut werden.
C04	SESSION_FAILED	Sessionwarmstart konnte nicht durchgeführt werden.
C05	STSN_CHANGED	SLU kann den vorgeschlagenen Aufsetzpunkt nicht akzeptieren.

&RCF2B	Bedeutung und evtl. Maßnahme	
CR00	Session und Connection gehören nicht zum selben LPAP. Maßnahme: Die KDCDEF-Generierung überprüfen.	
CR01	Beim Sessionwarmstart konnten die Sessions keinen gültigen Aufsetzpunkt vereinbaren. Die Anwendungen wurden möglicherweise mit verschiedenen KDCDEF-Ständen gestartet. Maßnahme: Überprüfen, ob eine der beiden Anwendungen neu generiert wurde.	
CR02	SLU hat einen neuen Aufsetzpunkt vorgeschlagen. Kommt die Session nicht zustande, wurden die Anwendungen wahrscheinlich mit verschiedenen KDCDEF-Ständen gestartet. Maßnahme: Überprüfen, ob eine der beiden Anwendungen neu generiert wurde.	
CR03	Quiet-Kommando wurde eingegeben.	
CR04	Keine geeignete Session verfügbar; Gründe: – Es wurden mehr Verbindungen als Sessions generiert. – KDCLSES-Kommando mit ACT=QUIET gegeben. – Eine Session wurde noch nicht abgebaut.	



<b>&amp;RCF2B</b>	<b>Bedeutung und evtl. Maßnahme</b>
CR05	Verbindungsaufbauanforderung von TRANSIT-CD <sup>1</sup> oder Partner-Anwendung erzeugt für diese Verbindung NEA-Protokolle.
CR06	Verbindungsaufbau in Bearbeitung.
CR07	Inkonsistenz bzgl. PLU-Angabe in der Verbindungsnachricht vom Partner und in der UTM-Generierung. Es wurden evtl. beide Partner als PLU generiert. Maßnahme: Generierungen der beiden Partner überprüfen.
CR08	Inkonsistenz bzgl. PLU-Angabe in der Verbindungsnachricht vom Partner und in der UTM-Generierung. Es wurden evtl. beide Partner als SLU generiert. Maßnahme: Generierungen der beiden Partner überprüfen.
CR09 CR10 CR11 CR12	Fehlerhafter Aufbau einer Verbindungsnachricht
CR13	Betriebsmittelengpass: kein Slot für dynamische Tabellen vorhanden.
CR14	In der Verbindungsnachricht ist der Name der PLU-Session falsch. Maßnahme: Generierungen der beiden Partner überprüfen
CR15 CR16 CR19 CR21	Session ist noch aufgebaut.
CR17 CR18 CR20	Keine geeignete Session verfügbar; Gründe: – Es wurden mehr Verbindungen als Sessions generiert. – KDCLSES-Kommando mit ACT=QUIET gegeben. – Eine Session wurde noch nicht abgebaut.
SC00	Kein Grund angegeben.
SC01 <sup>1</sup>	Der Partner meldet sich mit einem falschen FM-Profil. UTM unterstützt nur FM-Profile 18.
SC02 <sup>1</sup>	Der Partner meldet sich mit einem falschen TS-Profil. UTM unterstützt TS-Profile 4.
SC03 <sup>1</sup>	UTM kann eine Nachricht in kleineren Teilen (Request Units) bearbeiten. Deshalb muss PLU sog. „Multiple-RU-Chains“ unterstützen.
SC04 <sup>1</sup>	Eine evtl. angeforderte Antwort muss gekommen sein, bevor die nächste Anforderung geschickt werden darf. PLU muss deshalb im „Immediate-Request-Mode“ arbeiten.
SC05 <sup>1</sup>	Abhängig von dem Nachrichtentyp fordert UTM immer eine Quittung bzw. nur eine negative Quittung. PLU muss deshalb sowohl „definite“ als auch „exception response“ akzeptieren.
SC07 <sup>1</sup>	Bei der Nachrichtenübertragung wird Datenkomprimierung nicht unterstützt.
SC08 <sup>1</sup>	PLU muss immer die Möglichkeit haben, eine Bracket zu schließen.

&RCF2B	Bedeutung und evtl. Maßnahme
SC09 <sup>1</sup>	Entspricht Reason SC03, aber für SLU.
SC10 <sup>1</sup>	Entspricht Reason SC04, aber für SLU.
SC11 <sup>1</sup>	Entspricht Reason SC05, aber für SLU.
SC12 <sup>1</sup>	UTM unterstützt kein „2-Phase-Commit“.
SC13 <sup>1</sup>	Entspricht Reason SC07, aber für SLU.
SC14 <sup>1</sup>	Entspricht Reason SC08, aber für SLU.
SC15 <sup>1</sup>	FM-Header müssen erlaubt sein.
SC16	Obwohl ein Partner einen Session-Kaltstart für ausreichend hält, besteht die andere Session auf einem Session-Warmstart.
SC17	Der Partner hat die Session als nicht abgeschlossen (in Bracket) bezeichnet, obwohl er keinen Warmstart wünscht. Die Session wird als abgeschlossen angenommen (BETB).
SC18 <sup>1</sup>	Ein Vorgang wird erst abgeschlossen, wenn er von beiden Partnern als abgeschlossen bezeichnet ist. D.h. UTM benutzt die sog. „Bracket Termination Rule 1“.
SC19	Die USER-DATA im BIND sind nicht korrekt. Vermutlich sendet der Partner einen fehlerhaften BIND. Zur Diagnose wird der UTM-BCAM-Trace benötigt.
SC23 <sup>1</sup>	In UTM-D wird mit „Half-duplex flip-flop“ Sende- bzw. Empfangsmodus gearbeitet. D.h. zu jedem Zeitpunkt hat nur ein Partner Sendeerlaubnis.
SC24 <sup>1</sup>	Beide Partner müssen für einen evtl. Sessionwiederanlauf verantwortlich sein.
SC25	Inkonsistenz bezüglich Contention-Winner im BIND und der KDCDEF-Generierung. Maßnahme: Beide Generierungen müssen bezüglich der Angabe Contention-Winner überprüft werden. (KDCDEF-Steueranweisung SESCHA CONTWIN=Y/N)
SC26	Beim Session-Warmstart will PLU die Sendeberechtigung bekommen. Kommt die Session nicht zustande, wurden die Anwendungen wahrscheinlich mit verschiedenen KDCDEF-Ständen gestartet. Maßnahme: Überprüfen, ob eine der beiden Anwendungen neu generiert wurde.
SC27	Beim Session-Warmstart will SLU die Sendeberechtigung bekommen. Kommt die Session nicht zustande, wurden die Anwendungen wahrscheinlich mit verschiedenen KDCDEF-Ständen gestartet. Maßnahme: Überprüfen, ob eine der beide Anwendungen neu generiert wurde.
SC28 <sup>1</sup>	UTM unterstützt „Two-Stage-Pacing“.

<b>&amp;RCF2B</b>	<b>Bedeutung und evtl. Maßnahme</b>
SC29	Inkonsistenz der Pacing-Counter. Maßnahme: Beide Generierungen müssen bezüglich der Pacing-Counter (=Window-Size) überprüft werden. (KDCDEF-Steueranweisung SESCHA PACCNT=)
SC30 <sup>1</sup>	In der BIND-Response schlägt SLU als „maximale Teilnachrichtenlänge von PLU“ einen nicht erlaubten Wert vor.
SC31 <sup>1</sup>	In der BIND-Response schlägt SLU als „maximale Teilnachrichtenlänge von SLU“ einen nicht erlaubten Wert vor.
SC32	In dem BIND-Request schlägt PLU als „maximale Teilnachrichtenlänge von SLU“ einen nicht erlaubten Wert vor.
SC33	In dem BIND-Request schlägt PLU als „maximale Teilnachrichtenlänge von PLU“ einen Wert vor, der von SLU modifiziert wird.
SC34 <sup>1</sup>	Presentation-Service (PS) Usage field format muss als „basic format“ definiert sein.
SC35 <sup>1</sup>	Als Session-Type ist nur LU-Session Type 6 erlaubt.
SC36 <sup>1</sup>	Nachrichtenverschlüsselung wird nicht unterstützt.
SC37	In der BIND-Response ist der Name der PLU-Session falsch. Maßnahme: Beide Generierungen müssen bezüglich der Verknüpfung LSES - RSES überprüft werden. (KDCDEF-Steueranweisung LSES RSES = Name)
SC38	In dem BIND-Request ist der Name der PLU-Session falsch. Maßnahme: Siehe Reason SC37.
SC39	In der BIND-Response ist der Name der SLU-Session falsch. Maßnahme: Siehe Reason SC37.
SC40	In dem BIND-Request ist der Name der SLU-Session falsch. Maßnahme: Siehe Reason SC37.
SC41 <sup>1</sup>	User Request Correlation(URC) wird von UTM-D nicht unterstützt.
SY01	Session termination pending. Der Abbau der Session ist noch nicht abgeschlossen.
SY02	PET no send request. Die Session ist im Zustand PTC und es existiert keine Nachricht für diesen Partner.
SY03	PEND RS pending. Die PEND RS-Behandlung ist noch nicht abgeschlossen.
SY04	job-submitting conversation active. Der Auftraggeber-Vorgang ist aktiv.
SY05	job submitter not available. Der Auftraggeber ist nicht angemeldet oder der Auftraggeber-Vorgang ist gequeued.

<b>&amp;RCF2B</b>	<b>Bedeutung und evtl. Maßnahme</b>
SY06	RESTART_VIA_SEND_RQ. Die Session befindet sich im Zustand IN_BRACKET und es liegt keine Nachricht an diesen Partner vor.
SY07	Session active Die Session ist bereits aktiv.

<sup>1</sup> Nur bei heterogenen Kopplungen

**K033** USER / LSES aktiv : &PTRM/&PRNM/&BCAP/&LTRM/&USER &REST, &GLOBALSG

Die Meldung K033 wird ausgegeben:

- nach erfolgreichem Verbindungsaufbau, wenn ohne User gearbeitet wird,
- nach erfolgreichem Anmelden, wenn mit Usern gearbeitet wird,
- nach erfolgreichem Binden einer Session bei einer LU6.1-Verbindung.

<b>&amp;REST</b>	<b>Bedeutung</b>
Y	Vorgangswiederanlauf
N	kein Vorgangswiederanlauf
U	undefiniert, nur bei UTM-D

Meldet sich ein Benutzer über eine OSI TP Verbindung an, haben die ersten Inserts folgende Inhalte:

<b>Insert</b>	<b>Inhalt</b>
&PTRM	OSI-CON Name
&PRNM	8 Leerzeichen
&BCAP	ACCESS-POINT Name
&LTRM	OSI-LPAP Name

Das Insert &GLOBALSG 'Cluster Global Sign' ist nur in UTM-Cluster-Anwendungen versorgt und kann folgende Werte annehmen:

<b>&amp;GLOBALSG</b>	<b>Bedeutung</b>
Y	Cluster-globale Anmeldung
N	Knoten-lokale Anmeldung
A	Der Benutzer war bereits Cluster-global an diesem Knoten angemeldet.
' '	Anmeldung für einen LU6.1-Session-User oder einen Verbindungs-User.

**K034** Die Transaktion wurde zurueckgesetzt

**K035** Vorgangswiederanlauf ist in Bearbeitung

**K036** Verbindungsaufbau : &PTRM/&PRNM/&BCAP/&LTRM &RSLT , &REA1

<b>&amp;RSLT</b>	<b>Bedeutung</b>
Y	Verbindung aufgebaut
N	Verbindung wurde nicht aufgebaut; die Ursache wird in &REA1 angegeben.

<b>&amp;REA1</b>	<b>Bedeutung</b>
X'00'	Verbindung aufgebaut
X'01'	PTERM-/CON-Name unbekannt
X'02'	Prozessorname unbekannt
X'03'	PTYPE nicht akzeptiert
X'04'	kein LTERM zugeordnet
X'05'	falsches Verbindungspasswort angegeben
X'06'	STATUS=OFF für dieses PTERM/CON
X'07'	STATUS=OFF für dieses LTERM/LPAP
X'08'	Betriebsmittelengpass
X'0A'	SHUTDOWN der Anwendung
X'0B'	Partner bereits angeschlossen
X'0C'	Verbindungsabbau in Bearbeitung
X'0D'	negativer BCAM Returncode, siehe dazugehörige K065-Meldung
X'0E'	der Partner ist an einer anderen BCAM-Anwendung generiert
X'0F'	UTM-D Verbindungswunsch abgelehnt; z.B. wegen UTM-Generierungsfehler oder da QUIET-Kommando gegeben
X'10'	PTERM-Name unbekannt und kein LTERM-Pool für diesen Prozessornamen vorhanden
X'11'	Kein LTERM-Pool für diesen PTYPE vorhanden und der PTERM-Name ist unbekannt
X'12'	Kein weiterer freier Eintrag im Terminalpool vorhanden
X'13'	Partner ist bereits an den LTERM-Pool angeschlossen
X'14'	Verbindung abgelehnt wegen nicht hinreichenden Eigenschaften der Transportverbindung (GROS)
X'15'	MUX-Verbindung abgelehnt, weitergehende Beschreibung im BIND Response
X'16'	Benutzer ist bereits angemeldet
X'17'	Der Verbindungswunsch wurde wegen contention abgelehnt. Der Partner hat selbst einen Verbindungsaufbau initiiert
X'18'	BCAM-Letter length zu klein.

&REA1	Bedeutung
X'1B'	Beim Start der Anwendung konnte die IP-Adresse des Partner-Rechners nicht ermittelt werden.
X'1F'	LPAP ist auf quiet gesetzt
X'20'	Es ist keine geeignete Session verfügbar
X'21'	Die Session ist bereits aktiv
X'22'	Session Termination ist in Bearbeitung
X'23'	Lokale Seite ist im PTC und hat keine Ausgabe-Nachricht für diese Session; zum Sessionausbau muss auf den Wiederanlauf des lokalen Vorgangs gewartet werden.
X'24'	PEND RS ist in Bearbeitung
X'25'	Der AG-Vorgang ist aktiv
X'26'	Der AG-User ist nicht verfügbar
X'27'	Zum Sessionaufbau muss auf den Wiederanlauf des lokalen Vorgangs gewartet werden
X'29'	Der Terminalpool ist an einer anderen BCAM-Anwendung generiert
X'2A'	Verbindungswunsch von VTSU abgelehnt, z.B. wegen nicht unterstütztem PTYPE oder falschem Connection-Letter
X'2B'	Der Verbindungswunsch für einen mit PTYPE = *ANY generierten PTERM-Partner wird abgelehnt, weil der Typ des rufenden Terminals bzw. Druckers nicht unterstützt wird.
X'2C'	Verbindungswunsch eines Druckers für einen mit PTYPE = *ANY und USAGE=D generierten LTERM-Partner wird abgelehnt.
X'2D'	Es wird eine ungültige MUX-Protokollversion beim Aufbau der MUX-Verbindung vorgeschlagen
X'2E'	Die Verbindung ist noch nicht vollständig abgebaut bzw. die MUX-Session befindet sich noch im Zustand „RELEASE-PENDING“
X'44'	Verbindungsaufbau von RSO abgelehnt, z.B. weil der Drucker bei RSO nicht bekannt ist.
X'45'	Während des Wartens auf eine Bestätigung des Verbindungsaufbaus traf ein Verbindungsabbauwunsch ein.
X'46'	PTERM/CON wurde per Administration gelöscht.
X'48'	Es ist schon ein PTERM mit dem Namen des Multi-Pool-LTERM generiert.
X'55'	Abgelehnter Socket-Verbindungsaufbau.
X'58'	Verbindungsaufbaubestätigung abgelehnt, da die Verbindung zwischenzeitlich wieder abgebaut wurde.
X'5B'	Fehler beim Aufruf der ICO-Funktion zur Umsetzung der Rechnernamen (siehe K037)

Mit dieser Meldung, die standardmäßig auf SYSLOG ausgegeben wird, können Probleme beim Verbindungsaufbau erkannt werden.

Bei Ausgabe der Meldung wird das Insert &REA1 abdruckbar aufbereitet, während es an der Programmschnittstelle zu MSGTAC / SYSLOG als Hexadezimalwert übergeben wird.

**K037** Host name mapping: &HST1 to &HST2

Mit dieser Meldung informiert UTM über die Zuordnung von realen zu lokalen Rechnernamen bzw. umgekehrt. Diese Meldung wird bei jeder Namensumsetzung ausgegeben.

Die vollständige Meldung K037 in der SYSLOG-Datei besitzt folgende Inserts:

**&HST1**

langer oder kurzer Rechnername, für den der zugeordnete Name ermittelt werden soll

**&HST2**

kurzer oder langer Rechnername, der dem Namen in &HST1 zugeordnet ist.

Werden bei &HST2 statt eines Rechnernamens Gleichheitszeichen zurückgegeben, dann bedeutet dies, dass der Aufruf zur Namensumsetzung nicht erfolgreich war.

**&RSLT**

Anzeige, ob die Namensumsetzung erfolgreich war oder nicht:

Y: die Namensumsetzung war erfolgreich

N: die Namensumsetzung war nicht erfolgreich

**&RCHX**

falls &RSLT den Wert 'N' hat, dann wird in diesem Insert ein 4 Bytes langer Returncode der Systemfunktion ausgegeben.

Der Wert X'00018108' bei &RCHX bedeutet, dass eine BCAM-Version kleiner als V24 im Einsatz ist. Eine Namensumsetzung wird jedoch erst ab BCAM V24 unterstützt.

**K038** Syntaxfehler &SYN

**K039** Die Startparameter &STRTPAR1 und &STRTPAR2 schliessen sich gegenseitig aus!

**K040** Die Warnungsstufe &WLEV fuer &PGPOOL wurde unterschritten

**K041** Die Warnungsstufe &WLEV fuer &PGPOOL wurde ueberschritten

**K042** Fehler beim Lesen der Startparameter von SYSDTA

Diese Meldung wird auch ausgegeben, wenn kein zweites END-Kommando angegeben wurde.

**K043** DMS Fehler &DMSE fuer Datei &FNAM aufgetreten

Im Insert &DMSE wird der DMS-Fehlercode ausgegeben.

Für UTM-Cluster-Dateien kann das Insert &DMSE neben den vom Betriebssystem gemeldeten Fehlercodes folgende weitere Fehlercodes enthalten:

ILCK Es ist ein Fehler beim Initialisieren eines Datei-Locks aufgetreten.

LOCK Es ist ein Fehler bei Anfordern eines Datei-Locks aufgetreten.

ULCK Es ist ein Fehler bei der Freigabe eines Datei-Locks aufgetreten.

DLCK Es ist ein Fehler beim Zerstören eines Datei-Locks aufgetreten.

**K044** Bitte Startparameter eingeben**K045** --- Kein Meldungstext ---

Als einziges Meldungsziel dieser Meldung ist MSGTAC zugelassen; daher ist zu dieser Meldung kein Meldungstext definiert.

Die Ausgabe der Meldung erfolgt, wenn die Ausgabe einer Nachricht im Quittungsmodus mit einer positiven Abdruckquittung beendet wurde.

Standardmäßig ist kein Nachrichtenziel für diese Meldung definiert.



**K046** Fehler bei Druckausgabe: &PALTRM / &CID / &DPID / &ERPRT

Die Meldung wird nur für Drucker, die einer Druckersteuerstation zugeordnet sind, ausgegeben. Sie ist vor allem für eine programmierte Fehlerbehandlung mit Hilfe des MSGTAC-Teilprogramms bestimmt.

Weitere Inserts dieser Meldung sind: &PTRM, &PRNM, &BCAP, &LTRM, &IMSG2, &GNDATE, &GNTIME

Die Meldung wird in folgenden Fällen ausgegeben:

- Beenden einer Druckausgabe mit negativer Abdruckquittung,
- Wiederholung einer Druckausgabe (zusätzlich zur K022-Meldung),
- Verbindungsaufbau zu einem Drucker wird von BCAM abgelehnt (negative COMPLETION announcement).

Standardmäßig ist für diese Meldung kein Meldungsziel definiert.

Das Insert &ERPRT gibt Auskunft über den vorliegenden Fehlerfall, im Insert &IMSG2 werden abhängig davon weitere Informationen zur Verfügung gestellt.

<b>&amp;ERPRT</b>	<b>Bedeutung und Inhalt von &amp;IMSG2</b>
A	Negative Abdruckquittung. Wenn der Drucker eine logische Abdruckquittung liefert (Rückmeldung), so enthält &IMSG2 die ersten 32 Byte dieser Abdruckquittung (ggf. mit Leerzeichen ergänzt). Der genaue Aufbau dieser Abdruckquittung ist in den entsprechenden Geräte-Benutzerhandbüchern nachzulesen.
R	Wiederholung einer Druckausgabe. &IMSG2 ist ohne Bedeutung.
C	Verbindungsaufbau wird von BCAM abgelehnt (negative COMPL announcement). &IMSG2 enthält das BCAM-INFOWORD und den BCAM-USERREASON.

**K047** Ungültiger SVC 134 - Aufruf (SVC fuer UTM): Opcode = &OPCD1, Returncode = &RTCD

UTM gibt die Meldung K047 aus, wenn der UTM-SVC (SVC 134) nicht erfolgreich bearbeitet werden kann. Die Meldung enthält die Inserts &OPCD1 und &RTCD.

Insert &OPCD1 enthält den KDCS-Operationscode, 4 Byte lang und abdruckbar. Er ist von den KDCS-Aufrufen bekannt, zusätzlich gibt es UTM-interne Operationscodes, z.B. 'STRT' (siehe Beschreibung der UTM Diagarea).

Insert &RTCD ist der aufbereitete Returncode (8 Zeichen, hexadezimale Darstellung). Der Returncode wird von UTM oder vom DSSM gesetzt.

### Returncodes des UTM-SVC (SVC 134)

Returncodes mit einem Wert 00000nn stammen von UTM und haben folgende Bedeutung:

nn	Bedeutung
04	Die Adresse von ROOTDATA ist ungültig (UTM-Fehler <sup>1</sup> ).
08	Die Version von ROOTDATA ist ungültig. Mögliche Ursache: Aus SYSLNK.UTM.065 wurden Module in die Anwendung eingebunden, die nicht zur openUTM-Version in der UTM-Makrobibliothek passen.
0C	Die Adresse der Parameterliste ist ungültig (UTM-Fehler).
10	Die Adresse des KDCS-Parameterbereichs ist ungültig. Bei Verwendung der von openUTM zur Verfügung gestellten Include-Dateien für die KDCS-Parameterbereiche handelt es sich um einen UTM-Fehler. Werden im Teilprogramm selbst definierte Bereiche für den KDCS-Parameterbereich verwendet, handelt es sich um einen Anwenderfehler. Maßnahme: Ausrichtung und Adresse des Parameterbereiches überprüfen.
14	Die Adresse des KTA in ROOTDATA ist ungültig. Mögliche Ursachen: – UTM-Fehler – KDCS-Aufruf im START-EXIT-Teilprogramm – KDCS-Aufruf im SHUT-EXIT-Teilprogramm – KDCS-Aufruf nach UTM Term Application
18	Der Inhalt des KTA-Headers ist ungültig. Mögliche Ursachen: – UTM-Fehler – KDCS-Aufruf im START-EXIT-Teilprogramm – KDCS-Aufruf im SHUT-EXIT-Teilprogramm
1C	KTA und KAA stimmen bezüglich des Applikationsnamens nicht überein oder der KAA-Header ist ungültig (UTM-Fehler).
20	ROOTDATA und KTA stimmen bezüglich des Applikationsnamens nicht überein (UTM-Fehler).
24	Die Adresse der Parameterliste ist nicht auf Wortgrenze ausgerichtet (UTM-Fehler).

nn	Bedeutung
28	Der KDCS-Parameterbereich ist nicht ausgerichtet. Bei Verwendung der von openUTM zur Verfügung gestellten Include-Dateien für die KDCS-Parameterbereiche handelt es sich um einen UTM-Fehler. Werden im Teilprogramm selbst definierte Bereiche für den KDCS-Parameterbereich verwendet, handelt es sich um einen Anwenderfehler (Ausrichtung auf Halbwort wird geprüft). Maßnahme: Ausrichtung und Adresse des Parameterbereiches überprüfen.
2C	Das Feld SVC-Returncode ist nicht mit X'FFFFFFFF' vorbelegt (UTM-Fehler).
FF	Der SVC-Aufruf konnte nicht an das Subsystem UTM weitergereicht werden. Ursache: UTM wurde nicht als Subsystem definiert bzw. geladen.

<sup>1</sup> Bei Returncodes vom Typ UTM-Fehler: PM schreiben und Service informieren

### Returncodes der Form 00mmFFFF

Diesen Returncode setzt UTM oder das DSSM (siehe Handbuch „Makroaufrufe an den Ablaufteil“, Standard-Header). Mögliche Werte und Ursachen sind:

mm	Bedeutung
01	Beim UTM-SVC wurde im Standard-Header ein falscher Wert für UNIT oder FUNCTION angegeben (UTM-Fehler <sup>1</sup> ).
03	Die Schnittstellenversion im Standardheader wird vom Betriebssystem nicht unterstützt. Maßnahme: Subsystemdefinition überprüfen.
04	Die Parameterliste, die beim SVC-Aufruf mitgegeben wird, ist nicht auf Wortgrenze ausgerichtet (UTM-Fehler).
41	UTM wurde noch nicht als Subsystem geladen (CREATIM=ONCREA), oder die Ladebibliothek für UTM enthält nicht den vollständigen Systemcode von UTM V6.5.
82	Beim Start einer neuen Anwendung oder einer neuen Anwendungstask wird festgestellt, dass das Subsystem UTM vom Systemverwalter in den DELETE- oder HOLD-Zustand versetzt wurde.

<sup>1</sup> Bei Returncodes vom Typ UTM-Fehler: PM schreiben, Service informieren

**K048** Es kann kein weiterer UTM Task erzeugt werden

**K049** Fehler &RCCC2 bei Start der Anwendung aufgetreten

UTM gibt die Meldung K049 aus, wenn der Start einer UTM-Task wegen eines Fehlers abgebrochen wird. Die folgende Tabelle enthält die möglichen Fehlercodes &RCCC2, die Fehlerursache und mögliche Maßnahmen zur Behebung des Fehlers.

**Start-Fehlercodes in der Meldung K049**

Code	Fehlerursache	Maßnahme
02	Für den ROOTDATA-Bereich steht nicht ausreichend Speicherplatz zur Verfügung.	UTM-Generierung prüfen, Systemkonfiguration prüfen
03	Der ROOTDATA-Bereich ist nicht allokiert oder der String ' ROOTDATA ' ist nicht vorhanden.	Konsistente openUTM-Version herstellen
04	Die Versionen von KDCROOT und den UTM-Systemmoduln sind verschieden.	Konsistente openUTM-Version herstellen
05	Anwendung kann beim Name Manager nicht angemeldet werden.	Systemfehler, bitte PM schreiben und Service informieren
06	Der Start einer Folgetask wird abgebrochen, weil der Start der ersten Task ebenfalls abgebrochen wurde.	siehe Fehlercode der ersten Task
07	Das Meldungsmodul des Anwenders ist inkonsistent zu dem Meldungsmodul, der zum UTM-Systemcode gehört. (z.B. falsche Anzahl von Inserts)	Meldungsmodul in Ordnung bringen und Anwendung neu binden
08	Es wurde versucht, UTM-Tasks im Dialog- und im Batchmodus gemischt zu starten.	Folgetasks so wie 1. Task starten
09	Platz für fixen Teil des KAA oder für den KDCA-FCB reicht nicht.	Systemfehler, bitte PM schreiben und Service informieren
10	Eine UTM-Task fordert 4 KB Klasse 6 Speicher an, bekommt ihn aber nicht.	BS2000-Generierung/Speicherbedarf der Anwendung überprüfen
11	Fehler beim 1. OPEN Aufruf für die KDCA-Datei von KDCFILE, evtl. falscher FILEBASE-Name in den Startparametern. Mögliche Ursache in UTM-Cluster-Anwendungen: Für den zu startenden Knoten läuft noch ein Online-Import.	siehe DMS-Fehlercode
12	DMS-Fehler beim Lesen der ersten KAA-Seite von der KDCA-Datei	siehe DMS-Fehlercode
13	DMS-Fehler beim Lesen der ersten oder letzten Kontrollseite der KDCA-Datei, die Datei ist nicht verwendbar.	Kopie nehmen oder neu generieren mit KDCDEF

Code	Fehlerursache	Maßnahme
14	Die openUTM-Version in der KAA (KDC Application Area), die aus der KDCFILE gelesen wird, stimmt nicht mit Version des UTM-Systemcodes überein. Der Fehler kann auftreten, wenn Sie versuchen, zwei UTM-Anwendungen mit demselben Namen in verschiedenen openUTM-Versionen parallel zu betreiben.	Name einer Anwendung ändern (MAX APPLINAME=...)
15	Inkonsistenz zwischen KDCDEF und UTM-Code hinsichtlich KAA-Struktur	Neu generieren mit richtigem KDCDEF
16	Die Versionen von KDCDEF und UTM-Systemmoduln sind verschieden.	Konsistente Versionen herstellen
17	Vor dem Start hat das Programm KDCUPD auf der KDCFILE gearbeitet und sich nicht normal beendet.	Konsistente KDCFILE herstellen mit KDCUPD oder KDCDEF
18	Die KDCFILE einer Folge-Task hat eine andere Generierungszeit als die der ersten Task.	Folge-Task mit gleicher KDCFILE starten wie 1. Task
20	Wegen Adressraumangel gelingt es der 1.Task der UTM-Anwendung nicht, den Memory Pool für KAA im Kl. 5 Speicher einzurichten.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und MPOOL-Anweisung überprüfen
21	Wegen Adressraumangel gelingt es einer Folgetask der UTM-Anwendung nicht, sich an den KAA Memory Pool anzuschließen.	wie 20
22	DMS-Fehler beim Lesen einer NSR-Seite in KAA	siehe DMS-Fehlercode
23	Eine KAA NSR-Seite ist zerstört, die KDCA-Datei ist nicht mehr verwendbar.	Neu generieren mit KDCDEF
24	DMS-Fehler beim Lesen einer SR-Seite in KAA	siehe DMS-Fehlercode
25	Eine KAA SR-Seite ist zerstört, die KDCA-Datei ist nicht mehr verwendbar.	Neu generieren mit KDCDEF
26	Mindestens ein Programm für einen generierten Benutzerausgang oder einen ereignisgesteuerten Vorgang fehlt; d.h. das Programm ist weder eingebunden noch konnte es nachgeladen werden.	Programm in die bei der UTM-Generierung angegebene Bibliothek bringen, evtl. neu binden
27	KCSBKAA lieferte einen schlechten Returncode. Beim Initialisieren der SYSLOG-Daten im KAA durch KCSSLOG ist ein Fehler aufgetreten.	Problemmeldung schreiben
28	Beim (Folge-)Start der UTM-Anwendung wurde eine Inkonsistenz bzgl. der Datenbank-Generierung zwischen KDCFILE und ROOT festgestellt. Der Fehler tritt dann auf, wenn KDCFILE und ROOT-Source nicht zusammenpassen. UTM lehnt dies ab, um eventuelle Folgefehler zu verhindern.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen

Code	Fehlerursache	Maßnahme
29	<ul style="list-style-type: none"> <li>– Das Teilprogramm der Administration (KDCADM) fehlt oder</li> <li>– der TAC KDCSHUT ist nicht generiert.</li> </ul>	<ul style="list-style-type: none"> <li>– Programm KDCADM in die UTM-Generierung und ins UTM-Anwendungsprogramm aufnehmen</li> <li>– TAC KDCSHUT generieren</li> </ul>
30	Eine UTM-Task hat einen zusammenhängenden Bereich für die Task-spezifischen UTM-Tabellen (=KTA) im Kl.5 Speicher angefordert, ihn aber nicht bekommen.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf überprüfen, KTA kleiner machen durch Änderung von KDCDEF-Operanden (z.B. MAX TRACEREC= überprüfen)
31	Es wurde versucht, mehr Tasks zu starten als generiert sind bzw. durch die Administration vorgegeben wurden.	Anwenderfehler
32	Schlechter Returncode von KCSBKTA (eventuell konnte KCSSLOG die SYSLOG-Daten im KTA nicht initialisieren).	Problemmeldung schreiben
33	Der Start einer weiteren Task für die Anwendung wird von UTM abgelehnt, weil die Anwendung schon beendet wird (normale Beendigung oder Abbruch).	normales Verhalten
34	Die KDCFILE einer Folge-Task unterscheidet sich hinsichtlich der KAA-Größe von der ersten Task.	Folge-Task mit gleicher KDCFILE starten wie 1. Task
35	Beim Nachladen des Anwendungsprogramms stellt UTM fest, dass die Anwendung abgebrochen wird.	normales Verhalten
36	Fehler beim Name Manager Aufruf für eine Folgetask einer Anwendung	Systemfehler, bitte Service informieren
37	Die Anzahl zu startender Tasks ist für die PGWT-Bearbeitung bzw. für eine UTM-Cluster-Anwendung nicht ausreichend.	Mindestens 2 Tasks starten.
38	Die Task ist bereits beim Name Manager angemeldet, die Adresse für KAA oder KTA ist jedoch ungültig. Der Fehler tritt dann auf, wenn der Start der Anwendung fehlerhaft beendet wurde (xxxx) und anschließend unter der gleichen Task das Anwendungsprogramm nochmal gestartet wird. UTM beendet die Task abnormal, da vermutlich eine Schleife in der Startprozedur vorliegt.	Startprozedur überprüfen
39	Beim Nachladen des Anwendungsprogramms stellt UTM fest, dass mittlerweile die UTM-Generierung unzulässig geändert wurde.	Startprozedur und UTM-Generierung überprüfen

Code	Fehlerursache	Maßnahme
40	Anzahl der Einträge in der Programm-, Lademodul- oder Meldungsmodultabelle im ROOT und in der KDCFILE stimmen nicht überein (siehe K072-Meldung).	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
41	Ein Attribut eines Programms oder eines Lademoduls im ROOT und in der KDCFILE stimmt nicht überein (siehe K073-Meldung).	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
42	Die Task versucht, einen Slot für die Task-Vorgangstabelle anzulegen. Wegen Adressraummangels gelingt dies nicht. Der Slot soll in einem Memory Pool im Kl. 5 Speicher angelegt werden.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf der UTM-Anwendung überprüfen.
43	Die Reihenfolge der Benutzermeldungsmodule in ROOT und KDCFILE stimmt nicht überein.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
44	Die Generierungen von KDCFILE und ROOT stimmen hinsichtlich der maximalen Länge der Nachrichten an/von Klienten nicht überein. (KDCDEF-Anweisung MAX TRMSGLTH)	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
45	Fehler von KCSLKLC beim Setzen eines TAC-Klassen-Locks	UTM-Generierungsparameter MAX RESWAIT=(...,zeit2) größer machen
46	UTM-Cluster-Anwendung: Eine Folge-Task wurde mit anderen Angaben bezüglich Cluster-Filebase gestartet als die erste Task.	1. Task und Folge-Task mit gleichen Startparametern starten
47	Die Generierungen von KDCFILE und ROOT stimmen hinsichtlich der Nutzung von BLS nicht überein.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
48	Fehler beim Initialisieren des Buffer Managements in Start-Case 1 oder 2.	Den virtuellen Adressraum erhöhen; wenn das nicht hilft, PM schreiben
49	Die in einer Folgetask gestartete Phase hat eine andere Versionsnummer als die Phase der ersten Task.	Phase der ersten Task bereitstellen.
50	Die 1. UTM-Task versucht, den Cache Memory Pool einzurichten, oder eine Folgetask will sich anschließen. Wegen Adressraumangel gelingt dies aber nicht. Der Memory Pool soll im Kl.5 Speicher angelegt werden.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf der UTM-Anwendung überprüfen
51	Die Datei stammt nicht von KDCDEF. Dateiname siehe K082 Meldung. (evtl. KDCDEF aus alter Version)	Richtige KDCFILE beschaffen oder neu generieren

Code	Fehlerursache	Maßnahme
52	Die Datei wurde mit einer anderen KDCFILE Datei vertauscht, z.B. Pool-Datei statt KDCB-Datei. Dateiname siehe K082-Meldung.	wie 51
53	Die Datei passt nicht zur KDCA-Datei. Z.B: KDCDEF Uhrzeit verschieden. Dateiname siehe K082-Meldung. Ein Kurzprotokoll nach SYSOUT zeigt die genaue Ursache.	wie 51
54	Die Konsistenzprüfung ergibt, dass die Datei zerstört ist. Evtl. liegt ein Versionsmix vor. Dateiname siehe K083-Meldung.	wie 51, ggf. Versionen überprüfen
55	DMS-Fehler bei KDCA-/KDCB-Datei	siehe DMS-Fehlercode
56	DMS-Fehler bei Pagepool-Datei	siehe DMS-Fehlercode
57	DMS-Fehler bei Wiederanlauf-Datei	siehe DMS-Fehlercode
58	Fehler im Zusammenhang mit der vom Benutzer bereitzustellenden SYSLOG-Datei. Mögliche Ursachen: <ul style="list-style-type: none"> <li>– Es gibt weder den Link-Namen SYSLOG noch eine Dateigenerationsgruppe mit dem Namen basisname.SLOG. Es wird zusätzlich die Meldung K043 ausgegeben.</li> <li>– Es wurde Größenüberwachung generiert (MAX SYSLOG-SIZE), aber die zur Verfügung gestellte SYSLOG ist keine Dateigenerationsgruppe.</li> </ul>	Startprozedur, SYSLOG-Datei und UTM-Generierung überprüfen
59	Fehler beim Öffnen der SYSLOG-Datei	siehe DMS-Fehlercode
60	Fehler beim Anmelden der Anwendung oder einer Folgetask an BCAM bzw. Socket. Genaue Ursache siehe BCAM-Infoword in der K065-Meldung (Standardziel SYSLOG) bzw. K154-Meldung (Standardziel SYSLOG) für Anwendungsnamen mit T-PROT=SOCKET. Anwenderfehler oder Systemfehler	K065 bzw. K154 in der SYSLOG-Datei auswerten, zum Inhalt des BCAM-Infowords Systemdienst fragen
61	Eine Folgetask der Anwendung erhält bei der Anmeldung an BCAM bzw. Socket die Mitteilung, sie sei die 1. Task. Dieser Fall kann auftreten, wenn die UTM-Anwendung vorher durch ein BCAM- Kommando (z.B. BCLOSE) beendet wurde, aber eine Task der Anwendung sich noch nicht beenden konnte, z.B. weil sie eine Langläufertransaktion bearbeitet oder weil sie auf das Hochfahren der Datenbank wartet. (1. Task der Anwendung)	Kontrollieren, ob noch Tasks der Anwendung im System verblieben sind, ggf. Datenbank hochfahren. Anschließend erneut starten.



Code	Fehlerursache	Maßnahme
62	Der 1. Task der Anwendung entdeckt beim Warmstart inkonsistente Daten in KDCFILE.	Systemfehler, bitte PM schreiben und Service informieren. KDCFILE mit KDCDEF neu generieren
63	Die Anzahl der TCB-Entries im KAA und im ROOT stimmen nicht überein.	KDCFILE und/oder ROOT neu generieren
64	Die 1. Task der Anwendung fordert im Warmstart einen Arbeitsbereich von 8 KB im Klasse 6 Speicher an, er bekommt ihn aber nicht.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und UTM-Generierung überprüfen
65	wie 64 (Länge = Taskzahl * 2KB)	wie 64
70, 72, 73	Der 1. Task der Anwendung entdeckt beim Warmstart inkonsistente Daten in KDCFILE.	Systemfehler, bitte PM schreiben und Service informieren. KDCFILE mit KDCDEF neu generieren
74	Datenbankproblem bei DB-Statusabfrage im Wiederanlauf, Ursache KUxx wird ausgegeben, siehe KRCRDC-Fehlercodes.	Datenbank in Ordnung bringen, Anwendung neu starten
75, 76	Der 1. Task der Anwendung entdeckt beim Warmstart inkonsistente Daten in KDCFILE.	Systemfehler, bitte PM schreiben und Service informieren. KDCFILE mit KDCDEF neu generieren
77	Eine Benutzerkennung für einen offenen Asynchron-Vorgang wurde nicht mehr gefunden.	PM schreiben
79	Eine UTM-Task fordert Klasse 6 Speicher an, bekommt ihn aber nicht.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf der Anwendung überprüfen
80, 81	Die 1. Task der Anwendung entdeckt beim Aufbau der Pagepool Belegungstabelle inkonsistente Daten in KDCFILE.	Systemfehler, bitte PM schreiben und Service informieren. KDCFILE mit KDCDEF neu generieren
82	Datenbankproblem beim Löschen der DB-Statusinformation, Ursache KUxx wird ausgegeben, siehe KRCRDC-Fehlercodes.	Datenbank in Ordnung bringen, Anwendung neu starten
83	Datenbankproblem beim Rücksetzen einer DB-Transaktion, Ursache KUxx wird ausgegeben, siehe KRCRDC-Fehlercodes.	Datenbank in Ordnung bringen, Anwendung neu starten
84	Die Task versucht, einen Slot für eine OSI-Vorgangstabelle anzulegen. Wegen Adressraumangel gelingt dies nicht. Der Slot soll in einem Memory Pool im Klasse 5 Speicher liegen.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf der UTM-Anwendung überprüfen

Code	Fehlerursache	Maßnahme
85	Die Task versucht, einen Slot für einen offenen Vorgang anzulegen. Wegen Adressraumangel gelingt dies nicht. Der Slot soll in einem Memory Pool im Kl. 5 Speicher liegen.	Attribute der BS2000-Kennung prüfen, z.B. ADDRESS-SPACE-LIMIT, und Speicherbedarf der UTM-Anwendung überprüfen
86	Die 1. Task der Anwendung entdeckt freizugebende Pagepoolseiten noch im Cache (im Warmstart).	Systemfehler, bitte PM schreiben und Service informieren
89	Fehler beim Name Manager Aufruf für die 1. Task der Anwendung	Systemfehler, bitte PM schreiben und Service informieren
90	Fehler beim Starten eines offenen verteilten Asynchron-Vorgangs	Systemfehler, bitte PM schreiben und Service informieren
91	Fehler beim Start von XAP-TP. Der Fehler wird in der Meldung K124 näher beschrieben.	Systemfehler, bitte PM schreiben und Service informieren
92	Beim Wiederanlauf für OSI TP mit XAP-TP wurde keine freie NODE-Tabelle gefunden.	Systemfehler, bitte PM schreiben und Service informieren
93	Beim Wiederanlauf für OSI TP mit XAP-TP wurde eine Benutzerkennung für eine noch nicht beendete Transaktion nicht mehr gefunden.	Systemfehler, bitte PM schreiben und Service informieren
94	Beim Wiederanlauf für OSI TP mit XAP-TP wurde eine inkonsistente NODE-Tabelle gefunden.	Systemfehler, bitte PM schreiben und Service informieren
95	DMS-Fehler beim Lesen einer UTM-Cluster-Datei (Start-Case 1).	siehe DMS-Fehlercode
96	Eine UTM-Cluster-Datei passt nicht zur Cluster-Konfigurationsdatei; z.B. Erzeugungszeit verschieden. Dateiname siehe K082-Meldung. Ein Kurzprotokoll nach SYSOUT zeigt die genaue Ursache.	Richtige UTM-Cluster-Datei einsetzen oder UTM-Cluster-Anwendung neu generieren.
97	DMS-Fehler beim Schreiben einer Cluster-Datei (Opcode CL_FIRST_NODE).	siehe DMS-Fehlercode
98	Die KDCFILE passt nicht zur Cluster-Konfigurationsdatei; z.B. Erzeugungszeit verschieden. Dateiname siehe K082-Meldung. Ein Kurzprotokoll nach SYSOUT zeigt die genaue Ursache.	Richtige KDCFILE einsetzen oder KDCFILE neu generieren.
99	Datenbankproblem beim Start über IUTMDB: DB-System nicht verfügbar.	Generierte DB-Systeme überprüfen
100	Beim Starten der UTM-Anwendung wurde eine Inkonsistenz bzgl. der KB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen

<b>Code</b>	<b>Fehlerursache</b>	<b>Maßnahme</b>
101	Beim Start einer Folge-Task wurde eine Inkonsistenz bzgl. der KB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
102	Beim Nachladen des Anwendungsprogramms wurde eine Inkonsistenz bzgl. der KB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
103	Beim Starten der UTM-Anwendung wurde eine Inkonsistenz bzgl. der NB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
104	Beim Start einer Folge-Task wurde eine Inkonsistenz bzgl. der NB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
105	Beim Nachladen des Anwendungsprogramms wurde eine Inkonsistenz bzgl. der NB-Länge zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
106	Beim Starten der UTM-Anwendung wurde eine Inkonsistenz bzgl. der Verwendung der SQL-Features zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
107	Beim Start einer Folge-Task wurde eine Inkonsistenz bzgl. der Verwendung der SQL-Features zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
108	Beim Nachladen des Anwendungsprogramms wurde eine Inkonsistenz bzgl. der Verwendung der SQL-Features zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
109	Beim Starten der UTM-Anwendung wurde eine Inkonsistenz bzgl. der Anzahl der Datenbanken zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
110	Beim Start einer Folge-Task wurde eine Inkonsistenz bzgl. der Anzahl der Datenbanken zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
111	Beim Nachladen des Anwendungsprogramms wurde eine Inkonsistenz bzgl. der Anzahl der Datenbanken zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
112	Die erste Task der Anwendung (Start-Case-1) hat eine Inkonsistenz bzgl. der Datenbank-Typen festgestellt (nur bei UTM-S-Anwendungen und Folgestart möglich).	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
115	Beim Starten der UTM-Anwendung wurde eine Inkonsistenz bzgl. der Datenbank-Typen oder der Datenbank-Entries zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen

Code	Fehlerursache	Maßnahme
116	Beim Start einer Folge-Task wurde eine Inkonsistenz bzgl. der Datenbank-Typen oder der Datenbank-Entries zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
117	Beim Nachladen des Anwendungsprogramms wurde eine Inkonsistenz bzgl. der Datenbank-Typen oder der Datenbank-Entries zwischen KDCFILE und ROOT festgestellt.	KDCFILE und ROOT mit gleichen Generierungsparametern erzeugen
118	UTM-Cluster-Anwendung: Fehler beim Lesen des Filebase-Namens der KDCFILE aus der Cluster-Konfigurationsdatei Mögliche Ursachen: – Speicherengpass – Fehler bei Zugriff auf Cluster-File – ungültige Cluster-File – eigener Knoten nicht gefunden Näheres siehe K043- und K190-Meldung.	Siehe K043- oder K190-Meldung; Ggf. Startparameter oder UTM-Generierung korrigieren
119	UTM-Cluster-Anwendung: Die KDCFILE wurde als UTM-Cluster-Anwendung generiert, aber es wurde kein Startparameter CLUSTER-FILEBASE angegeben.	Startparameter korrigieren
120	UTM-Cluster-Anwendung: Die KDCFILE wurde nicht als UTM-Cluster-Anwendung generiert, aber es wurde ein Startparameter CLUSTER-FILEBASE angegeben.	Startparameter korrigieren
122	UTM-Cluster-Anwendung: Fehler beim Bearbeiten der Cluster-Konfigurationsdatei zum Initialisieren des KAA	Siehe K043- und/oder K190-Meldung
123	UTM-Cluster-Anwendung: Fehler beim Registrieren des Knotens	Siehe K043- und/oder K190-Meldung
125	UTM-Cluster-Anwendung: Die Reihenfolge der Knoten-Namen in der KDCFILE ist verschieden von der in der Cluster-Konfigurationsdatei.	UTM-Generierung korrigieren; KDCFILE und ggf. UTM-Cluster-Dateien neu erzeugen
126	UTM-Cluster-Anwendung: Der lokale Rechner gehört keinem oder einem anderen XCS-Verbund an.	Konfiguration des lokalen Rechners überprüfen. Siehe K171-Meldung.
127	UTM-Cluster-Anwendung: Fehler beim Öffnen der Cluster-User-Datei beim Start der ersten Task der Anwendung (Start-Case 1)	Siehe K043- und/oder K190-Meldung
128	UTM-Cluster-Anwendung: Fehler beim Öffnen der Cluster-User-Datei beim Start einer Folge-Task (Start-Case 2)	Siehe K043- und/oder K190-Meldung

Code	Fehlerursache	Maßnahme
129	UTM-Cluster-Anwendung: Fehler beim Öffnen der Cluster-User-Datei beim Neuladen des Anwendungsprogramms nach einem Programmaustausch (Start-Case 3)	Siehe K043- und/oder K190-Meldung
130	UTM-Cluster-Anwendung: Fehler beim Anfordern des Start-Locks zur Serialisierung durch KCSCONS (Start-Case 1)	Siehe K043- und/oder K190-Meldung
131	UTM-Cluster-Anwendung: Fehler beim Öffnen der Dateien des Administrations-Journals beim Start einer Folge-Task (Start-Case 2)	Siehe K043-Meldung und K190-Meldung
132	UTM-Cluster-Anwendung: Fehler beim Öffnen der Dateien des Administrations-Journals beim Neuladen des Anwendungsprogramms nach einem Programmaustausch (Start-Case 3)	Siehe K043-/K190-Meldung
133	UTM-Cluster-Anwendung: Fehler beim Anlegen der Dateien des Administrations-Journals beim Start der ersten Task der Anwendung (Start-Case 1)	Siehe K043-/K190-Meldung
134	UTM-Cluster-Anwendung: Fehler beim Schreiben der Online-Kopie beim Start der ersten Task der Anwendung (Start-Case-1)	Siehe K043- und/oder K190-Meldung
135	UTM-Cluster-Anwendung: Fehler beim Einarbeiten der Online-Kopie beim Start der ersten Task der Anwendung (Start-Case-1)	Siehe K043-/K190- und/oder K174-Meldung
136	UTM-Cluster-Anwendung: Fehler beim Bearbeiten der Cluster-User-Datei beim Start der ersten Task der Anwendung (Start-Case-1)	Siehe K043- und/oder K190-Meldung
137	UTM-Cluster-Anwendung: Der Aufruf von KCCCTRL mit Operationscode REGISTER_NODE lieferte einen unbekanntem Returncode.	Systemfehler, bitte PM schreiben und Service informieren.
138	UTM-Cluster-Anwendung: Der Aufruf von KCCJCTL mit Operationscode WRITE_JOURNAL_PI lieferte einen schlechten Returncode.	Systemfehler, bitte PM schreiben und Service informieren.
139	UTM-Cluster-Anwendung: Sperre zur Start-Serialisierung wurde nicht angefordert.	Systemfehler, bitte PM schreiben und Service informieren.
140	UTM-Cluster-Anwendung: Fehler beim Abschluss der Registrierung des Knotens am Cluster	Siehe K043- oder K190-Meldung
141	UTM-Cluster-Anwendung: Der Aufruf von KCCJCTL mit Operationscode JFCT_SET_KAA_INFO lieferte einen schlechten Returncode.	Systemfehler, bitte PM schreiben und Service informieren.

Code	Fehlerursache	Maßnahme
142	UTM-Cluster-Anwendung: Der Aufruf von KCCJCTL mit Operationscode JFCT_GET_KAA_INFO lieferte einen schlechten Returncode.	Systemfehler, bitte PM schreiben und Service informieren.
143	UTM-Cluster-Anwendung: Fehler beim Öffnen der Dateien des Administrations-Journals beim Start der ersten Task der Anwendung (Start-Case 1)	Siehe K043-Meldung und K190-Meldung
144	UTM-Cluster-Anwendung: Fehler beim Einarbeiten des Administrations-Journals	Systemfehler, bitte PM schreiben und Service informieren.
145	UTM-Cluster-Anwendung: Der Aufruf von KCCJCTL mit Operationscode JFCT_SET_KAA_INFO lieferte einen schlechten Returncode.	Systemfehler, bitte PM schreiben und Service informieren.
146	UTM-Cluster-Anwendung: Die Laufzeit-Konfiguration der startenden Knoten-Anwendung passt nicht zur Laufzeit-Konfiguration der laufenden Knoten-Anwendungen.	Siehe K174-Meldung
147	UTM-Cluster-Anwendung: Im Warmstart sollen globale Administrationsaktionen eingearbeitet werden. Die Dateien des Administrations-Journals können aber nicht geöffnet werden.	Prüfen, ob die Dateien des Administrations-Journals existieren, siehe K043-/K190-Meldung
148	UTM-Cluster-Anwendung: Im Warmstart sollen globale Administrationsaktionen eingearbeitet werden. Interner Fehler beim Lesen der Journaldateien.	Systemfehler, bitte PM schreiben und Service informieren.
149	UTM-Cluster-Anwendung: Interner Fehler beim Bereinigen Knoten-spezifischer Information in den Journaldateien.	Systemfehler, bitte PM schreiben und Service informieren.
150	UTM-Cluster-Anwendung: Fehler beim Öffnen der Dateien des Administrations-Journals beim Start der ersten Task der Anwendung (Start-Case 1).	Siehe K043-Meldung und K190-Meldung
151	UTM-Cluster-Anwendung: Fehler beim Anfordern der Sperre zur Start-Serialisierung der Knoten (im Warmstart)	Siehe K043- oder K190-Meldung
152	UTM-Cluster-Anwendung: Fehler beim Anfordern der Sperre zur Start-Serialisierung der Knoten	Siehe K043- oder K190-Meldung
153	UTM-Cluster-Anwendung: Verwaltungsdatei des Cluster-Pagepools fehlerhaft.	Siehe K190-Meldung
154	UTM-Cluster-Anwendung: Fehler beim Öffnen der LOCK-Datei beim Start einer Folge-Task (Start-Case 2)	Siehe K043-Meldung
155	UTM-Cluster-Anwendung: Fehler beim Öffnen der LOCK-Datei beim Start der ersten Task der Anwendung (Start-Case 1)	Siehe K043-Meldung

Code	Fehlerursache	Maßnahme
156	UTM-Cluster-Anwendung: Fehler beim Öffnen der LOCK-Datei beim Start einer Folge-Task (Start-Case 3)	Siehe K043-Meldung
157	UTM-Cluster-Anwendung: Fehler beim Prüfen der Cluster-GSSB-Datei	Siehe K043- oder K190-Meldung
158	UTM-Cluster-Anwendung: Fehler beim Prüfen der Cluster-ULS-Datei.	Siehe K043- oder K190-Meldung
159	UTM-Cluster-Anwendung: Fehler beim Löschen aller leeren GSSBs.	Siehe K043- oder K190-Meldung
161	UTM-Cluster-Anwendung: Fehler beim Aufheben der ULS-Locks in der Cluster-ULS-Datei	Siehe K190-Meldung
162	UTM-Cluster-Anwendung: Fehler beim Aufheben der GSSB-Locks in der Cluster-GSSB-Datei	Siehe K190-Meldung
164	UTM-Cluster-Anwendung: Fehler beim Öffnen der LOCK-Datei beim Warmstart	Siehe K043-Meldung
165	UTM-Cluster-Anwendung: Beim Warmstart einer Knoten-Anwendung konnte die Cluster-Lock-Datei nicht in der generierten Zeit gesperrt werden.	Start der Anwendung wiederholen
166	UTM-Cluster-Anwendung: Interner Fehler beim Rücksetzen einer Transaktion im Zustand PTC bei der Knoten-Recovery.	Systemfehler, bitte PM schreiben und Service informieren.
167	UTM-Cluster-Anwendung: Interner Fehler bei der Ausgabe vorhandener Transaktionen im Zustand PTC bei der Knoten-Recovery.	Systemfehler, bitte PM schreiben und Service informieren.
168	Wie 167.	Wie 167.
169	UTM-Cluster-Anwendung: Es wurde eine Knoten-Recovery für eine normal beendete Knoten-Anwendung versucht.	Prüfen, ob im Startparameter der falsche Knoten angegeben wurde.
170	UTM-Cluster-Anwendung: Cluster-Pagepool und KDCFILE mit unterschiedlicher BLKSIZE generiert.	Alle UTM-Cluster-Dateien neu generieren.
171	UTM-Cluster-Anwendung: Es wurde eine Knoten-Recovery im Dialog gestartet.	Knoten-Recovery im Batch starten.
172	UTM-Cluster-Anwendung: Fehler beim Sperren der Cluster-User-Datei beim Start der ersten Task der Anwendung (Start-Case 1)	Siehe K043-Meldung und K190-Meldung

Im Fehlerfall werden ein UTM-Dump mit REASON=STnnnn und ein CDUMP erzeugt, falls es für Diagnosezwecke sinnvoll ist.

**K050** Warmstart fuer Anwendung &APPL mit UTM &VERS / &OST1 erfolgreich

**K051** Kaltstart fuer Anwendung &APPL mit UTM &VERS / &OST1 erfolgreich

Die Meldung K050 bzw. K051 wird nach einem erfolgreichen Start der Anwendung ausgegeben. Dabei enthalten die Inserts &VERS die openUTM-Version, &OST1 den Typ des Betriebssystems und &BMD1 den Bit-Modus (32/64).

Die beiden Meldungen besitzen die zusätzlichen Inserts &AMOD, &TERM, &ATYP und &FNOD, die nicht im Standardmeldungstext enthalten sind; diese haben die folgende Bedeutung:

&AMOD enthält den Application Mode der Anwendung:

"S" bei UTM-S und

"F" bei UTM-F

&TERM enthält den beim Start der Anwendung in der KDCFILE eingetragenen Termination Type; folgende Werte sind möglich:

"D": Die KDCFILE wurde neu erstellt mit KDCDEF.

"U": Die KDCFILE wurde mit KDCUPD aktualisiert.

"N": Der letzte Anwendungslauf wurde normal beendet.

"A": Der letzte Anwendungslauf wurde abnormal beendet.

&ATYP enthält :

"C" bei einer UTM-Cluster-Anwendung

"S" bei einer stand-alone-Anwendung

&FNOD enthält:

" " (Leerzeichen) bei einer stand-alone-Anwendung

"Y" beim Start der ersten Knoten-Anwendung einer UTM-Cluster-Anwendung

"N" beim Start jeder weiteren Knoten-Anwendung einer UTM-Cluster-Anwendung

**K052** Start beendet - Task &TASK fuer Anwendung &APPL aktiviert; System-Task: &STSK

&STSK gibt an, ob es sich um eine System-Task handelt (Y) oder nicht (N).

**K053** Neue Userlogdatei wurde erzeugt; die alte Userlogdatei enthaelt &CNTR Saetze**K054** Die Kopien der Userlogdateien sind nicht identisch**K055** Asynchron-Vorgang &ATAC1 durch UTM beendet; KCRCCC=&RCCC;  
KCRCDC=&R CDC; Benutzer=&USER; LTERM=&LTRM**K056** Die Task &TASK wurde beendet**K057** Der Anwendungslauf wurde beendet**K058** Die Task &TASK wurde abnormal beendet**K059** Der Anwendungslauf wurde abnormal beendet



**K060** Der Anwendungslauf wurde abgebrochen; die Ursache ist &TRMA.

UTM erzeugt einen Speicherauszug, wenn eine UTM-Anwendung abnormal beendet wird oder ein Speicherauszug angefordert wird, und zwar für jede Task der Anwendung (siehe auch [Kapitel „Der UTM-Dump“ auf Seite 55](#)).

Das Insert &TRMA beschreibt den Grund (TRMA-Reason) des Abbruchs. Diesen sechsstelligen Code finden Sie im Dateinamen der Dump-Datei (siehe [Kapitel „Der UTM-Dump“ auf Seite 55](#)).

Die TRMA-Reasons der einzelnen Tasks einer UTM-Anwendung sind i.A. nicht identisch. Der TRMA-Reason der Task, die als erstes den Anwendungsabbruch einleitet, bezeichnet dabei den Grund für den Abbruch der Anwendung. Die TRMA-Reasons der Folge-Tasks, sogenannte Folge-Dumps, haben untergeordnete Bedeutung.

Eine Liste der Abbruch-Codes, eine Erklärung und ggf. Empfehlungen zu Änderungen an der Konfiguration der Anwendung finden Sie im [Abschnitt „Fehlercodes im UTM-Dump“](#).

Zur weiteren Diagnose beachten Sie bitte auch die letzten K190-Meldungen.

**K061** Die Dumpdatei &FNAM wurde erzeugt

**K062** Die Dumpdatei konnte nicht erzeugt werden

**K063** Formatierungsfehler &RCF1A &RCF2A bei Format &FMTN fuer asynchrone Nachricht auf LTERM = &LTRM

**K064** Die Nachricht wurde verworfen : &PTRM/&PRNM/&BCAP/&LTRM &DEVC &FIL1A &FIL2A &FIL3 &VTRC &CBRC &IMSG2 &REA1

### Bedeutung

1. An der Workbourse wurde ein ungültiges oder unerwartetes Announcement empfangen. In diesem Fall hat das Insert &REA1 den Wert X'05' und das Insert &IMSG2 enthält das empfangene Announcement.
2. Von einem LTERM-Partner wurde eine Nachricht empfangen, die aus Sicht von UTM logisch inkonsistent ist, z.B. weil der strenge Dialog nicht eingehalten wurde. Oder eine Nachricht kann nicht ausgegeben werden, weil der VTSU sie zurückweist.

Die Meldung enthält als Insert einen 4 Byte langen Returncode, den VTSU an UTM übergibt, sowie zusätzlich den Returncode, den VTSU im VTSUCB übergibt.

Wenn das Insert &REA1 den Wert X'06' hat, enthält die Meldung als Insert den Returncode, den das BS2000-Subsystem ASECO an UTM übergibt.

3. Von einem LU6.1 Partner wurde eine Nachricht empfangen, die keinen gültigen Transmission Header enthält.

Zur Diagnoseunterstützung werden die ersten 32 Zeichen der Nachricht ausgegeben. Damit können z.B. geblockte Eingabe-Nachrichten erkannt werden.

*Vorsicht:* Diese Zeichen können teilweise Überreste einer früheren Nachricht sein, bei Länge < 32 oder K-Taste.

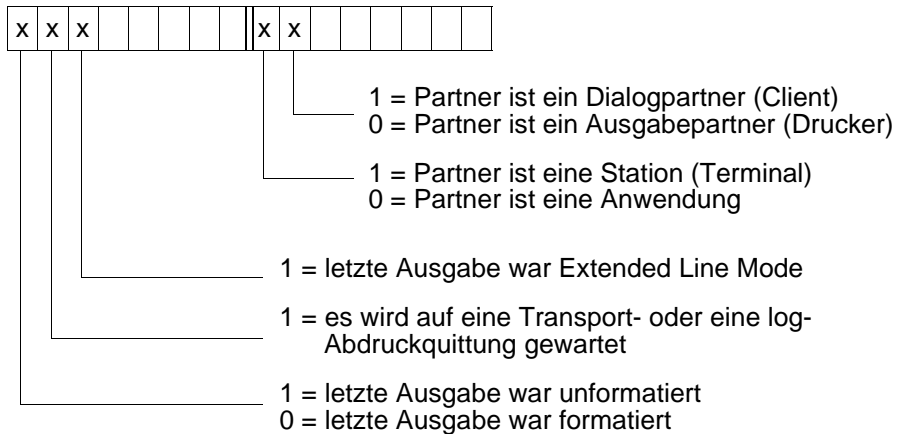
Als Folge dieser Meldung wird die Verbindung abgebaut oder ein automatischer KDCDISP durchgeführt.

Der standardmäßig auf SYSLOG ausgegebene Satz hat folgenden Aufbau:

INSERTS	Bedeutung
&DEVC	Terminal- bzw. Druckertyp, die Codes können dem Makro DCSTA mit TYPE=TCHAR entnommen werden.
&FIL1A	Status der Anwendung mit den Werten X'02' = Startphase X'03' = normaler Lauf X'04' = shut warn X'05' = shut grace X'07' = fast shut X'08' = term application

INSERTS	Bedeutung
&FIL2A	Status des Client bzw. Druckers: X'00' = Verbindung wird abgebaut X'01' = Verbindung nicht aufgebaut X'02' = UTM wartet auf Abschluss des Verbindungsaufbaus X'03' = Verbindung aufgebaut, Benutzer noch nicht angemeldet X'04' = UTM wartet auf Eingabe eines Passwortes X'05' = UTM wartet auf Eingabe eines Passwortes X'06' = UTM wartet auf Eingabe eines neuen Passwortes X'07' = UTM wartet auf Eingabe eines Ausweises X'08' = Benutzer akzeptiert X'09' = 2. Teil des Anmelde-Vorgangs X'0A' = für LTERM mit USAGE=O: Drucker verbunden. Sonst: Benutzer ist angemeldet
&FIL3	Status des physikalischen Terminals bzw. Druckers (2 Byte), dient zur Diagnose in Fehlerfällen. Die Bedeutung der einzelnen Bits finden Sie unten.
&VTRC	VTSU-Returncode
&CBRC	Returncode im VTSUCB
&IMSG2	die ersten 32 Zeichen der Nachricht (bei Stationen enthalten die ersten 8 Zeichen den Header des Stationsprotokolls)
&REA1	Grund der K064-Meldung: X'01' = schlechter VTSU-Returncode X'02' = ungültiger Funktions-Key X'03' = keine positive Abdruckquittung X'04' = schlechter Returncode vom ISLP X'05' = inkonsistente Eingabe-Nachricht X'06' = LU6.1: Die Anno-Länge ist kürzer als das DCF im Transmission-Header X'07' = Eingabe-Nachricht für disconnectierte MUX-Verbindung X'08' = inkonsistente Nachricht von einem LU6.1 Partner X'09' = UPIC-Eingabe-Nachricht mit ungültigem Protokoll X'0A' = Inkonsistente physikalische oder logische Quittung X'0B' = Eingabe-Nachricht von einem als Ausgabepartner generierten Socket-Client X'0C' = Es wurde ein nicht erwartetes GO-Signal für eine UPIC- oder Socket-Verbindung empfangen

Die einzelnen Bits des Inserts &FIL3 haben folgende Bedeutung:



**K065** Netzmeldung : &PTRM/&PRNM/&BCAP/&LTRM &FIL1B &FIL2B

Die Inserts &FIL1B und &FIL2B haben dabei folgende Bedeutung:

Inserts	Bedeutung
&FIL1B	BCAM-Call oder BCAM-, UTM- oder RSO-Announcement-Typ
&FIL2B	BCAM-Infoword (siehe Tabelle <a href="#">Seite 182</a> ), UTM-Announcement-Code oder RSO-Returncode

&FIL1B wird abdruckbar ausgegeben. Dabei kennzeichnen die Werte X'E0', X'E2' und X'F0' bis X'FF' ein UTM-(Timer-)Announcement. Die Werte werden in der folgenden Tabelle erklärt:

&FIL1B	Bedeutung	Bedeutung von &FIL2B
X'01' - X'4C'	BCAM-Call oder BCAM- Announcement <sup>1</sup>	BCAM-Infoword (abdruckbar)
X'56'	RSO-Announcement	asynchroner RSO-Returncode (abdruckbar, siehe K150), Wert und Bedeutung sind in der Beschreibung der Meldung K150 nachzu- lesen oder beim Systemdienst zu erfragen.
X'E0'	Idle-Timeout für eine LU6.1- Session (siehe KDCDEF- Anweisung SESCHA, Parameter IDLETIME)	ohne Bedeutung

&FIL1B	Bedeutung	Bedeutung von &FIL2B
X'E2'	Timeout beim Empfang von Nachrichtenfragmenten auf einer Socket-Verbindung, da die Nachricht nicht innerhalb der erlaubten Zeitspanne von 10 Minuten (interner Wert) komplett empfangen wurde	ohne Bedeutung
X'F0'	UTM-Anno	Ist das erste Byte von &FIL2B X'FD', dann wurde von MSCF ein Shutdown-Announcement empfangen und die Anwendung wird beendet.
X'F1' - X'FA'	UTM-Anno	ohne Bedeutung
X'FB'	UTM-Anno: Timeout beim Warten auf eine Quittung von einem Drucker oder von einer TS-Anwendung nach dem Senden einer Asynchron-Nachricht (siehe KDCDEF-Anweisung MAX, Parameter LOGACKWAIT)	ohne Bedeutung
X'FC' - X'FE'	UTM-Anno	ohne Bedeutung
X'FF'	UTM-Anno: Timeout	Das erste Byte von &FIL2B gibt die Ursache des Timeout an: X'21' Ablauf des Timers, der den Aufbau einer LU6.1-Session überwacht. Siehe KDCDEF-Anweisung UTMD, Parameter CONCTIME=(time1,...).
		X'22' Ablauf des Timer, der den Empfang der Quittung für eine über eine LU6.1-Session gesendete Asynchron-Nachricht überwacht. Siehe KDCDEF-Anweisung UTMD, Parameter CONCTIME=(...,time2).
		X'30' Ablauf des PEND-KP-Timer. Siehe KDCDEF-Anweisung MAX, Parameter TERMWAIT oder PGWTTIME.

&FIL1B	Bedeutung	Bedeutung von &FIL2B
		X'40' Ablauf des PTERM-Idle-Timer. Siehe bis KDCDEF-Anweisung PTERM bzw. X'43' TPOOL, Parameter IDLETIME oder KDCDEF-Anweisung MAX, Parameter PGWTTIME.

<sup>1</sup> Wert und Bedeutung von BCAM-Call und -Announcement können Sie den BCAM-Diagnoseunterlagen entnehmen bzw. beim Systemdienst erfragen

### Werte vom BCAM-Infoword

Die folgende Tabelle zeigt Codes vom BCAM-Infoword, deren Bedeutung sowie mögliche Maßnahmen zur Fehlerbeseitigung.

Es gibt zwei Varianten des BCAM-Infoword. In der ersten Spalte der Tabelle finden Sie die Werte des BCAM-Infoword (im 4. Byte aufsteigend sortiert), die bei einem BCAM-Call (&FIL1B < X'40') als Returncode gesetzt werden, und in der zweiten Spalte die Werte des BCAM-Infoword (im 1. Byte aufsteigend sortiert), die in einem BCAM-Anno (&FIL1B ≥ X'40') gesetzt werden.

Wert (Call)	Wert (Anno)	Bedeutung	mögliche Maßnahmen
00820004	04000000	BCAM nicht aktiv	BCEND - DCSTART
0001xx08	08000000	falscher Aufbau der Parameterleiste:	Softwarewartung verständigen
00010014		Letter too long	
00010108		CONNB nicht angegeben	
00010208		NAME falsch	
00010308		APID nicht angegeben	
00010408		DID nicht angegeben	
00010508		NAME nicht angegeben	
00010608		PNAME nicht angegeben	
00010708		CID nicht angegeben	
00010808		LETAR nicht angegeben	
00010908		TELAR nicht angegeben	
00010A08		NBUFS größer als das Maximum	
00010B08		distribution code length=0	
00010C08		distribution code length größer als das Maximum	
00010D08		NGBS nicht angegeben	

Wert (Call)	Wert (Anno)	Bedeutung	mögliche Maßnahmen
00010E08 00010F08 00011008 00011108 00011208 00011308 00011408 00011508 00011608 00011708 00011808 00011908 00011A08 00011B08 00011C08	08000000	GROB nicht angegeben distribution code nicht angegeben distribution bourse ID nicht angegeben APREF nicht angegeben BLTH nicht angegeben user buffer length = 0 TAKAR nicht angegeben RTACK falsch telegram length = 0 LANNO falsch NBUFS nicht angegeben EVENT falsch RCCOD falsch Anzahl Routen außerhalb des erlaubten Bereichs OWNPN falsch	
0080000C	0C000000	Zeitablauf	Partner überprüfen
00400010	10000000	keine Nachricht angekommen	UTM-BCAM-Trace erzeugen
00010014	14000000	Verbindungs-Nachricht zu lang (Letter too long)	Aufbau der Verbindungs-Nachricht überprüfen
00010018	18000000	DISTAB-Fehler	Softwarewartung verständigen
0480xx1C 0480011C	1C040000	Betriebsmittelengpass data buffer	Aufruf wiederholen/mit BCMOD Speicher vergrößern/Datenflussprobleme durch Softwarewartung untersuchen lassen

Wert (Call)	Wert (Anno)	Bedeutung	mögliche Maßnahmen
0480021C	1C040000	Reference	Bei wiederholtem Auftreten Softwarewartung verständigen
0480031C		ACONCB oder REQCB	
0480041C		APPCB	
0480051C		DISTAB	
0480061C		Name Manager	
0480071C		interface flow control	
0480081C		bourse	
0480091C		ENACB	
04800A1C		ADDRCB-P	
04800B1C		CID	
0840011C	1C080000	Warnung: BCAM wird beendet	
0C40021C	1C0C0000	BCAM wird beendet	
1082001C	1C100000	maximale Anzahl der vordefinierten Anwendungen wurde erreicht	/BCMOD_MAXNPA=
1440001C	1C140000	Warten auf das GO-Signal	
04010020	20000400	Applikation noch nicht bei BCAM bekannt	die Applikation bei BCAM eröffnen
08010020	20000800	Applikation bei BCAM unter einer anderen Task bekannt	Aufruf mit „richtiger“ Task wiederholen
0C010020	20000C00	Applikation exklusiv durch eine andere Task bei BCAM angemeldet	
10010020	20001000	Applikation unter dieser Task bereits bei BCAM angemeldet	
14400220	20001400	erzwungenes Schließen der Anwendung	Applikation neu eröffnen
18400120	20001800	erzwungenes Schließen der Anwendung mit Warnung	Applikation neu eröffnen
1C010020	20001C00	Applikation ist nicht aktiv	/BCACT Applikationsname
20010010	20002000	Applikation darf aktiv keine Verbindung aufbauen	
24820020	20002400	für diese Applikation sind keine weiteren Verbindungen erlaubt	Anzahl der Verbindungen durch das BCMOD-Kommando erhöhen
28820020	20002800	keine nicht vordefinierten Anwendungen für diese Task erlaubt	Mit BCMOD MAXNPT erhöhen
2C010020	20002C00	Applikation-Passwort-Fehler	Passwort berichtigen



Wert (Call)	Wert (Anno)	Bedeutung	mögliche Maßnahmen
30010020	20003000	Aufruf wurde schon durch eine andere Task vorgenommen,	
34010020	20003400	Applikation darf nicht mit hoher Priorität arbeiten	Aufruf mit niedriger Priorität wiederholen
04010024	24000004	Partner hat keine Verbindung	Verbindung aufbauen
08010024	24000008	Partner hat schon eine Verbindung	
0C010024	2400000C	Die Anwendung macht einen weiteren Verbindungsaufbauwunsch zu einem Partner, aber dieser hat noch nicht reagiert	
10400024	24000010	Partner versucht schon, die Verbindung aufzubauen	eigenen Verbindungswunsch vergessen und Verbindungswunsch des Partners akzeptieren
14010024	24000014	Verbindungswunsch hat nicht bestanden oder Zeitablauf	Verbindungswunsch innerhalb des Zeitablaufs akzeptieren
18010024	24000018	Partner nicht bekannt	Partner richtig generieren
1C820024	2400001C	Partnerrechner nicht bekannt	Generierung überprüfen, /BCIN Partnerrechner
1C820124		Partnerrechner nicht aktiv	/BCACT
1C820224		ROUTE nicht bekannt	Generierung überprüfen
1C820324		ROUTE nicht aktiv	/BCACT
	2400002C	Partner nimmt den Verbindungswunsch nicht an	DCM.BCAM.CON-Trace erstellen
30010024	24000030	Applikation darf aktiv keine Verbindung aufbauen	
	24000034	der Partner hat die Verbindung verloren, er wird sie sofort wieder aufbauen	auf Verbindungsaufbau warten
	24000038	Verbindungsabbau durch den Partner	
	2400003C	erzwungenes Schließen der Anwendung durch den Operator	

Wert (Call)	Wert (Anno)	Bedeutung	mögliche Maßnahmen
4001xx24	24000040	Verbindungsparameter inkonsistent:	Verbindungsparameter korrigieren
40010124		TELAR angefordert	
40010224		telegrams nicht erlaubt	
40010324		vordefinierte Verbindung: Aufruf verlangt GROS=F oder THRU=N	
40010424		Benutzerdaten nicht erlaubt	
00800028	28000000	Ein-/Ausgabe später versuchen	Quittung für Verbindungsprotokoll fehlt: Ursache im Zielrechner suchen, UTM-Parameter BRETRYWAIT und BRETRYNR im KDCDEF verändern
0040002C	2C000000	Nachricht nach Zeitablauf verworfen	Anwenderproblem / mehr Tasks bei UTM einsetzen
0020xx30	30000000	BCAM-Fehler	Dump ziehen, ASTRID TOTAL Abzug erstellen
	34000000	Netzadresse inkonsistent	Verbindung neu aufbauen
00800038	38000000	eine andere Ein-/Ausgabe läuft gerade	Aufruf wiederholen, bei wiederholtem Auftreten: DCM.BCAM.CON-Trace einschalten und Abzug erzeugen
0001003C	3C000000	ungültiger Aufruf	Softwarewartung verständigen
0001FFFF		ungültige Einheit/Funktion	
0002FFFF		Funktion nicht verfügbar	
0003FFFF		ungültige Version	
0004FFFF		Parameterleiste falsch ausgerichtet	

**K066** Der Pflichtparameter FILEBASE fehlt

**K067** Fehler in Verbindung mit Language-Connection-Modul &MOD: Errorcode = &ERCD1 &ERCD2; Opcode = &OPCD2

Die Inserts der Meldung haben folgende Bedeutung:

<b>&amp;MOD</b>	<b>Bedeutung</b>
KDCCC	Verbindungsmodul für C
KDCCCOB	Verbindungsmodul für COBOL
KDCCASS	Verbindungsmodul für Assembler
KDCCFOR1	Verbindungsmodul für FORTRAN
KDCCILC	Verbindungsmodul für ILCS
KDCCSPL	Verbindungsmodul für SPL
UTMPL1	Verbindungsmodul für PLI
UTMPAS	Verbindungsmodul für PASCAL

<b>&amp;OPCD2</b>	<b>Bedeutung</b>
INITIALISE	Initialisierung der Sprachumgebung
DEACTIVATE	Freigabe der Sprachumgebung
STXIT-HANDLING	STXIT-Behandlung durch die Sprache
S; XXXXXXXX	Starten des Teilprogramms ' XXXXXXXX'
E; XXXXXXXX	Durchführung der Ende-Behandlung für das Teilprogramm ' XXXXXXXX'

Der ERRORCODE setzt sich aus zwei mal 4 Zeichen zusammen.

<b>&amp;ERCD1</b>	<b>Bedeutung und evtl. Maßnahme</b>
0000	Auftrag wurde korrekt durchgeführt
0004	Auftrag wurde nicht korrekt durchgeführt. Fehlerunterlagen erstellen und an Service schicken
0008	Fehler im Anwendungs-Programm
00FF	Das Language-Connection-Modul für eine generierte Sprache ist beim Start der Anwendung nicht eingebunden

Die Inserts in &ERCD2 liefern die Bedeutung zu aufgetretenen Fehlern und dienen nur zu Diagnosezwecken. Die Bedeutung ist den entsprechenden Handbüchern zu den Programmiersprachen zu entnehmen.

**K068** Datenbank Verbindungsmodul &DBCON Version &DBV1 kann nicht mit KDCCDB-Makro Version &DBV2 zusammenarbeiten

**K069** Verbindungsabbau : &PTRM/&PRNM/&BCAP/&LTRM/&REA4/&REA6/&COTM

Das Insert &REA4 zeigt an, wer den Verbindungsabbau initiiert hat. Das Insert &REA6 enthält die Ursache für den Verbindungsabbau.

&REA4	Bedeutung
B	Das Transportsystem meldet Verbindungsverlust
L	Verbindungsverlust für Socket-Verbindung
sonst	Die Verbindung wurde von openUTM abgebaut, Ursache siehe &REA6

&REA6	Bedeutung
X'00'	Grund nicht spezifiziert
X'08'	Betriebsmittelengpass
X'09'	Verbindung abgebaut
X'0A'	SHUTDOWN der Anwendung
X'0D'	Negativer Returncode vom Transportsystem, siehe zugehörige K065-Meldung.
X'1A'	Zu viele ungültige Anmeldeversuche.
X'2A'	Verbindungsabbau nachdem auf einer verschlüsselten Verbindung eine nicht verschlüsselte Nachricht vom Terminal empfangen wurde.
X'2F'	Abbau der MUX-Session, nachdem die MUX-Transportverbindung schon abgebaut wurde.
X'30'	Verbindungsabbau durch KCSMUX
X'31'	Verbindungsabbau nach KDCOFF
X'32'	Verbindungsabbau nach SIGN OF
X'33'	Verbindungsabbau durch die Druckadministration
X'34'	Verbindungsabbau durch die Administration
X'35'	Keine Nachrichten für Drucker mit PLEV > 0
X'36'	Pagepool-Warnstufe 2 überschritten beim Empfangen einer asynchronen VTV-Nachricht.
X'37'	Verbindungsabbau nach Empfangen einer falschen Status-Nachricht vom Terminal
X'38'	Schlechter Returncode von VTSU-B oder unverschlüsselte Nachricht empfangen, obwohl für die Verbindung permanente Verschlüsselung vereinbart war
X'39'	Ungültige Funktionstaste
X'3A'	Verbindungsabbau durch VTV bei noch nicht aufgebauter Session
X'3B'	Verbindungsabbau durch UPIC
X'3C'	Verbindungsabbau nach erfolgreichem Anmeldeversuch

&REA6	Bedeutung
X'3D'	Verbindungsabbau nach erfolglosem Anmelde-Vorgang
X'3E'	Verbindungsabbau nach Timeout
X'3F'	Nachrichtenfragment empfangen
X'40'	Negative Transport- oder Abdruckquittung
X'41'	Unerwartetes Protokoll-Element
X'42'	Inkonsistente Eingabe-Nachricht
X'43'	Verbindungsabbau durch neuen Verbindungsaufbauwunsch
X'44'	Verbindungsabbau wegen negativer Quittung von RSO.
X'47'	Verbindungsabbau nach Ablauf des IDLE-Timers.
X'49'	Socket - Verbindungsabbau wegen ungültiger Länge der eingehenden Nachricht (negativ oder >32000)
X'4A'	Socket - Verbindungsabbau wegen ungültiger Version im Protokoll Header
X'4B'	Socket - Verbindungsabbau wegen ungültigem Typ im Protokoll Header
X'4C'	Socket - Verbindungsabbau nach einem Fehler beim Schreiben der Nachricht in die Message Queue
X'4D'	Socket -Verbindungsabbau nach Timeout
X'4E'	Socket - Verbindungsabbau wegen Längenproblem: die generierte Nachricht ist größer als die generierte maximale Länge (vgl. Operand MAX TRMSGLTH in der KDCDEF Anweisung)
X'4F'	Socket - Verbindungsabbau wegen unbekanntem Identifier im Protokoll-Header
X'50'	Socket - Verbindungsabbau wegen ungültiger Version Minor im Protokoll-Header
X'51'	Socket - Verbindungsabbau wegen ungültigem Flag im Protokoll-Header
X'52'	Socket - Verbindungsabbau wegen Überschreitung der maximal möglichen Anzahl von Fragmenten
X'53'	Socket - Verbindungsabbau wegen falschem Messagetype FRAGMENT zu Beginn einer Nachricht
X'54'	Socket - Verbindungsabbau wegen fehlendem letzten Teil eines Fragments.
X'56'	Abbau der Verbindung zum RSO-Drucker, da der KDCS-Aufruf FPUT/DPUT RP von der aktuellen RSO-Version nicht unterstützt wird
X'57'	Abbau der Verbindung zum Client, da nach Timeout an der PGWT-Börse nicht genug Tasks aktiv sind, um den Vorgang normal fortzusetzen.
X'59'	Verbindungsabbau durch VTV bei aufgebauter Session.
X'5C'	Inkonsistenz zwischen der an der Socket-Schnittstelle angekündigten Länge der Daten und der Länge der tatsächlich erhaltenen Daten.
X'60'	Socket-Verbindungsabbau wegen nicht mehr unterstützter Protokoll-Version 1.0 (Version Minor X'00') im Protokoll-Header.

**K070** USER / LSES inaktiv : &USER, &GLOBALSG

Das Insert &GLOBALSG 'Cluster Global Sign' ist nur in UTM-Cluster-Anwendungen versorgt und kann folgende Werte annehmen:

<b>&amp;GLOBALSG</b>	<b>Bedeutung</b>
Y	Cluster-globale Abmeldung
N	Knoten-lokale Abmeldung
T	Die Cluster-globale Abmeldung war nicht erfolgreich, weil die Cluster-User-Datei innerhalb der generierten Zeit nicht gesperrt werden konnte.
F	Die Cluster-globale Abmeldung war nicht erfolgreich, weil der Benutzer nicht in der Cluster-User-Datei gefunden wurde.
S	Die Cluster-globale Abmeldung war nicht erfolgreich, weil der Benutzer nicht Cluster-global angemeldet war.
O	Die Cluster-globale Abmeldung war nicht erfolgreich, weil der Benutzer nicht an diesem Knoten Cluster-global angemeldet war.
' '	Abmeldung für einen LU6.1-Session-User oder einen Verbindungs-User.

Die Meldung wird ausgegeben, wenn sich ein Benutzer abmeldet. Sie besitzt zusätzlich folgende Inserts:

&COTM	Anzahl Sekunden seit dem Anmelden von &USER auf dieser Verbindung
&CPTM	Verbrauchte CPU-Zeit in msec in der Benutzererkennung &USER, wenn unter der Benutzererkennung &USER zur Zeit nur noch ein Benutzer angemeldet war. Wenn sich vorher Benutzer unter mehreren Verbindungen unter &USER angemeldet haben, dann enthält der Wert auch ihren CPU-Verbrauch. Anschließend wird der Wert wieder auf 0 zurückgesetzt.

Wenn unter der Benutzererkennung &USER noch weitere Benutzer angemeldet sind, dann ist der Wert = 0.

**K071** Interner Fehler im Datenbankanschluss &DBCON: Operation=&OPCD1, Fehler=&ERCD3, DB-Trace-Info=&DBTRAC

Die Inserts der Meldung haben folgende Bedeutung:

&DBCON	Name des Verbindungsmoduls zum DB-System
&OPCD1	DB-Operationscode, siehe <a href="#">Abschnitt „DB Diagarea“ auf Seite 100</a> .
&ERCD3	Fehlercode KCR CDC mit den Werten KU04 bis KU24, siehe <a href="#">Seite 393</a> .

**&DBTRA** DB-Trace-Information. Die möglichen Werte und deren Bedeutung sind bei dem jeweiligen Datenbanksystem beschrieben:  
 Bei UDS-DB-System: Interpretation der DB-Trace-Information, siehe Beschreibung der Returncodes im Handbuch „UDS/SQL Meldungen“.  
 Bei SESAM-DB-System: siehe Handbuch „SESAM/SQL Datenbankbetrieb“, Abschnitt „Diagnoseinformationen in der DB Diagarea von openUTM“ im Kapitel „Fehlerbehandlung“.

Bei einem Anschluss des DB-Systems über die XA-Schnittstelle haben die 4 Bytes der DB-Trace-Information folgende Bedeutung:

Byte 1	Index des Resource-Managers (vgl. Reihenfolge in den Startparametern)
Byte 2 und 3	Opcode des XA-Aufrufs Ausgabe hexadezimal und plattformabhängig (EBCDIC bzw. ASCII), z.B. im BS2000-System: (EBCDIC) X'9697' = C'op' op = xa_open, en = xa_end etc.
Byte 4	Returncode des XA-Aufrufs, Ausgabe hexadezimal und plattformabhängig z.B. im BS2000-System: FD = -3 = XAER_RMERR

**K072** Die Anzahl der &STMT-Einträge im ROOT und in der KDCFILE stimmen nicht überein

Diese Meldung wird ausgegeben, wenn beim Start der Anwendung erkannt wird, dass die Generierungen von ROOT und KDCFILE hinsichtlich der Anzahl von Programmen bzw. Lademodulen nicht übereinstimmen. Der Start der Anwendung bzw. der Task wird mit der Meldung K049, Code 40 abgebrochen.

Maßnahme: ROOT und/oder KDCFILE neu erzeugen

**K073** Das Attribut &ATTR des &STMTs &PROG im ROOT und in der KDCFILE stimmt nicht überein

Diese Meldung wird ausgegeben, wenn beim Start einer Task festgestellt wird, dass die Einträge der LOAD-MODULE Tabelle oder der PROGRAM Tabelle im ROOT und in der KDCFILE nicht übereinstimmen. Das erste Insert gibt das Attribut an, in dem die Einträge differieren. Der Start der Anwendung bzw. der Task wird mit Meldung K049, Code 41 abgebrochen.

Maßnahme: ROOT und/oder KDCFILE neu erzeugen

**K074** Programmaustausch abgeschlossen; &CTYP &PROG &PVER

**K075** Programmaustausch von Task &TASK abgebrochen; &CTYP &PROG &PVER  
 K074 bzw. K075 werden nach positivem bzw. negativem Abschluss eines Programmaustauschs ausgegeben. Die Inserts geben Information darüber welcher Teil des Anwendungsprogramms ausgetauscht wurde bzw. nicht ausgetauscht werden konnte. Es bedeuten:

&TASK: TSN der Task, von dem der Programmaustausch abgebrochen wurde.

&CTYP: APPL=Austausch des gesamten Anwendungsprogramms  
 LMOD=Austausch eines Lademoduls

&PROG: Name des Lademoduls (nur bei CTYP=LMOD)

&PVER: Versionsnummer des Lademoduls (nur bei CTYP=LMOD)

**K076** Fehler bei asynchroner Administration mit Transaktionscode &ADTC;  
 KCRCCC=&RCCC, KCRCDC=&RCDC

**K077** Interne UTM Cluster Kommunikation mit &PTRM/&PRNM/&BCAP/&LTRM erfolgreich: &CLSIGT

Die K077-Meldung wird nur bei eingeschaltetem Test-Modus (siehe [Seite 44](#)) ausgegeben.

In der internen Cluster-Kommunikation wurde eine Nachricht &CLSIGT an eine andere Knoten-Anwendung gesendet. &CLSIGT kann folgende Werte annehmen:

&CLSIGT	Bedeutung
1	Prüfen auf Verfügbarkeit der überwachten Knoten-Anwendung (check alive request)
2	Prüfen auf Änderungen in der Cluster-Konfiguration
4	Administrationsauftrag
8	Auftrag zum Wecken einer auf einen GSSB oder ULS wartenden Task
16	Antwort an die überwachende Knoten-Anwendung auf eine Verfügbarkeits-Anfrage (check alive response)

Es ist keine Maßnahme notwendig.



**K078** &ERRNAME &ERRCODE: in &REA3

Die Meldung K078 wird von UTM in mehreren Varianten ausgegeben, abhängig vom aufgetretenen Ereignis. Die Ereignisse können in folgende Kategorien unterteilt werden:

- [ILCS-Meldungen](#)
- [Meldung beim Verbindungsaufbau zum DB-System](#)
- [Meldungen beim Laden bzw. Austauschen von Programmen](#)
- [STXIT-Meldungen](#)
- [Meldung \(Warnung\) bei Startparameter-Verarbeitung](#)

*ILCS-Meldungen***K078** WRONG SPL-VERSION (NO ILCS)

Das bedeutet, dass eine falsche SPL-Version eingebunden ist ( $\leq V1.5$ ): SPL baut bei der Initialisierung nicht ILCS-konforme Stacks auf.

Maßnahme: Neu binden mit SPL-Version  $\geq V2.0$ .

**K078** subcode: ERROR HANDLING IN modul, RC = returncode

Das bedeutet, dass in der Verwaltung von ILCS-Stacks oder bei Aufrufen von ILCS-internen Funktionen ein Fehler aufgetreten ist. Bei Aufrufen von ILCS-Funktionen wird der Returncode mitausgegeben.

Anhand des *subcodes* können Sie erkennen, in welcher Phase der Fehler aufgetreten ist:

*subcode* = IL01 - IL09: Fehler bei Initialisierung

*subcode* = IL20 - IL29: Fehler bei TP-Start/ Ende

*subcode* = IL30 - IL39: Fehler bei Stackverwaltung

Die folgende Tabelle beschreibt die Fehler, bei denen *subcode* einen Anwenderfehler beschreibt. In allen anderen Fällen müssen Sie eine Problemmeldung (PM) schreiben und sich an den Service wenden.

subcode	Modul	RC ausgegeben?	genauere Bedeutung von &ERCD2
IL02	ITOINIT	ja	Fehler bei Initialisierung von ILCS. Anwenderfehler, wenn vor dem UTM-Startcoding schon ILCS-Coding durchlaufen wurde. Maßnahme: ENTRY-POINT der Anwendung auf KDCRTST setzen.
IL08	KDCCILCI	nein	Es wurde eine falsche ILCS-Version eingebunden. Es wird mindestens Version 1.5 benötigt.

K078 xxxx : UNRECOVERABLE ILCS ERROR OCCURRED - ILCS FUNCTION WITHIN NON-ILCS PROGRAM UNIT.

xxxx = SEUB/SELB/ASID/AHID

Bedeutung:

Diese Meldung wird bei der Beendigung eines Teilprogrammablaufs ausgegeben. In einem Teilprogramm mit COMP=ASSEMB/COB1 ist ein ILCS-Objekt (Source-Modul) mit einem ILCS-fähigen Compiler übersetzt worden. Das ILCS-Objekt verändert jedoch im zentralen ILCS-Datenbereich (ITOPCD) eines der Felder, wofür xxxx steht. Daher kann openUTM nicht mehr auf diesen Daten aufsetzen, das Anwendungsprogramm wird beendet.

Maßnahmen:

Sie müssen entweder aus Ihrem ASSEMB/COB1-Teilprogramm ein ILCS-fähiges Teilprogramm erzeugen oder das ILCS-fähige Objekt so anpassen, dass es das ILCS nicht benutzt.

Variante der K078-Meldung zur Prüfung der in das Anwendungsprogramm eingebundenen Oracle-Schnittstelle:

```
"K078 *BIND -- MISMATCH -- : in KDCRTDB:
    old ORACLE IUTMDB interface linked in XA environment"
"K078 *BIND -- MISMATCH -- : in KDCRTDB:
    start of application will be terminated."
```

Bedeutung:

Beim Anwendungsstart einer UTM-Oracle-XA-Anwendung wird geprüft, ob versehentlich noch die alte IUTMDB-Schnittstelle eingebunden wurde. Dabei wurde das (alte) Verbindungsmodul ORADBCON der IUTMDB-Schnittstelle im Anwendungsprogramm gefunden. openUTM beendet den Anwendungsstart.

Maßnahme:

Das Anwendungsprogramm muss neu gebunden werden, siehe auch Oracle® Database User's Guide (BS2000/OSD).

*Meldung beim Verbindungsaufbau zum DB-System*

K078 Invalid DBCON-Adress in KDCRTDB:  
KDCRTDB: application running in lower adress-space!Term in Start

Beim Nachladen des DBCON-Moduls wird festgestellt, dass das DBCON-Modul in den oberen Adressraum geladen wurde, die Anwendung aber im unteren Adressraum abläuft.

K078 KDCRTDB ERROR : in SESAM : INSTALLATION-UNIT /  
 VERSION = db-system / version  
 K078 GETINSP 00400013: in KDCRTDB: error accessing pathname for logical-id yyyyyy  
 K078 KDCRTDB ERROR : in KDCRTDB -  
 START OF APPLICATION WILL BE TERMINATED!

Fehler beim Nachladen des DB-Verbindungsmoduls, wenn das Verbindungsmodul mit LIB=LOGICAL-ID (yyyyyy) über den IMON-Installationspfad nachgeladen wird. Der angegebene Name yyyyyy ist fehlerhaft.

K078 \*BIND 0C010600: in KDCRTDB ENTRYNAME = entryname  
 K078 \*BIND ERROR---: in KDCRTDB: LOAD-LIBRARY = : library-name  
 K078 \*BIND ERROR---: in KDCRTDB -  
 START OF APPLICATION WILL BE TERMINATED!

Fehler beim Nachladen des DB-Verbindungsmoduls, wenn das Verbindungsmodul mit LIB=LOGICAL-ID (SYSLNK) über den IMON-Installationspfad nachgeladen wird. Das Nachlademodul ist nicht in der Bibliothek, die über LOGICAL-ID(SYSLNK) gefunden wurde.

K078 RTIMER event A0 : in KDCRTDI - event in KDCRTDB while connecting to database  
 Beim Start der UTM-Anwendung wird der Verbindungsaufbau zum DB-System wegen Timer-Ablauf (Realzeit) abgebrochen. Anschließend wird die Task beendet.  
 Die Meldung wird nur ausgegeben, wenn der Startparameter DB-CONNECT-TIME gesetzt ist (Minimum = 60 Sek., Maximum = 3600 Sek.). In DB-CONNECT-TIME legen Sie den Timer fest, der den Verbindungsaufbau zum DB-System in der Startphase einer UTM-Anwendung überwacht.

#### *Meldungen beim Laden bzw. Austauschen von Programmen*

K078 xxxxxxxx yyyyyyyy: IN modulname text

Diese Variante wird ausgegeben in Verbindung mit Programmen und mit Common Memory Pools im Klasse 6 Speicher, und zwar in folgenden Situationen:

- bei mit BLS erzeugtem Anwendungsprogramm und Nachladen oder Einzelprogrammaustausch mittels BLS
- beim Programmaustausch mittels KDCPROG

Die einzelnen Inserts bedeuten:

xxxxxxx

enthält eine Kurzbezeichnung des aufgetretenen Fehlers (siehe Tabelle ab [Seite 196](#)). Bei den Fehlern mit xxxxxxxx = (KDCS00,LDRTnn) wird das Feld yyyyyyyy nicht mit ausgegeben.

yyyyyyyy

enthält den Returncode des aufgerufenen BS2000-Makros. Weitere Informationen siehe Handbuch „Makroaufrufe an den Ablaufteil“. Bei UTM-internen Fehlercodes entfällt dieses Insert.

modulname

enthält den Modulnamen des Moduls, in dem dieser Fehler aufgetreten ist (für interne Diagnose)

text

enthält zusätzliche Informationen bei bestimmten Fehlern; mögliche Texte:

```
' WRONG OPCODE IN KDCRTL: WRONG OPCODE AT CALL OF KDCRTL '
' LMOD: STAT ERR IN KDCRTL: LMOD: <name> / <version> '
```

Diese Texte deuten auf interne Fehler in UTM hin.

```
' LINK ERR OF AREAS in KDCRTL: FOR AREA: <areaname> '
```

In diesem Fall wurde die angegebene Area vom Anwender nicht eingebunden (UTM-Generierung und Lademodul-Erzeugung prüfen).

Maßnahme: Anwendungsprogramm richtig binden.

xxxxxxx	Fehlerursache	Maßnahme
ASHARE	Fehler beim Laden eines Lademoduls aus Memory-Pools. Weitere Informationen gibt das Insert ' LMOD: name / version '	siehe Returncode.
BIND	<ul style="list-style-type: none"> <li>– Fehler beim Laden eines Lademoduls: Mehr Informationen gibt das Insert ' LMOD: name / version ' oder</li> <li>– Fehler beim Nachladen des Verbindungsmoduls zum DB-System (vgl. Insert ' ENTRYNAME = ')</li> <li>– Fehler beim Nachladen des Root-Tabellenmoduls (vgl. Inserts &amp;object und &amp;lib)</li> <li>– Fehler beim Nachladen des benutzereigenen Meldungsmoduls (vgl. Inserts &amp;object und &amp;lib)</li> </ul>	siehe Returncode.
CSTAT	Der Status einer Seite eines MP's kann nicht auf 'read only' geändert werden. Das zusätzliche Insert ' PAGENUMBER = ' gibt weitere Informationen	Systemfehler.
DEQAR	Fehler bei der Freigabe einer Serialisierungskennung.	siehe Returncode (eventuell Poolname gesperrt; UTM-Generierung ändern).

xxxxxxx	Fehlerursache	Maßnahme
DMS	Fehler bei der Zusammenarbeit mit dem DMS. Das zusätzliche Insert ' FILENAME = ' gibt weitere Informationen.	siehe DMS-Fehlercode.
ENAMP	Fehler beim Einrichten eines Memory-Pools. Zusätzlich werden SCOPE und NAME des betroffenen Memory-Pools angegeben.	siehe Returncode, z.B. Poolname schon von anderer Task belegt; UTM-Generierung ändern.
ENQAR	Fehler beim Zugriff auf die Serialisierungskennung.	siehe Returncode (wie DEQAR) evtl. UTM-Generierung ändern und Wert für MAX MP-WAIT erhöhen.
ITABL	Fehler bei Adressermittlung von Shareable Programmen. Vermutlich handelt es sich um unresolved EXTRNS. Das zusätzliche Insert ' ENTRYNAME = ' gibt weitere Informationen.	Systemfehler.
JOINFOA	Fehler bei der Abfrage des Benutzereintrags.	siehe Returncode.
KDCALME	Beim Anfordern von KL6-Speicher kann nicht genug Speicherplatz bereitgestellt werden (INSUFFICIENT MEMORY und Anzahl Bytes werden mit ausgegeben).	Evtl. wurde versucht die Anwendung im unteren Adressraum zu starten (nicht genug Speicher!).
KDCS00	CALL „KDCS“ im START-Exit, im SHUT-Exit oder im INPUT-Exit.	Teilprogramm korrigieren.
LDRT00	TABLIB-Startparameter fehlt, aber kein ROOT-Modul eingebunden	Startparameter anpassen oder ROOT fest einbinden
LDRT01	Die Version der UTM-Makro-Bibliothek stimmt nicht mit der Version der UTM-Module oder des Meldungsmoduls überein.	ROOT neu generieren, mit aktueller Bibliothek SYSLNK.UTM.nnn binden oder Meldungsmodul neu erzeugen.
LDRT08	Der Parameter CATID wurde verwendet, obwohl der Filebase Operand mit catid spezifiziert wurde.	Startprozedur ändern.
LDRT12	Das Root-Tabellenmodul soll dynamisch nachgeladen werden, es wurde aber kein Lademodul generiert.	Anwendung neu erzeugen (mit mindestens einem LOAD MODULE in Bibliothek SYSLIB, vgl. Parameter ROOT in der KDCDEF-Generierung).
LINK	Fehler beim Laden mit LINK Makro. Das zusätzliche Insert ' ENTRYNAME = ' gibt weitere Informationen.	UTM-Generierung überprüfen.
MINF	Speicherbelegungstabelle ist nicht verfügbar.	Systemfehler.
RELM	Fehler beim Freigeben von KL6-Speicher. Das zusätzliche Insert ' PAGENUMBER = ' gibt weitere Informationen	Systemfehler.

xxxxxxx	Fehlerursache	Maßnahme
REQM	Fehler beim Belegen eines Memorypools.	Systemfehler.
REQMP	Fehler beim Anfordern von KL6-Speicher. Bei Errorcode X'0C' kann es sich um Überschneidung mit einem durch DSSM belegten und mit SCOPE=GLOBAL definierten Memorypool handeln	BS2000-Generierung überprüfen, evtl. UTM Anwendung verkleinern. Bereich freigeben mit Kommando: /SET-SUBSYSTEM-OPTIONS SUBSYSTEM-NAME=NON-PRIVILEGED.
SETIC	Der Intervallzeitgeber konnte nicht gesetzt werden.	Systemfehler.
STXIT	Der STXIT-Aufruf wurde nicht erfolgreich durchgeführt.	Systemfehler.
TABLE	Fehler bei Makro TABLE	KDCSHARE Modul überprüfen.
TMODE	Fehler bei der Abfrage der Task-Eigenschaften.	Systemfehler.
UNBIND	Fehler beim Entladen eines Lademoduls. Weitere Informationen gibt das Insert ' LMOD: name / version '	siehe Returncode.
VSVI	Fehler bei Makro VSVI	siehe Returncode
VSVI1	Fehler beim Ermitteln der Programm-Adressen. Weitere Informationen gibt das Insert ' FOR PROGRAM: name'	siehe Returncode.
\$APCB	Fehler bei der Übergabe des TU-PCs an UTM-TPR, mögliche Returncodes: – 02: \$APCB wird in dieser BS2000-Version nicht unterstützt – sonst: siehe Macro-Returncode	UTM-Fehler. siehe Returncode.

### STXIT-Meldungen

K078 STXIT 01 STXIT(ILCS\_STXIT\_CONTROL) started at xxxxxxxx with IW=yyyyyyyy

K078 STXIT 02 STXIT(STANDARD) started at xxxxxxxx with IW=yyyyyyyy

Die Meldungen „STXIT 01“ und „STXIT 02“ haben folgenden Zweck:

Sie werden von UTM immer beim Start der UTM STXIT-Routinen ausgegeben und protokollieren, dass ein STXIT-Ereignis aufgetreten ist. Diese Meldungen werden ab openUTM Version 5.1 ausgegeben.

STXIT 01	Die STXIT-Routine für ILCS wurde gestartet.
STXIT 02	Die Standard STXIT-Routine wurde gestartet.

Für die Diagnose ist die Angabe des P-Counters der STXIT-Routine wichtig; es kann auf dieser Adresse ein Haltepunkt gesetzt werden und dann mit AID der Fehler analysiert werden.

Im allgemeinen hat ein STXIT-Ereignis einen Abbruch des UTM-Vorgangs zur Folge.

Es können STXIT-Ereignisse auftreten, die nicht zu einem Abbruch des Vorgangs führen. Dies ist dann der Fall, wenn ein STXIT-Ereignis auftritt, an die ILCS-Laufzeitumgebung weitergereicht und dort, entweder durch ILCS selber oder ein Teilprogramm, behandelt wird, und dieses Ereignis dann nicht an UTM weitergegeben wird.

Beim Auftreten eines STXIT-Ereignisses wegen Timer-Ablauf (z.B. IW=A0) ist es möglich, dass zwar die STXIT-Meldung ausgegeben wird, aber keine STXIT-Routine gestartet wird, da mittlerweile das Teilprogramm schon beendet wurde. Sie sollten hier dennoch ggf. den Timerwert für diesen Transaktionscode erhöhen. Der TAC kann der im Anschluss ausgegebenen K099 Meldung entnommen werden:

```
K099 STXIT occured in TAC: tacname at date: xx-xx-xx time: xx:xx:xx
```

Datum und Uhrzeit in der K099 beziehen sich auf den Start des Teilprogramms.

Das Ereignis IW=90 (TERM) kann auftreten, wenn sich eine Task der Anwendung beendet.

Maßnahme:

Bei einem Abbruch des Vorgangs den erzeugten Dump analysieren.

Wenn ein Fehler in UTM vorliegt, dann Unterlagen sicherstellen und PM schreiben, andernfalls den Fehler im Teilprogramm korrigieren.

Wenn der Vorgang nicht abgebrochen wurde, dann zeigt die Meldung nur an, dass die Unterbrechung durch das Laufzeitsystem oder das Teilprogramm behandelt wird, es liegt kein Fehler vor.

*Meldung (Warnung) bei Startparameter-Verarbeitung*

K078 WARNING : in KDCRTBF! Syntax-error in SYSPROT-Parameter!  
Default value used!

K078 WARNING : in KDCRTBF! SYSPROT-Parameter: prefix too long!  
Default value used!

In beiden Fällen wurde ein fehlerhafter Startparameter SYSPROT erkannt, der Default ist wirksam. Gegebenenfalls den Startparameter SYSPROT verbessern!

K078 WARNING : in KDCRTBF! No database generated, but startparameters given!

Es wurde ein Startparameter an ein Datenbank-System angegeben, es ist aber keine Datenbank generiert. UTM-Generierung oder Startparameter anpassen!

Es ist nicht mehr möglich, den Anschluss an eine Oracle-Datenbank mittels der IUTMDB-Schnittstelle zu generieren. Die entsprechende DATABASE-Anweisung wird bereits im KDCDEF-Lauf mit Fehler quittiert. Wenn die Anwendung dennoch mit einem Startparameter ".ORACLE ..." gestartet wird, wird der Start der Anwendung abgebrochen.

Die folgende K078-Meldung wird ausgegeben:

```
K078 Error in start params : in UTM V6.5 Oracle is supported only via
XA-Interface. Task gets terminated.
```

### **K079** Accounting-Problem - Ursache: &REA2

Beim Versuch, Accounting-Information zu schreiben, meldet die BS2000 Accounting-Routine einen Fehler. Die Anwendung läuft weiter.

Im Insert &REA2 wird die Fehlerursache angezeigt:

<b>&amp;REA2</b>	<b>Bedeutung</b>
04	Das BS2000 Accounting ist nicht eingeschaltet
08	Im BS2000 Accounting ist der entsprechende Satztyp nicht eingeschaltet (Satztyp UTMA oder UTMK)
20	Fehler im BS2000-Accounting: Slot-Mangel
28	Fehler im Schreibtask des BS2000-Accounting

Bei REA2=08 schaltet openUTM entsprechend dem Satztyp die Abrechnungsphase, die Kalkulationsphase oder beide aus.

In den übrigen Fällen schaltet openUTM beide Phasen aus.

Maßnahme: BS2000-Systemverwalter informieren. Nach Behebung des Fehlers kann der UTM-Administrator die Kalkulations- und/oder Abrechnungsphase mit dem Kommando KDCAPPL wieder einschalten.

### **K080** KDCMON ist nicht eingeschaltet

Mögliche Fehlerursachen:

- Das Subsystem KDCMON wurde nicht gestartet
- Der UTM-Messmonitor KDCMON wurde nicht gestartet oder hat sich inzwischen wieder beendet

Siehe openUTM-Handbuch „Einsatz von UTM-Anwendungen auf BS2000-Systemen“, Stichwort "KDCMON starten".



**K081** Statistik: &IMSG1/&OMSG1/&CONU/&ATAC2/&LWRT/&HITR/&WTBF

Inserts	Bedeutung
&IMSG1	Anzahl Terminal Input Messages 1)
&OMSG1	Anzahl Terminal Output Messages 1)
&CONU	Anzahl connected Users
&ATAC2	Anzahl unprocessed ATAC' s
&LWRT	Anzahl Log Writes 1)
&HITR	Cache Hit Rate
&WTBF	Cache Waits for Buffer 1)

Die Meldung wird zu jeder vollen Stunde und bei normaler Beendigung der Anwendung geschrieben. Die mit 1) markierten Werte werden danach wieder auf 0 gesetzt.

**K082** Falsche Datei &FNAM**K083** Die Datei &FNAM ist zerstört**K084** &OBJ1 &VER1 &OST1 und &OBJ2 &VER2 &OST2 sind nicht vertraglich**K085** Die &FNKT Funktionen sind nicht verfügbar**K086** UTM-D Fehlerinformation: &PTRM, &PRNM, &BCAP, &LTRM, &USER, &SYSD, &USSD, &FMH7, &AGUS

Inserts	Bedeutung
&PTRM	Name der LU6.1-Verbindung (CON)
&PRNM	Prozessorname
&BCAP	BCAM Anwendungsname
&LTRM	Name des LPAP-Parameters
&USER	Benutzerkennung
&SYSD	SNA sense code (system)
&USSD	SNA sense code (user)
&FMH7	Fehler-Nachricht von der entfernten Anwendung
&AGUS	Benutzerkennung des Auftraggebers

Die Inserts &SYSD und &USSD können bei homogener Kopplung folgende Werte annehmen:

&SYSD&USSD	Bedeutung
08120000	Betriebsmittelengpass (z.B. Pagepool voll)
08130000 081B0000	Contention: Eine Sessionbelegungsanforderung von der Contention-Loser-Anwendung wurde abgelehnt, da die Contention-Winner-Anwendung die Session für einen Auftrag belegt hat.
08641003	ungültiger oder gesperrter TAC

<b>&amp;SYSD&amp;USSD</b>	<b>Bedeutung</b>
0864C5E2	PEND ER durch UTM
0864C5D9	PEND ER vom Anwendungsteilprogramm
0864D9E2	PEND RS durch UTM
0864D9E4	PEND RS vom Anwendungsteilprogramm
0866D9E2	PEND RS durch UTM
0866D9E4	PEND RS vom Anwendungsteilprogramm
10030000	ungültiger oder gesperrter TAC

**K087** Fehler in der Zusammenarbeit mit dem Formatierungssystem: Opcode = &OPCD3, Errorcode = &ERCD4 &ERCD5

<b>&amp;OPCD3</b>	<b>Bedeutung</b>
STPA	Startparameter-Eingabe
INIT	Initialisierung des Formatierungs-Systems
INPI	Anlegen bzw. Vergrößern des PI (Presentation Image)
ISOUT	Einarbeiten Ausgabe-Nachrichtenteil ins PI (bei MPUT)
CRM	Erstellen Ausgabe-Nachricht (bei PEND)
ISINP	Einarbeiten Eingabe-Nachricht ins PI
RDMP	Lesen Eingabe-Nachrichtenteil aus PI (bei MGET)
REST	Restart-Formatierung für Bildschirmwiederanlauf
USER	Benutzer-Aufruf an das Formatierungssystem
CUR	Cursor-Positionierung
CRMA	Erstellen asynchrone Ausgabe-Nachricht

Das Insert &ERCD4 enthält den Fehlercode KCR CDC der Form FHxx:

<b>&amp;ERCD4</b>	<b>Bedeutung</b>
FH01	Ein Fehler bei der Formatierung ist aufgetreten. Siehe zusätzlichen Returncode von FHSCON bzw. FHS im Feld KCRMF.
FH02	Hardware-Funktion nicht vorhanden.
FH04	Schwerer Fehler beim Initialisieren des Formatierungssystems FHS. openUTM bricht den Start der Anwendung ab.

Das Insert in &ERCD5 liefert Zusatzinformationen zu den aufgetretenen Fehlern. Die Bedeutung der Zusatzinformationen können Sie dem Handbuch zum Formatierungssystem entnehmen.

**K088** UTM-D Session Start : &LSES/&RSES/&LPAP  
 SR-STATE: &SRFG PET &PSQN  
 SAVED : &ESQS &EBSS  
 ACT. : &ESQR &ESRR &EBSR

UTM-D Sessionaufbau.

Die Meldung enthält folgende Diagnoseinformationen.

<b>Inserts</b>	<b>Bedeutung</b>
&LSES	LSES-Name
&RSES	RSES-Name
&LPAP	LPAP-Name
&SRFG	Status Bits aus dem LSES-Tabelleneintrag (SR)
&PSQN	PET-Sequenznummern zweimal 2 Bytes: – Sequence number sent saved – Sequence number sent current
&ESQS	gesicherte Sequenznummern zweimal zwei Bytes: – SLU TO PLU – PLU TO SLU
&EBSS	gesicherter Bracketzustand – "BETB" = BETWEEN Bracket – "INBR" = IN Bracket Receive (lokale Seite ohne Senderecht) – "INBS" = IN Bracket Send (lokale Seite mit Senderecht)
&ESQR	aktuelle Sequenznummern (aus STSN request) Insgesamt 5 Bytes: – SEC_TO_PRI_SQN (2 Byte) – PRI_TO_SEC_SQN (2 Byte) – ACTION_CODE_SEC_TO_PRI (2 Bit) 00: ignore 01: set 10: test 11: set and test – ACTION_CODE_PRI_TO_SEC (2 Bit) 00: ignore 01: set 10: test 11: set and test – RESERVED (4 Bit)

Inserts	Bedeutung
&ESRR	aktuelle Sequenznummern RSP(STSN) Insgesamt 5 Bytes: <ul style="list-style-type: none"> <li>– SEC_TO_PRI_SQN (2 Byte)</li> <li>– PRI_TO_SEC_SQN (2 Byte)</li> <li>– ACTION_CODE_SEC_TO_PRI (2 Bit) <ul style="list-style-type: none"> <li>00: ignore</li> <li>01: set</li> <li>10: test</li> <li>11: set and test</li> </ul> </li> <li>– ACTION_CODE_PRI_TO_SEC (2 Bit) <ul style="list-style-type: none"> <li>00: ignore</li> <li>01: set</li> <li>10: test</li> <li>11: set and test</li> </ul> </li> <li>– RESERVED (4 Bit)</li> </ul>
&EBSR	aktueller Bracketzustand <ul style="list-style-type: none"> <li>– "BETB" = BETWEEN Bracket</li> <li>– "INBR" = IN Bracket Receive (lokale Seite ohne Senderecht)</li> <li>– "INBS" = IN Bracket Send (lokale Seite mit Senderecht)</li> </ul>

**K089** Der Auftrag zum Loeschen der asynchronen Nachricht (&GNDATE/&GNTIME) an &DEST wurde angenommen.  
&GNUSER / &USER / (&DLDATE/&DLTIME) / &CHAIN

&CHAIN	Bedeutung
NO	Zu der gelöschten Nachricht war keine negative Folgenachricht vorhanden.
DEL	Die negative Folgenachricht soll ebenfalls gelöscht werden.
ACT	Die negative Folgenachricht soll aktiviert, d.h. in die Nachrichtenkette ihres Empfängers eingekettet werden.

Eine evtl. vorhandene positive Folgenachricht wird bei Ausführung des Auftrags in jedem Fall gelöscht.

**K090** Der Auftrag zum Loeschen aller asynchronen Nachrichten an &DEST wurde angenommen.  
&USER / (&DLDATE/&DLTIME)

**K091** Wegen Betriebsmittelengpass ist zur Zeit kein Anmelden an Anwendung &BCAP moeglich - bitte versuchen Sie es spaeter.

Diese Meldung besitzt zusätzlich folgendes Insert:

&REA7            Ursache warum die Anmeldung fehlgeschlagen ist. Die Werte in &REA7 sind bei der Meldung K147 auf [Seite 236](#) beschrieben.

**K092** Bitte Passwort eingeben

&gt; &amp;PAS1 &lt;

Falls Aenderung des Passworts gewuenscht, bitte neues Passwort eingeben

&gt; &amp;PAS2 &lt;

und neues Passwort wiederholen

&gt; &amp;PAS3 &lt;

**K093** Vorgangskellerung zur Zeit nicht moeglich.

Siehe auch KDCDEF-Steueranweisung MAX, Operand NRCONV.

**K094** Sequenz von erfolglosen Anmeldeversuchen.

Mit dieser Meldung löst openUTM einen stillen Alarm aus, um auf eine Folge von erfolglosen Anmeldeversuchen eines Benutzers oder von einem Client aus hinzuweisen. In der Anwendung kann dann z.B. über das MSGTAC-Teilprogramm auf diese Situation reagiert werden. Über die Anweisung SIGNON ...SILENT-ALARM = *nnn* des Dienstprogramms KDCDEF kann eingestellt werden, nach wievielen erfolglosen Versuchen openUTM die Meldung K094 erzeugen soll.

Die Meldung enthält folgende Inserts:

Inserts	Bedeutung
&PTRM	Stationsname
&PRNM	Prozessorname
&BCAP	Anwendungsname
&LTRM	logischer Stationsname
&USER	Benutzer-Identifikation
&RCF1B	Grund für die Ablehnung
&REA4	Verursacher der Ablehnung (L=LTERM, U=USER, B=BOTH)

Das Insert &USER enthält die Benutzer-Identifikation des letzten erfolglosen Anmeldeversuchs der Sequenz. Dies kann eine für die Anwendung generierte Benutzerkennung oder aber eine beliebige Zeichenfolge sein, die UTM als Benutzer-Identifikation übergeben wurde.

Das Insert &RCF1B enthält in drei Zeichen den Grund für die Zurückweisung des letzten Anmeldeversuchs der Sequenz. Es wird von UTM so versorgt wie die Rückgabefelder KCRSIGN1 und KCRSIGN2 des KDCS-Aufrufs SIGN ST.

Das Insert &REA4 zeigt, ob der Benutzer (U), das Client-LTERM (L) oder beide (B) Verursacher der Meldung sind. Der Zähler für die fehlerhaften Anmeldeversuche des Verursachers wird auf 0 zurückgesetzt.

**K095** KDCOFF ist wirksam - Bitte <DUE> eingeben**K096** Rueckkehr zum unterbrochenen Vorgang - Bitte <DUE> eingeben.

**K097** Die Angaben zum neuen Passwort sind nicht verwendbar - Bitte KDCSIGN

**K098** Der Input-Exit lieferte die Fehlerinformation &RCF1C &RCF2B - Bitte Eingabe  
Die Meldung wird dann erzeugt, wenn beim Aufruf des INPUT-Exits von UTM oder vom Exit selbst Fehler entdeckt wurden.

Insert &RCF1C zeigt an, was UTM bei oder nach dem Aufruf des INPUT-Exits entdeckt:

&RCF1C	Bedeutung
IN00	Aufruf des INPUT-Exits aus Sicht von UTM o.k.
IN01	Fehler beim Aufruf des INPUT-Exits über IUTMHLL
IN02	KDCS-Aufruf im INPUT-Exit
IN03	Ausgabeparameter passen nicht zu KCICCD
IN04	unzulässiger Wert in KCICCD
IN05	DB-USER-CALL im INPUT-Exit
IN06	STXIT im INPUT-Exit

Insert &RCF2B zeigt an, was der INPUT-Exit im Fall KCICCD = 'ER' in den Ausgabeparameter KCIERRCD eingetragen hat.

**K099** &MSG

Diese Meldung gibt openUTM entweder bei Auftreten eines STXIT-Ereignisses oder als Ergänzung zur Meldung K078 oder bei zu langem Startparameter ENTER-PROC-INPUT aus. Im ersten Fall enthält das Insert &MSG:

```
STXIT occured in TAC: tacname at date: jj-mm-tt time: hh:mm:ss
```

Im zweiten Fall enthält das Insert &MSG zusätzliche Returncodes.

Im dritten Fall enthält das Insert &MSG den Text:

```
KDCRTBF: maximum length(2000 byte) of ENTER-PROC-INPUT exceeded!
```

Zusätzlich kommt diese Meldung auch als Info-Meldung bei generierten Memory Pools. Sie gibt dann Hinweise auf die tatsächliche Belegung des Memory Pools durch in diesen Pool geladene shared Teile des Anwendungsprogramms. Zwei K099-Meldungen werden ausgegeben: Die erste benennt den betroffenen Memory Pool, die zweite den Belegungsgrad durch generierte shared Teile, falls dieser Pool mit Schreibschutz generiert ist.

**K101** Betriebsmittelengpass - Bitte Eingabe wiederholen

Die Meldung wird bei folgenden Ereignissen ausgegeben:

- Wenn eine Dialog-Eingabe-Nachricht wegen einer TAC-Klassenwartesituation im Pagepool zwischengespeichert werden soll und der Platz im Pagepool nicht mehr ausreicht.
- Wenn eine für ein Asynchron-Programm, eine TAC-Queue oder eine temporäre Queue bestimmte Eingabe-Nachricht auf Pagepool gesichert werden soll und entweder der Warnlevel 2 überschritten ist oder der Platz auf Pagepool nicht ausreicht.
- Wenn eine Eingabe-Nachricht für ein Asynchron-Programm, eine TAC-Queue oder eine temporäre Datei bestimmt ist und die Anzahl der gesicherten Nachrichten für diesen Asynchron-TAC, diese TAC-Queue oder diese temporäre Queue den generierten Wert QLEV schon erreicht oder überschritten hat und die temporären Queues mit QMODE=STD generiert wurden.

**K102** Unterbrechungsgewicht &CIW bei Befehlszaehler &CPC &CDSPL.

Befehlslaengencode : &CILC Programmmaske : &CPM Condition Code : &CCC  
 AMODE : &CAMODE  
 R0 &CR0 R1 &CR1 R2 &CR2 R3 &CR3 R4 &CR4

Die Meldungen K102 und K103 werden zusammen ausgegeben (siehe Erläuterung bei der Meldung K103).

**K103** R5 &CR5 R6 &CR6 R7 &CR7 R8 &CR8 R9 &CR9  
 R10 &CR10 R11 &CR11 R12 &CR12 R13 &CR13 R14 &CR14  
 R15 &CR15

Ausgabe des Kontextes beim Auftreten eines STXIT-Ereignisses, siehe auch Tabelle im [Abschnitt „Inserts in Meldungen“ auf Seite 341](#).

Unterbrechungsgewicht 48:

Tritt das STXIT-Ereignis 48 bei Befehlszähler KDCS + 8 auf, dann bedeutet dies, dass beim Aufruf der KDCS-Schnittstelle eine Adresse für den KDCS-Parameterbereich angegeben wurde, auf die nicht zugegriffen werden kann.

Maßnahme: Anwendungs-Teilprogramm ändern!

Unterbrechungsgewicht 90:

Bei einem Terminierungsversuch in einem Teilprogramm (z.B. durch die Anweisung „exit (1)“ in einem C++ oder C-Programm) werden erst mehrere Laufzeitmodule aufgerufen, bevor openUTM wieder die Kontrolle über das Anwendungsprogramm erhält. Daher wird bei der Darstellung des Registersatzes der zuletzt zur Verfügung stehende Registersatz interpretiert.

Das für die Terminierung verantwortliche Modul finden Sie dann über die Stackverkettung, wenn Register 13 auf eine gültige Save Area zeigt.

**K104** UTM-D &UTMDEVT (&RCVDANNO): &LSES , &LPAP , &AGUS  
; alter Status: ( &OCVST, &OTAST ); Aktion: &ACTION; neuer Status: ( &NCVST,  
&NTAST ).

Bei Kommunikation über LU6.1 wird K104 bei folgenden Ereignissen ausgegeben:

- Bei Ablauf eines der folgenden Timer:
  - Session-Belegungs-Timer für einen Dialog-Auftrag (siehe Wert *time1* bei der KDCDEF-Anweisung LTAC, Parameter WAITTIME).  
Das Insert &LSES enthält in diesem Fall Leerzeichen.  
*Ausnahme:* Timer-Ablauf nach Belegung einer Contention-Loser-Session.  
*Sonderfall:* Ist der Session-Belegungs-Timer einer Contention-Winner-Session gleich 0 und bei Teilprogrammende keine Session frei, wird dies wie der Ablauf eines Timer behandelt.
  - Antwort-Timer (siehe Wert *time2* bei der KDCDEF-Anweisung LTAC, Parameter WAITTIME).  
In diesem Fall enthalten &LSES und &AGUS unterschiedliche Namen.
  - Prepare-to-Commit-Timer (siehe KDCDEF-Anweisung UTMD, Parameter PTCTIME). &LSES und &AGUS sind in diesem Fall gleich.
- Bei Verbindungsverlust, falls die Session belegt ist
  - von einem Vorgang mit offener Transaktion oder
  - von einem Vorgang, der das Transaktionsende eingeleitet hat.
- Bei Sessionwiederanlauf, falls die Session von einem Vorgang belegt ist, der das Transaktionsende eingeleitet hat.

Bei Kommunikation über OSI TP wird die Meldung bei Ablauf eines der folgenden Timer ausgegeben. &LSES enthält immer Leerzeichen.

- Association-Belegungs-Timer (für Dialogaufträge siehe Wert *time1* bei KCDEF-Anweisung LTAC, Parameter WAITTIME, für Asynchronaufträge interner Wert von 60 Sekunden).  
Bei einem Asynchron-Auftrag enthält &ACTION den Wert ASYNCH.
- Antwort-Timer (ür Dialogaufträge siehe Wert *time2* bei KCDEF-Anweisung LTAC, Parameter WAITTIME, für Asynchronaufträge siehe Wert *time2* bei KCDEF-Anweisung UTMD, Parameter CONCTIME).  
Bei einem Dialog-Auftrag wird auf die Antwort, bei einem Asynchron-Auftrag auf die Quittung von Auftragnehmer gewartet. Bei einem Asynchron-Auftrag enthält &ACTION den Wert ASYNCH.
- PEND-KP- und PGWT-KP-Timer (siehe KDCDEF-Anweisung MAX, Parameter TERMWAIT bzw. PGWTTIME).



- Ready-Timer (siehe KDCDEF-Anweisung UTMD mit Parameter PTCTIME oder KDCDEF-Anweisung MAX mit Parameter PGWTTIME)

Inserts	Bedeutung
&UTMDEVT	Ereignis, bei dem die Meldung ausgegeben wurde: RESTART Sessionwiederanlauf DISCON Verbindungsverlust TIMEOUT Timeout
&RCVDANNO	Zuletzt an der UTM-Börse empfangenes Announcement. Bei &UTMDEVT = TIMEOUT haben die ersten beiden Bytes in &RCVDANNO folgende Bedeutung: X'F330' Timer-Ablauf im OSI-TP-Auftragnehmer-Vorgang nach dem X'F331' Senden einer Nachricht an den Auftraggeber in einer Transaktion (siehe KDCDEF-Anweisung MAX, Parameter TERMWAIT oder PGWTTIME). X'F332' Timer-Ablauf im OSI-TP-Auftraggeber-Vorgang nachdem alle Auftragnehmer zum Einleiten des Transaktionsendes aufgefordert wurden. Der Timer entspricht dem größten generierten Wert von <i>time</i> in der KDCDEF-Anweisung MAX PGWTTIME= und von <i>time2</i> in der KDCDEF-Anweisung LTAC WAITTIME= der beteiligten LTACs. X'F333' Ablauf des Ready-Timer im OSI-TP-Auftragnehmer-Vorgang (KDCDEF-Anweisung UTMD, Parameter PTCTIME oder KDCDEF-Anweisung MAX PGWTTIME=). X'F400' Ablauf des OSI-TP-Association-Belegungs-Timer für einen Dialog-Auftrag ( <i>time1</i> in KDCDEF-Anweisung LTAC, Parameter WAITTIME). X'F520' Ablauf des OSI-TP-Association-Belegungs-Timer für einen Asynchron-Auftrag (interner Timer von 60 Sekunden). X'F522' Ablauf des Timer, der den Empfang der Quittung für eine Asynchron-Nachricht überwacht, die über eine OSI-TP-Association gesendet wurde ( <i>time2</i> in KDCDEF-Anweisung UTMD, Parameter CONCTIME). X'F534' Ablauf des OSI-TP-Antwort-Timer im Auftraggeber-Vorgang für einen Dialog-Auftrag. Der Timer entspricht dem größten generierten Wert von <i>time</i> in der KDCDEF-Anweisung MAX PGWTTIME= und von <i>time2</i> in der KDCDEF-Anweisung LTAC WAITTIME= der beteiligten LTACs. X'F800' Ablauf des LU6.1-Session-Belegungs-Timer für einen Dialog-Auftrag ( <i>time1</i> in KDCDEF-Anweisung LTAC, Parameter WAITTIME)).

Inserts	Bedeutung
X'F933'	Ablauf des Prepare-to-Commit-Timer im LU6.1-Auftragnehmer-Vorgang (KDCDEF-Anweisung UTMD, Parameter PTCTIME oder KDCDEF-Anweisung MAX, Parameter PGWTTIME).
X'F934'	Ablauf des LU6.1-Antwort-Timers in einem Dialog-Auftrag (Der Timer entspricht dem größten generierten Wert von <i>time</i> in der KDCDEF- Anweisung MAX PGWTTIME= und von <i>time2</i> in der KDCDEF-Anweisung LTAC WAITTIME= der beteiligten LTACs).
X'F935'	Ablauf des LU6.1-Antwort-Timers in einem Dialog-Auftrag nach erneutem Aufbau der LU6.1-Session (Der Timer entspricht dem größten generierten Wert von <i>time</i> in der KDCDEF- Anweisung MAX PGWTTIME= und von <i>time2</i> in der KDCDEF-Anweisung LTAC WAITTIME= der beteiligten LTACs.).
X'F936'	Ablauf des LU6.1-Antwort-Timer beim Vorgangwiederanlauf nach Aufbau einer Session ( <i>time2</i> in KDCDEF-Anweisung LTAC, Parameter WAITTIME)
&AGUS	Name des Auftraggebers (Benutzer, Session oder Association)
&ACTION	Maßnahme von openUTM: COMMIT Beenden der Transaktion RESET Rücksetzen der Transaktion WAIT nichts STPROG Folgeprogrammstart ASYNCH Erneute Ausführung des Asynchron-Auftrag.
&OCVST &OTAST &NCVST &NTAST	Vorgangs- und Transaktionsstatus von dem in &AGUS angegebenen Auftraggeber (vor und nach der in &ACTION angegebenen Aktion). Mögliche Werte: O O Vorgang und Transaktion sind offen. O P Der Vorgang ist offen und hat Transaktionsende eingeleitet. O C Der Vorgang ist offen, die Transaktion beendet. C C Vorgang und Transaktion wurden beendet. O R Die Transaktion wird zurückgesetzt, der Vorgang bleibt offen. Z R Die Transaktion wird zurückgesetzt, der Vorgang beendet.

**K105** UTM-D &SYST mismatch; &LSES , &LPAP , &AGUS

&SYST enthält das System, mit dem ein Mismatch entstanden ist:

DB Datenbank

LPAP andere Anwendung.

**K106** Die Ausgabenachricht wurde verworfen.

**K107** Der Terminaltyp &TTYP wird von VTSU nicht unterstützt oder die Verbindungsnachricht ist falsch.

**K108** Fehler &KRRC bei der Kerberos Authentifizierung.

Die Meldung besitzt folgende Inserts:

Inserts	Bedeutung
&KRLH	Länge des Client- oder Server-Principal
&KRPR	Die ersten 100 Zeichen des Client- oder Server-Principal. Ist die Länge des Principal kleiner als 100, ist der Inhalt der restlichen Zeichen undefiniert.
&KRRC	Returncode

Der Returncode KRRC ist gemäß dem BS2000-Standardheader aufgebaut und hat folgende Bedeutung:

SC2	SC1	MC	Bedeutung
	X'00'	X'0000'	Funktion ohne Fehler ausgeführt
	X'01' X'02'	X'0018'	Der Ausgabebereich reicht nicht aus, um den kompletten Kerberos-Principal abzulegen.
		X'FFFE'	Parameterfehler.
		X'FFFF'	Funktion nicht verfügbar.
		X'FFFF'	Funktion nicht verfügbar.
	X'03'	X'FFFF'	Version nicht unterstützt.
	X'20'	X'00FF'	Interner Fehler von SECOS. Es wurde ein SERSLOG-Eintrag erstellt.
	X'40'	X'0001'	Das Ticket des Verbindungspartner ist nicht mehr gültig. Im Ausgabebereich steht ein gültiger Principal des Client.
		X'0002'	Das Ticket des Verbindungspartners ist zu alt. Im Ausgabebereich steht der Client.
		X'0003'	Server- und Client-Realm sind unterschiedlich, was aber per Systemeinstellung verboten wurde. Im Ausgabebereich steht der Client.
		X'0014'	In der Key-Tabelle existiert kein Eintrag mit der angegebenen ENTRY-ID. Kein Eintrag im Ausgabebereich.
	X'41'	X'FFFF'	Subsystem SECOS ist nicht geladen.
	X'42'	X'FFFF'	Subsystem SECOS ist nicht verbunden.
	X'81'	X'FFFF'	Subsystem SECOS ist zeitweise nicht erreichbar.
	X'82'	X'FFFF'	Subsystem SECOS wurde angehalten oder gestoppt.

SC2	SC1	MC	Bedeutung
X'01'	X'40'	X'0008'	Das Ticket des Verbindungspartners ist ungültig. Im Ausgabebereich steht der Principal des Servers.
		X'0009'	Das Ticket des Verbindungspartner ist mit einem Algorithmus verschlüsselt, für den kein Schlüssel im BS2000-System vorliegt. Im Ausgabebereich steht der Principal des Servers.
		X'000A'	Für den im Ticket angegebenen Server liegt im BS2000-System kein Schlüssel vor. Im Ausgabebereich steht der Principal des Servers.
		X'000B'	Die im Ticket angegebene Schlüsselversion liegt im BS2000-System nicht vor. Im Ausgabebereich steht der Principal des Servers.
X'02'	X'40'	X'0010'	Problem bei VTSU-B. Mögliche Ursachen sind: <ul style="list-style-type: none"> <li>– Der Verbindungspartner unterstützt keine Kerberos-Authentifizierung.</li> <li>– Der Anwender hat mit einer K-Taste den Authentisierungs-Vorgang unterbrochen.</li> </ul> Kein Eintrag im Ausgabebereich.
		X'0011'	Der Verbindungspartner hat ein Ticket geliefert, das nicht interpretiert werden konnte. Kein Eintrag im Ausgabebereich.
		X'0012'	Die Terminal-Emulation benutzt eine Protokollversion, die nicht unterstützt wird. Kein Eintrag im Ausgabebereich.
		X'0013'	Während der Authentisierung wurde die Verbindung abgebaut. Kein Eintrag im Ausgabebereich.

**K109** Fehler &KRRC bei der Kerberos Authentifizierung - Bitte KDCSIGN

Inserts siehe Meldung [K108](#).

**K110** Ungültiger Kerberos Principal (&KRPR) - Bitte KDCSIGN

Inserts siehe Meldung [K108](#).

**K111** Anmeldung wegen fehlender Kerberos-Unterstützung nicht möglich - Bitte KDCSIGN

**K112** Das Passwort kann z.Zt. nicht geändert werden

**K115** MUX-Protokollfehler: &PTRM/&PRNM/&BCAP/&LTRM  
&SNPT/&SNPR/&SNLT &CCC &REA1  
&ANNO

<b>Inserts</b>	<b>Bedeutung</b>
&PTRM	PTRM-Name der Transportverbindung
&PRNM	Prozessor-Name der Transportverbindung
&BCAP	BCAM-Anwendungsname
&LTRM	LTRM-Name der Transportverbindung
&SNPT	PTRM-Name der Session
&SNPR	Prozessor-Name der Session
&SNLT	LTRM-Name der Session
&CCC	Fehlerkontext (siehe folgende Tabelle)
&REA1	Fehlerursache
&ANNO	BCAM-Announcement

<b>&amp;CCC</b>	<b>Bedeutung</b>
X'01'	Der Fehler kann einer bestehenden Session zum Multiplexanschluss zugeordnet werden.
X'02'	Fehler beim BIND-Request.
X'03'	Der Fehler kann nicht zugeordnet werden.

Das Insert &REA1 wird nur zur Systemdiagnose ausgegeben. Die Felder &SNPT, &SNLT, &SNPR können nur versorgt werden, wenn der Fehler einer bestehenden Session zugeordnet werden kann (&CCC=X'01') bzw. beim Sessionaufbau bereits eine Session identifiziert werden konnte (also ggfs. bei &CCC=X'02').

**K116** MUX-Sessionabbau: &PTRM/&PRNM/&BCAP/&LTRM  
&SNPT/&SNPR/&SNLT/&USER &REA1

<b>&amp;REA1</b>	<b>Bedeutung</b>
X'01'	Der Benutzer hat die Session durch Eingabe von KDCCOFF abgebaut.
X'02'	Der Benutzer hat die Session durch Eingabe von KDCCOFF BUT abgebaut.
X'03'	Ein UTM-internes Ereignis hat zum Sessionabbau geführt (z.B. Timeout für die Session, Administration, ...).
X'04' bis X'07'	Protokollfehler
X'10'	Die Session wurde von außen abgebaut.
X'0E'	Nachricht konnte nicht gelesen werden (LETTER_LOST).

Das Insert &USER bezeichnet den User, der der Session zugeordnet ist. Die Bedeutung der anderen Inserts entspricht deren Bedeutung bei der K115-Meldung.

**K117** MUX-BIND abgelehnt: &PTRM/&PRNM/&BCAP/&LTRM  
&SNPT/&SNPR/&SNLT/&USER &REA1

<b>&amp;REA1</b>	<b>Bedeutung</b>
X'01'	Die angegebene Schnittstellenversion wird nicht unterstützt.
X'02'	Der Partner ist für das Signon-Handling verantwortlich. Die mitgelieferten Signon-Daten waren aber nicht komplett bzw. nicht korrekt.
X'03'	Die maximale Anzahl der Sessions ist belegt, d.h. über diesen Multiplexanschluss kann keine weitere Session belegt werden.
X'04'	Der BIND-Request wurde abgelehnt (allgemeiner Fehlercode).
X'05'	Die angegebene Station ist schon angemeldet.
X'06'	Der angegebene Benutzer ist schon angemeldet.
X'08'	Die Stationsangaben im BIND stimmen nicht mit der UTM-Generierung überein.
X'09'	Beim BIND wird ein falsches Passwort geliefert.
X'0A'	Der BIND-Request konnte nicht gelesen werden (LETTER_LOST).
X'0B'	Protokollfehler.
X'0C'	Die angegebene Station wird vom Kommunikationspartner nicht unterstützt (UTM: VTV-Station, keine Dialogstation, PTYPE=APPLI, ANNOAMSG=N).
X'0D'	Die angegebene Station wird vom Kommunikationspartner nicht unterstützt (UTM: PROTOCOL=N ist generiert).
X'0E'	Protokollfehler.
X'10'	Ausweisinformation beim BIND ist falsch.
X'11'	Ungültiger USER beim BIND angegeben.
X'12'	KSET für Station fehlt.

<b>&amp;REA1</b>	<b>Bedeutung</b>
X'13'	USER gesperrt.
X'15'	Protokollfehler.
X'16'	Protokollfehler.
X'17'	Release Pending, d.h. der Session-Abbau ist in Bearbeitung, aber noch nicht beendet.
X'18'	Transaction recovery - kein Signon für den Benutzer
X'1C'	Passwort wird geprüft
X'1D'	Das neue Passwort ist nicht komplex genug
X'1E'	Fehler im neuen Passwort: neues Passwort = altes Passwort
X'1F'	Das Passwort wird benötigt
X'20'	KDCSIGN zur Zeit nicht erlaubt; Benutzer ist bereits angemeldet
X'21'	Es dürfen sich keine weiteren Benutzer mehr anmelden; die zulässige Anzahl (MAX CONN-USERS) ist erreicht
X'22'	Kein neues Passwort angegeben
X'23'	Gültigkeit des Passworts ist abgelaufen
X'24'	Das neue Passwort ist zu kurz
X'25'	Betriebsmittelengpass
X'26'	Die Anwendung wird beendet; SHUT WARN wurde gegeben
X'27'	Dem LTERM-Partner des Terminals fehlt der Keycode (KDCDEF-Anweisung KSET), der zur Fortsetzung des Vorgangs nötig ist
X'28'	Nach Übertragung des Passworts durch KDCUPD ist das Passwort nicht mehr komplex genug oder zu kurz
X'29'	PTERM-Name unbekannt
X'2A'	Prozessorname unbekannt
X'2B'	STATUS=OFF für dieses PTERM
X'2C'	STATUS=OFF für dieses LTERM
X'2D'	Verbindungsabbau in Bearbeitung
X'2E'	LTERM/Verbindung ist an einer anderen BCAM-Anwendung generiert
X'2F'	Prozessorname unbekannt und auch kein LTERM-Pool-Name
X'30'	Kein LTERM-Pool zu Stationstyp vorhanden oder PTERM-Name unbekannt
X'31'	Kein weiterer freier Eintrag im LTERM-Pool vorhanden
X'32'	Terminal bereits am LTERM-Pool angeschlossen
X'33'	Der LTERM-Pool ist an einer anderen BCAM-Anwendung generiert
X'34'	Der Verbindungswunsch wurde wegen contention abgelehnt

&REA1	Bedeutung
X'35'	Verbindungswunsch von VTSU abgelehnt, z.B. wegen nicht unterstütztem PTYPE oder falschem Connection-Letter
X'36'	BCAM-Letterlength zu klein
X'37'	BCAM-Returncode nicht in Ordnung

Im Insert &USER ist in einigen Fällen der im BIND-Request angegebene USER eingetragen, ansonsten enthält &USER Leerzeichen.

Die übrigen Inserts haben die gleiche Bedeutung wie bei der Meldung K115.

**K118** Global Transaction ID (Laenge, Daten): (&LTHGTRID,&GTRID)  
Atomic Action ID (Laenge, Daten): (&AAIS,&AAID)



**K119** OSI-TP Fehlerinformation: &OSLPAP, &USER, &TAC, &DIA1, &DIA2, &DIA3

Die Erklärungen in den folgenden Tabellen beziehen sich auf Protokollelemente (Primitive) an der XAP-TP Schnittstelle. Diese Datenelemente haben folgende Bedeutung:

<b>Datenelement</b>	<b>Bedeutung</b>
APM_ALLOCATE_CNF	Antwort (Confirmation) auf die Anforderung einer Association für einen OSI-TP-Dialog. Eine negative Antwort bedeutet, dass keine Association bereitgestellt werden konnte.
TP_BEGIN_DIALOGUE_REQ	Anforderung (Request) an den Partner, einen OSI-TP-Dialog zu beginnen.
TP_BEGIN_DIALOGUE_IND	Anzeige (Indication), dass der Partner einen OSI-TP-Dialog beginnen möchte.
TP_BEGIN_DIALOGUE_CNF	Antwort auf die Anforderung, einen OSI-TP-Dialog zu beginnen. Eine negative Antwort bedeutet, dass der Partner den Dialog abgelehnt hat.
TP_END_DIALOGUE_REQ	Anforderung an den Partner, den OSI-TP-Dialog zu beenden.
TP_END_DIALOGUE_IND	Anzeige, dass der Partner den OSI-TP-Dialog beenden möchte.
TP_U_ABORT_IND	Anzeige, dass der ferne TP Service User <sup>1</sup> den OSI-TP-Dialog abgebrochen hat.
TP_P_ABORT_IND	Anzeige, dass der entfernte TP Service Provider <sup>2</sup> den OSI-TP-Dialog abgebrochen hat.
TP_DATA_IND	Anzeige, dass Daten vom Partner übergeben wurden.
TP_U_ERROR_IND	Anzeige, dass der Partner einen Fehler gemeldet hat.
TP_HANDSHAKE_IND	Anzeige einer Synchronisierungs (Handshake)-Anforderung durch den Partner.
APUTM_ABORT_REQ	Anforderung zum Abbruch einer Verbindung.

<sup>1</sup> TP Service User: UTM-Anwendung

<sup>2</sup> TP Service Provider: Systemteil, der den OSI-TP-Service bereitstellt. Bei openUTM ist das der XAP-TP-Baustein.

Das Insert &DIA1 enthält die Ursache für die Ausgabe der Meldung K119.

&DIA1	Bedeutung
1	Es wurde eine negative APM_ALLOCATE_CNF empfangen.
2	Es wurde eine negative TP_BEGIN_DIALOGUE_CNF empfangen.
3	Eine TP_BEGIN_DIALOGUE_IND wurde von openUTM abgelehnt.
4	Es wurde eine TP_U_ABORT_IND bzw. eine TP_P_ABORT_IND empfangen.
5	Es wurde eine TP_U_ERROR_IND empfangen.
6	Empfang von TP_HANDSHAKE_IND für eine Dialog-Nachricht wird von openUTM nicht unterstützt.
7	Empfang von TP_END-DIALOGUE_IND vom Client für eine Dialog-Nachricht wird von openUTM nicht unterstützt, evtl. wurde bei einem asynchronen LTAC ein Dialog-RTAC generiert.
8	Bei der Functional Unit Unchained Transactions fordert UTM, dass die Transaktion mit dem Dialog beginnt.
9	Die Association wurde mit APUTM_ABORT_REQ abgelehnt.
10	Betriebsmittelengpass in openUTM

Die Inserts &DIA2 und &DIA3 enthalten je nach Wert von &DIA1 folgende Zusatzinformationen:

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
1	Verursacher der Ablehnung: 0 ACSE Service User 1 ACSE Service Provider	Diagnoseinformation: -1 Kein Grund angegeben 2 Application Context Name wird nicht unterstützt 3 Application Process Title (APT) des Auftraggebers ist unbekannt 4 Application Identifier (APID) des Auftraggebers ist unbekannt 5 Application Entity Qualifier (AEQ) des Auftraggebers ist unbekannt 6 Application Entity Identifier (AEID) des Auftraggebers ist unbekannt 7 Application Process Title (APT) des Auftragnehmers ist unbekannt 8 Application Identifier (APID) des Auftragnehmers ist unbekannt 9 Application Entity Qualifier (AEQ) des Auftragnehmers ist unbekannt 10 Application Entity Identifier (AEID) des Auftragnehmers ist unbekannt

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
1 (Forts.)	2 Presentation Service Provider	nicht relevant
	6 TP Service Provider	-1 kein Gund angegeben. 0 CCR Version 2 nicht verfügbar. 1 TP Protokoll Version nicht kompatibel 2 Keine Contention Winner Association verfügbar. Mögliche Ursache: In der Partner-Anwendung ist die Anzahl der Associations kleiner als in der lokalen UTM-Anwendung.
	7 Association Pool Manager	-1 kein Gund angegeben. 4 Association Pool Manager kann keine weitere Association aufbauen. 6 Lokaler oder ferner Application Entity Title (AET) ist unbekannt. 7 Keine passende Association für angegebenen AET gefunden.
2	Verursacher der Ablehnung:	Diagnoseinformation:
	2 TP Service Provider 3 TP Service User	0 Kein Grund angegeben. 1 TPSU title nicht bekannt. 2 TPSU title permanent nicht verfügbar. 3 Mögliche Gründe sind: <ul style="list-style-type: none"> <li>- TPSU title ist temporär nicht verfügbar.</li> <li>- In der Partner-Anwendung ist keine freie OSI-TP-Instanz-Tabelle für den ACCESS-POINT verfügbar.</li> <li>- Es ist keine freie Dialog-Tabelle verfügbar.</li> <li>- Es besteht ein anderer Betriebsmittelengpass in openUTM. Die Meldung K119 mit &amp;DIA1=10 auf Auftragnehmer-Seite enthält nähere Informationen über die Art des Engpasses.</li> </ul> Maßnahme: In der Partner-Anwendung für die OSI-LPAP-Partner des ACCESS-POINT eine größere Anzahl Associations generieren.

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
2 (Forts.)	2 TP Service Provider 3 TP Service User	4 Kein TPSU title angegeben. 5 Functional Unit nicht unterstützt. 6 Kombination von Functional Units nicht unterstützt. 7 Association reserviert für Partner. 8 Partner-AEI unbekannt. 11 Functional Unit Shared Control wird von openUTM nicht unterstützt. 12 TPSU title nicht decodierbar. 13 Typ des TPSU title nicht unterstützt. 14 Länge des TPSU title ist zu groß. 15 Decodierfehler für Benutzerdaten. 16 Decodierfehler für Security-Daten. 17 Abstrakte Syntax (UTMSEC) nicht bekannt bzw. nicht generiert. 18 Benutzerkennung unbekannt bzw. Partner-Anwendung ohne USER generiert oder Benutzerkennung abgelehnt, z.B. wegen eines falschen Passworts. 19 Partner nicht verfügbar, z.B. gesperrt (STATUS=OFF) oder wegen QUIET-Kommando. 20 Benutzerkennung oder Passwort zu lang. 21 Warnlevel 2 für den Pagepool wurde überschritten. 22 Mögliche Gründe sind: – Ungültiger Transaktionscode – Transaktionscode gesperrt 23 – Administrationsberechtigung fehlt. Transaktionscode nicht mit CALL=FIRST generiert. 24 Asynchron-Vorgang kann nicht gestartet werden. Der für den Asynchron-TAC generierte Queue Level (maximale Anzahl der Nachrichten in der Message Queue) ist bereits erreicht. 25 Dialog-Vorgang kann nicht gestartet werden, weil der Transaktionscode als Asynchron-TAC generiert ist (KDCDEF-Anweisung TAC TYPE=A). 26 Kombination von Functional Unit und Restart-Funktionalität (KDCDISP) wird nicht unterstützt.

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
3	Verursacher der Ablehnung: 3 TP Service User	Diagnoseinformation: 4 Kein TPSU title angegeben 11 Functional Unit Shared Control wird von openUTM nicht unterstützt. 12 TPSU title ist nicht decodierbar. 13 Typ des TPSU title nicht unterstützt. 14 Länge des TPSU title zu groß. 15 Decodierfehler für Benutzerdaten. 16 Decodierfehler für Security-Daten. 17 Abstrakte Syntax (UTMSEC) nicht bekannt bzw. nicht generiert. 18 Benutzerkennung unbekannt bzw. Partner-Anwendung ohne USER generiert oder Benutzerkennung abgelehnt, z.B. wegen eines falschen Passworts. Genauere Informationen können Sie der vorhergehenden Meldung K147 entnehmen. 19 Partner nicht verfügbar, z.B. gesperrt (STATUS=OFF) oder wegen QUIET-Kommando. 20 Benutzerkennung oder Passwort zu lang. 21 Warnlevel 2 für den Pagepool wurde überschritten.
4	Verursacher der Ablehnung: 2 TP Service Provider 3 TP Service User	Diagnoseinformation: 0 Kein Grund angegeben. 1 Ein permanenter Fehler ist aufgetreten. 2 TP_BEGIN_TRANSACTION_REQ abgelehnt. 3 Ein temporärer Fehler ist aufgetreten. 4 Ein Protokollfehler ist aufgetreten. 5 Kollision zweier TP_END_DIALOGUE_REQ Primitive. 6 Kollision von TP_BEGIN_TRANSACTION_REQ und TP_END_DIALOGUE_REQ.

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
9	Verursacher der Ablehnung: 3 TP Service User	Diagnoseinformation: 21 Warnlevel 2 für den Pagepool erreicht. Die Asynchron-Nachricht konnte nicht gespeichert werden. 31 Die KDCFILE ist voll: Die Nachricht konnte nicht gespeichert werden. 32 Decodierfehler für UDT-Daten eines TP_DATA_IND Primitives In allen Fällen wird die Association abgebaut.
10	Betroffenes Betriebsmittel: 11 In der Anwendung des Auftraggeber-Vorgangs ist keine freie OSI-TP-Instanz-Tabelle für diesen ACCESS-POINT verfügbar. Ursache ist eine Nachricht an einen OSI-TP-Partner, aber alle OSI-TP-Instanz-Tabellen sind belegt. Die Nachricht wird in eine Warteschlange eingereicht. Maßnahme: Für die OSI-LPAP-Partner des Access Point eine größere Anzahl Associations generieren <sup>1</sup> .	Diagnoseinformation: nicht relevant
	12 In der Anwendung des Auftraggeber-Vorgangs ist keine freie OSI-TP-Instanz-Tabelle für diesen OSI-LPAP-Partner verfügbar. Ursache ist eine Nachricht an den OSI-TP-Partner, aber alle OSI-TP-Instanz-Tabellen sind belegt. Die Nachricht wird in eine Warteschlange eingereicht. Maßnahme: Für die OSI-LPAP-Partner des Access Point eine größere Anzahl Associations generieren <sup>1</sup> .	Index OSI-LPAP

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
10 (Forts.)	13 In der Anwendung des Auftraggeber-Vorgangs ist die Instanz in XAPTP belegt. Die Nachricht wird in eine Warteschlange eingereicht. Maßnahme: Für die OSI-LPAP-Partner des Access Point eine größere Anzahl Associations generieren <sup>1</sup> .	Index UTM-Instanz
	14 In der Anwendung des Auftraggeber-Vorgangs trat ein Fehler beim Lesen der zu sendenden Asynchron-Nachricht auf. Die Nachricht wird in eine Warteschlange eingereicht.	nicht relevant
	15 Keine freie OSI-TP-Node-Tabelle verfügbar. Dieser Fehler kann sowohl einer in Anwendung eines Auftraggeber-Vorgangs als auch in einer Anwendung eines Auftragnehmer-Vorgangs auftreten. In einer Anwendung eines Auftraggeber-Vorgangs wird die Nachricht in eine Warteschlange eingereicht. Maßnahme: Für die OSI-LPAP-Partner eine größere Anzahl Associations generieren <sup>1</sup> .	nicht relevant
	16 In der Anwendung des Auftragnehmer-Vorgangs ist keine freie OSI-TP-Vorgangs-Tabelle verfügbar. Maßnahme: Für die OSI-LPAP-Partner eine größere Anzahl Associations generieren <sup>1</sup> .	nicht relevant

&DIA1	&DIA2 / Bedeutung	&DIA3 / Bedeutung
	17 In der Anwendung des Auftragnehmer-Vorgangs ist keine freie OSI-TP-Benutzer-Tabelle für den OSI-LPAP-Partner verfügbar. Mögliche Ursache: Für den OSI-LPAP-Partner ist noch ein Dialog-Teilprogramm aktiv, dessen Association zwischenzeitlich abgebaut wurde.	Index OSI-LPAP

<sup>1</sup> KDCDEF-Generierung OSI-LPAP ASSOCIATIONS=

**K120** Die Gültigkeit des Passworts ist abgelaufen - Bitte KDCSIGN

**K122** Das Passwort ist nur noch &NUMDAYS Tag(e) gültig.

**K123** LTERM hat nicht die Berechtigung den Vorgang fortzusetzen.  
- Bitte KDCSIGN

**K124** Fehler: &RCXAPTP beim Start von XAP-TP in Phase: &PHAXAPTP aufgetreten  
Die Meldung wird als Diagnosehilfe bei Fehlern während des Starts von XAP-TP ausgegeben.

Bedeutung der Inserts:

**&PHAXAPTP** Phase beim Starten von XAP-TP. Mögliche Werte:  
INIT (Übergabe der Generierungswerte an XAP-TP)  
START/RECOVERY (OPEN und BIND der Instanzen sowie Recovery nicht beendeter Transaktionen).

**&RCXAPTP** Returncode, der den Fehler genauer spezifiziert. Je nach Phase (Wert von &PHAXAPTP) haben die Returncodes unterschiedliche Bedeutungen. Sofern der Returncode nicht in der folgenden Tabelle aufgelistet ist, handelt es sich um einen internen Fehler (in diesem Falle müssen Sie eine Fehlermeldung schreiben).

&PHAXAPTP	&RCXAPTP	Bedeutung
INIT	18	Die OSS-Version ist kleiner als 4.0A20.
	20	KDCFILE und ROOT passen nicht zusammen. Mögliche Ursache: Es wurde eine neue KDCFILE mit KDCDEF erzeugt (neu: mit OSI TP, alt: ohne OSI TP); die UTM-Anwendung wurde aber nicht neu gebunden bzw. die ROOT-Source wurde nicht neu übersetzt.



&PHAXATP	&RCXATP	Bedeutung
	16 60 85	Es konnte nicht genug Speicherplatz angefordert werden, um die OSI-TP-spezifischen Generierungswerte abzudecken.
	21	Der KCSALME Aufruf zum Anlegen des Übergabepuffers für die Transfer Syntaxen liefert einen schlechten Returncode.
	41	Die „apext_init“ Sequenz wurde bereits früher durchgeführt.
	44	Der KCSALME Aufruf zum Anlegen des Übergabepuffers für die Abstrakten Syntaxen liefert einen schlechten Returncode.
	63	Der KCSALME Aufruf zum Anlegen des Übergabepuffers für die Access Points liefert einen schlechten Returncode.
	99	Eine von „apext_att“ aufgerufene Funktion hat einen schlechten Returncode geliefert. Dieser Returncode kann auch bedeuten, dass der Adressraum den OSS für sein Shared Memory in einer Folgetask belegen will, bereits mit einem anderen Shared Memory (z.B. KAA, XATP-Globalmemory) belegt ist. Maßnahme: Versuchen weitere Tasks nachzustarten.
	102	Fehler bei attach. Maßnahme: Prüfen Sie, ob auf diesem Rechner unter diesem Namen bereits eine Anwendung angemeldet ist.
	106	XATP-Aufruf „apext_att“ Returncode: APEXT_ATTACH_INVEREF Bedeutung: Die beim OSS-Aufruf o_attach angegebene Waiting Point Referenz ist ungültig bzw. nicht (mehr) bekannt. Mögliche Ursache: Folgetask kann nicht gestartet werden, da die Anwendung schon beendet wird.

**K125** Passwort nicht komplex genug - Administrator verstaendigen oder KDCOFF eingeben

Die Komplexität des Passworts ist nicht ausreichend. Es entspricht nicht der in der KDCDEF-Anweisung USER...,PROTECT-PW=(,..,stufe,..) generierten Komplexitätsstufe.

Maßnahme: Administrator muss neues Passwort vergeben

**K126** Negativer SAT-Returncode: &SATRC

Das Insert &SATRC enthält die Ursache für die Ausgabe der Meldung K126.

Die Hexa-Zeichen des Returncodes (XXSSMMMM) haben folgende Bedeutung

SS	Bedeutung
01 - 40	Systemfehler
41 - 42	SATCP nicht verfügbar
80 - 82	SAT-Funktion zur Zeit nicht ausführbar

## Zusätzliche Returncodes ab SATCP V2.0A

SS	MMMM	Bedeutung
00	0001	SAT-Protokollsatz zwischengespeichert
00	0101	SATCP in Start-Phase
00	0102	SATCP im Hold-Mode
00	0103	SATCP in Shutdown-Phase
00	0104	Ereignis nicht protokolliert
00	0105	Ereignis zurückgewiesen von Systemexitroutine #110

Falls kein Systemfehler vorliegt, sollte der BS2000-Sicherheitsbeauftragte informiert werden.

**K127** Interner Fehler in UTM - ROOT: &ERCD6

Das Insert &ERCD6 enthält die Ursache für die Ausgabe der Meldung K127.

<b>&amp;ERCD6</b>	<b>Bedeutung</b>
RT04	fehlerhafter Aktionsindex in KDCRTMM
LD01	ungültiger Lademodus eines Lademoduls
LD02	ungültiger Operationscode bei Aufruf von KDCRTL
LD03	ungültiger Ladestatus eines Lademoduls
LD04	die Area-Tabelle im Root ist überschrieben
LD05	Area in nicht geladenem Lademodul (ONCALL)

**K128** UTM-D Auftrag abgelehnt: &CON/&PRNM/&BCAP/&LPAP &LSES &REA1 &RCDC &TAC

Handelt es sich um einen OSI TP UTM-D Auftrag, haben die Inserts folgende Bedeutung:

&CON:           OSI-CON-Name  
 &PRNM:         Acht Leerzeichen  
 &BCAP          ACCESS-POINT-Name  
 &LPAP          OSI-LPAP-Name

Das Insert &REA1 enthält die Ursache für die Ausgabe der Meldung K128.

<b>&amp;REA1</b>	<b>Bedeutung</b>
X'01'	Ungültiger Transaktionscode – LU6.1: DPN oder PRN im FMH-5 oder FMH-6 oder die ersten 8 Zeichen der Nachricht – OSI TP: Recipient-tpsu-title in TP-BEGIN-DIALOGUE-RI &RCDC enthält einen Fehlercode KRCDC
X'02'	Transaktionscode nicht mit CALL=FIRST generiert – LU6.1: DPN oder PRN im FMH-5 oder FMH-6 oder die ersten 8 Zeichen der Nachricht – OSI TP: Recipient-tpsu-title in TP-BEGIN-DIALOGUE-RI
X'03'	Ein asynchroner Vorgang soll gestartet werden, TAC ist mit TYPE=D generiert – LU6.1: Nachricht mit Begin Bracket und End Bracket – OSI TP: Empfang eines TP-END-DIALOGUE-RI Protokollelements
X'04'	Ein Dialog-Vorgang soll gestartet werden, TAC ist mit TYPE=A generiert – LU6.1: Nachricht mit Begin Bracket und Change Direction – OSI TP: Empfang von TP-GRANT-CONTROL-RI oder TP-HANDSHAKE-AND-GRANT-CONTROL-RI

&REA1	Bedeutung
X'05'	Ein asynchroner Vorgang soll gestartet werden, aber der QLEV des Asynchron-TAC ist bereits erreicht. – LU6.1: Die Verbindung wird abgebaut. – OSI TP: Die Verbindung wird abgebaut
X'06'	Nur bei heterogener Kopplung über LU6.1: Eine asynchrone Nachricht wurde mit RQE empfangen. openUTM erwartet RQD2. Maßnahme: Bei CICS muss im START-Kommando NOCHECK und PROTECT angegeben werden
X'07'	Nur bei heterogener Kopplung über LU6.1: Eine Nachricht wurde mit EC und RQD2 empfangen, aber weder CD noch EB sind gesetzt. Maßnahme: CICS- bzw. IMS-Programm verbessern.

**K129** &CMD - Startparameter zur Zeit nicht erlaubt! Anweisung wird ignoriert

**K130** Task-Prioritaet &TPRIO ist fuer Task &TASK nicht erlaubt! Prioritaet nicht geaendert.

**K132** &OBJ1 und &OBJ2 stammen nicht aus dem gleichen KDCDEF-Lauf

**K133** Programm fuer &EXIT fehlt!

**K134** Nachricht an &DEST wurde in die Dead Letter Queue gestellt. &NMSG Nachricht(en) in der Dead Letter Queue

Sobald der generierte Schwellwert erreicht ist, wird der Administrator mit dieser Meldung über Nachrichten informiert, die nach abnormaler Verarbeitung in die Dead Letter Queue gestellt wurden.

Die Inserts haben folgende Bedeutung:

Insert	Bedeutung
&DEST	Ursprüngliches Ziel der Nachricht, die zum Erreichen des Schwellwertes in der Dead Letter Queue führte.
&NMSG	Generierter Schwellwert für Anzahl Nachrichten in der Dead Letter Queue

**K135** UPIC-Meldung :  
&PTRM/&PRNM/&BCAP/&LTRM/&UPCREAS/&UPCSTAT/&UPCROT/&UPVEN  
C1/&UPPENC2

Bei der Zusammenarbeit mit einem UPIC-Client ist ein Problem aufgetreten.

Der Wert des Inserts &UPCREAS zeigt die Ursache des Problems an. Die möglichen Werte von &UPCREAS und ihre Bedeutung sind in der folgenden Tabelle beschrieben.

<b>&amp;UPCREAS</b>	<b>Bedeutung</b>	<b>Ursache / Maßnahme</b>
01	UPIC-Client sendet ohne Senderecht.	Fehler in UPIC oder UTM
02	UPIC-Client sendet zu kurzes Protokoll.	Fehler in UPIC
03	UPIC-Client schlägt beim Verbindungsaufbau eine nicht unterstützte Protokoll-Version vor.	UPIC-Version passt nicht zu openUTM-Version
04	UPIC-Client sendet ungültiges Protokoll.	Fehler in UPIC
05	UPIC-Client hat Abbruch der Conversation eingeleitet.	Wirkung der UPIC-Funktion Deallocate() (CMDEAL())
06	UPIC-Client hat keinen TAC geschickt.	Fehler in UPIC oder Folgefehler nach einem unterdrückten Vorgangswiederanlauf
07	der vom UPIC-Client gesendete TAC: – ist nicht generiert – LTERM/USER hat keinen Key – ist Admin.TAC und USER ist kein Admin – existiert und Programm fehlt	Anwenderfehler (Side info oder SETTP)
08	UPIC-Client hat TAC geschickt, der mit CALL=NEXT generiert ist.	Anwenderfehler
09	Der vom UPIC-Client gesendete TAC ist gesperrt/gelöscht, oder besitzt zu geringe Verschlüsselung.	Anwenderfehler (Side info oder SETTP)
0A	UPIC-Client hat zu kurzen TAC geschickt.	Fehler in UPIC
0B	UPIC-Client hat TAC bei offenem Vorgang geschickt.	Fehler in UPIC
0C	UPIC-Client hat TAC bei Folgeteil geschickt.	Fehler in UPIC
0D	UTM-Pagepool ist zu klein für Eingabemessage.	Pagepool der Anwendung ist zu klein generiert (Anweisung MAX PGPOOL)
0E	UPIC-Client gibt bei offener Nachricht nur Senderecht ab.	Fehler in UPIC

&UPCREAS	Bedeutung	Ursache / Maßnahme
0F	UPIC-Client gibt bei offener Nachricht Senderecht ab ohne aktuellen Nachrichtenteilabschluss.	Fehler in UPIC
10	UPIC-Client hat TAC geschickt, der mit TYPE=A (Asynchron-TAC) generiert ist.	Anwenderfehler
11	UPIC-Client ist beim Verbindungsaufbau einem LTERM mit STATUS=OFF (gesperrt) zugeordnet.	Normales Verhalten
12	UPIC-Client ist beim Nachrichtempfang einem LTERM mit STATUS=OFF (gesperrt) zugeordnet.	Normales Verhalten Administrator hat LTERM gesperrt
13	UPIC-Client hat USER in ungültiger Länge geschickt.	Fehler in UPIC
14	UPIC-Client hat ungültigen USER oder PASSWORD geschickt.	Anwenderfehler
15	UPIC-Client hat ungültige Daten für Vorgangswiederanlauf geschickt, z.B. KDCDISP ohne USER oder mit zusätzlichen Anwenderdaten.	Anwenderfehler
16	UPIC-Client hat ungültige Daten für Vorgangswiederanlauf geschickt, z.B. KDCDISP für USER mit RESTART=NO.	Anwenderfehler
17	UPIC-Client hat einen USER mitgeschickt, aber die Anwendung ist ohne USER generiert.	Anwenderfehler
18	Länge des empfangenen Protokolltoken ungültig.	Fehler in UPIC
19	Das empfangene Protokolltoken ist ungültig.	Fehler in UPIC
1A	Die Gesamtlänge der empfangenen Daten ist inkonsistent.	Fehler in UPIC oder Transportsystem-Fehler
1B	Das Protokolltoken für Daten war nicht das letzte Protokolltoken.	Fehler in UPIC
1C	Es wurden zwei Protokolltoken für Service empfangen.	Fehler in UPIC
1D	Es wurden zwei Protokolltoken für Userid empfangen.	Fehler in UPIC
1E	Es wurden zwei Protokolltoken für Password empfangen.	Fehler in UPIC
1F	Es wurden zwei Protokolltoken für Format empfangen.	Fehler in UPIC

&UPCREAS	Bedeutung	Ursache / Maßnahme
20	Es wurden zwei Protokolltoken für Function Key empfangen.	Fehler in UPIC
21	Es wurden keine Protokolltoken für Daten empfangen.	Fehler in UPIC
22	Es wurden inkonsistente SIGNON-Daten empfangen.	Fehler in UPIC
23	Es wurde ein Protokolltoken für Userid empfangen, aber nicht für Service.	Fehler in UPIC
24	Ungültiger Function Key.	Fehler in UPIC
25	Protokolltoken für Format in Datenfragment empfangen.	Fehler in UPIC
26	Protokolltoken für Function Key in Datenfragment empfangen.	Fehler in UPIC
27	Ungültiger Encryption-Level.	Fehler in UPIC bzw. UTM-Fehler
28	Länge der empfangenen User Message ist ungültig.	Fehler in UPIC
29	Es ist ein Fehler bei der Protokolldiskussion aufgetreten.	Fehler in UPIC bzw. UTM-Fehler
2A	Es wurden zwei Protokolltoken für verschlüsselte Daten empfangen.	Fehler in UPIC
2B	Es wurden zwei Protokolltoken für ein verschlüsseltes Passwort empfangen.	Fehler in UPIC
2C	Es wurden zwei Protokolltoken für einen RSA Schlüssel empfangen.	Fehler in UPIC
2D	Es wurden zwei Protokolltoken für einen DES- oder AES-Schlüssel empfangen.	Fehler in UPIC
2E	Es wurden zwei Protokolltoken für einen Cursor empfangen.	Fehler in UPIC
2F	Es wurden zwei Protokolltoken für eine Protokoll Diskussion empfangen.	Fehler in UPIC
30	Es wurde kein Protokolltoken für verschlüsselte Daten empfangen.	Fehler in UPIC
31	Es wurde ein Protokolltoken für ein Passwort und ein verschlüsseltes Passwort empfangen.	Fehler in UPIC
32	Es ist ein Fehler bei der Passwortentschlüsselung aufgetreten.	Fehler in UPIC bzw. UTM-Fehler

&UPCREAS	Bedeutung	Ursache / Maßnahme
33	Das entschlüsselte Passwort ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Passwörter
34	Es ist ein Fehler beim Lesen des DES Schlüssels aufgetreten.	Fehler in UPIC bzw. UTM-Fehler
35	Der RSA Schlüssel kann nicht versendet werden.	UTM-Fehler
36	Es ist ein Fehler beim Lesen des RSA Schlüssels aufgetreten.	UTM-Fehler
37	Es ist ein Fehler bei der Datenverschlüsselung aufgetreten.	UTM-Fehler
38	Es ist ein Fehler bei der Datenentschlüsselung aufgetreten.	Fehler in UPIC bzw. UTM-Fehler
39	Die Nachricht war mit einem niedrigeren Encryption-Level verschlüsselt als für den Vorgang ausgehandelt wurde.	Fehler in UPIC
3A	Die Protokoll Diskussion konnte nicht abgeschlossen werden.	UTM-Fehler
3B	Es ist ein Fehler beim Schlüsselaustausch aufgetreten.	UTM-Fehler
3C	Der UPIC-Client unterstützt keine Verschlüsselung.	Normales Verhalten, UTM-Generierung ändern
3D	Es wurden zwei Protokolltoken für den Transaktionsstatus empfangen.	Fehler in UPIC
3E	Es wurden zwei Protokolltoken für den Client Context empfangen.	Fehler in UPIC
3F	Die Länge der Usermessage im Connectionletter stimmt nicht mit der Protokollversion überein	Fehler in UPIC
40	openUTM unterstützt keine Verschlüsselung.	Normales Verhalten, UTM-Generierung ändern
41	Zwei Protokolltoken für neues Passwort empfangen.	Fehler in UPIC
42	Zwei Protokolltoken für verschlüsseltes neues Passwort empfangen.	Fehler in UPIC
43	Protokolltoken für neues Passwort und verschlüsseltes neues Passwort empfangen.	Fehler in UPIC



&UPCREAS	Bedeutung	Ursache / Maßnahme
44	Unerlaubter Operationscode beim Aufruf der internen Funktion PASSWD_EN-CRYPT_PROC zum Entschlüsseln eines Passwortes aufgetreten.	UTM-Fehler
45	Zwei Protokolltoken für den Client Typ empfangen.	Fehler in UPIC
46	Es konnte kein RSA-Schlüssel gefunden werden.	Einen der RSA-Schlüssel aktivieren
47	UPIC-Client schlägt beim Verbindungsaufbau keine Protokoll-Version vor.	Kein UPIC-Client oder Fehler in UPIC
48	UPIC-Client sendet nicht ausgehandelte Protokoll-Version.	Fehler in UPIC
49	Das vom Client empfangene Passwort ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Passwörter
4A	Das vom Client empfangene neue Passwort ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Passwörter
4B	Das entschlüsselte neue Passwort ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Passwörter
4C	Der vom Client empfangene TAC ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere TACs
4D	Der vom Client empfangene USER ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere USER
4E	Das vom Client empfangene Format ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Formate
4F	Der vom Client empfangene Function Key ist länger als von openUTM in dieser Version unterstützt.	Fehler in UPIC oder die UPIC-Version unterstützt längere Function Keys
50	Der vom Client empfangene DES- oder AES-Schlüssel ist zu lang.	Fehler in UPIC
51	Das vom Client empfangene verschlüsselte Passwort ist zu lang.	Fehler in UPIC
52	Das vom Client empfangene verschlüsselte neue Passwort ist zu lang.	Fehler in UPIC

&UPCREAS	Bedeutung	Ursache / Maßnahme
53	Es wurden zwei Protokolltoken für GTRID empfangen.	Fehler in JConnect
54	Die vom Client empfangene GTRID ist zu lang.	Fehler in JConnect
55	Es wurde ein Protokolltoken für GTRID empfangen, obwohl noch eine Transaktion offen ist.	Fehler in JConnect

Die Inserts &UPCSTAT und &UPCROT dienen der Diagnose in Fehlerfällen.

Die Inserts &UPVENC1 und &UPPENC2 dienen der Diagnose von Datenverschlüsselung.

Das erste Byte von &UPVENC1 gibt den ENCRYPTION\_LEVEL der Message aus, das zweite Byte von &UPVENC1 gibt den ENCRYPTION\_LEVEL der Conversation aus.

Das erste Byte von &UPPENC2 gibt den ENCRYPTION\_LEVEL der Session aus, das zweite Byte von &UPPENC2 gibt den ENCRYPTION\_LEVEL des Partners aus.

#### **K136** (Erste) SYSLOG-Datei ist &FNAM

openUTM gibt diese Meldung in der Startphase aus. &FNAM enthält den Namen der SYSLOG-Datei. Ist die SYSLOG als Dateigenerationsgruppe angelegt, dann enthält &FNAM den Namen der ersten Dateigeneration, die von openUTM beschrieben wird.

#### **K137** SYSLOG umgeschaltet auf Datei &FNAM

openUTM hat erfolgreich auf eine neue SYSLOG-Dateigeneration umgeschaltet. Das Umschalten wurde durch die Administration oder durch die automatische Größenüberwachung ausgelöst. &FNAM enthält den Namen der neuen SYSLOG-Dateigeneration.

#### **K138** SYSLOG-Datei &FNAM geschlossen

Es sind zwei Fälle zu unterscheiden:

- die SYSLOG wird als einfache Datei geführt:  
Die letzte UTM-Task der Anwendung hat die SYSLOG-Datei geschlossen. &FNAM enthält den Namen der SYSLOG-Datei.
- die SYSLOG wird als SYSLOG-FGG geführt:  
Eine SYSLOG-Dateigeneration wurde von der letzten UTM-Task (also vollständig) geschlossen. Sie können jetzt frei über diese Dateigeneration verfügen. Sie wird nicht mehr von openUTM benötigt. &FNAM enthält den Namen der geschlossenen Dateigeneration.

- K139** Fehler beim Umschalten der SYSLOG-Datei! Es wird weiterhin die Datei &FNAM benutzt  
Der Versuch, auf eine neue SYSLOG-Dateigeneration umzuschalten, ist fehlgeschlagen. openUTM arbeitet weiterhin mit der Dateigeneration &FNAM. Den Grund für den Fehler beim Umschalten können Sie evtl. dem DMS-Fehlercode in der vorausgegangenen Meldung K043 entnehmen.
- K140** Im Bereich &MXP1 bis &MXP2 wird keine MUX-Protokollversion unterstuetzt
- K141** Die MUX-Protokollversion &MXP1 wird nicht unterstuetzt
- K142** Release-Pending-Timeout fuer Session. PTERM: &PTRM MUX-PTERM: &MXPT
- K143** UTM-D: STSN Sequencenumbers Response ungleich Request. Request: &STS1, &STS2 Response: &STS3, &STS4
- K144** VTSU-B Returncode &VTRC &CBRC bei Aufbereitung der asynchronen Ausgabemessage mit Editprofil oder Format &FMTN und Zeichensatz &CCSN an LTERM = &LTRM
- K145** Wegen Transaktions-Recovery kann Benutzer &USER zur Zeit nicht angemeldet werden - Bitte KDCSIGN
- K146** Erfassung BCAM Wartezeit. OPCODE= &BCMOPCD, RTCODE= &BCMRTCD, Standardheader= &STDHEAD, TSN= &TASK, BCAM-Anwendungsname= &BCAP  
Die Meldung wird erzeugt, wenn:
- openUTM die Erfassung der BCAM-Wartezeit ein- oder ausschaltet, oder
  - BCAM beim Lesen der BCAM-Wartezeit einen Fehler meldet. In diesem Fall setzt openUTM die Verarbeitung ohne Erfassung der BCAM-Wartezeit fort.

Die Werte in OPCODE und RTCODE haben folgende Bedeutung:

OPCODE	Bedeutung
00000000	Einschalten der Zeiterfassung
00000001	Ausschalten der Zeiterfassung
00000002	Wartezeit einer Nachricht lesen

RTCODE	Bedeutung
00000000	Auftrag durch BCAM ausgeführt
00000001	Funktionen zur Zeiterfassung nicht verfügbar
00000002 oder größer	BCAM meldet Fehler, Ursache in openUTM oder BCAM

**K147** Anmeldung fuer &USRTYPE User &USER nicht erfolgreich.  
&PTRM/&PRNM/&BCAP/&LTRM Grund: U&REA7

&USRTYPE enthält folgende Werte:

- CONNECTION beim Anmelden der Verbindungs-Benutzerkennung einer Verbindung zu einem UPIC- oder TS-Client
- CLIENT bei Anmeldung einer echten Benutzerkennung über TS-Anwendung, einem UPIC-Client oder einem OSI TP Partner.

Die Werte in &REA7 haben folgende Bedeutung:

&REA7	Bedeutung
U1	Der angegebene USER existiert nicht.
U2	Der angegebene USER ist gesperrt.
U3	Mit diesem USER hat sich bereits jemand angemeldet.
U4	Das als „altes“ Passwort angegebene Passwort ist falsch.
U5	Angaben zum neuen Passwort nicht verwendbar
U6	Es ist kein Kartenleser vorhanden
U7	Die Karteninformation ist falsch
U8	Die Anmeldung ist zur Zeit nicht möglich: – wegen Betriebsmittelengpass. – weil die Maximalzahl gleichzeitig anmeldbarer Benutzer bereits erreicht ist. – weil ein Passwort nicht geändert werden konnte, da gerade ein inverser KDCDEF läuft.
U9	Eine Anmeldung ist wegen fehlender Kerberos-Unterstützung nicht möglich.
U10	Der aktuelle LTERM-Partner hat nicht die Berechtigung, den Vorgang fortzusetzen.
U11	Die Gültigkeitsdauer des Passworts wurde überschritten. Der Administrator muss das Passwort ändern. Bei Anmeldung des Benutzers über eine OSI TP Association kann das Passwort nicht geändert werden (auch nicht in einer Anwendung mit Grace-Sign-On).
U12	Das neue Passwort erfüllt nicht die Anforderungen der generierten Komplexitätsstufe
U13	Das neue Passwort ist zu kurz
U14	Das von KDCUPD übertragene Passwort erfüllt nicht die Anforderungen der generierten Komplexitätsstufe oder es ist zu kurz.
U15	Für den angegebenen USER ist ein Transaktionswiederanlauf erforderlich.
U16	Der offene Vorgang kann von diesem Partnertyp aus nicht fortgesetzt werden.
U17	Vom Administrator wurde SHUT WARN eingegeben; für normale Anwender ist kein Anmelden bei der Anwendung mehr möglich (ein Administrator darf sich noch anmelden).

<b>&amp;REA7</b>	<b>Bedeutung</b>
U18	Auf der Verbindung ist der für die Fortsetzung des offenen Vorgangs nötige Verschlüsselungsmechanismus nicht verfügbar
U19	Die Gültigkeitsdauer des Passwortes ist abgelaufen. Da Grace-Sign-On generiert ist, kann die Anmeldung durch Übergabe eines neuen Passwortes wiederholt werden.
U22	Der angegebene USER existiert nicht in der Cluster-User-Datei.
U23	Mit diesem USER hat sich bereits jemand an einem anderen Knoten angemeldet.
U24	Anmeldung ist z.Z. nicht möglich, weil die Cluster-User-Datei innerhalb der generierten Zeit (Anweisung CLUSTER, Parameter FILE-LOCK-TIMER-SEC, Parameter FILE-LOCK-RETRY) nicht gesperrt werden konnte.
U25	Anmeldung an dieser Knoten-Anwendung nicht möglich, weil ein an eine andere Knoten-Anwendung gebundener Vorgang des Benutzers existiert, der nicht beendet werden darf.
U26	Anmeldung abgelehnt, da der offene Vorgang des Benutzers eine Transaktion im Zustand PTC hat, aber kein Vorgangs-Wiederanlauf angefordert wurde.

**K149** Interne Diagnoseinformation &DIA5

Es ist ein Ereignis eingetreten, das mit einer Diagnoseinformation protokolliert wird. &DIA5 ist die UTM-interne Diagnoseinformation.

**K150** Subsystem RSO meldet Fehler &RSORC &RSO2RC &RSOMSG beim Aufruf mit Operationscode &RSOOPC

Das Insert &RSOOPC beschreibt den RSO-Aufruf, bei dem der Fehler aufgetreten ist, und hat folgende Bedeutung:

<b>RSOOPC</b>	<b>Bedeutung</b>
X'01'	Reserve Printer: Drucker reservieren
X'02'	Print Message: Nachricht drucken
X'03'	Release Printer: Drucker freigeben
X'04'	Invalidate Bourse: Anwendung abmelden
X'05'	Receive Info: Informationen über Drucker bzw. Druckauftrag
X'06'	Empfangen der OLTP Schnittstellen Versionsnummer

Die Inserts &RSORC, &RSO2RC und &RSOMSG beschreiben den synchronen und asynchronen Returncode und die Fehlermeldung von RSO. Die RSO-Fehlermeldung ist im RSO-Handbuch nachzulesen.

Wenn der asynchrone Returncode ungleich X'00000000' ist, ist der Fehler bei Ausführung des vorhergehenden Auftrages aufgetreten. In diesem Fall ist der Operationscode des vorhergehenden Aufrufs in Byte 4 und der Returncode in den Bytes 5 bis 8 des Inserts RSOANNO zu finden:

X'56'	undef	undef	Opcode	R e t u r n c o d e
-------	-------	-------	--------	---------------------

Wenn der synchrone Returncode ungleich X'00000000' ist, ist der Fehler beim aktuellen Aufruf &RSOACT aufgetreten. Der Inhalt des Inserts &RSOANNO ist dann irrelevant.

Wenn sowohl der synchrone als auch der asynchrone Returncode gleich X'00000000' sind, weist die Fehlermeldung von RSO auf ein Problem hin.

Der Returncode ist aus den Bestandteilen Subcode2 (SC2, immer X'00'), Subcode 1 (SC1) und Maincode (MC) aufgebaut.

SC1	MC	Bedeutung
X'00'	X'0000'	Die Funktion wurde ohne Fehler ausgeführt.
X'01'	X'0003'	Funktionsaufruf mit ungültiger Aktion
	X'0004'	Länge des vom UTM-Teilprogramm beim FPUT/DPUT RP übergebenen RSO-Makros OLTPRQX falsch.
	X'0005'	Version des vom UTM-Teilprogramm beim FPUT/DPUT RP übergebenen RSO-Makros OLTPRQX falsch.
	X'0006'	Function/Unit des vom UTM-Teilprogramm beim FPUT/DPUT RP übergebenen RSO-Makros OLTPRQX falsch.
	X'0011'	Zeichensatzname (CCSNAME) fehlerhaft
	X'000A'	RSO-Puffergröße zu klein (siehe RSO-Kommando MODIFY-SPOOL-PARAMETER).
	X'FFFF'	Einheit oder Funktion nicht unterstützt.
X'02'	X'0006'	Drucker befindet sich im Freigabe-Prozess. Er wird demnächst verfügbar sein.
	X'FFFF'	Funktion nicht verfügbar
X'03'	X'FFFF'	Version wird nicht unterstützt.

SC1	MC	Bedeutung
X'20'	X'0001'	Fehler bei einer Speicheranforderung
	X'0003'	Dieser Drucker ist bereits durch eine andere Anwendung reserviert (ab RSO V3.2A).
	X'0004'	Dieser Drucker ist bereits reserviert
	X'0005'	Fehler beim Erzeugen des OLTP-Controller
	X'0009'	Beim Benutzen der Börsenschnittstelle trat ein Fehler auf.
	X'000F'	Fehler beim Benutzen des Memory Pool
	X'0010'	Inkonsistenz der internen Tabellen
X'40'	X'0007'	Drucker ist momentan belegt
	X'0012'	Zeichensatzname (CCSNAME) unbekannt
	X'0013'	XHCS-Fehler in RSO
	X'000B'	Fehler beim Zugriff auf den Drucker
	X'000D'	UTM hat keine Zugriffsrechte für das Gerät (siehe Kommando MODIFY-PRINTER-OUTPUT)
X'41'	X'0002'	Drucker nicht von openUTM reserviert oder nicht gestartet (siehe RSO-Kommando START-PRINTER-OUTPUT)
	X'0006'	Drucker wird gerade freigegeben, weil das Kommando STOP-PRINTER-OUTPUT gegeben wurde
	X'0008'	Drucker wurde aufgrund eines internen Fehlers freigegeben (Recovery)
	X'000C'	Drucker kann nicht reserviert werden wegen Recovery
	X'FFFF'	Subsystem RSO ist nicht geladen
X'42'	X'FFFF'	Subsystem RSO nicht konnektiert
X'80'	X'000E'	Beim Zugriff auf den Drucker trat ein vorübergehender Fehler auf
X'81'	X'FFFF'	Subsystem RSO zeitweise nicht erreichbar
X'82'	X'FFFF'	Subsystem RSO angehalten oder gestoppt

**K151** Lauf des inversen KDCDEF beendet. Returncode: &IDFRC &DMSE &FNAM

Das Insert &IDFRC enthält den 16-stelligen Returncode des inversen KDCDEF. Der Returncode setzt sich aus zwei 8-stelligen, abdruckbaren Zahlen zusammen.

Die ersten 8 Zeichen haben folgende Bedeutung:

Erste 8 Zeichen	Bedeutung
00000000	Der inverse KDCDEF wurde ordnungsgemäß beendet.
00000001	Bei der Dateibehandlung ist ein Fehler aufgetreten.
00000002	Ein Speicherengpass ist aufgetreten.
00000003	Es wurden keine KDCDEF-Anweisungen generiert.

Die Zeichen 9 - 16 sind nur relevant, wenn in den ersten 8 Stellen 00000001 steht. In diesem Fall enthalten die Zeichen 9 - 16 eine nähere Beschreibung des aufgetretenen Dateifehlers.

In allen anderen Fällen sind die Zeichen 9 - 16 mit 00000000 belegt.

Zeichen 9 - 16	Bedeutung
00000001	Der Dateiname für eine Ausgabedatei ist unzulässig.
00000002	Eine Ausgabedatei konnte nicht eingerichtet werden.
00000003	Eine Ausgabedatei konnte nicht geöffnet werden.
00000004	Eine Ausgabedatei konnte nicht geschrieben werden.
00000005	Eine Ausgabedatei existiert bereits und ist nicht vom Dateityp SAM.

Das Insert &DMSE enthält den abdruckbaren DMS-Errorcode, falls bei der Dateibearbeitung ein Fehler aufgetreten ist und in den Zeichen 9 - 16 des Inserts &IDFRC einer der Werte 00000002, 00000003 oder 00000004 steht.

Das Insert &FNAM enthält den Namen der Datei, bei der der Fehler aufgetreten ist.



**K152** Heuristikmeldung: &COND &MTYPE &OSLPAP &USER &LTAC &AAIS &AAID

Die Inserts haben folgende Bedeutung

<b>&amp;COND</b>	<b>Bedeutung</b>
MIX	Der Server hat eine Dateninkonsistenz gemeldet. Bei einer asynchronen Nachricht an eine UTM-Anwendung ist dieser Fall nicht möglich.
HAZ	Eine Dateninkonsistenz ist möglich. Die Verbindung zum OSI-TP-Auftragnehmer (Subordinate) ging verloren, nachdem ein PREPARE gesendet wurde und der OSI-TP-Auftraggeber (Superior) noch keine Datensicherung durchgeführt hat. Beim Auftraggeber wird die Transaktion zurückgesetzt, eine asynchrone Nachricht erneut gesendet. In einem UTM-Auftragnehmer wird die Transaktion ebenfalls zurückgesetzt, bzw. eine asynchrone Nachricht wird verworfen (sofort nach dem Wiederanlauf), so dass die Daten in einem konsistenten Zustand sind.

<b>&amp;MTYPE</b>	<b>Bedeutung</b>
DIAL	Es wurde eine Dialog-Nachricht gesendet.
ASYN	Es wurde eine asynchrone Nachricht gesendet.

&OSLPAP: Name der Partner-Anwendung.

&USER: Name des Benutzers, der den Auftrag erteilt hat.

&LTAC: bei einer asynchronen Nachricht: LTAC-Namen des Auftrags bei einer Dialog-Nachricht: TAC der Transaktion, für die eine Inkonsistenz möglich ist.

&AAIS: Größe des Atomic Action Identifiers in Bytes.

&AAID: enthält den encodierten Atomic Action Identifier (max. 64 Bytes).

**K153** Info fuer FHS-Meldung: &PTRM/&PRNM/&BCAP/&LTRM/&USER

Diese Meldung wird erzeugt, wenn FHS die Ausgabe einer Meldung auf SYSOUT von openUTM anfordert. Die Meldung dient der Zuordnung von Benutzer und LTERM-Partner zur FHS-Meldung. Diese Information kann bei Problemen mit OMNIS (mit und ohne MUX-Verbindungen) nötig sein, um den Verursacher des Problems zu ermitteln.

**K154** SOCKET Meldung: &PTRM/&PRNM/&BCAP/&LTRM &TCPCL &TCPRC

Dabei haben die Inserts der Meldung folgende Bedeutung:

Insert	Bedeutung
&TCPCL	interne Socket-Funktion oder internes Socket-Event
&TCPRC	Diagnostic Word

&TCPCL wird abdruckbar ausgegeben:

&TCPCL	Bedeutung
OPEN_EVENTGROUP	Eröffnen der Eventgruppe
ATTACH IPv4	Erzeugen eines Sockets für die Adressfamilie AF_INET
ATTACH IPv6	Erzeugen eines Sockets für die Adressfamilie AF_INET6
ATTACH AMBIGUOUS	Es wurde schon ein Socket mit derselben Portnummer erzeugt.
EVENT	Abholen eines Ereignisses
CONNECTION_REQ	Verbindungsaufbau anfordern
CONNECTION_RSP	Verbindungsaufbaubestätigung
DISCONNECT_REQ	Verbindung abbauen
DATA_REQ	Daten senden
DATA_IND	Daten empfangen
GET_HOST_BY_ADDR	Rechnername zu einer IPv4-Adresse bestimmen
GET_IPNODE_BY_ADDR	Rechnername zu einer IPv6-Adresse bestimmen
GET_HOST_BY_NAME	IPv4-Adresse eines Rechners bestimmen
GET_IPNODE_BY_NAME	IPv6-Adresse zu einem Rechnernamen bestimmen

Die Werte des Diagnostic Words sind 4 Bytes lang und hexadezimal angegeben. Sie entsprechen den Returncodes der „YDBSOC“ Schnittstelle.

Allgemeine Returncodes:

Folgende Returncodes können bei jedem Aufruf der TCP/IP-Funktionen auftreten.

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	01	FF	FF	Ungültige Unit bzw. Funktion im Standard-Header
00	02	FF	FF	Angeforderte Funktion nicht verfügbar
00	03	FF	FF	Ungültige Versionsnummer im Standard-Header
00	41	FF	FF	Subsystem nicht vorhanden oder nicht gestartet
00	20	00	30	Interner BCAM-Fehler
00	82	00	04	BCAM ist nicht aktiv
0C	40	02	1C	BCAM-Shutdown läuft
14	40	02	20	SOCKET von BCAM-Administrator geschlossen
00	04	00	44	Parameterliste nicht zugreifbar

&TCPCL = OPEN\_EVENTGROUP

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	OPEN-EVENTGROUP erfolgreich durchgeführt
04	80	06	1C	Kein Speicher für SUB-TCB vorhanden
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
04	80	10	1C	Kein Speicher für Lokalen Ereignisgruppen-Kontrollblock vorhanden
04	80	11	1C	Kein Speicher für Globalen Ereignisgruppen-Kontrollblock vorhanden
04	80	14	1C	Kein Speicher für Ereignisgruppennamen vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	02	51	Ereignisgruppe bereits von dieser Task eröffnet
00	01	10	51	USERID des Eigentümers der Ereignisgruppe konnte nicht ermittelt werden

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	01	11	51	Angabe von EVENTGROUP_NAME ist für einen TU Anwender nicht erlaubt

## &amp;TCPCL = CLOSE\_EVENTGROUP

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	01	00	00	CLOSE_EVENTGROUP erfolgreich durchgeführt
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	80	00	28	Ereignisgruppe kann im Moment nicht geschlossen werden (Aufruf später wiederholbar)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	04	51	Task nicht an die Ereignisgruppe angeschlossen
00	01	07	51	Ungültige EVENTGROUP_ID
00	01	13	51	Ereignisgruppe kann im Moment nicht geschlossen werden, da sie noch in Benutzung ist

&amp;TCPCL = ATTACH IPv4 oder ATTACH IPv6

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	SOCKET erfolgreich durchgeführt
00	01	33	08	Angegebene Portnummer wird bereits genutzt
04	80	04	1C	Kein Speicher für APPCB vorhanden
04	80	06	1C	Kein Speicher für SUB-TCB vorhanden
04	80	09	1C	Kein Speicher für ENACB vorhanden
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
04	80	0D	1C	Keine freie ABID vorhanden
04	80	0E	1C	Keine freie Portnummer vorhanden
04	80	13	1C	Kein Nameserver Entry vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
10	82	00	1C	Globaler Grenzwert für die Anzahl der eröffneten SOCKETS erreicht
10	01	00	20	SOCKET bereits von dieser Task eröffnet
18	40	01	20	SOCKET wird gerade zwangsweise durch den BCAM Administrator geschlossen
28	82	00	20	Tasklokaler Grenzwert für die Anzahl der eröffneten SOCKETS
00	80	00	28	SOCKET momentan nicht möglich (Aufruf später wiederholbar)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	01	50	Unbekannter Host
00	01	02	50	Host nicht aktiv
00	01	04	51	Task nicht an der Ereignisgruppe angeschlossen
00	01	07	51	Ungültige EVENTGROUP_ID
00	01	10	53	USER_ID konnte nicht ermittelt werden

## &amp;TCPCL = DETACH

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	CLOSE_SOCKET erfolgreich durchgeführt
00	01	03	08	SOCKET_ID nicht angegeben
04	08	0C	1C	Kein Speicher für Layer 4 CB vorhanden
04	01	00	20	SOCKET nicht von dieser Task eröffnet
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings

## &amp;TCPCL = EVENT

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	GET-EVENT erfolgreich durchgeführt
00	80	00	28	GET-EVENT kann im Moment nicht durchgeführt werden (Aufruf später wiederholbar)
00	01	04	51	Task nicht an die Ereignisgruppe angeschlossen
00	01	07	51	Ungültige EVENTGROUP_ID
00	80	09	51	Kein Ereignis eingetroffen

## &amp;TCPCL = CONNECTION\_REQ

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	CONNECT erfolgreich durchgeführt
00	01	03	08	SOCKET_ID nicht angegeben
04	80	02	1C	Keine freie Transportreference vorhanden
04	80	03	1C	Kein Speicher für ACONCB vorhanden
04	80	0A	1C	Kein Speicher für ADDRCP-P vorhanden
04	80	0B	1C	Keine freie CONNECTION_ID vorhanden
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
04	80	0E	1C	Keine freie Portnummer vorhanden
04	80	15	1C	Kein Speicher für EVOL vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
04	01	00	20	SOCKET nicht von dieser Task eröffnet
18	40	01	20	SOCKET wird gerade zwangsweise durch den BCAM Administrator geschlossen
20	01	00	20	CONNECT für SOCKET nicht erlaubt
24	82	00	20	Keine weitere Verbindungen für diesen SOCKET erlaubt
08	01	00	24	Verbindung bereits aufgebaut
0C	01	00	24	Verbindung wird bereits aufgebaut
18	01	00	24	Partner nicht bekannt
1C	82	00	24	Partner Prozessor nicht bekannt
1C	82	01	24	Partner Prozessor nicht aktiv
1C	82	04	24	Partner IP Adresse nicht bekannt
1C	82	06	24	Verbindungsaufbauwunsch zu Broadcastadresse
30	01	00	24	SOCKET nicht zum Verbindungsaufbau berechtigt
40	01	05	24	Geforderte Interfacefunktionalität nicht unterstützt
40	01	07	24	Interfacefunktionalität der Partner passt nicht zusammen
40	01	08	24	Level 4 Adresse nicht vorhanden
60	40	00	24	Portnummer wird bereits genutzt
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	04	51	Task nicht an die Ereignisgruppe angeschlossen
00	01	07	51	Ungültige EVENTGROUP_ID

## &amp;TCPCL = CONNECTION\_RSP

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	ACCEPT erfolgreich durchgeführt
00	01	07	08	CONNECTION_ID nicht angegeben
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
04	01	00	20	SOCKET nicht von dieser Task eröffnet
18	40	01	20	SOCKET wird gerade zwangsweise durch den BCAM Administrator geschlossen
04	01	00	24	Ungültige CONNECTION_ID
14	01	00	24	Keine CONNECTION Request anstehend
40	01	05	24	Geforderte Interfacefunktionalität nicht unterstützt
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	04	51	Task nicht an die Ereignisgruppe angeschlossen
00	01	07	51	Ungültige EVENTGROUP_ID

## &amp;TCPCL = DISCONNECT\_REQ

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	CLOSE-CONNECTION erfolgreich durchgeführt
00	01	07	08	CONNECTION_ID nicht angegeben
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
04	01	00	20	SOCKET nicht von dieser Task eröffnet
18	40	01	20	SOCKET wird gerade zwangsweise durch den BCAM Administrator geschlossen
04	01	00	24	Ungültige CONNECTION_ID
64	40	00	24	Verbindung wird bereits abgebaut
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings



&amp;TCPCL = DATA\_REQ

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	SEND erfolgreich durchgeführt
14	00	00	00	SEND erfolgreich durchgeführt, aber weiteres Senden durch Interface Flow Control gesperrt
00	01	07	08	CONNECTION_ID nicht angegeben
00	01	26	08	LENGTH_OF_DATA ungültig
00	01	13	08	User Buffer length = 0
00	01	31	08	User Buffer nicht zugreifbar
00	01	00	14	Benutzerdatenanlage zu groß
04	80	01	1C	Kein Speicher für Datenpuffer vorhanden
04	80	07	1C	Senden durch Interface Flow Control gesperrt
14	40	00	1C	Warten auf DATA_GO_INDICATION
04	01	00	20	SOCKET nicht von dieser Task eröffnet
04	01	00	24	Ungültige CONNECTION_ID
48	01	00	24	Verbindung ist nicht in der Datentransferphase (noch nicht komplett aufgebaut)
00	80	00	28	SEND kann im Moment nicht durchgeführt werden (Aufruf später wiederholen)

&amp;TCPCL = DATA\_IND

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	READ erfolgreich durchgeführt
00	01	07	08	CONNECTION_ID nicht angegeben
00	01	26	08	length_of_data ungültig
00	01	2D	08	Keine Daten vorhanden
00	01	31	08	User Buffer nicht zugreifbar
00	40	00	10	Keine Daten eingetroffen
04	01	00	20	SOCKET nicht von dieser Task eröffnet
04	01	00	24	Ungültige CONNECTION_ID
48	01	00	24	Verbindung ist nicht in der Datentransferphase (noch nicht komplett aufgebaut)
00	80	00	28	READ kann im Moment nicht durchgeführt werden (Aufruf später wiederholbar)
00	40	00	2C	Benutzerdaten verloren gegangen

&amp;TCPCL = GET\_HOST\_BY\_ADDR

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	GET_HOST_BY_ADDR erfolgreich durchgeführt
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	80	00	28	GET_HOST_BY_ADDR momentan nicht möglich (Aufruf später wiederholbar)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	01	53	Ungültige IP-Adresse

## &amp;TCPCL = GET\_HOST\_BY\_NAME

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	GET_HOST_BY_NAME erfolgreich durchgeführt
00	01	30	08	Ungültiger Socket Hostname
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	80	00	28	GET_HOST_BY_NAME momentan nicht möglich (Aufruf später wiederholbar)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings

## &amp;TCPCL = GET\_IPNODE\_BY\_ADDR

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	GET_IPNODE_BY_ADDR erfolgreich durchgeführt
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	80	00	28	GET_IPNODE_BY_ADDR momentan nicht möglich (Aufruf später wiederholen)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings
00	01	01	53	Ungültige IPv6-Adresse

&TCPCL = GET\_IPNODE\_BY\_NAME

Returncode				Bedeutung
Subcode		Maincode		
2	1	2	1	
00	00	00	00	GET_IPNODE_BY_NAME erfolgreich durchgeführt
00	01	30	08	Ungültiger Socket-Hostname
04	80	0C	1C	Kein Speicher für Layer 4 CB vorhanden
08	40	01	1C	BCAM-Shutdown angekündigt
00	80	00	28	GET_IPNODE_BY_NAME momentan nicht möglich (Aufruf später wiederholen)
00	20	01	30	System Error beim Anstarten des CONHAND processings
00	20	02	30	System Error beim Warten auf die Beendigung des CONHAND processings

**K155** Die Gültigkeit des Passworts ist abgelaufen

Bitte bisheriges Passwort eingeben

> &PAS1 <

Bitte neues Passwort eingeben

> &PAS2 <

und neues Passwort wiederholen

> &PAS3 <

**K156** UTM-UPIC Verschlüsselungskomponente &CRYP verfügbar: &RSLT

**K158** Inkonsistenter CPU-Verbrauchswert: &CPUTEXT

Bei der Berechnung der für den Benutzer aufgelaufenen CPU-Zeit ergibt sich ein inkonsistenter Wert.

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
PTRM	Name des aktuellen PTERMs.
PRNM	Name des aktuellen Prozessors.
BCAP	Name der aktuellen BCAM-Anwendung.
LTRM	Name des aktuellen LTERMs.
USER	Name des aktuellen Benutzers.
CPUTEXT	NEGATIVE: ein negativer Wert wurde beobachtet. OVERFLOW: die für den Benutzer aufgelaufene Zeit wurde zu groß.
CPUBEGIN	CPU-Verbrauchswert am Anfang des beobachteten Zeitraums.

Insert	Bedeutung
CPUEND	CPU-Verbrauchswert am Ende des beobachteten Zeitraums.
CPUUSED	Berechneter CPU-Verbrauchswert für den beobachteten Zeitraum.
CPUCLNT	Aufgelaufener CPU-Verbrauchswert für den aktuellen Benutzer.
CPUREAS	Interner Diagnosewert.

### K159 Passwort-Änderung fuer Benutzer & USER durchgeführt

Für den angegebenen Benutzer wurde das Passwort geändert. Die Meldung wird bei jeder Passwort-Änderung gesendet, unabhängig davon, wie das Passwort geändert wurde (administrativ, über SIGN CP, ...).

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
USER	Name des Benutzers, dessen Passwort geändert wurde.
ENCPW	Geändertes Passwort in verschlüsselter Form

Diese Meldung kann dazu dienen, geänderte Benutzerpasswörter in eine Standby-Anwendung zu übertragen. Dazu muss im privaten Meldungsmodul das Meldungsziel MSGTAC oder USER-DEST für diese Meldung definiert werden. Das neue Benutzerpasswort kann mittels programmierter Administration KC\_MODIFY\_OBJECT, Objekttyp USER in verschlüsselter Form durch Setzen von pw\_encrypted='Y' in der Standby-Anwendung gesetzt werden.

#### *Hinweise:*

- Es ist zu beachten, dass im MSGTAC-Teilprogramm keine UTM-D-Kommunikation erlaubt ist. Das geänderte Passwort kann somit nicht direkt aus dem MSGTAC-Teilprogramm über eine UTM-D-Verbindung an die StandBy-Anwendung gesendet werden.
- Wenn in der Produktiv-Anwendung der Anwendungslauf beendet wird, und wenn in dieser Zeit noch Passwort-Änderungen erfolgen, dann ist es möglich, dass die entsprechenden Meldungen an das MSGTAC- Programm nicht mehr empfangen werden, sondern verloren gehen.
- UTM erzeugt die neue Meldung bei jeder Art der Passwort-Änderung, also insbesondere auch dann, wenn das Passwort in der StandBy-Anwendung geändert wird. Es ist darauf zu achten, dass die StandBy-Anwendung das Passwort nicht wieder an die Produktiv-Anwendung überträgt, damit kein Ping-Pong-Effekt entsteht.

- Es ist zu beachten, dass UTM auf verschiedenen Plattformen verschiedene Algorithmen zur Verschlüsselung verwendet; somit kann die Funktion nicht plattformübergreifend eingesetzt werden.
- In UTM(BS2000) sollte auf den beteiligten Rechnern derselbe Host-Code verwendet werden, damit Sonderzeichen in den Passwörtern immer den gleichen EBCDIC-Code haben.
- Die Passwörter sollten in den beteiligten Anwendungen mit derselben Komplexitätsstufe generiert sein, weil UTM beim Eintragen des verschlüsselten Passworts die Komplexität nicht prüft.

**K160** Die &TACNTR. Transaktion des Vorgangs &TCVG wurde durch &RBCAUSER zurueckgesetzt (&RCCC/&RCDC)

Die Meldung besitzt folgende Inserts:

<b>Insert</b>	<b>Bedeutung</b>
PTRM	Name des aktuellen PTERMs
PRNM	Name des aktuellen Prozessors
BCAP	Name der aktuellen Anwendung
LTRM	Name des aktuellen LTERMs
USER	Name des aktuellen Benutzers
TCVG	Name des Vorgangs-TACs, in dem die Transaktion zurückgesetzt wurde
TAC	Name des TACs, in dem die Transaktion zurückgesetzt wurde.
TACNTR	Nummer der zurückgesetzten Transaktion innerhalb des Vorgangs

Insert	Bedeutung
RBCAUSER	<p>Verursacher des Rücksetzens. Es gibt es folgende Werte:</p> <ul style="list-style-type: none"> <li>– User PEND RS: Das Rücksetzen wurde durch einen PEND RS im Teilprogramm ausgelöst.</li> <li>– System PEND RS: Das Rücksetzen wurde durch einen von UTM initiierten PEND RS ausgelöst.</li> <li>– User PGWT RB: Das Rücksetzen wurde durch einen PGWT RB im Teilprogramm ausgelöst.</li> <li>– System PGWT RB: Das Rücksetzen wurde durch einen von UTM initiierten PGWT RB ausgelöst.</li> <li>– User PEND ER: Das Rücksetzen wurde durch einen PEND ER/FR im Teilprogramm ausgelöst.</li> <li>– System PEND ER: Das Rücksetzen wurde durch einen von UTM initiierten PEND ER ausgelöst.</li> <li>– DBSTATUS ERROR: Fehler bei der Datenbankstatus-Abfrage während der Knoten-Recovery.</li> <li>– DBSTATUS OPEN: Das Rücksetzen wurde im Warmstart wegen nicht erfolgreicher Datenbankstatus-Abfrage ausgelöst.</li> <li>– DB CONFIRM PTC: Fehler beim XA Datenbankaufruf CONFIRM PREPARE TO COMMIT STATE während der Knoten-Recovery.</li> <li>– DB CATA VTV: Fehler beim Datenbankaufruf CANCEL TRANSACTION während der Knoten-Recovery.</li> <li>– JOURNAL: Das Rücksetzen wurde im Warmstart wegen nicht vollständig geschriebenem Journalsatz ausgelöst.</li> </ul>
RCCC	Wert des kompatiblen Returncodes KCRCCC
RCDC	Wert des inkompatiblen Returncodes KCRCDC

**K161** Synchroner Periodic Write gestartet fuer Task &TASK

**K162** Lange IO fuer Task &TASK: &IOPG Seiten, &IOMS Millisekunden

**K163** Langer Periodic Write fuer Task &TASK: &IOPG Seiten, &IOMS Millisekunden

**K164** Die System-Datei SYSOUT wird umgeschaltet auf Datei &FNAM.

**K165** Die System-Datei SYSLST wird umgeschaltet auf Datei &FNAM.

**K166** Fehler &RCHX beim Umschalten der System-Datei &SYFILE auf Datei &FNAM.

**K167** Die System-Dateien SYSLST und SYSOUT wurden erfolgreich umgeschaltet.

Wurde die UTM-Anwendung im Dialog gestartet, so bestätigt die Meldung K167 das erfolgreiche Umschalten der SYSLST-Dateien für alle Tasks der Anwendung. SYSOUT bleibt unverändert den Terminals zugeordnet.

**K169** Aktion: &ACTION; Knoten: &NNM1/&HST1/&STATE; ueberwachender Knoten: &NNM2/&HST2; ueberwachter Knoten: &NNM3/&HST3

Die Meldung wird beim Anmelden und Abmelden einer Knoten-Anwendung an einem Cluster ausgegeben, sowie beim Ausfall einer Knoten-Anwendung. Sie gibt Informationen zu den Überwachungsbeziehungen in der UTM-Cluster-Anwendung aus.

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
&ACTION	beschreibt die Situation, in der die Meldung ausgegeben wird. Es enthält einen der folgenden Werte: – ADD – REMOVE – CHANGE – FAIL
&NNM1, &NNM2, &NNM3	enthalten die Knoten-Namen.
&HST1, &HST2, &HST3	enthalten die Rechnernamen.
&STATE	enthält den Status der Knoten-Anwendung, die hinzugefügt bzw. entfernt wurde. Beim Hinzufügen wird der alte Status dieser Knoten-Anwendung ausgegeben, beim Entfernen der neue Status dieser Knoten-Anwendung. Das Insert &STATE kann folgende Werte enthalten: G: generated      Knoten-Anwendung war noch nie gestartet R: registered      Knoten-Anwendung ist aktiv T: terminated      Knoten-Anwendung wurde normal beendet A: abterm          Knoten-Anwendung wurde abnormal beendet F: failure          Knoten-Anwendung als ausgefallen markiert



**K170** Cluster: Neue KDCFILE erkannt; Erzeugungszeit: &DTTM, Status: &STATE

Die Meldung wird ausgegeben, wenn anhand der Erzeugungszeit eine neu generierte KDCFILE erkannt wurde, die als Basis für zukünftige Starts von Anwendungsinstanzen definiert wird.

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
&DTTM	Generierungszeit der neuen KDCFILE
&STATE	Start-Typ der lokalen Anwendung. Das Insert &STATE kann folgende Werte enthalten: G: generated      Knoten-Anwendung war noch nie gestartet R: registered      Knoten-Anwendung ist aktiv T: terminated      Knoten-Anwendung wurde normal beendet A: abterm          Knoten-Anwendung wurde abnormal beendet F: failure          Knoten-Anwendung als ausgefallen markiert

**K171** Cluster: Knoten &HST1 gehoert zu keinem oder zu einem anderen XCS-Verbund (&XCS1,&XCS2).

Das Insert &XCS1 enthält den Namen des XCS-Verbunds von Knoten &HST1. Wenn &XCS1 Leerzeichen enthält, gehört der Knoten &HST1 keinem XCS-Verbund an.

Das Insert &XCS2 enthält den Namen des XCS-Verbunds des Clusters oder zusätzliche Informationen. Ist der XCS-Verbund nicht aktiv, dann wird in &XCS2 der XCS-Status in der Form Status: *n* zurückgegeben. Die Werte für *n* haben folgende Bedeutung:

- 1:      XCS\_INACTIVE
- 3:      XCS\_RECONF\_JOIN
- 4:      XCS\_RECONF\_LEAVE
- 5:      XCS\_RECONF\_FAIL
- 6:      XCS\_INTERRUPTED
- 7:      XCS\_SUSPENDED

Wenn &XCS2 Leerzeichen enthält, ist der Name des XCS-Verbunds noch nicht in die Cluster-Konfigurationsdatei eingetragen, da noch keine Knoten-Anwendung läuft.

**K174** Inkompatible Aenderung der Konfiguration: &DIA1

Die Konfiguration der startenden Knoten-Anwendung passt nicht zur Konfiguration der laufenden UTM-Cluster-Anwendung.

Das Insert &DIA1 beschreibt die Fehlerursache:

- Werte  $\leq 10$ : Fehler in der Laufzeit-Konfiguration
- Werte  $> 10$ : Fehler in der KDCDEF-Konfiguration

Werte des Inserts &DIA1 und deren Bedeutung:

<b>&amp;DIA1</b>	<b>Bedeutung</b>
1	Mischung von System-Protokolldatei SYSLOG als einfache Datei und als Datei-generationsgruppe FGG
2	Die Verschlüsselungsfähigkeit der startenden Knoten-Anwendung stimmt nicht mit der Verschlüsselungsfähigkeit der laufenden Knoten-Anwendungen überein
11	Mischung aus UTM-F- und UTM-S-Anwendungen (MAX APPLIMODE)
12	Mischungen aus einfacher und doppelter Dateiführung (zweiter Parameter von MAX KDCFILE, MAX USLOG)
13	Mischung aus Anwendungen mit und ohne Benutzern
14	Mischung aus Anwendungen mit und ohne erlaubter Mehrfach-Anmeldung (SIGNON MULTI-SIGNON)
15	Mischung aus Anwendungen mit und ohne Formatierungs-System
16	Die Anzahl der Datenbank-Systeme wurde verändert (DATABASE, RMXA).
17	Die Reihenfolge der Datenbank-Anweisungen (DATABASE) wurde geändert.
18	Die Passwort-Historie wurde geändert (SIGNON PW-HISTORY)
19	Die Anzahl der LSSBs wurde verändert (MAX LSSB)
20	Die Anzahl der GSSBs wurde verändert (MAX GSSB)
21	Die Anzahl der ULS wurde verändert (ULS)
22	Ein ULS der alten KDCFILE existiert in der neuen KDCFILE nicht mehr.
23	Die Maximalzahl von Vorgängen, die ein Benutzer kellern darf, wurde verringert (MAX NRCONV)
24	Die Maximalzahl von Asynchron-Vorgängen, die gleichzeitig offen sein dürfen, wurde verringert (2. Wert bei MAX ASYNTASKS)
25	Die maximale Länge der Ausweisinformation wurde verringert (MAX CARDLTH)
26	Die maximale Länge des Kerberos-Prinzipals wurde verringert (MAX PRINCIPAL-LTH)
27	Die Größe des Pagepool wurde verkleinert (1. Parameterwert bei MAX PGPOOL)
28	Die Größe des taskspezifischen Puffers für die Wiederanlaufdaten wurde verkleinert (2. Parameterwert bei MAX RECBUF)

<b>&amp;DIA1</b>	<b>Bedeutung</b>
29	Die Länge des Kommunikationsbereichs wurde verändert (MAX KB)
30	Die Länge des Standard Primären Arbeitsbereichs wurde verringert (MAX SPAB)
31	Die Länge des Nachrichtenbereichs wurde verringert (MAX NB)
32	Die maximale Länge von physikalischen Ausgabe-Nachrichten wurde verändert (MAX TRMSGLTH)
33	Die maximale Länge der Benutzerdaten in LPUT-Sätzen wurde verringert (MAX LPUTLTH)
34	Der Wert des Parameters ABORT-BOUND-SERVICE der Anweisung CLUSTER wurde geändert.

**K175** Datei &FNAM erzeugt

**K176** Prozedur &PRCN gestartet. &MSG2 RC = &RCHX

Die konfigurierte Prozedur wurde beim Erkennen eines Ausfalls einer Knoten-Anwendung gestartet.

Das Insert &PRCN enthält die gestartete Prozedur einschließlich der übergebenen Parameter.

Das Insert &MSG2 enthält zusätzliche Return-Information. Im BS2000-System enthält &MSG2 den abdruckbaren Returncode des CMD-Makros.

Werte des Inserts &RCHX1 und deren Bedeutung:

<b>&amp;RCHX1</b>	<b>Bedeutung</b>
0	Die Prozedur konnte erfolgreich gestartet werden
-1	Die Prozedur konnte nicht gestartet werden. &MSG2 enthält zusätzliche Informationen.

**K178** Cluster Journal Dateien: Aktion &ACTION; Stati (&STATE,&STA2); belegte Seiten (&PGS1,&PGS2); Umschaltvorgaenge &SWNR

Diagnoseinformation zu den beiden Dateien des Administrations-Journals wird ausgegeben.

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
&ACTION	<p>Aktion des Administrations-Journals:</p> <p><b>CHANGE</b> Die Datei des Administrations-Journals, in die aktuell nicht geschrieben wird, wurde von allen laufenden Knoten-Anwendungen vollständig eingearbeitet und eine Online-Kopie der Verwaltungsdaten wurde gezogen; die Einträge der Datei werden nicht mehr benötigt.</p> <p><b>CREATE</b> Die Dateien des Administrations-Journals wurden (neu) erzeugt. Dies erfolgt beim Start der ersten Knoten-Anwendung, wenn die Knoten-Anwendung zuvor neu generiert wurde.</p> <p><b>EXTEND</b> Die Datei des Administrations-Journals, in die aktuell geschrieben wird, wurde vergrößert, da die andere Datei des Administrations-Journals noch nicht von allen laufenden Knoten-Anwendungen eingearbeitet wurde oder die Online-Kopie der Verwaltungsdaten noch nicht abgeschlossen wurde.</p> <p><b>OPEN</b> Die Dateien des Administrations-Journals wurden geöffnet. Wird beim Start jeder UTM-Task ausgegeben.</p> <p><b>SWITCH</b> Die Datei des Administrations-Journals, in die aktuell geschrieben wird, wurde gewechselt.</p>
&STATE / &STA2	<p>Status einer Datei des Administrations-Journals: Zu einer Zeit darf nur eine Datei Status C haben, die andere Datei darf Status O oder R haben.</p> <p><b>C (Current)</b> In diese Datei wird aktuell geschrieben.</p> <p><b>O (Old)</b> Diese Datei wird noch von laufenden Knoten-Anwendungen eingearbeitet oder die Online-Kopie der Verwaltungsdaten ist noch nicht abgeschlossen.</p> <p><b>R (Reusable)</b> Die Daten dieser Datei werden nicht mehr benötigt und können nach Umschalten der Dateien des Administrations-Journals überschrieben werden.</p>
&PGS1 / &PGS2	<p>Gültige Seiten einer Datei des Administrations-Journals: Gibt die Anzahl der logisch gültigen Seiten einer Datei des Administrations-Journals aus. Nach dem Umschalten der Datei, in die aktuell geschrieben wird, enthält diese nur eine gültige Seite: Die erste Seite mit Kontrollinformation. Weitere Seiten mit alten ungültigen Journalsätzen werden nicht physikalisch freigegeben, sondern bleiben belegt.</p>
&SWNR	Anzahl der Umschaltungen des Administrations-Journals (ACTION SWITCH)

**K179** Die zu importierenden Daten wurden vollständig vom Online Import importiert

**K180** Die zu importierenden Daten wurden teilweise vom Online Import importiert

- K181** Mit der Datei &FNAM wurde noch keine Anwendung gestartet, somit sind keine Daten fuer den Online Import vorhanden.
- K182** Abnormale Beendigung des Online Import aus Datei &FNAM.
- K183** Normale Beendigung des Online Import aus Datei &FNAM.
- K184** Es ist bereits ein Online Import oder ein KDCUPD-Lauf fuer die alte KDCFILE aktiv.
- K185** Beim Online Import stimmen die Generierungszeiten der KDCFILEs nicht ueberein
- K186** Die Quell-Anwendung gehoert nicht zu dieser Cluster-Anwendung.
- K187** Start des Online Import aus Datei &FNAM.
- K188** Journal Erzeugungszeit ist unterschiedlich. In &OBJ1: &DTTM; in &OBJ2: &DTM2  
Für UTM-Cluster-Anwendungen wird diese Warnmeldung zur Diagnose ausgegeben, wenn in Dateien unterschiedliche Erzeugungszeiten für das Administrations-Journal gefunden wurden.

Die Meldung besitzt folgende Inserts:

Insert	Bedeutung
&OBJ1 / &OBJ2	Dateikennzeichen.
	jkaa   JKAA-Datei
	journal   JRN1-/JRN2-Datei
	kdcfile   KDCFILE
&DTTM / &DTM	Journal-Erzeugungszeitpunkt, der in Datei 1 bzw. Datei 2 gefunden wurde.

- K189** Anmeldung abgelehnt - Vorgangswiederanlauf in Knoten-Anwendung auf Rechner &HST1 erforderlich

**K190** Diagnosehinweis &DIA1 zu der Cluster &SUFF-Datei: &INF1: &INF2

Das Insert &SUFF bezeichnet die Datei, auf die sich die Meldung bezieht.

Das Insert &DIA1 beschreibt die Fehlerursache; die Inserts &INF1 und &INF2 geben zusätzliche Informationen.

Die Spalte **Grp.** (Gruppe) in den folgenden Tabellen beschreibt, welcher Ursachen-  
gruppe der Fehlercode angehört. Es gibt folgende Gruppen:

- A Ursache ist ein Anwenderfehler, z.B. ein Fehler beim
  - Generieren und Administrieren von UTM-Anwendungen
  - Betreiben von UTM-Anwendungen
  - Generieren des Systems (z.B. Aufteilung des Adressraums)
- U Ursache ist ein Fehler im UTM-Code.
- S Ursache ist ein Fehler in einer anderen Systemkomponente (Software oder Hardware).
- M Ursache ist ein Speicherengpass.
- I Die Meldung dient zur Information.

**K190-Meldung für die Cluster-Konfigurationsdatei**

Ist &DIA1 im Bereich 1000 bis 1099, dann ist &SUFF gleich 'CFG' und die Meldung bezieht sich auf die Cluster-Konfigurationsdatei.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-Konfigurationsdatei und deren Bedeutung, sowie Werte und Bedeutung der Inserts &INF1 und &INF2 in Abhängigkeit von &DIA1:

&DIA1	Grp.	Bedeutung	&INF1	&INF2
1001	A	APPLNAME_MISMATCH Die Anwendungsnamen in Cluster-Konfigurationsdatei und KDCFILE stimmen nicht überein.	Anwendungsname aus der Cluster-Konfigurationsdatei	Anwendungsname aus der KDCFILE
1002	A	HOSTNAME_NOT_FOUND Der eigene Rechnername wurde in der Cluster-Konfigurationsdatei nicht gefunden.	Lokaler Host-Name	Host-Name des Cluster-Knotens oder Leerzeichen
1003	A	INVALID_KDCDEF_TIME Die Erzeugungszeit der KDCFILE ist zu alt; es wurde bereits eine andere Knoten-Anwendung mit neuerer KDCFILE gestartet.	KDCDEF-Zeit aus der Cluster-Konfigurationsdatei	KDCDEF-Zeit aus der KDCFILE

&DIA1	Grp.	Bedeutung	&INF1	&INF2
1004	I	<p>OLD_STATE_INVALID (Warnung)            Der alte Status des Knotens ist "registered" und es handelt sich nicht um einen Warm-Start.</p>	<p>G generated:            Knoten war noch nie gestartet            R registered:            Knoten ist aktiv            T terminated:            Knoten wurde normal beendet            A abterm:            Knoten wurde abnormal beendet            F failure:            Knoten als ausgefallen markiert</p>	<p>D: Start nach KDCDEF-Lauf            C: Kaltstart            W: Warmstart            U: Start nach KDCUPD-Lauf</p>
1006	A	<p>CLUSTER_FILE_CHANGED            Die Cluster-Konfigurationsdatei wurde während des Anwendungslaufs neu erzeugt.</p>	<p>Generierungszeit aus der Cluster-Konfigurationsdatei oder KDCFILE-Filebase aus der Cluster-Konfigurationsdatei</p>	<p>Generierungszeit aus der KDCFILE oder KDCFILE-Filebase aus der KDCFILE</p>
1008	U	<p>NODE_NOT_REGISTERED            Die lokale Anwendung ist im KAA nicht als registriert eingetragen.</p>		
1010	AS	<p>NODE_START_SERIALIZATION            Die Sperre zur Serialisierung des Starts verschiedener Knoten-Anwendungen konnte nicht initialisiert oder nicht gesetzt werden.</p>	<p>INIT: Fehler beim Initialisieren der Sperre            LOCK: Fehler beim Setzen der Sperre</p>	<p>1: Timeout            2: Locked            3: Try later</p>
1014	A	<p>INVALID_FILE_VERSION            Die Datei-Version der Cluster-Konfigurationsdatei passt nicht zum UTM-Systemcode.</p>	<p>Erwartete Version der Cluster-Konfigurationsdatei</p>	<p>Tatsächliche Version der Cluster-Konfigurationsdatei</p>
1015	ASU	<p>CLUSTER_FILE_CORRUPTED            Die Beginn- und Ende-Marker der Cluster-Konfigurationsdatei konnten nicht gefunden werden; vermutlich handelt es sich bei der Datei um keine Cluster-Konfigurationsdatei.</p>		

&DIA1	Grp.	Bedeutung	&INF1	&INF2
1016	A	NODE_KDCFILE_MISMATCH Mit dieser KDCFILE wurde schon eine Knoten-Anwendung auf einem anderen Rechner gestartet.	Rechnername aus der Cluster-Konfigurationsdatei, auf dem mit dieser KDCFILE schon gestartet wurde.	Rechnername aus der Cluster-Konfigurationsdatei, auf dem mit dieser KDCFILE nun gestartet werden soll.
1017	A	NODE_GT_KDCFILE_START Die Knoten-Anwendung wurde zuletzt nicht mit dieser KDCFILE gestartet.	Zeitpunkt des letzten Starts mit dieser KDCFILE	Zeitpunkt des letzten Starts dieser Knoten-Anwendung.
1018	A	NODE_START_GT_KDCFILE_DEF Die neu generierte KDCFILE wurde schon vor dem Start der Knoten-Anwendung generiert.	Generierungszeitpunkt der KDCFILE	Zeitpunkt des letzten Starts dieser Knoten-Anwendung.
1019	A	NODE_NAME_OF_OWN_HOST Die Knoten-Recovery der Knoten-Anwendung des eigenen Rechners ist nicht erlaubt.		
1020	A	NODE_NAME_NOT_FOUND Knotenname für Knoten-Recovery existiert nicht in der Cluster-Konfigurationsdatei.	Knotenname in der Cluster-Konfigurationsdatei	Knotenname in der KDCFILE
1021	A	NODE_NAME_INCONSISTENCY Die Reihenfolge der Knotennamen in der Cluster-Konfigurationsdatei und in der KDCFILE stimmt nicht überein.		

### K190-Meldung für die Cluster-User-Datei

Ist &DIA1 im Bereich 1100 bis 1199, dann ist &SUFF gleich 'USER' und die Meldung bezieht sich auf die Cluster-User-Datei. Die Meldung kann in diesem Fall auch vom Dienstprogramm KDCDEF ausgegeben werden.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-User-Datei und deren Bedeutung:

&DIA1	Grp.	Bedeutung
1101	AS	TIMEOUT_FOR_USERFILE Beim Öffnen oder Schließen der Cluster-User-Datei kam es bei Anforderung der Datei-Sperre zu einem Timeout.
1102	A	NO_USER_FILE Die erste Seite der Datei ist keine CONS Seite einer Cluster-User-Datei.



&DIA1	Grp.	Bedeutung
1103	A	INVALID_FILE_VERSION Die Versionsnummer in der Cluster-User-Datei passt nicht zu der Versionsnummer im UTM-Systemcode.
1104	A	APPLNAME_MISMATCH Der Anwendungsname in der Cluster-User-Datei ist ungleich dem Anwendungsnamen aus der KDCFILE.
1105	A	GEN_TIME_MISMATCH Die Angaben in der Cluster-User-Datei und in der KDCFILE zu der Generierungszeit der Cluster-User-Datei stimmen nicht überein.
1106	A	CLUSTER_FILEBASE_MISMATCH Die Angaben in der Cluster-User-Datei und in der KDCFILE zu der Cluster Filebase stimmen nicht überein.
1107	ASU	LAST_PAGE_CORRUPTED Die letzte Seite der Datei ist keine CONS Seite einer Cluster-User-Datei
1108	U	PAGE_COUNTERS_INCONSISTENT Die Zähler zu freien und benutzten Einträgen in der Cluster-User-Datei sind inkonsistent.
1109	U	FREE_ANCHOR_INCONSISTENT Die Felder zur Verwaltung der Freikette in der Cluster-User-Datei sind inkonsistent.
1110	U	ENTRY_NOT_FOUND Beim Adressieren eines Eintrags in der Cluster-User-Datei ist ein Fehler aufgetreten.
1111	U	FREE_ANCHOR_BROKEN Die Verkettung des ersten Eintrags in der Freikette ist inkonsistent.
1112	U	USER_STATE_INVALID Der Status eines User-Eintrags in der Freikette ist ungültig.
1113	U	ENTRY_NOT_FREE Ein Eintrag in der Freikette ist nicht als frei gekennzeichnet.
1114	U	END_OF_CHAIN_NOT_FOUND Beim Durchsuchen der Freikette wurde das letzte Element nicht gefunden.
1115	U	FREE_CHAIN_BROKEN Die Anzahl der Elemente in der Freikette unterscheidet sich von dem Zähler in der Verwaltungsseite.
1116	AU	MAX_FILE_SIZE_REACHED Die Cluster-User-Datei kann nicht mehr erweitert werden, da sie bereits die maximale Anzahl von Einträgen enthält.

&DIA1	Grp.	Bedeutung
1118	A	VERSION_MISMATCH Die Cluster-User-Datei und die KDCFILE stammen von verschiedenen openUTM-Versionen.

### K190-Meldung für die Cluster-JKAA-Datei

Ist &DIA1 im Bereich 1200 bis 1399, dann ist &SUFF gleich 'JKAA' und die Meldung bezieht sich auf die Cluster-JKAA-Datei.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-JKAA-Datei und deren Bedeutung:

&DIA1	Grp.	Bedeutung
1202	AU	FILE_IS_EMPTY Die JKAA-Datei ist leer.
1203	AU	FILE_DOES_NOT_EXIST Die JKAA-Datei existiert nicht.
1204	AU	DMS_ERROR DMS-Fehler für die JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1205	M	MEMORY_INSUFFICIENT Fehler beim Allokieren von Speicher für FCB und Lese-Puffer. Maßnahme: Virtuellen Speicher vergrößern.
1206	AU	FILE_OPEN_ERROR DMS-Fehler beim Öffnen der JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1207	ASU	LOCK_INIT_TIMEOUT Timeout bei Initialisierung des Datei-Locks (KCCGFLI).
1208	SU	LOCK_INIT_ERROR Fehler bei Initialisierung des Datei-Locks (KCCGFLI).
1209	ASU	FILE_LOCK_TIMEOUT Timeout beim Anfordern des Datei-Locks (KCCGFLK).
1210	SU	FILE_LOCK_ERROR Fehler beim Anfordern des Datei-Locks (KCCGFLK).
1211	AU	INVALID_FILE_TYPE Die Datei ist keine JKAA-Datei.
1212	AU	FILE_CORRUPTED Die zweite CONS-Seite hat ein ungültiges Format.
1213	ASU	FILE_UNLOCK_TIMEOUT Timeout bei Freigabe des Datei-Locks (KCCGFLK).

&DIA1	Grp.	Bedeutung
1214	SU	FILE_UNLOCK_ERROR Fehler bei Freigabe des Datei-Locks (KCCGFLK).
1215	ASU	DESTROY_LOCK_TIMEOUT Timeout beim Zerstören des Datei-Locks (KCCGFLK).
1216	SU	DESTROY_LOCK_ERROR Fehler beim Zerstören des Datei-Locks (KCCGFLK).
1217	SU	FILE_CLOSE_ERROR DMS-Fehler beim Schließen der JKAA-Datei.
1218	AU	READ_FILE_ERROR DMS-Fehler beim Lesen der KAA-Seiten der JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1219	A	VERSION_MISMATCH Die openUTM-Versionen von KDCFILE und JKAA-Datei stimmen nicht überein. Insert &INF1 enthält die openUTM-Version der KDCFILE. Insert &INF2 enthält die openUTM-Version der JKAA-Datei.
1221	A	APPLNAME_MISMATCH Die Anwendungsnamen von KDCFILE und JKAA-Datei stimmen nicht überein. Insert &INF1 enthält den Anwendungsnamen der KDCFILE. Insert &INF2 enthält den Anwendungsnamen der JKAA-Datei.
1223	A	DATETIME_NOT_EQUAL Die Generierungszeit des KAA der JKAA-Datei stimmt nicht mit der Generierungszeit des KAA der KDCFILE überein, obwohl die startende Knoten-Anwendung dieselbe Generierungszeit hat wie die laufende UTM-Cluster-Anwendung. Insert &INF1 enthält den Generierungszeitpunkt des KAA der KDCFILE. Insert &INF2 enthält den Generierungszeitpunkt des KAA der JKAA-Datei.
1224	A	DATETIME_NOT_LATER Die Generierungszeit des KAA der KDCFILE ist nicht später als die Generierungszeit des KAA der JKAA-Datei, obwohl die startende Knoten-Anwendung neu generiert wurde. Insert &INF1 enthält den Generierungszeitpunkt des KAA der KDCFILE. Insert &INF2 enthält den Generierungszeitpunkt des KAA der JKAA-Datei.
1225	A	KCSANT_ERROR Beim Aufnehmen eines zu erzeugenden KSET in den Objektbaum der lokalen Anwendung hat die Komponente KCSANT einen schlechten Returncode geliefert.

&DIA1	Grp.	Bedeutung
1226	A	<b>OBJECT_TYPE_ERROR</b> Beim Erzeugen eines Benutzers (USER) bzw. einer LU6.1-Session in der lokalen Anwendung wurde festgestellt, dass unter dem Index in der lokalen Anwendung schon eine LU6.1-Session bzw. ein Benutzer existiert.
1301	A	<b>BLKSIZE_ERROR</b> Die Blocksize der KDCFILE ist ungleich 4K. Maßnahme: KDCFILE mit MAX BLKSIZE=4K generieren.
1302	A	<b>BUFFER_RQ_ERROR</b> Fehler beim Anfordern des Schreib-/Lese-Puffers. Maßnahme: Virtuellen Speicher vergrößern.
1303	A	<b>FCB_RQ_ERROR</b> Fehler beim Anfordern von Speicher für den FCB. Maßnahme: Virtuellen Speicher vergrößern.
1304	AU	<b>FILE_CMD_ERROR</b> DMS-Fehler beim Einrichten der JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1305	AU	<b>OPEN_ERROR</b> DMS-Fehler beim Öffnen der JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1306	U	<b>WRITE_NSR_1_ERROR</b> DMS-Fehler beim Schreiben der ersten NSR-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1307	U	<b>WRITE_NSR_ERROR</b> DMS-Fehler beim Schreiben von Folge-NSR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1308	U	<b>WRITE_NSR_END_ERROR</b> DMS-Fehler beim Schreiben des letzten Blocks NSR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1309	AU	<b>PWRT_LOCK_ERROR</b> Fehler beim Anfordern des Periodic-Write-Locks.
1310	U	<b>READ_DSR_1_ERROR</b> DMS-Fehler beim Lesen der ersten DSR-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.

&DIA1	Grp.	Bedeutung
1311	U	WRITE_SR_ERROR DMS-Fehler beim Schreiben von SR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1312	U	WRITE_SR_END_ERROR DMS-Fehler beim Schreiben des letzten Blocks SR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1313	U	READ_SR_1_ERROR DMS-Fehler beim Lesen der ersten SR-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1314	U	READ_SR_ERROR DMS-Fehler beim Lesen von Folge-SR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1315	U	SR_PAGE_TYPE_ERROR Falscher Seiten-Typ einer gelesenen SR-Seite. Insert &INF1 enthält den Typ der gelesenen Seite. Insert &INF2 enthält die Halfpage-Number.
1316	U	READ_DSR_ERROR DMS-Fehler beim Lesen von Folge-DSR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1317	U	DSR_PAGE_TYPE_ERROR Falscher Seiten-Typ einer gelesenen DSR-Seite. Insert &INF1 enthält den Typ der gelesenen Seite. Insert &INF2 enthält die Halfpage-Number.
1318	U	WRITE_DSR_ERROR DMS-Fehler beim Schreiben von DSR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1319	U	WRITE_DSR_END_ERROR DMS-Fehler beim Schreiben des letzten Blocks DSR-Seiten. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1320	U	READ_CONS_ERROR DMS-Fehler beim Lesen der ersten CONS-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.

<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1321	AU	WRITE_CONS_1_ERROR DMS-Fehler beim Schreiben der ersten CONS-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1322	AU	WRITE_CONS_2_ERROR DMS-Fehler beim Schreiben der letzten CONS-Seite. Insert &INF1 enthält den DMS-Fehlercode. Insert &INF2 enthält die Halfpage-Number.
1323	U	CLOSE_ERROR DMS-Fehler beim Schließen der JKAA-Datei. Insert &INF1 enthält den DMS-Fehlercode.
1324	SU	GFLI_ERROR Fehler bei Initialisierung des Datei-Locks KCCGFLI. Insert &INF1 enthält den Returncode von KCCGFLI.
1325	ASU	GFLI_TIMEOUT Timeout bei Initialisierung des Datei-Locks KCCGFLI. Insert &INF1 enthält den Returncode von KCCGFLI.
1326	SU	GFLK_LOCK_ERROR Fehler beim Anfordern des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1327	AU	GFLK_LOCK_TIMEOUT Timeout beim Anfordern des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1328	SU	GFLK_UNLOCK_ERROR Fehler bei Freigabe des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1329	ASU	GFLK_UNLOCK_TIMEOUT Timeout bei Freigabe des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1330	SU	GFLK_DESTROY_ERROR Fehler beim Zerstören des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1331	ASU	GFLK_DESTROY_TIMEOUT Timeout beim Zerstören des Datei-Locks KCCGFLK. Insert &INF1 enthält den Returncode von KCCGFLK.
1332	U	JFCT_START_ERROR Fehler beim Schreiben des Copy-Status (Started) in die Journaldatei.
1333	U	JFCT_END_ERROR Fehler beim Schreiben des Copy-Status (Completed) in die Journaldatei.

**K190-Meldung für die Dateien des Administrations-Journals**

Ist &DIA1 im Bereich 1400 bis 1499, dann ist &SUFF gleich 'JRN' und die Meldung bezieht sich auf die Dateien des Administrations-Journals.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Dateien des Administrations-Journals und deren Bedeutung:

<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1401	AU	Eine oder beide Journaldateien existieren nicht. Siehe vorherige K043-Meldung(en) Maßnahme: Anwendungen neu generieren.
1402	AU	Fehler beim Öffnen einer Journaldatei. Siehe vorherige K043-Meldung.
1403	AU	Fehler beim Erzeugen einer Journaldatei. Siehe vorherige K043-Meldung.
1404	SU	Fehler beim Einrichten der Sperre für die Journaldateien.
1405	SU	Fehler beim Schließen einer Journaldatei. Siehe vorherige K043-Meldung.
1406	ASU	Fehler beim Freigeben reservierter Seiten einer Journaldatei. Siehe vorherige K043-Meldung.
1407	A	Fehlerhafte Journaldatei(en): Falsches Kennzeichen. &INF1/&INF2: Kennzeichen in Datei mit Suffix JRN1/JRN2. Maßnahme: Dateien löschen, Anwendungen neu generieren.
1408	A	Fehlerhafte Journaldatei(en): Falsches Datei-Kennzeichen. &INF1/&INF2: Kennzeichen in Datei mit Suffix JRN1/JRN2.
1409	A	Fehlerhafte Journaldatei(en): Falscher UTM Anwendungsname. &INF1/&INF2: Anwendungsname in JRN1-Datei / KDCFILE.
1410	A	Fehlerhafte Journaldatei(en): Falscher UTM Anwendungsname. &INF1/&INF2: Anwendungsname in JRN2-Datei / KDCFILE.
1411	A	Fehlerhafte Journaldatei(en): Falsche Cluster-Filebase. &INF1/&INF2: Cluster-Filebase in JRN1-Datei / KDCFILE.
1412	A	Fehlerhafte Journaldatei(en): Falsche Cluster-Filebase. &INF1/&INF2: Cluster-Filebase in JRN2-Datei / KDCFILE.
1413	A	Fehlerhafte Journaldatei(en): Erzeugungszeit verschieden. &INF1/&INF2: Erzeugungszeit Datei mit Suffix JRN1/JRN2.
1414	AU	Fehlerhafte Journaldatei(en): Datei-Status ungültig. &INF1/&INF2: Datei-Status in Datei mit Suffix JRN1/JRN2.
1415	SU	Fehler beim Zerstören der Sperre für die Journaldateien.
1416	AU	Fehlerhafte Journaldatei(en): Falsche Dateisequenznummern. &INF1/&INF2: Dateisequenznummer in JRN1/JRN2-Datei.

&DIA1	Grp.	Bedeutung
1417	A	Fehlerhafte Journaldatei(en): Falsche Dateiversion. &INF1/&INF2: Version der JRN1-Datei / erwartete Version.
1418	A	Fehlerhafte Journaldatei(en): Falsche Dateiversion. &INF1/&INF2: Version der JRN2-Datei / erwartete Version.

### K190-Meldung für die Verwaltungsdatei des Cluster-Pagepools

Ist &DIA1 im Bereich 1500 bis 1599, dann ist &SUFF gleich 'CPMD' und die Meldung bezieht sich auf die Verwaltungsdatei des Cluster-Pagepools.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Verwaltungsdatei des Cluster-Pagepools und deren Bedeutung:

&DIA1	Grp.	Bedeutung
1501	A	Falsche Dateiversion der Verwaltungsdatei. &INF1/&INF2: Version der CPMD-Datei / erwartete Version Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen.
1502	A	Ein Cluster-Update mit Übertragung in die CPMD-Datei wurde abnormal beendet. Maßnahme: Cluster-Konfigurationsdateien neu erzeugen.
1503	A	Die Größe des Cluster-Pagepools wurde verkleinert. &INF1/&INF2: Größe in CPMD-Datei / Größe in KDCFILE. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen.
1504	A	Die Anzahl der Dateien des Cluster-Pagepools wurde geändert. &INF1/&INF2: Anzahl in CPMD-Datei / Anzahl in KDCFILE. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen.
1505	A	Fehler beim Versuch, die Dateien des Cluster-Pagepools zu vergrößern. &INF1: 4K Seitennummer, die nicht geschrieben werden konnte. &INF2: Generierte Größe des Cluster-Pagepools in 4K Seiten. Maßnahme: Genügend Plattenspeicherplatz für UTM-Cluster Dateien zur Verfügung stellen.
1506	AU	Ungültige Größe der Freiseiten-Kontingente des Cluster-Pagepools. &INF1: Kontingentgröße in Anzahl UTM-Seiten. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen oder Problem-meldung schreiben.
1507	AU	Ungültige Anzahl von Seiten der Freiseiten-Kontingente des Cluster-Pagepools. &INF1: Anzahl der Seiten aller Freiseiten-Kontingente. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen oder Problem-meldung schreiben.



<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1508	AU	Fehlerhafte Verwaltung der Freiseiten-Kontingente des Cluster-Pagepools. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen oder Problem-meldung schreiben.
1509	AU	Ungültiges Kennzeichen zur Datei-Vergrößerung des Cluster-Pagepools. &INF1: Kennzeichen zur Datei-Vergrößerung. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen oder Problem-meldung schreiben.
1510	AU	Ungültiges Kennzeichen zum Lauf eines Cluster-Updates mit Übertragung in die CPMD-Datei. &INF1: Kennzeichen zum Lauf des Cluster-Updates. Maßnahme: Alle UTM-Cluster-Dateien neu erzeugen oder Problem-meldung schreiben.
1511	A	MAX APPLIMODE der Knoten-Anwendung wurde geändert. Für einen Wechsel zwischen Secure und Fast müssen alle UTM-Cluster-Dateien neu erzeugt werden. &INF1/&INF2 : APPLIMODE in CPMD-Datei / in KDCFILE
1512	A	MAX VGMSIZE der Knoten-Anwendung wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : VGMSIZE in CPMD-Datei / in KDCFILE
1513	A	MAX KB der Knoten-Anwendung wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : KB in CPMD-Datei / in KDCFILE
1514	A	MAX NB der Knoten-Anwendung wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : NB in CPMD-Datei / in KDCFILE
1515	A	MAX LSSBS der Knoten-Anwendung wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : LSSBS in CPMD-Datei / in KDCFILE
1516	A	MAX TRMSGLTH der Knotenanwendung wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : TRMSGLTH in CPMD-Datei/KDCFILE
1517	A	Die Anzahl der generierten Datenbanken wurde geändert. Alle UTM-Cluster-Dateien müssen neu erzeugt werden. &INF1/&INF2 : Anzahl in CPMD-Datei/KDCFILE
1518	A	Unzulässige Mischung mit und ohne Benutzerkennungen. Alle UTM-Cluster-Dateien müssen neu erzeugt werden.
1519	A	Unzulässige Mischung mit und ohne Formatierungssystem. Alle UTM-Cluster-Dateien müssen neu erzeugt werden.

<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1520	A	Änderung von CLUSTER ABORT-BOUND-SERVICE. Alle UTM-Cluster-Dateien müssen neu erzeugt werden.
1521	A	Änderung der generierten Datenbanksysteme. Alle UTM-Cluster-Dateien müssen neu erzeugt werden.

### **K190-Meldung für die Cluster-ULS-Datei**

Ist &DIA1 im Bereich 1600 bis 1699, dann ist &SUFF gleich 'ULS' und die Meldung bezieht sich auf die Cluster-ULS-Datei, in der in UTM-Cluster-Anwendungen die Verwaltungsdaten zu ULS-Bereichen gehalten werden.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-ULS-Datei und deren Bedeutung:

<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1601	AU	Die Datei ist keine Cluster-ULS-Datei oder die Datei ist korrupt.
1602	A	Die Cluster-ULS-Datei hat eine ungültige Dateiversion.
1603	A	Die Anwendungsnamen aus Cluster-ULS-Datei und KAA stimmen nicht überein.
1604	AU	Die letzte Seite der Cluster-ULS-Datei ist zerstört.
1605	U	UTM-Generierungsfehler: Die Anzahl der ULS-Blöcke aus der Cluster-ULS-Datei stimmt nicht überein mit der Anzahl der ULS-Blöcke in der KDCFILE.
1606	U	UTM-Generierungsfehler: Ein ULS aus der Cluster-ULS-Datei fehlt in der KDCFILE.
1607	ASU	Beim Anfordern einer Sperre auf die Cluster-ULS-Datei kam es zu einem Timeout.

**K190-Meldung für die Cluster-Lock-Datei**

Ist &DIA1 im Bereich 1700 bis 1799, dann ist &SUFF gleich 'LOCK' und die Meldung bezieht sich auf die Cluster-Lock-Datei für die Sperren von globalen Betriebsmitteln in UTM-Cluster-Anwendungen (GSSB und ULS).

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-Lock-Datei und deren Bedeutung:

<b>&amp;DIA1</b>	<b>Grp.</b>	<b>Bedeutung</b>
1700	AU	Beim Öffnen der bestehenden Cluster-Lock-Datei ist ein Fehler aufgetreten. Siehe vorherige K043-Meldung.
1701	AU	Beim Anlegen der Cluster-Lock-Datei ist ein Fehler aufgetreten. Siehe vorherige K043-Meldung.
1702	SU	Fehler beim Schreiben der Cluster-Lock-Datei. Siehe vorherige K043-Meldung.
1703	SU	Fehler beim Schließen der Cluster-Lock-Datei. Siehe vorherige K043-Meldung.
1704	SU	Fehler beim Öffnen der Cluster-Lock-Datei. Siehe vorherige K043-Meldung.
1705	SU	Fehler beim Ersetzen der Cluster-Lock-Datei. Siehe vorherige K043-Meldung.
1706	I	Informationsmeldung - Cluster-Lock-Datei wurde erweitert. Es wird die Anzahl der neuen Seiten ausgegeben..
1707	SU	Fehler beim Schließen der Cluster-Lock-Datei. Siehe vorherige K043-Meldung.

### K190-Meldung für die Cluster-GSSB-Datei

Ist &DIA1 im Bereich 1800 bis 1899, dann ist &SUFF gleich 'GSSB' und die Meldung bezieht sich auf die Cluster-GSSB-Datei, in der in UTM-Cluster-Anwendungen die Verwaltungsdaten für GSSBs gehalten werden.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 für die Cluster-GSSB-Datei und deren Bedeutung, sowie Werte und Bedeutung der Inserts &INF1 und &INF2 in Abhängigkeit von &DIA1:

&DIA1	Grp.	Bedeutung	&INF1	&INF2
1800	ASU	GSSBFILE_LOCK_ERROR Beim Anfordern oder Aufheben einer Sperre auf die Cluster-GSSB-Datei kam es zu einem Fehler oder Timeout.	LOCK: Fehler beim Anfordern der Sperre UNLOCK: Fehler beim Freigeben der Sperre	TIMEOUT: Timeout (Returncode vom KCSGFLK) TRY LATER: Try later (Returncode vom KCSGFLK) LOCKED: Datei ist bereits vom eigenen Task gesperrt NOT LOCKED: Datei ist nicht gesperrt
1801	A	INVALID_FILE_VERSION Die Cluster-GSSB-Datei hat eine ungültige Dateiversion.	Dateiversion in der GSSB-Datei	Erwartete Dateiversion
1802	AU	HEADER_PAGE_CORRUPTED Die Verwaltungsseite der Cluster-GSSB-Datei ist zerstört.	Fehlerhafter Feldinhalt in der GSSB-Datei	Erwarteter Feldinhalt
1803	AU	FREE_PAGES_ANCHOR_CORRUPTED Die Freiseitenverkettung der Cluster-GSSB-Datei ist korrupt.		
1804	AU	NO_GSSBFILE Die Datei ist keine Cluster-GSSB-Datei oder die Datei ist korrupt.	Dateityp auf der CONS-Seite der GSSB-Datei	Erwarteter Dateityp
1805	AU	PAGE_COUNTER_INCONSISTENT Die Seitenzähler der Cluster-GSSB-Datei sind korrupt.		
1806	AU	LAST_PAGE_INCONSISTENCY Die Dateigröße auf CONS- und Verwaltungsseite der Cluster-GSSB-Datei stimmen nicht überein.		
1808	SU	CONS_PAGE_READ_ERROR Fehler beim Lesen der ersten CONS-Seite der Cluster-GSSB-Datei.		

&DIA1	Grp.	Bedeutung	&INF1	&INF2
1809	SU	CONS_PAGE_WRITE_ERROR Fehler beim Schreiben einer CONS-Seite der Cluster-GSSB-Datei.		
1810	SU	READ_ERROR_GSSBFILE Fehler beim Lesen einer Seite der Cluster-GSSB-Datei.		
1811	SU	WRITE_ERROR_GSSBFILE Fehler beim Schreiben einer Seite der Cluster-GSSB-Datei.		
1812	U	RELEASE_BUFFER_ERROR Fehler beim Freigeben einer Seite der Cluster-GSSB-Datei.		
1814	U	FREE_ENTRY_CHAIN_CORRUPTED Die Verkettung der freien GSSB-Einträge in der Cluster-GSSB-Datei ist korrupt.		
1815	U	HASH_CHAIN_CORRUPTED Die Hash-Verkettung der GSSB-Einträge in der Cluster-GSSB-Datei ist korrupt.		
1816	U	FREE_PAGE_CHAIN_CORRUPTED Die Freiseitenverkettung in der Cluster-GSSB-Datei ist korrupt.		
1817	U	GSSB_CNTR_ERROR Der Zähler über die belegten GSSB-Einträge in der Cluster-GSSB-Datei ist korrupt.		
1818	AU	MAX_GSSB_ERROR Der Wert von MAX GSSB in der KDCFILE stimmt nicht mit dem Wert in der Cluster-GSSB-Datei überein.	MAX_GSSB in der GSSB-Datei	MAX_GSSB in der KDCFILE

### K190-Meldung für UTM-Cluster-Dateien

Ist &DIA1 im Bereich 1900 bis 1949, so bezieht sich die Meldung auf die Anforderung einer Sperre für eine UTM-Cluster-Datei und dient nur zur Diagnose.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 und deren Bedeutung.

&DIA1	Grp.	Bedeutung
1900	I	Die Lock-Anforderung konnte in der generierten Zeit (Operand FILE-LOCK-TIMER-SEC in der CLUSTER-Anweisung) nicht durchgeführt werden. Die Lock-Anforderung wird in Abhängigkeit von dem im Operanden FILE-LOCK-RETRY in der CLUSTER-Anweisung generierten Wert wiederholt.  &INF1: interner Fehlercode der Lock-Anforderung. &INF2: Entweder der Status des XCS-Verbundes oder FILE LOCKED, wenn Sperre gehalten wird, siehe Tabelle unten.
1901	I	Diese Meldung kommt bei Freigabe des Datei-Locks, wenn die Sperre von dieser Task mindestens so lange wie die halbe generierte Zeit des im Operanden FILE-LOCK-TIMER-SEC in der CLUSTER-Anweisung generierten Wertes gehalten wurde.  &INF1: Status der angeforderten Sperre. &INF2: Lock-Haltedauer der Task in Sekunden.

&SUFF gibt das Suffix der UTM-Cluster-Datei an, für die die Sperre angefordert wurde. Ausnahme: Der Wert SLCK steht nicht für eine Datei, sondern für die logische Sperre, um die Starts der Knoten-Anwendungen zu serialisieren.

Die folgende Tabelle gibt für &DIA1=1900 eine Übersicht über die Werte des Inserts &INF2 und deren Bedeutung.

&INF2	Bedeutung
FILE LOCKED	Die UTM-Cluster-Datei ist zur Zeit von einer anderen Task gesperrt.
1	XCS_INACTIVE
2	XCS_ACTIVE
3	XCS_RECONF_JOIN
4	XCS_RECONF_LEAVE
5	XCS_RECONF_FAIL
6	XCS_INTERRUPTED
7	XCS_SUSPENDED

### K190-Meldung für Ausfall einer Knoten-Anwendung

Ist &DIA1 im Bereich 2001 bis 2099, so ist &SUFF gleich 'KDCA' und die Meldung bezieht sich auf die KDCFILE einer Knoten-Anwendung. Diese Meldung wird ausgegeben, wenn der Ausfall eines Knotens festgestellt wird.

Die folgende Tabelle gibt eine Übersicht über die Werte des Inserts &DIA1 und deren Bedeutung:

&DIA1	Grp.	Bedeutung
2001	I	Die Anwendung ist nicht aktiv.
2002	S	Beim Anfordern von Speicher kam es zu einem Fehler.
2003	ASU	Vom Dateisystem wurde ein Fehler gemeldet; siehe auch K043-Meldung
2004	SU	Beim Initialisieren des Locks kam es zu einem Fehler.
2005	U	Der Knoten konnte in der UTM-Cluster-Datei nicht gefunden werden

&INF1 enthält die Knotennummer des Knotens, dessen Ausfall erkannt wurde.

&INF2 enthält den Dateinamen der KDCFILE des Knotens, dessen Ausfall erkannt wurde.

**K191** Dump wird ohne &SUFF-Datei geschrieben, da nicht genügend Speicher (&RQM Bytes) allokiert werden konnte.

**K192** Knoten-Recovery fuer Knoten &NNM1 auf Rechner &HST1, RESET-PTC=&RSPTC

Diese Meldung wird zu Beginn einer Knoten-Recovery auf SYSOUT und SYSLST ausgegeben.

**K193** PTC gefunden : ID=&PTCID, USER=&USER, LPAP=&LPAP, LSES=&LSES, USER-Typ=&USTYPPTC

Diese Meldung wird bei einer Knoten-Recovery für jede Transaktion im Status PTC einmal ausgegeben.

**K194** Nach Knoten-Recovery: Anzahl noch gesperrter GSSB: &GBLNBR, Anzahl noch gesperrter ULS: &ULLNBR

Diese Meldung wird am Ende der Knoten-Recovery ausgegeben.

Hält die ausgefallene Knoten-Anwendung, für die eine Knoten-Recovery durchgeführt wurde, noch Sperren auf GSSBs, kann dies die laufende UTM-Cluster-Anwendung beeinträchtigen.

**K199** ROOT-Beendigung eingeleitet mit TERM-UNIT: &TRMTYP, Ursache: &ROOTTRMR

Die Ausgabe der Meldung K199 dient zu Diagnosezwecken. In einigen Fällen enthalten vorangegangene Meldungen zusätzliche Informationen.

Die Inserts haben folgende Bedeutung:

**&TRMTYP** bezeichnet die Art der Task-Beendigung:

PROG: Anwendungsprogramm soll nachgeladen werden.

STEP: Die Task soll beendet werden.

**&ROOTTRMR** Ursache für Beendigung. Folgende Werte sind möglich:

<b>&amp;ROOTTRMR</b>	<b>Bedeutung</b>
01	Dynamischer ROOT generiert aber keine Parameter dazu angegeben
02	Bindefehler beim Nachladen des ROOT-Moduls
03	Ungültiges/falsches ROOT-Modul eingebunden
04	Falsche Version des ROOT-Moduls
06	Bindefehler beim Laden des Meldungsmoduls
07	Falsche Version des Meldungsmoduls
10	Aufruf zum Ermitteln Task-Info liefert Fehler
11	Fehler beim Anfordern Speicher für Startparameter
12	Keine DB generiert aber ORACLE-Startparameter
13	Cluster-Filebase und Filebase gleichzeitig angegeben
14	Cluster-file und CATID angegeben
15	Catid-Parameter und Filebase enthält einen “:“
16	STARTNAME- und ENTER-PROC-Parameter gleichzeitig angegeben
17	Maximale Länge für ENTER-PROC-Parameter überschritten
18	Fehler beim Einlesen der Startparameter
20	Fehler bei Anfordern Speicher für UTM Diagarea
21	Fehler bei Anfordern Speicher für DB Diagarea
25	Ein STXIT-Ereignis ist aufgetreten
26	KDCTRMA wurde aufgerufen
27	Fehler beim Initialisieren der Sprachumgebung
28	Fehler beim Deaktivieren der Sprachumgebung
30	Bindefehler für das FORMCON-Modul
31	Ungültiger Opcode bei Aufruf KDCRTFH



&ROOTTRMR	Bedeutung
32	Fehler bei Aufruf Format-Startparameter
33	Fehler bei Initialisierung der Formatierung
40	Normales Anwendungs-Ende
41	KDCS-Aufruf in SHUT-Exit
43	Fehler beim Aufheben Schreibschutz (ROOTDATA)
44	Fehler beim Setzen Schreibschutz (ROOTDATA)
45	Programm entladen (Austauschen)
46	Fehlerhafte ROOT-Aktions-Tabelle
47	Fehler beim Schließen der STXIT
48	Fehler beim Anfordern Puffer für LOGGING-Exit
49	Fehler beim Nachinitialisieren des Sprachanschlusses
50	Fehler beim Aufruf START-Exit
51	Fehler beim Starten der Überwachung der Programmlaufzeit
52	Fehler beim Laden des Anwendungsprogramms
53	Fehler beim Austausch des Anwendungsprogramms
54	Falscher Index beim Laden eines Lademoduls
55	Ungültiger Index für ROOT-Aktions-Tabelle
56	Ungültige ROOT-Aktions-Tabelle
57	Fehler beim Aufruf START-Exit
58	Fehler bei Teilprogramm-Beendigung
59	SVC-Fehler
69	Falscher DB-Parameter
70	Fehler bzgl. der Startparameter für das Datenbanksystem
71	UTM-Fehler bei der Behandlung der Startparameter für das Datenbanksystem
72	Datenbank-Fehler beim Starten einer Transaktion
73	Datenbank Aufruf in INPUT-Exit
74	Datenbank Aufruf in START-Exit
75	Fehlerhafter Opcode in KDCRTDB
76	Timer-STXIT während Datenbank-Aufruf
77	Fehler beim Anfordern von Speicher für Transaktionsverwaltung
78	Datenbank nicht aktiv oder Datenbank Admin-Fehler
79	Fehlerhaftes UTM-Verhalten beim Anmelden an die Datenbank

<b>&amp;ROOTTRMR</b>	<b>Bedeutung</b>
80	Canceln einer Datenbank Transaktion aus einer STXIT Behandlung
81	Fehler beim Anmelden an die Datenbank
82	Fehler bei Get install Path für BIND (DBCON)
83	Fehler beim Laden des Verbindungsbausteins zur Datenbank (wird wiederholt)
84	Fehler bei Get Install Path DB-Modlib
85	Falsche Version des Verbindungsbausteins zur Datenbank
86	Fehler bei Datenbankkopplung über XA (mismatch)
87	Fehler beim Nachladen des XA-Anschluss-Bausteins
88	Das Datenbank-Verbindungsmodul wurde falsch geladen (oberer statt unterer Adressraum).
90	PEND ER aufgerufen vom Anwendercode / UTM-Systemcode
91	Verbindungsverlust bei PGWT
95	KCSSTRT-Error-Routine aktiviert

**K201** XA( &TSNPID) Resource Manager Anbindung ueber X/Open &XASPEC

&XASPEC bezeichnet die Version des XA-Anschlusses. Das Insert kann die Werte XA-CAE-Spec oder XA-P-Spec annehmen.

**K202** XA( &TSNPID) RM-Instanz &INSTNUM ,&TEXT32, &RMSTAT

Die Meldung informiert über den Status der XA-Verbindung.

Die Inserts haben folgende Bedeutung:

**&INSTNUM** Laufende Nummer der Resource Manager-Instanz, beginnend mit 1. Die Nummerierung entspricht der Reihenfolge der Startparameter.

**&TEXT32** Name des Resource Managers.

**&RMSTAT** Status der Verbindung. Folgende Werte sind möglich:

<b>&amp;RMSTAT</b>	<b>Bedeutung</b>
opened	Der Resource Manager wurde geöffnet
reopened	Der Resource Manager wurde nach einem Wiederanlauf geöffnet
closed	Der Resource Manager wurde geschlossen

**K203** XA( &TSNPID) RM &TEXT32,&INSTNUM; Wiederhergestellte Transaktionen:&RTAANZ

Die Inserts haben folgende Bedeutung:

&TEXT32 Name des Resource Managers.

&INSTNUM Laufende Nummer der Resource Manager-Instanz, beginnend mit 1. Die Nummerierung entspricht der Reihenfolge der RMXA-Startparameter.

&RTAANZ Anzahl der wiederhergestellten Transaktionen.

**K204** XA( &TSNPID) Precommit erfordert generelles Ruecksetzen; Ursache:&XATXT  
TA=&INTTAID

Precommit ist fehlgeschlagen, die globale Transaktion wird zurückgesetzt.

Die Inserts haben folgende Bedeutung:

&XATXT Grund für das Fehlschlagen des Precommit.

Folgende Werte sind möglich:

<b>&amp;XATXT</b>	<b>Bedeutung</b>
XA_RBROLLBACK	Rollback aus nicht näher spezifiziertem Grund
XA_RBCOMMFAIL	Rollback wegen eines internen Kommunikationsfehlers im Resource Manager
XA_RBDEADLOCK	Rollback wegen Deadlocks
XA_RBINTEGRITY	Rollback wegen Ressourcen-Inkonsistenz
XA_RBOTHER	Rollback aus nicht näher spezifiziertem Grund
XA_RBPROTO	Rollback wegen eines internen Protokollfehlers beim Resource Manager
XA_RBTIMEOUT	Rollback wegen Zeitüberschreitung der Transaktionsdauer
XA_RBTRANSIENT	Rollback wegen eines vorübergehenden Fehlers

&INTTAID Beschreibung der UTM-Transaktion, die das globale Rücksetzen der anderen Transaktionen ausgelöst hat.

Byte 0 enthält die Instanznummer des Resource Managers.

Wenn die XID (= XA-Transaktionsbezeichner), in einer OSI-TP-Auftragnehmer-Transaktion angelegt wurde, folgt hier der AAID (Atomic Action Identifier).

Anderenfalls enthalten die nachfolgenden Bytes:

Byte	Bedeutung
1 - 8	Hostname
9 - 16	Anwendungsname
17	Typ der BS2000-Hardware-Plattform (7 ≙ /390, 8 ≙ RISC, 9 ≙ SPARC)
18	Bit-Modus (1 ≙ 32-Bit-Plattform)
19	Byte-Format (2 ≙ Bigendian)
20	Nummer der openUTM-Session
21 - 22	Nummer der Transaktion innerhalb des Vorgangs
23 - 24	reserviertes Feld
25 - 28	Vorgangsnummer
29 - 32	Vorgangsindex

**K205** XA( &TSNPID) Transaktion vorgesetzt; Ursache: &XATXT  
TA=&INTTAID

Die Inserts haben folgende Bedeutung:

&XATXT Grund für das Vorsetzen der Transaktion (Commitment). Mögliche Gründe: „Recovery“ oder „Int.Event“.

&INTTAID Beschreibung der UTM-Transaktion, die vorgesetzt wurde (Inhalt siehe [K204](#)).

**K206** XA( &TSNPID) Transaktion zurueckgesetzt; Ursache: &XATXT  
TA=&INTTAID

Die Inserts haben folgende Bedeutung:

&XATXT Grund für den Rollback. Mögliche Gründe: „Recovery“ oder „Int.Event“.

&INTTAID Beschreibung der UTM-Transaktion, die vorgesetzt wurde (Inhalt siehe [K204](#)).

**K207** XA( &TSNPID) Transaktion ist dem RM &INSTNUM nicht bekannt  
TA=&INTTAID

Die Inserts haben folgende Bedeutung:

&INSTNUM Instanz-Nummer des Resource Managers, siehe [K203](#).

&INTTAID Beschreibung der UTM-Transaktion, die dem RM nicht bekannt ist (Inhalt siehe [K204](#)).

**K210** XA( &TSNPID) Returncode: &XATXT; Open RM: &TEXT32,&INSTNUM

**K211** XA( &TSNPID) Returncode: &XATXT; Close RM: &TEXT32,&INSTNUM

Die Inserts der Meldungen K210, K211 und K216 haben folgende Bedeutung:

**&TEXT32** Name des Resource Managers.

**&INSTNUM** Instanz-Nummer des Resource Managers, siehe [K203](#).

**&XATXT** Schlüsselwort für den Returncode der aufgerufenen XA-Funktion.  
Mögliche Werte sind:

<b>&amp;XATXT</b>	<b>Bedeutung</b>
XA_OK	Normale Ausführung
XA_NOMIGRATE	Die Transaktion wurde suspendiert, aber der Kontext kann nicht migriert und die Transaktion nur in derselben Task fortgesetzt werden
XA_HEURHAZ	Aufgrund möglicher RM-interner Fehler kann die Transaktion heuristisch abgeschlossen worden sein
XA_HEURRB	Aufgrund einer heuristischen Entscheidung des Resource Managers wurde die Transaktion intern zurückgesetzt
XA_HEURMIX	Aufgrund einer heuristischen Entscheidung des Resource Managers wurde die Transaktion intern teilweise vor- und teilweise zurückgesetzt
XA_RETRY	Der Aufruf der Routine muss wiederholt werden
XA_RDONLY	Die Transaktion war „read_only“ und wurde vorgelesen
XA_RBROLLBACK	Rollback aus nicht näher spezifiziertem Grund
XA_RBCOMMFAIL	Rollback wegen Kommunikationsfehlers
XA_RBDEADLOCK	Rollback wegen Deadlocks
XA_RBINTEGRITY	Rollback wegen Ressourcen-Inkonsistenz
XA_RBOTHER	Rollback aus nicht näher spezifiziertem Grund
XA_RBPROTO	Rollback wegen eines RM-internen Protokollfehlers
XA_RBTIMEOUT	Rollback wegen Zeitüberschreitung der Transaktionsdauer
XA_RBTRANSIENT	Rollback wegen vorübergehenden Fehlers
XAER_ASYNC	Eine asynchrone Operation steht noch aus
XAER_RMERR	Im Resource Manager ist ein nicht wiederherstellbarer Fehler aufgetreten. Mögliche Ursache: Der Resource Manager wurde nicht oder falsch initialisiert
XAER_NOTA	Der Transaktions-Identifikator ist ungültig bzw. dem Resource Manager nicht bekannt
XAER_INVALID	Ungültige Funktionsargumente wurden angegeben. Möglicherweise enthalten der Open- oder Close-String falsche Parameter.

<b>&amp;XATXT</b>	<b>Bedeutung</b>
XAER_PROTO	Die Routine wurde RM-intern in unpassendem Kontext aufgerufen
XAER_RMFAIL	Der Resource Manager ist nicht mehr verfügbar
XAER_DUPID	Der Transaktions-Identifikator existiert bereits im Resource Manager. openUTM beendet den Vorgang. Zur Beseitigung dieser „XID-Leiche“ bei XAER_DUPID aus dem Resource Manager sollte der Datenbank-Administrator diese Transaktion entfernen. Mögliche Maßnahme: Datenbank herunterfahren und neu starten.
XAER_OUTSIDE	Der Resource Manager arbeitet außerhalb der Transaktion.

**K212** XA( &TSNPID) xa\_start(&XAFLAG) - Returncode: &XATXT  
TA=&INTTAID

**K213** XA( &TSNPID) xa\_end(&XAFLAG) - Returncode: &XATXT  
TA=&INTTAID

**K214** XA( &TSNPID) xa\_commit() - Returncode: &XATXT  
TA=&INTTAID

**K215** XA( &TSNPID) xa\_rollback() - Returncode: &XATXT  
TA=&INTTAID

Die Inserts der Meldungen K212 bis K215 haben folgende Bedeutung:

**&XATXT** Schlüsselwort für den Returncode der aufgerufenen XA-Funktion.  
Mögliche Werte siehe [K211](#).

**&XAFLAG** Schlüsselwort, das die Flags beim Aufruf von xa\_start() bzw. xa\_end() beschreibt.

**&INTTAID** Beschreibung der UTM-Transaktion, bei der der Returncode aufgetreten ist (Inhalt siehe [K204](#)).

**K216** XA( &TSNPID) Returncode: &XATXT; Recover PTC-Liste, RM:  
&TEXT32,&INSTNUM

Bedeutung und Inhalt der Inserts siehe [K211](#).

**K217** XA( &TSNPID) xa\_prepare() - Returncode: &XATXT  
TA=&INTTAID

Zur Bedeutung der Inserts siehe [K211](#) (&XATXT) und [K215](#) (&INTTAID).

**K218** XA( &TSNPID) xa\_forget() - Returncode: &XATXT  
TA=&INTTAID

Zur Bedeutung der Inserts siehe [K211](#) (&XATXT) und [K215](#) (&INTTAID).

- K220** XA( &TSNPID) Fehler: xa\_switch Definition fuer spezifizierten RM nicht gefunden: &TEXT32
- Ein Resource Manager (RM), für den RMXA-Startparameter angegeben wurden, wurde bei der KDCDEF-Generierung nicht definiert.  
&TEXT32 enthält den Namen des Resource Managers.
- K221** XA( &TSNPID) Fehler: Startparameter fuer definierten RM nicht gefunden: &TEXT32
- Für einen in der KDCDEF-Generierung enthaltenen Resource Manager (RM) sind keine Startparameter angegeben.  
&TEXT32 enthält den Namen des Resource Managers .
- K222** XA( &TSNPID) Fehler: Gebundener RM ist nicht &XASPEC kompatibel: &TEXT32
- Der Resource Manager (RM) benutzt nicht die XA-Schnittstelle, die bei der KDCDEF-Generierung angegeben wurde.  
&TEXT32 enthält den Namen des Resource Managers, &XASPEC die bei der UTM-Generierung angegebene XA-Schnittstelle.
- K223** XA( &TSNPID) Syntaxfehler in Startparameter:
- Die folgende Zeile enthält die fehlerhafte Zeile aus der Startparameter-Datei.
- K224** XA( &TSNPID) &XACALL - Returncode &XATXT der RM-Instanz &INSTNUM, &TEXT32 entspricht nicht der XA( CAE)-Spezifikation
- Der Resource Manager (RM) liefert einen unerwarteten Returncode.  
Die Inserts haben folgende Bedeutung:
- |          |  |
|----------|--|
| &XACALL  | Name des Aufrufs an der XA-Schnittstelle, z.B. xa_start().                                   |
| &XATXT   | Schlüsselwort für den Returncode der aufgerufenen XA-Funktion (siehe <a href="#">K211</a> ). |
| &INSTNUM | Instanz-Nummer des Resource Managers, siehe <a href="#">K203</a> .                           |
| &TEXT32  | Name des Resource Managers.  |
- K225** XA( &TSNPID) rekursiver Aufruf: &XADBC1 - Fehler/Signal im DB/XA-Anschluss bei &XADBC2
- Die Inserts haben folgende Bedeutung:
- |         |   |
|---------|---|
| &XADBC1 | Name der aufgerufenen Funktion im UTM/XA-Anschlussmodul.                          |
| &XADBC2 | Name der Funktion im UTM/XA-Anschlussmodul, in der ein Fehler festgestellt wurde. |
- K230** XA( &TSNPID) Int.Fehler: &TEXT32

- K231** XA( &TSNPID) Int.Fehler: PETA wird nicht unterstuetzt  
Das vorläufige Transaktionsende (Preliminary End of Transaction) wird nicht unterstüzt.
- K232** XA( &TSNPID) Int.Fehler: DBSTAT sekundaerer Operationscode inkonsistent  
Der sekundäre Operationscode des primären Operationscodes DBSTAT ist nicht konsistent.
- K233** XA( &TSNPID) Heur. Entsch. RM=&INSTNUM in &XACALL(&DBCALL), xa-rc=&XATXT, LTHGTRID=&LTHGTRID, GTRID=&GTRID  
Die Inserts haben folgende Bedeutung:
- |           |   |
|-----------|---|
| &INSTNUM  | Instanz-Nummer des Resource Managers (RM), siehe <a href="#">K203</a> .   |
| &XACALL   | Name des Aufrufs an der XA-Schnittstelle, z.B. xa_start().  |
| &DBCALL   | Name der aufgerufenen Funktion der UTM-DB-Schnittstelle   |
| &XATXT    | Schlüsselwort für den Returncode der aufgerufenen XA-Funktion (siehe <a href="#">K211</a> ).                                    |
| &GTRID    | Globale Transaktions-ID (gtrid) der UTM-Transaktion entsprechend der XA/CAE-Spezifikation (Inhalt siehe <a href="#">K204</a> ). |
| &LTHGTRID | Länge der globalen Transaktions-ID.   |
- K234** LMS Fehler (PLA&PLARC, DMS&DMSRC) fuer Bibliothek: &FNAM  
Element: &ELEM  
Version: &EVER  
Typ: &LMSTYPE  
Die Inserts haben folgende Bedeutung:
- |          |  |
|----------|--|
| &PLARC   | PLAM-Returncode. Der Hilfetext kann mittels /HELP PLA&PLARC erfragt werden. 0000 bedeutet, dass der PLAM-Aufruf erfolgreich war. |
| &DMSRC   | DMS-Returncode. Der Hilfetext kann mittels /HELP DMS&DMSRC erfragt werden. 0000 bedeutet, dass der DMS-Aufruf erfolgreich war.   |
| &FNAM    | Name der PLAM-Bibliothek, auf die sich die Meldung bezieht.  |
| &ELEM    | Name des LMS-Elements, auf das sich die Meldung bezieht.   |
| &EVER    | LMS-Version des LMS-Elements.  |
| &LMSTYPE | LMS-Typ des LMS-Elements .   |



- K235** Die Namensauflösung fuer &PRNM dauerte &TCPMS Millisekunden (Socket-Aufruf: &TCPCL, Returncode: &TCPRC, IP-Adresse: &IPADDR).
- K236** Die Adressauflösung fuer IP-Adresse &IPADDR dauerte &TCPMS Millisekunden (Socket-Aufruf: &TCPCL, Returncode: &TCPRC, Hostname: &PRNM).  
Die Inserts haben folgende Bedeutung:
- |         |  |
|---------|--|
| &IPADDR | IPv4 oder IPv6 Adresse   |
| &TCPMS  | Anzahl Millisekunden   |
| &TCPCL  | Socket-Funktion (siehe Beschreibung bei Meldung <a href="#">K154</a> ) |
| &TCPRC  | Diagnostic Word (siehe Beschreibung bei Meldung <a href="#">K154</a> ) |
| &PRNM   | Rechnername  |
- K237** XA( &TSNPID) Fehler bei &DBCALL: Oracle-Passwort und -User koennen nur per KDCDEF-Generierung festgelegt werden.  
&TSNPID ist die TSN und &DBCALL hat immer den Wert DBSTPA (DB-Startparameter Analyse). Der Start der Anwendung wird abgebrochen.
- K238** XA( &TSNPID) Warnung bei &DBCALL: Klartext User oder Passwort fuer Oracle-DB in Startparameter.
- K239** Die asynchrone Nachricht von &GNUSER an &LPAP konnte wegen eines permanenten Fehlers nicht zugestellt werden und wurde geloescht.
- K251** Version V&IMPVER von Filebase &FBASUPD stimmt nicht mit der aktuellen Version V&DEFVER ueberein.
- K252** &UPDERR Fehler.  
Einer der folgenden Fehler ist aufgetreten:
- Es ist ein Fehler beim Anfordern des Speicherplatzes beim "Online-Import" aufgetreten.
  - Fehler im Dienstprogramm KDCUPD.
- Wenn &UPDERR den Wert PARAM hat, dann liegt ein Syntaxfehler vor.  
KDCUPD bricht ab und setzt den Schalter 3 auf ON.  
Maßnahme: Richtiges Kommando eingeben!  
Ansonsten ist ein Fehler bei Speicherplatzanforderung, beim Einlesen etc. aufgetreten  
Maßnahme: PM schreiben mit Unterlagen
- K255** DMS-Fehler &DMSE fuer Datei &FNAM.  
Maßnahme: Abhängig von DMS-Fehlermeldung.

- K256** Datei &FNAM entspricht nicht KDCFILE - Konventionen.  
Maßnahme: Richtige Datei zur Verfügung stellen.
- K257** Der Anwendungslauf wurde nicht ordnungsgemäß beendet.
- K258** Mit der Datei &FNAM ist schon gestartet worden.
- K260** Unbekannte Version &DEFVER von Filebase &FBASUPD.  
Maßnahme: Richtige Datei zur Verfügung stellen oder Problemmeldung.
- K261** Inkonsistente Datei &FNAM.  
Maßnahme: Datei wiederherstellen oder neu generieren.
- K262** Falsche Datei &FNAM.
- K263** Mit der Datei &FNAM ist noch nicht gestartet worden.
- K269** &OBJ1 &OST1 und &OBJ3 &OST2 sind nicht vertraglich
- K273** Fehler: &TRMA in Modul &UPDMODUL.  
Der KDCUPD-Lauf wurde abgebrochen, der Fehlercode &TRMA gibt Auskunft über die Ursache.

<b>&amp;TRMA</b>	<b>Ursache</b>	<b>Maßnahme</b>
BFMMER	Fehler beim Initialisieren des Buffer-Managements (Cluster)	ggf. virtuellen Adresraum erhöhen oder PM
CFGERR	Fehler beim Bearbeiten der Cluster-Konfigurationsdatei	vorherige Diagnose-Meldungen oder falls keine: inkonsistente Reihenfolge Knotennamen
CONSER	Interner Fehler	PM schreiben und Unterlagen erstellen
DMSERR	Ein DMS-Fehler ist aufgetreten	siehe K255
DxxxRD	DMS Fehler xxx beim Lesen der ersten Seite	je nach DMS-Code
DxxxWR	DMS Fehler xxx beim Schreiben der ersten Seite	je nach DMS-Code
xxxx01	DMS-Fehler xxxx beim Lesen der ersten KAA-Seite	je nach DMS-Code
GMDTER	Fehler beim Prüfen der Cluster-GSSB-Datei	siehe vorherige Diagnose-Meldungen
INCONS	Fehler bei Konsistenzprüfungen	siehe zusätzliche Meldung K261/K262
LOOKFL LOOKFU	Fehler bei Suche nach Verursacher (LTERM, USER)	PM schreiben und Unterlagen erstellen

<b>&amp;TRMA</b>	<b>Ursache</b>	<b>Maßnahme</b>
LOOKLT	Fehler bei Suche nach LTERM	PM schreiben und Unterlagen erstellen
LOOKTC	Fehler bei Suche nach TAC	PM schreiben und Unterlagen erstellen
LOOKUS	Fehler bei Suche nach USER	PM schreiben und Unterlagen erstellen
READxx	ein DMS-Fehler ist aufgetreten	s. Startfehlercode xx in K049
REQKTA	Fehler bei Speicheranforderung für KTA	zuwenig Speicherplatz vorhanden, bei geringerer Systembelastung erneut versuchen
REQM01	Fehler bei Request Memory für 1. Seite	evtl. System neu generieren
REQM02	Speichermangel beim Anlegen von Memory Pool für UTM-Cache	BS2000-Generierung überprüfen oder Operand CACHESIZE kleiner machen
OPWRON	Interner Fehler	PM schreiben und Unterlagen erstellen
PPMMER	Verwaltungsdatei des Cluster Pagepools fehlerhaft.	siehe vorherige Diagnose-Meldungen
UMDTER	Fehler beim Prüfen der Cluster-ULS-Datei	siehe vorherige Diagnose-Meldungen
UPDSTA	Datei wurde bereits mit KDCUPD bearbeitet	richtige Datei zuweisen
USFERR	Fehler beim Öffnen der Cluster-User-Datei	siehe vorherige Diagnose-Meldungen
WRDB01	Abbruch im KDCUPD Schreibmodul. Ursache: Anzahl der DB-Systeme in der neuen KDCFILE kleiner als in der alten KDCFILE. Kann nur auftreten, wenn alte und neue KDCFILE mit UTM-S generiert sind. Zusätzlich kommt K307.	Generierung der neuen KDCFILE anpassen
WRDB02	Abbruch im KDCUPD Schreibmodul. Ursache: Typ der Datenbank(en) in der neuen KDCFILE anders als in der alten KDCFILE. Kann nur auftreten, wenn alte und neue KDCFILE mit UTM-S generiert sind. Zusätzlich kommt K307.	Generierung der neuen KDCFILE anpassen

<b>&amp;TRMA</b>	<b>Ursache</b>	<b>Maßnahme</b>
WRDB03	Abbruch im KDCUPD Schreibmodul. Ursache: mindestens ein DB-System in der alten KDCFILE und mehr DB-Systeme in der neuen KDCFILE als in der alten KDCFILE. Kann nur auftreten, wenn alte und neue KDCFILE mit UTM-S generiert sind. Zusätzlich kommt K307.	Generierung der neuen KDCFILE anpassen
WRFORM	Abbruch im KDCUPD Schreibmodul. Ursache: Alte KDCFILE ist mit Formatierung und neue KDCFILE ohne Formatierung generiert, beide KDCFILES sind mit Variante UTM-S generiert.	Generierung der neuen KDCFILE anpassen
WRITxx	ein DMS-Fehler ist aufgetreten	s. Startfehlercode xx in K049
...GSB ....GB	Fehler bei GSSB-Behandlung	PM schreiben und Unterlagen erstellen
...TLS ....TL	Fehler bei TLS-Behandlung	PM schreiben und Unterlagen erstellen
...ULS ....UL	Fehler bei der ULS- Behandlung	PM schreiben und Unterlagen erstellen
..DIAL	Fehler beim Übertragen von geketteten Vorgängen	PM schreiben und Unterlagen erstellen
...ASY ....AS	Fehler bei Behandlung von FPUTs oder DPUTs an LTERM	PM schreiben und Unterlagen erstellen
...IMS ....IM	Fehler bei Behandlung von FPUTs oder DPUTs an TAC	PM schreiben und Unterlagen erstellen
...UPI ....UP	Fehler bei Behandlung einer Dialognachricht für UPIC	PM schreiben und Unterlagen erstellen
...SOC ....SO	Fehler bei Behandlung einer Dialognachricht für SOCKET	PM schreiben und Unterlagen erstellen
...UMS ....UM	Fehler bei Behandlung einer Nachricht einer User-Queue	PM schreiben und Unterlagen erstellen
...QMS ....QM	Fehler bei Behandlung einer Nachricht einer Queue	PM schreiben und Unterlagen erstellen
...LSB	Fehler bei LSSB-Behandlung	PM schreiben und Unterlagen erstellen

Weitere Fehlercodes können die Fehlercodes bei abnormaler Beendigung von UTM sein (Beschreibung siehe Abschnitt „[Der Anwendungslauf wurde abgebrochen; die Ursache ist &TRMA.](#)“ auf Seite 177). In diesem Fall sollten Sie immer eine PM schreiben und Unterlagen erstellen.

- K274** Programm Ende mit Dump.
- K277** Mindestens eine Knoten-Anwendung wurde nicht ordnungsgemäss beendet.  
Näheres siehe SYSOUT-Protokoll.
- K278** Mindestens eine Knoten-Anwendung wurde schon einmal gestartet.
- K279** Fehler bei Cluster-Bearbeitung, siehe (mögliche) vorherige Diagnosemeldung
- K300** &UPDPRO Prozent des &PGPOOL belegt.
- K303** &UKCOP-Daten uebertragen. KCRN = &UKCRN, Typ = &UPDTYP, KCLM = &UKCLM.
- K304** Vorgangs-Daten uebertragen fuer USER &USER. Vorgangs-Typ = &TACTYPE, Stapelhoehoe = &UKCHSTA
- K305** Belegte Seiten des &PGPOOL fuer &UPDTYP: &PGS1 insgesamt, &PGS2 fuer &UKCRN.
- K306** Belegte Seiten des &PGPOOL fuer &UPDTYP: &PGS1.

#### *Hinweis*

Bei allen folgenden Meldungen von KDCUPD, bei denen der Text mit einem „\*“ (Stern) beginnt, konnten die Daten nicht uebertragen werden.

- K310** \* &UPDTYP &UKCRN nicht gefunden.  
Siehe auch [Hinweis](#), oben.
- K311** \* &UPDTYP &UKCRN nicht gefunden. Daten nicht uebertragen.  
Siehe auch [Hinweis](#), oben.
- K314** \* Warnung! LPUT-Daten gehen verloren.  
Siehe auch [Hinweis](#), oben.
- K317** \* &UKCOP-Daten nicht uebertragen. KCRN= &UKCRN, Typ = &UPDTYP, KCLM= &UKCLM, KCRCCC= &RCCC, KCRCDC= &RCDC . Sender ist LTERM = &LTRM , USER = &USER.  
Siehe auch [Hinweis](#), oben.

**K318** \* &UPDTYP &UKCRN : Sender der Asynchron-Nachricht nicht gefunden. LTERM = &LTRM , USER = &USER.

Siehe auch [Hinweis](#), oben.

**K320** \* Vorgangs-Daten fuer USER &USER nicht uebertragen. Vorgangs-Typ = &TACTYPE, Grund: &UERRCODE, &UERRINFO, &RCDC

Siehe auch [Hinweis](#), oben.

Das Insert &UERRCODE gibt an, warum die Vorgangsdaten für den USER &USER nicht übertragen wurden. Das Insert &UERRINFO enthält eine Zusatzinformation.

<b>&amp;UERRCODE</b>	<b>&amp;UERRINFO</b>	<b>Bedeutung</b>
EDIT	name	Das Editprofil ist nicht generiert.
KBLTH	bytes	Der Generierungs Parameter MAX KB in der neuen KDCFILE ist zu klein. Der Vorgang benötigt „bytes“ Bytes.
LOCK ERR	tac	Der Benutzer hat nicht den nötigen Schlüssel für den Transaktionscode „tac“ in seinem Schlüsselbund
LSSBS	anzahl	Die Anzahl „anzahl“ LSSBs in einem Vorgang des Benutzers ist größer als der Generierungswert (MAX LSSBS = ).
LTERM	lterm	Das Sender-LTERM „lterm“ ist nicht definiert.
MISSMATCH		Der Benutzer ist in der neuen KDCFILE mit RESTART=NO generiert oder sein Typ hat sich geändert (Verbindungs-Benutzerkennung ja/nein).
NRCONV	anzahl	Die Anzahl „anzahl“ der gekellerten Vorgänge des Benutzers ist größer als der Generierungswert (MAX NRCONV = ).
PAGEPOOL		Der Pagepool ist zu klein (MAX PAGEPOOL = ).
RESSHORT		Die UTM-Tabelleneinträge zur Verwaltung von Vorgangsdaten reichen nicht aus. In der neuen KDCFILE wurden mehr Benutzerkennungen, Sessions und/oder Tasks generiert als in der alten oder der Wert von ASYNTASKS ist kleiner als in der alten KDCFILE.
TAC	name	Der Transaktionscode „name“ ist ungültig.
TRMSGLTH		Die Nachricht für ein Folgeprogramm ist zu groß (MAX TRMSGLTH).
VGMSIZE	größe	Das Vorgangsgedächtnis („größe“ in KB) ist größer als der Generierungswert in der neuen KDCFILE (MAX VGMSIZE=)

## 5.2 Meldungen des XAP-TP-Providers

Die Meldungen des XAP-TP-Providers beginnen mit dem Buchstaben 'P'. Die Werte der Inserts sind entweder im Anschluss an die Meldung erläutert oder - für mehrfach vorkommende Inserts - in [Abschnitt „Allgemeine Inserts der XAP-TP-Meldungen“ auf Seite 314](#) aufgeführt.

**P001** Fehler beim OSS Aufruf (&XPFUNC): &ACPNT, &XPRET, &XPERR, &XP1INFO, &XP2INFO

Diese Meldung wird ausgegeben, wenn der Aufruf einer OSS-Funktion (*&XPFUNC*) einen Fehler liefert. Handelt es sich um einen vom Transportsystem gemeldeten Fehler, wird zusätzlich die Meldung P012 ausgegeben.

Die Inserts haben folgende Bedeutung:

&XPFUNC	Name der OSS-Funktion
&ACPNT	Name des lokalen ACCESS-POINT
&XPRET	siehe Tabelle auf <a href="#">Seite 314</a>
&XPERR	siehe Tabelle auf <a href="#">Seite 314</a>
&XP1INFO	OSS-Zusatzinformation
&XP2INFO	OSS-Zusatzinformation

**P002** Fehler beim Associationaufbau (&XPFUNC): &ACPNT, &OSLPAP, &XPRET, &XPERR, &XP1INFO, &XP2INFO

Diese Meldung wird ausgegeben, wenn der Aufruf einer OSS-Funktion (*&XPFUNC*), die zum Aufbau einer Association benötigt wird, einen Fehler liefert. Handelt es sich um einen vom Transportsystem gemeldeten Fehler, wird zusätzlich die Meldung P012 ausgegeben. Ist es kein vom Transportsystem gemeldeter Fehler, wird die Anwendung mit „Termapplication“ beendet.

Die Inserts haben folgende Bedeutung:

&XPFUNC	Name der OSS-Funktion
&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPRET	siehe Tabelle auf <a href="#">Seite 314</a>
&XPERR	siehe Tabelle auf <a href="#">Seite 314</a>
&XP1INFO	OSS-Zusatzinformation
&XP2INFO	OSS-Zusatzinformation

**P003** Association abgelehnt (a\_assin() ): &ACPNT, Grund: &XPRJCT, Laenge: &XPLTH

Diese Meldung wird ausgegeben, wenn der Aufbau einer Association von außen abgelehnt wird.

Die Inserts haben folgende Bedeutung:

&ACPNT        Name des lokalen ACCESS-POINT

&XPLTH        fehlerhafte Länge

&XPRJCT       siehe Tabelle auf [Seite 317](#)

**P004** Association abgelehnt (a\_assin() ): &ACPNT, &OSLPAP, Grund: &XPRJCT

Diese Meldung wird ausgegeben, wenn der Aufbau einer Association von außen abgelehnt wird.

Die Inserts haben folgende Bedeutung:

&ACPNT        Name des lokalen ACCESS-POINT

&OSLPAP       Name des Partners in der lokalen Anwendung

&XPRJCT       siehe Tabelle auf [Seite 317](#)

Mögliche Ursachen bei &XPRJCT = 34 oder &XPRJCT = 35 (NO\_MORE\_CONTENTION\_LOSER / WINNER\_ASSOCIATIONS, [Seite 317](#)):

- Associations wurden abgebaut, ohne dass die lokale UTM-Anwendung darüber informiert wurde. Die Associations sind aus Sicht der lokalen UTM-Anwendung noch aufgebaut. Die OSI-TP-Partner-Anwendung versucht diese Associations wieder aufzubauen. Diese können aber erst wieder aufgebaut werden, wenn entweder die betroffenen Associations durch Ablauf des Idle-Timers oder alle Associations zum betroffenen Partner administrativ abgebaut werden.
- In der OSI-TP-Partner-Anwendung sind mehr Associations generiert als in der lokalen UTM-Anwendung.
- die Anzahl der Contention Loser/Winner in der lokalen und in der OSI-TP-Partner-Anwendung passt nicht zusammen.

**P005** Association abgelehnt (a\_assin() ): &ACPNT, Grund: Partner unbekannt

N-SEL: &XPNSEL, T-SEL: &XPTSEL

S-SEL: (&XPLSSEL,&XPCSSEL,&XPHSSEL)

P-SEL: (&XPLPSEL,&XPCPSEL,&XPHPSEL)

Diese Meldung wird ausgegeben, wenn der Aufbau einer Association von außen abgelehnt wird, weil der entfernte Partner in der lokalen Anwendung nicht bekannt ist.

Die Inserts haben folgende Bedeutung:



&ACPNT	Name des lokalen ACCESS-POINT
&XPNSEL	Network Selektor (BCAM-Prozessorname) des entfernten Partners
&XPTSEL	Transport Selektor (BCAM-Anwendungsname) des entfernten Partners
&XPLSSEL	Länge des Session Selectors des entfernten Partners
&XPCSSEL	Session Selector (abdruckbar) des entfernten Partners
&XPHSSEL	Session Selector (hexadezimal) des entfernten Partners
&XLPLSEL	Länge des Presentation Selectors des entfernten Partners
&XPCPSEL	Presentation Selector (abdruckbar) des entfernten Partners
&XPXPSEL	Presentation Selector (hexadezimal) des entfernten Partners

**P006** Association abgelehnt (a\_assin() ): &ACPNT, &OSLPAP, Grund: Falscher Application Context Name ( &XP0OBID, &XP1OBID, &XP2OBID, &XP3OBID, &XP4OBID, &XP5OBID, &XP6OBID, &XP7OBID, &XP8OBID, &XP9OBID )

Diese Meldung wird ausgegeben, wenn der Aufbau einer Association von außen abgelehnt wird. Der Application Context Name für den entfernten Partner stimmt nicht mit dem in der lokalen Anwendung für diesen Partner generierten Application Context Namen überein.

Die Inserts haben folgende Bedeutung:

&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XP0OBID - &XP9OBID	sind die (maximal) zehn Elemente des Objektbezeichners, die den Application Context Name des entfernten Partners bilden. Für nicht belegte Elemente wird <i>-1</i> ausgegeben.

**P007** Fehler beim Associationaufbau (a\_assrs() ): &ACPNT, &OSLPAP, &XPRET, &XPERR, &XP1INFO, &XP2INFO

Diese Meldung wird ausgegeben, wenn der Aufruf der OSS-Funktion *a\_assrs()*, mit der der Associationaufbau von außen beantwortet wird, einen Fehler liefert. Handelt es sich um einen vom Transportsystem gemeldeten Fehler, wird zusätzlich die Meldung P012 ausgegeben.

Die Inserts haben folgende Bedeutung:

&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPRET	siehe Tabelle auf <a href="#">Seite 314</a>

&XPERR      siehe Tabelle auf [Seite 314](#)

&XP1INFO    OSS-Zusatzinformation

&XP2INFO    OSS-Zusatzinformation

**P008** Association (&XPOSAS ) aufgebaut: &ACPNT, &OSLPAP

Diese Meldung wird ausgegeben, wenn eine Association aufgebaut wurde.

Die Inserts haben folgende Bedeutung:

&XPOSAS    Index der betroffenen Association

&ACPNT      Name des lokalen ACCESS-POINT

&OSLPAP    Name des Partners in der lokalen Anwendung

**P009** Association (&XPOSAS ) abgelehnt (a\_asscf() ): &ACPNT, &OSLPAP, Grund: &XPRJCT, Laenge: &XPLTH

Diese Meldung wird ausgegeben, wenn der aktive Aufbau einer Association abgelehnt wird, weil die Bestätigung des Partners nicht akzeptiert werden kann.

Die Inserts haben folgende Bedeutung:

&XPOSAS    Index der betroffenen Association

&ACPNT      Name des lokalen ACCESS-POINT

&OSLPAP    Name des Partners in der lokalen Anwendung

&XPRJCT    siehe Tabelle auf [Seite 317](#)

&XPLTH      eventuell eine fehlerhafte Länge

**P010** Association (&XPOSAS ) abgelehnt (a\_asscf() ): &ACPNT, &OSLPAP, Grund: Partner unbekannt

N-SEL: &XPNSEL, T-SEL: &XPTSEL

S-SEL: (&XPLSSEL,&XPCSSEL,&XPHSSEL)

P-SEL: (&XPLPSEL,&XPCPSEL,&XPHPSEL)

Diese Meldung wird ausgegeben, wenn der aktive Aufbau einer Association abgelehnt wird, weil der entfernte Partner bei der Bestätigung des Associationaufbaus mit einer Adresse (&XPADDR) antwortet, die in der lokalen Anwendung nicht bekannt ist.

Die Inserts haben folgende Bedeutung:

&XPOSAS    Index der betroffenen Association

&ACPNT      Name des lokalen ACCESS-POINT

&OSLPAP    Name des Partners in der lokalen Anwendung

&XPNSEL	Network Selector (BCAM-Prozessorname) des entfernten Partners
&XPTSEL	Transport Selektor (BCAM-Anwendungsname) des entfernten Partners
&XPLSSEL	Länge des Session Selectors des entfernten Partners
&XPCSSEL	Session Selector (abdruckbar) des entfernten Partners
&XPHSSEL	Session Selector (hexadezimal) des entfernten Partners
&XPLPSEL	Länge des Presentation Selectors des entfernten Partners
&XPCPSEL	Presentation Selector (abdruckbar) des entfernten Partners
&XPXPSEL	Presentation Selector (hexadezimal) des entfernten Partners

**P011** Association (&XPOSAS ) abgelehnt (a\_asscf() ): &ACPNT, &OSLPAP, Grund: Falscher Application Context Name ( &XP0OBID, &XP1OBID, &XP2OBID, &XP3OBID, &XP4OBID, &XP5OBID, &XP6OBID, &XP7OBID, &XP8OBID, &XP9OBID )

Diese Meldung wird ausgegeben, wenn der aktive Aufbau einer Association abgelehnt wird, weil der entfernte Partner bei der Bestätigung des Association-aufbaus mit einem Application Context Name antwortet, der nicht mit dem in der lokalen Anwendung für diesen Partner konfigurierten Application Context Name übereinstimmt.

Die Inserts haben folgende Bedeutung:

&XPOSAS      Index der betroffenen Association

&ACPNT        Name des lokalen ACCESS-POINT

&OSLPAP      Name des Partners in der lokalen Anwendung

&XP0OBID - &XP9OBID

sind die (maximal) zehn Elemente des Objektbezeichners, die den Application Context Name des entfernten Partners bilden.  
Für nicht belegte Elemente wird -1 ausgegeben.

**P012** CMX Diagnoseinformation: &XPCTYPE, &XPCCLS, &XPCVAL, &XPBCAM

Diese Meldung wird ausgegeben, wenn eine vorhergehende Meldung auf Grund eines vom Transportsystem gemeldeten Fehlers ausgegeben wurde. Es wird der Diagnosecode des Transportsystems aufbereitet. In den folgenden Tabellen werden einige Werte für &XPCTYPE, &XPCCLS und &XPCVAL beschrieben. Eine vollständige Auflistung enthält die zu CMX gehörende Include-Datei `cmx.h`.

<b>XPCTYPE</b>	<b>Bedeutung (CMX-Fehlertyp)</b>
0	T_CMXTYPE: CMX-Fehler, der von der CMX-Bibliothek erkannt wurde
2	T_DSTEMPERR: Temporärer TNS-Fehler
3	T_DSCALL_ERR: TNS-Aufruffehler
4	T_DSPERM_ERR: Permanenter TNS-Fehler
5	T_DSWARNING: TNS-Warnung
>15	CMX-Fehler auf Grund von Fehlercodes aus dem Transportsystem

<b>XPCCLS</b>	<b>Bedeutung (CMX-Fehlerklasse, gültig für &amp;XPCTYPE &lt; 15)</b>
0	T_CMXCLASS: CMX-Klasse
2	T_DSNOT_SPEC: TNS-Klasse nicht spezifiziert
3	T_DSPAR_ERR: TNS-Parameterfehler
4	T_DSILL_VERS: Ungültige TNS-Version
5	T_DSSYS_ERR: TNS-Systemfehler
6	T_DSINT_ERR: Interner TNS-Fehler
7	T_DSMESSAGE: TNS-Hinweis

<b>XPCVAL</b>	<b>Bedeutung (CMX-Fehlerwert)</b>
0	T_NOERROR: Kein Fehler
5	T_EIO: Momentaner Engpass bzw. Fehler im Transportsystem
14	T_EFAULT: IO_Area nicht allokiert
100	T_UNSPECIFIED: Nicht näher spezifizierter Fehler, i. a. Fehler bei einem Systemaufruf
101	T_WSEQUENCE: Unzulässige Aufrufreihenfolge
103	T_WPARAMETER: Ungültiger Parameter
104	T_WAPPLICATION: Die Anwendung ist im TNS nicht bekannt oder die Task ist zur Anmeldung der Anwendung nicht berechtigt oder die Anwendung ist von dieser Task bereits eröffnet.

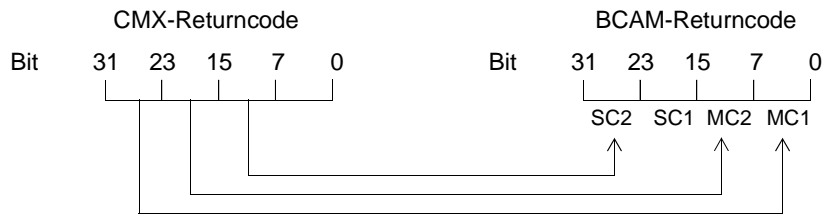
XPCVAL	Bedeutung (CMX-Fehlerwert)
105	T_WAPP_LIMIT: Der Grenzwert für gleichzeitig aktive Anwendungen ist erreicht.
106	T_WCONN_LIMIT: Der Grenzwert für gleichzeitig aktive Verbindungen ist erreicht.
107	T_WTREF: Ungültige Transportreferenz oder Transportverbindung bereits abgebaut.
111	T_NOCCP: Das Transportsystem unterstützt die gewünschte Anwendung oder Verbindung nicht.
114	T_CCP_END: Das Transportsystem wurde beendet oder die Anwendung wurde vom Administrator geschlossen.
255	T_WLIBVERSION: Kein Anschluss an das CMX-Subsystem möglich.
-100	T_INVREF: Die evid ist ungültig. CMX kann den Aufruf keinem Wartepunkt zuordnen.

&XPBCAM Wenn der Wert von &XPCTYP größer als 15 ist, wird mit &XPBCAM das BCAM-Infoword ausgegeben.

### CMX Returncode

Informieren Sie sich im Anhang des CMX-Handbuchs über den Aufbau des CMX-Returncodes. Bei der Auswertung des CMX-Returncodes sind zwei Fälle zu unterscheiden:

1. Der CMX-Fehlertyp hat einen Wert kleiner als 15: Es liegt ein Systemfehler vor.
2. Der CMX-Fehlertyp hat einen Wert größer als 15: Das Transportsystem hat einen Fehler gemeldet, der BCAM-Returncode ist im CMX-Returncode enthalten. Dabei enthalten die Bits 0-7 des CMX-Returncodes den CMX-Fehlerwert (seine Bedeutung ist im CMX-Handbuch beschrieben). Den BCAM-Returncode erhält man aus dem Rest des CMX-Returncodes mit Hilfe des folgenden Schemas. Dabei muss SC1 des BCAM-Returncodes (S-RTC1) immer auf 00 gesetzt werden.



Dabei bedeutet:

SC1/2 = S-RTC1/2 (Subcodes)

MC1/2 = M-RTC1/2 (Maincodes)

Die Bits 24-31 aus dem CMX-Returncode werden in das Feld MC1 des BCAM-Returncodes übertragen.

Die Bits 16-23 aus dem CMX-Returncode werden in das Feld MC2 des BCAM-Returncodes übertragen.

Die Bits 8-15 aus dem CMX-Returncode werden in das Feld SC2 des BCAM-Returncodes übertragen.

Die Bedeutung der Diagnosecodes ist im Anhang des CMX-Handbuches beschrieben.

**P013** Association (&XPOSAS ) abgelehnt (a\_asscf() ): &ACPNT, &OSLPAP, Grund:  
 &XPCRES, &XPSRC, &XPNDIA  
 CCR V2 = &XP1BOOL, Version Incompatibility = &XP2BOOL  
 ContWin Assignment rejected = &XP3BOOL  
 Bid mandatory rejected = &XP4BOOL, No reason = &XP5BOOL

Diese Meldung wird ausgegeben, wenn der aktive Aufbau einer Association vom entfernten Partner abgelehnt wird.

Die Inserts haben folgende Bedeutung:

&XPOSAS      Index der betroffenen Association

&ACPNT        Name des lokalen ACCESS-POINT

&OSLPAP        Name des Partners in der lokalen Anwendung

&XPCRES        gibt an, ob die Ablehnung vorübergehend oder dauerhaft ist:  
 0= permanent reject (dauerhafte Ablehnung)  
 1= transient reject (vorübergehende Ablehnung)

- &XPCSRC gibt an, wer den Aufbau abgelehnt hat:  
 0 = ACSE Service User  
 1 = ACSE Service Provider  
 2 = Presentation Service Provider
- &XPNDIA siehe Tabelle auf [Seite 321](#)
- &XP1BOOL - &XP5BOOL können die Werte TRUE und FALSE annehmen. Die mit TRUE belegten Werte geben an, welche Gründe der Partner für die Ablehnung des Associationaufbauwunsches mitgeschickt hat:
- &XP1BOOL: Die CCR Version 2 ist nicht verfügbar  
 &XP2BOOL: Die TP-Protokollversionen sind nicht kompatibel  
 &XP3BOOL: Die Contention-Winner-Zuordnung wird abgelehnt  
 &XP4BOOL: Die Festlegung „Bidding ist Pflicht“ oder „Bidding ist nicht Pflicht“ wird abgelehnt  
 &XP5BOOL: Es wird kein Grund angegeben

Mögliche Ursachen dafür, dass die Contention-Winner-Zuordnung abgelehnt wird (&XP3BOOL=TRUE):

- In der OSI-TP-Partner-Anwendung sind weniger Associations generiert als in der lokalen UTM-Anwendung.
- Die Anzahl der Contention Loser/Winner in der lokalen und in der Partner-Anwendung passt nicht zusammen.
- Associations wurden abgebaut, ohne dass die OSI-TP-Partner-Anwendung darüber informiert wurde. Die OSI-TP-Partner-Anwendung lehnt den Aufbau der Associations ab, weil diese aus Sicht der OSI-TP-Partner-Anwendung noch aufgebaut sind.  
 In der OSI-TP-Partner-Anwendung wird in dieser Situation die Meldung P004 mit &XPRJCT = 34 oder &XPRJCT = 35 erzeugt (NO\_MORE\_CONTENTION\_LOSER / WINNER\_ASSOCIATIONS, [Seite 317](#)).

**P014** Fehler beim Associationabbau (&XPOSAS ) ( &XPFUNC): &ACPNT, &OSLPAP, &XPRET, &XPERR, &XP1INFO, &XP2INFO

Diese Meldung wird ausgegeben, wenn der Aufruf einer OSS-Funktion, die zum Aufbau einer Association benötigt wird, einen Fehler liefert. Handelt es sich um einen vom Transportsystem gemeldeten Fehler, wird zusätzlich die Meldung P012 ausgegeben. Ist es kein vom Transportsystem gemeldeter Fehler, wird die Anwendung mit "Termapplication" beendet.

Die Inserts haben folgende Bedeutung:

&XPOSAS Index der betroffenen Association

&XPFUNC	Name der OSS-Funktion
&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPRET	siehe Tabelle auf <a href="#">Seite 314</a>
&XPERR	siehe Tabelle auf <a href="#">Seite 314</a>
&XP1INFO	OSS-Zusatzinformation
&XP2INFO	OSS-Zusatzinformation, zur Zeit immer Null.

**P015** Association (&XPOSAS ) abgebaut (&XPFUNC): &ACPNT, &OSLPAP, &XPLNK, &XPSRC, &XPNDIA, &XPINI, &XP1INFO, &XP2INFO

Diese Meldung wird ausgegeben, wenn eine Association abgebaut wird.

Die Inserts haben folgende Bedeutung:

&XPOSAS	Index der betroffenen Association
&XPFUNC	Name der OSS-Funktion
&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPLNK	repräsentiert den internen Zustand der Association 0 = Association nicht verknüpft 1 = Association mit Channel verknüpft 2 = Association mit Instanz verknüpft
&XPCSRC	Verursacher des Abbaus 0 = ACSE Service User 1 = ACSE Service Provider 2 = Presentation Service Provider
&XPNDIA	siehe Tabelle auf <a href="#">Seite 321</a>
&XP1INFO	OSS-Zusatzinformation
&XP2INFO	OSS-Zusatzinformation
&XPINI	siehe folgende Tabelle:

XPINI	Bedeutung
0	Die Association wurde von innen abgebaut.
401	O_LOC_TRAN Der Verursacher ist das lokale Transportsystem. &XP1INFO enthält den CMX-Returncode. Dieser wird in der Folgemeldung P012 detailliert ausgegeben.



XPINI	Bedeutung
402	<p>O_REM_TRAN</p> <p>Der Verursacher ist das entfernte Transportsystem. &amp;XP1INFO enthält den Grund des CMX-Events t_disin. Die Werte sind in cmx.h definiert. Nachfolgend einige ausgewählte Werte von &amp;XP1INFO und ihre Bedeutung. Dabei werden folgende Abkürzungen verwendet:</p> <ul style="list-style-type: none"> <li>– CCP (<b>C</b>ommunication <b>C</b>ontrol <b>P</b>rogram) für das Anwendungsprogramm, das die Kommunikation steuert</li> <li>– TSAP (<b>T</b>ransport <b>S</b>ervice <b>A</b>ccess <b>P</b>oint) für den Zugriffspunkt zum Transport-Service,</li> <li>– NSAP (<b>N</b>etwork <b>S</b>ervice <b>A</b>ccess <b>P</b>oint) für den Zugriffspunkt zum Netzwerk-Service .</li> <li>– PDU (<b>P</b>rotocol <b>D</b>ata <b>U</b>nit) für Datenelemente einer Protokoll-Schicht.</li> </ul> <p>&amp;XP1INFO kann z.B. folgende Werte annehmen:</p> <p>0 (T_USER) Der Abbau erfolgte durch den Kommunikationspartner, u. U. auch durch einen Benutzerfehler auf der Partnerseite.</p> <p>1 (T_TIMEOUT) Wegen Inaktivität der Verbindung gemäß Parameter t_timeout wurde die Verbindung lokal durch CMX abgebaut.</p> <p>2 (T_RADMIN) Wegen Außerbetriebnahme des CCP durch die Administration wurde die Verbindung lokal durch CMX abgebaut.</p> <p>3 (T_CCPEND) Wegen CCP-Ausfall wurde die Verbindung lokal durch CMX abgebaut.</p> <p>256 (T_RUNKOWN) Der Partner oder das CCP hat die Verbindung abgebaut. Ein Grund für den Abbau wurde nicht angegeben.</p> <p>257 (T_RSAP_CONGEST) Wegen eines TSAP-spezifischen Engpasses hat das Partner-CCP die Verbindung abgebaut.</p> <p>258 (T_RSAP_NOTATT) Das Partner-CCP hat die Verbindung abgebaut, weil der adressierte TSAP dort nicht angemeldet ist.</p> <p>259 (T_RUNSAP) Das Partner-CCP hat die Verbindung abgebaut, weil der adressierte TSAP dort nicht bekannt ist.</p> <p>261 (T_RPERMLOST) Abbau durch die Netzadministration oder Administration des Partner-CCP</p> <p>262 (T_RSYSERR) Fehler im Netz</p>

XPINI	Bedeutung
402 (Forts.)	<p>385 (T_RCONGEST) Das Partner-CCP hat die Verbindung wegen Betriebsmittelengpass abgebaut.</p> <p>386 (T_RCONNFAIL) Das Partner-CCP hat die Verbindung wegen Misslingens des Verbindungsaufbaus abgebaut.</p> <p>387 (T_RDUPREF) Weil für ein NSAP-Paar eine zweite Verbindungsreferenz vergeben wurde (Systemfehler), wurde die Verbindung vom Partner-CCP abgebaut.</p> <p>388 (T_RMISREF) Das Partner-CCP hat die Verbindung wegen einer nicht zuzuordnenden Verbindungsreferenz (Systemfehler) abgebaut.</p> <p>389 (T_PROTERR) Das Partner-CCP hat die Verbindung wegen eines Protokollfehlers (Systemfehler) abgebaut.</p> <p>391 (T_PREFOFLOW) Das Partner-CCP hat die Verbindung wegen Verbindungsreferenz-Überlauf abgebaut.</p> <p>392 (T_RNOCONN) Das Partner-CCP hat den Aufbau der Netzverbindung abgelehnt.</p> <p>394 (T_RINLNG) Das Partner-CCP hat die Verbindung wegen falscher Header- oder Parameterlänge (Systemfehler) abgebaut.</p> <p>448 (T_RLCONGEST) Das lokale CCP hat die Verbindung wegen Betriebsmittelengpass abgebaut.</p> <p>449 (T_RLNOQOS) Das lokale CCP hat die Verbindung abgebaut, weil "Quality of Service" nicht mehr unterstützt wird.</p> <p>451 (T_RILLPWD) Ungültiges Verbindungspasswort.</p> <p>452 (RNETACC) Netzzugang wurde verweigert.</p> <p>464 (T_RLPROTERR) Das lokale CCP hat die Verbindung wegen eines Transportprotokollfehlers (Systemfehler) abgebaut.</p> <p>465 (T_RLINTIDU) Das lokale CCP hat die Verbindung abgebaut, weil es eine zu lange Schnittstellen-Dateneinheit erhalten hat (Systemfehler).</p> <p>466 (T_RLNORMFLOW) Das lokale CCP hat die Verbindung wegen Verletzung der Flusskontrollregeln für Normaldaten (Systemfehler) abgebaut.</p>

XPINI	Bedeutung
402 (Forts.)	<p>467 (T_RLEXFLOW) Das lokale CCP hat die Verbindung wegen Verletzung der Flusskontrollregeln für Vorrangdaten (Systemfehler) abgebaut.</p> <p>468 (T_RLINSAPID) Das lokale CCP hat die Verbindung abgebaut, weil es eine ungültige TSAP-Identifikation erhalten hat (Systemfehler).</p> <p>469 (T_RLINCEPID) Das lokale CCP hat die Verbindung abgebaut, weil es eine ungültige TCEP-Identifikation (<b>T</b>ransport <b>C</b>onnection <b>E</b>nd <b>P</b>oint, Endpunkt einer Transportverbindung) erhalten hat (Systemfehler).</p> <p>470 (T_RLINPAR) Das lokale CCP hat die Verbindung wegen eines unzulässigen Parameterwerts (z. B. Benutzerdaten zu lang oder Vorrangdaten nicht zugelassen) abgebaut.</p> <p>480 (T_RLNOPERM) Die Administration des lokalen CCP hat den Verbindungsaufbau verhindert.</p> <p>481 (T_RLPERMLOST) Die Administration des lokalen CCP hat die Verbindung abgebaut.</p> <p>482 (T_RLNOCNN) Weil keine Netzverbindung verfügbar ist, konnte das lokale CCP den Verbindungsaufbau nicht durchführen.</p> <p>483 (T_RLCONNLOST) Das lokale CCP hat die Verbindung wegen Verlust der Netzverbindung abgebaut. Häufigste Ursache: Generierungsfehler auf CCP-Seite, z. B. unstimme Link-Adressen. Als Problemursache kommen außerdem in Frage: Partner ist nicht vorhanden, Modem ist defekt oder falsch eingestellt, DÜ-Anschluss ist nicht gesteckt, DFÜ-Board defekt.</p> <p>484 (T_RLNORESP) Das lokale CCP kann die Verbindung nicht aufbauen, weil der Partner nicht auf die Verbindungsanforderung (CONRQ) antwortet.</p> <p>485 (T_RLIDLETRAF) Das lokale CCP hat die Verbindung wegen Verbindungsverlust (Idle Traffic Timeout) abgebaut.</p> <p>486 (T_RLRESYNC) Das lokale CCP hat die Verbindung abgebaut, weil die Resynchronisierung (mehr als zehn Wiederholungen) erfolglos war.</p>
	<p>487 (T_RLEXLOST) Das lokale CCP hat die Verbindung abgebaut, weil der Vorrangdatenkanal defekt ist (mehr als drei Wiederholungen).</p>

XPINI	Bedeutung
403	<p>O_LOC_SESS Der Versursacher ist der lokale Session Provider. &amp;XP1INFO kann folgende Werte annehmen:</p> <p>4 (S_PROTERROR) Protokollfehler: fehlerhafter Aufbau der Session PDU oder fehlerhafter SPDU Parameter</p> <p>16 (S_PICSREST) Verstoß gegen implementierungsbedingte Beschränkungen.</p>
404	<p>O_REM_SESS Der Verursacher ist der entfernte Session Provider. &amp;XP1INFO kann folgende Werte annehmen:</p> <p>1 (S_TCDISCON) Abbau der Transportverbindung (transport disconnect)</p> <p>4 (S_PROTERROR) Protokollfehler</p> <p>8 (S_UNDEFINED) Undefiniert</p> <p>16 (S_PICSREST) Verstoß gegen implementierungsbedingte Beschränkungen</p>
405	<p>O_LOC_PRES Der Verursacher ist der lokale Presentation Provider. &amp;XP1INFO kann folgende Werte annehmen:</p> <p>0 (P_ARRNO) reason not specified</p> <ul style="list-style-type: none"> <li>– Ein intern angeforderter Dekodierpuffer kann wegen Speichermangel nicht bereitgestellt werden.</li> <li>– Überlauf des internen Datenpuffers beim Reassemblieren von fragmentierten Nachrichten.</li> <li>– Ein unbekanntes Session-Ereignis wurde angezeigt.</li> <li>– Systemengpass oder Systemfehler.</li> </ul> <p>1 (P_ARNRPDU) unrecognized PPDU</p> <ul style="list-style-type: none"> <li>– Es sind keine Session-Userdaten vorhanden oder es kann deren Presentation-Teil nicht dekodiert werden (Systemfehler).</li> </ul> <p>4 (P_ARNRPAR) unrecognized PPDU parameter</p> <ul style="list-style-type: none"> <li>– Fehler beim Dekodieren der ACSE-, Presentation- oder einer Benutzersyntax.</li> </ul> <p>5 (P_ARNEPAR) unexpected PPDU parameter</p> <ul style="list-style-type: none"> <li>– PPDU-Parameter nicht im normal mode.</li> </ul>

XPINI	Bedeutung
405 (Forts.)	6 (P_ARNIPAR) invalid PPDU parameter <ul style="list-style-type: none"> <li>– Ungültiger context identifier beim Dekodieren.</li> <li>– Ungültiger PPDU-Parameter, z. B. falsche Länge. Dieser „abort“ kann vom UTM-Benutzer durch die Angabe von ungültigen Presentation-/Session-Selektoren ausgelöst werden.</li> </ul>
406	O_REM_PRES Der Verursacher ist der entfernte Presentation Provider. &XP1INFO kann folgende Werte annehmen: -1 (O_NOVALUE) optional parameter is not present 0 (P_ARNNO) reason not specified 1 (P_ARNRPDU) unrecognized PPDU 2 (P_ARNEPDU) unexpected PPDU 3 (P_ARNESSP) unexpected session service primitive 4 (P_ARNRPAR) unrecognized PPDU parameter 5 (P_ARNEPAR) unexpected PPDU parameter 6 (P_ARNIPAR) invalid PPDU parameter value
407	O_LOC_ACSE Der Verursacher ist der lokale ACSE Provider &XP1INFO hat immer den folgenden Wert: 1 (A_ABSASP) ACSE-service-provider initiated the abort Es wird die Instanz angegeben, die aus ACSE-Sicht den “abort” ausgelöst hat (“abort source”).
408	O_REM_ACSE Der Verursacher ist der ferne ACSE-Service-Provider. &XP1INFO kann folgende Werte annehmen: 0 (A_ABSASU) ACSE-service-user initiated the abort 1 (A_ABSASP) ACSE-service-provider initiated the abort

**P016** Association (&XPOSAS ) abgebaut (a\_relin() ): &ACPNT, &OSLPAP, &XPLNK, &XPNDIA

Diese Meldung wird ausgegeben, wenn eine Association abgebaut wird, weil eine "release indication" empfangen wurde.

Die Inserts haben folgende Bedeutung:

&XPOSAS	Index der betroffenen Association
&ACPNT	Name des lokalen ACCESS-POINT
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPLNK	repräsentiert den internen Zustand der Association 0 = Association nicht verknüpft 1 = Association mit Channel verknüpft 2 = Association mit Instanz verknüpft
&XPNDIA	siehe Tabelle auf <a href="#">Seite 321</a>

**P017** OSS Dekodierfehler: &XPPDU, &XP1DIA, &XP2DIA, &XP3DIA

Diese Meldung wird ausgegeben, falls OSS beim Dekodieren einer TP-, CCR- oder Benutzerdaten-PDU einen Fehler erkennt. Das Insert &XPPDU kennzeichnet den Typ der betroffenen PDU.

Die Inserts haben folgende Bedeutung:

XPPDU	Bedeutung
0	Unbekannter PDU-Typ (PDU_UNKNOWN)
1	TP_BEGIN_DIALOGUE_RI
2	TP_BEGIN_DIALOGUE_RC
3	TP_BID_RI
4	TP_BID_RC
5	TP_END_DIALOGUE_RI
6	TP_END_DIALOGUE_RC
7	TP_U_ERROR_RI
8	TP_U_ERROR_RC
9	TP_ABORT_RI
10	TP_GRANT_CONTROL_RI
11	TP_REQUEST_CONTROL_RI
12	TP_HANDSHAKE_RI
13	TP_HANDSHAKE_RC
14	TP_HSK_AND_GRT_CTRL_RI

<b>XPPDU</b>	<b>Bedeutung</b>
15	TP_HSK_AND_GRT_CTRL_RC
16	TP_DEFER_RI
17	TP_PREPARE_RI
18	TP_HEURISTIC_REPORT_RI
19	TP_TOKEN_GIVE_RI
20	TP_TOKEN_PLEASE_RI
21	TP_RECOVER_RI
22	TP_INITIALIZE_RI
23	TP_INITIALIZE_RC
24	CCR_INITIALIZE_RI
25	CCR_INITIALIZE_RC
26	CCR_BEGIN_RI
27	CCR_BEGIN_RC
28	CCR_PREPARE_RI
29	CCR_READY_RI
30	CCR_COMMIT_RI
31	CCR_COMMIT_RC
32	CCR_ROLLBACK_RI
33	CCR_ROLLBACK_RC
34	CCR_RECOVER_RI
35	CCR_RECOVER_RC
50	PDU_ANY
51	PDU_UASE_RI

XP1DIA / XP2DIA	Bedeutung
1	not supported parameter was received and skipped
2	received data truncated
4	required transfer syntax name missing in user data or not specified in AVX list, error codes in &XP2DIA
6	no transfer syntax name in user data though presentation negotiation was not completed
7	transfer syntax name encoded in user data not found in AVX list
10	invalid value in data structure
11	invalid object identifier in data structure
12	invalid length or count in data structure
13	invalid index in data structure (EXTERNAL, CHOICE)
14	invalid value of ax_typtag in corresponding syntax table

&XP3DIA      zugehöriger Index in der Syntaxtabelle

**P018** FSM Protokollfehler: &ACPNT, &OSLPAP, &XPPTYP, &XPFSMN

Diese Meldung wird ausgegeben, wenn die Finite-State-Machine einen Fehler meldet.

Die Inserts haben folgende Bedeutung:

&ACPNT      Name des lokalen ACCESS-POINT

&OSLPAP      Name des Partners in der lokalen Anwendung

&XPPTYP      Typ des Service-Protokollelements

&XPFSMN      Name der Finite-State-Machine

**P019** APDU enthaelt ungueltigen Wert: &ACPNT, &OSLPAP, &XPAPDU, &XP3INFO

Diese Meldung wird ausgegeben, wenn eine ungültige APDU empfangen wurde.

Die Inserts haben folgende Bedeutung:

&ACPNT      Name des lokalen ACCESS-POINT

&OSLPAP      Name des Partners in der lokalen Anwendung

&XPAPDU      Typ der APDU

&XP3INFO      Zusatzinformation zum Fehler



**P020** OTRACE implizit ausgeschaltet. Grund: &XPTRFAIL

Diese Meldung wird ausgegeben, wenn der Versuch, einen Trace-Record zu schreiben, fehlgeschlagen ist. Als Folge des Fehlers wurde der OSS-Trace implizit ausgeschaltet. Nachdem der Fehler behoben worden ist, kann über die Administration der OSS-Trace wieder eingeschaltet werden.

Die Inserts haben folgende Bedeutung:

XPTRFAIL	Bedeutung
1	Die OSS-Funktion o_wutr() lieferte den Returncode O_ERROR. In der vorangegangenen P001-Meldung finden Sie nähere Informationen zu dem Fehler.
2	Die OSS-Funktion o_wutr() lieferte den Returncode O_INVEREF.
3	Die OSS-Funktion o_wutr() lieferte einen unbekanntem Returncode.

**P021** Unerwartetes Event &XPEVT fuer Association (&XPOSAS) aufgetreten, Event wird ignoriert: &ACPNT, &OSLPAP, &XPASST

Diese Meldung wird ausgegeben, wenn ein Ereignis auftritt, das nicht zum momentanen Zustand der Association passt. XAPTP berücksichtigt dieses Ereignis nicht.

Die Inserts der Meldung haben folgende Bedeutung:

&XPEVT	Typ des aufgetretenen Ereignisses
&XPOSAS	Index der betroffenen Association
&ACPNT	Name des lokalen Access Point
&OSLPAP	Name des Partners in der lokalen Anwendung
&XPASST	Zustand der betroffenen Association

## 5.2.1 Allgemeine Inserts der XAP-TP-Meldungen

XPRET	Bedeutung
2	Not first process of application
-1	Function call not successful due to permanent error.
-2	Function call not successful due to transient error. Retry the call later
-3	Function call not successful, data flow stopped Continue after event GO
-4	Session call: Expedited function call stopped due to expedited data flow control shortage Continue after event S_XGO/S_GO  Presentation call: Function call not successful, apref invalid  Local function call: Invalid connection reference  ACSE call: Function call not successful, apref resp. aref invalid
-5	Invalid waiting point reference
-6	Invalid application reference
-7	Waiting period to obtain a lock on a shared association expired

XPERR	Bedeutung
1	No memory available (temporary)
100	Call sequence error
101	Application not attached
102	Sending of data not allowed; wait for GO event
103	Internal error
104	Shared association is not locked
200	Missing ACSE/presentation reference
201	Invalid ACSE/presentation reference
202	Presentation call: missing AVX list (o_attach) ACSE call: missing application reference
203	Presentation call: invalid AVX list ACSE call: invalid application reference

<b>XPERR</b>	<b>Bedeutung</b>
204	Presentation call: invalid abstract syntax name in AVX ACSE call: missing ACSE parameters
205	Presentation call: invalid decoding mode in AVX ACSE call: missing presentation parameters
206	Presentation call: invalid user data length ACSE call: missing session parameters
207	Presentation call: invalid context id in p_udl ACSE call: missing application context name
208	Presentation call: invalid next parameter in p_udl ACSE call: invalid application context name
209	Presentation call: invalid pdv parameter in p_udl ACSE call: invalid calling AP Title
210	Presentation call: invalid chaining parameter ACSE call: invalid calling AE Qualifier
211	Presentation call: missing token parameter ACSE call: invalid called AP Title
212	Presentation call: invalid token parameter ACSE call: invalid called AE Qualifier
213	Presentation call: missing rtype parameter ACSE call: invalid responding AP Title
214	Presentation call: invalid rtype parameter ACSE call: invalid responding AE Qualifier
215	Presentation call: missing type parameter ACSE call: missing called p_address
216	Presentation call: invalid type parameter ACSE call: invalid called p_address
217	Presentation call: invalid syncp parameter ACSE call: missing calling p_address
218	Presentation call: missing syncp parameter ACSE call: missing responding p_address
219	Presentation call: invalid ctxlst parameter ACSE call: no mode parameter
220	Presentation call: invalid number of abstract syntaxes passed to OSS ACSE call: invalid mode parameter
221	Presentation call: invalid transfer syntax name ACSE call: missing result
222	Presentation call: invalid number of transfer syntaxes ACSE call: invalid result

XPERR	Bedeutung
223	Presentation call: invalid number of abstract syntaxes ACSE call: missing result source
224	Presentation call: same abstract syntax occurred already in transparent or non-transparent mode ACSE call: invalid result source
225	Presentation call: invalid data separation parameter ACSE call: invalid diagnostic
226	ACSE call: missing reason
227	ACSE call: invalid reason
228	ACSE call: missing provider reason
229	ACSE call: invalid provider reason
230	ACSE call: missing abort source
231	ACSE call: invalid p-requirements
232	ACSE call: invalid s-requirements
233	ACSE call: invalid syntax identifier
234	ACSE call: invalid p-context identifier
235	ACSE call: invalid p-context definition list
236	ACSE call: invalid p-context definition result list
237	ACSE call: invalid result in p-context definition result list
238	ACSE call: invalid default p-context result
239	ACSE call: invalid default p-context name
240	ACSE call: invalid user data length
241	ACSE call: invalid quality of service
242	ACSE call: invalid sync point serial number
243	ACSE call: invalid tokens
244	ACSE call: invalid SS-user reference
245	ACSE call: invalid SS-common reference
246	ACSE call: invalid SS-additional reference
250	Presentation call: ASN encoding error ACSE call: ASN encoding error
251	Presentation call: ASN decoding error ACSE call: ASN decoding error
252	Presentation call: ASN: invalid value in data struct ACSE call: ASN: invalid value in data struct

<b>XPERR</b>	<b>Bedeutung</b>
253	Presentation call: ASN: invalid object id in data struct ACSE call: ASN: invalid object id in data struct
254	Presentation call: ASN: invalid length in data struct ACSE call: ASN: invalid length in data struct
255	Presentation call: ASN: invalid index in data struct ACSE call: ASN: invalid index in data struct
256	Presentation call: ASN: invalid tag in syntax table ACSE call: ASN: invalid tag in syntax table
300	Presentation call: invalid protocol state ACSE call: invalid protocol state Local function call: error on system call
301	Presentation call: protocol error ACSE call: protocol error Local function call: error on transport system call
302	Local function call: error on local function call
305	Local function call: error on session call
306	Local function call: error on presentation call
307	Local function call: error on ACSE call

<b>XPRJCT</b>	<b>Bedeutung</b>
0	NO_REJECT
1	APPLICATION_CONTEXT_NAME_TOO_LONG Der vom Partner empfangene Objektbezeichner, der den Application Context Name bildet, enthält mehr Elemente, als von UTM unterstützt.
2	CALLING_APT_TOO_LONG Bei der Association Indication wird für den Application Process Title eine Länge angegeben, die UTM nicht unterstützt.
3	CALLING_AEQ_TOO_LONG Bei der Association Indication wird für den Application Entity Qualifier eine Länge angegeben, die UTM nicht unterstützt.
4	CALLED_APT_TOO_LONG Der gerufene Application Process Title ist länger als von UTM unterstützt.
5	CALLED_AEQ_TOO_LONG Der gerufene Application Entity Qualifier ist länger als von UTM unterstützt.
6	CONTEXT_DEFINITION_LIST_TOO_LONG Bei der Association Indication werden mehr abstrakte Syntaxen mitgegeben, als UTM unterstützt.

XPRJCT	Bedeutung
7	CONTEXT_RESULT_LIST_TOO_LONG Die beim Associationaufbau (Association indication bzw. confirmation) mitgegebene Liste der unterstützten abstrakten Syntaxen enthält mehr Elemente, als von UTM unterstützt.
9	ADDRESS_NO_PSAPINFO Die bei Association indication bzw. confirmation mitgegebene Adresse enthält keine Informationen zum PSAP.
10	ADDRESS_NO_INFOVERS_0_PSAP Die bei Association indication bzw. confirmation mitgegebene Adresse enthält für die PSAP-Information eine falsche Version.
11	ADDRESS_INVALID_P_SEL_LENGTH Die bei Association indication bzw. confirmation mitgegebene Adresse enthält für den Presentation-Selektor eine ungültige Länge.
12	ADDRESS_NO_SSAPINFO Die bei Association indication bzw. confirmation mitgegebene Adresse enthält keine Information zum SSAP.
13	ADDRESS_NO_INFOVERS_0_SSAP Die bei Association indication bzw. confirmation mitgegebene Adresse enthält für die SSAP-Information eine falsche Version.
14	ADDRESS_INVALID_S_SEL_LENGTH Die bei Association indication bzw. confirmation mitgegebene Adresse enthält keinen gültigen Teil für den Session-Selektor.
15	ADDRESS_NO_PARTNER_MODE Die bei Association indication bzw. confirmation mitgegebene Adresse enthält keinen gültigen Teil für den Network- und Transport-Selektor.
16	ADDRESS_TNSX_ERROR Die bei Association indication bzw. confirmation mitgegebene Adresse wird vom TNS abgewiesen.
17	UNKNOWN_PARTNER Die bei Association indication bzw. confirmation mitgegebene Adresse ist in der lokalen Anwendung nicht bekannt.
18	WRONG_APPLICATION_CONTEXT_NAME Der bei Association indication bzw. confirmation mitgegebene Application Context Name entspricht nicht dem in der lokalen Anwendung generierten Application Context Namen.
19	ABSTRACT_SYNTAX_MISSING Bei Association indication bzw. confirmation werden weniger abstrakte Syntaxen unterstützt, als in der lokalen Anwendung generiert wurden.
20	OSITP_SYNTAX_MISSING Bei Association indication bzw. confirmation wird die abstrakte Syntax für OSI TP nicht unterstützt.

<b>XPRJCT</b>	<b>Bedeutung</b>
21	NO_TP_INITIALIZE Bei Association indication bzw. confirmation wird keine TP-INITIALIZE-RI/RC PDU mitgeliefert.
22	OSITP_NO_VERSION_1 Der Partner unterstützt die Version 1 des OSI-TP-Protokolls nicht.
23	OSITP_RCH_WRONG_LENGTH Der bei der TP-INITIALIZE-indication bzw. TP-INITIALIZE-confirmation mitgegebene Recovery Context Handle hat eine Länge, die von UTM nicht unterstützt wird.
24	NO_CCR_INITIALIZE Die CCR-INITIALIZE-RI PDU fehlt.
25	CCR_NOT_VERSION_2 Der Partner unterstützt die Version 2 des CCR-Protokolls nicht.
26	SESSION_NO_FDX Die Session-Funktionalität "Full Duplex" ist nicht gesetzt.
27	SESSION_NO_DATA_SEPARATION Die Session-Funktionalität "Data Separation" ist nicht gesetzt, obwohl CCR im Kontext ist.
28	SESSION_NO_TYPED_DATA Die Session-Funktionalität "Typed Data" ist nicht gesetzt, obwohl CCR im Kontext ist.
29	SESSION_NO_MINOR_SYNCHRONIZE Die Session-Funktionalität "Minor Synchronize" ist nicht gesetzt, obwohl CCR im Kontext ist.
30	SESSION_NO_RESYNCHRONIZE Die Session-Funktionalität "Resynchronize" ist nicht gesetzt, obwohl CCR im Kontext ist.
31	TOKEN_CONTENTION_WINNER_AND_NO_TOKEN Die lokale Anwendung ist Contention Winner, aber nicht im Besitz des "Tokens" (nur wenn CCR im Kontext ist).
32	TOKEN_CONTENTION_LOSER_AND_TOKEN Die lokale Anwendung ist Contention Loser, aber im Besitz des "Tokens" (nur wenn CCR im Kontext ist).
33	INITIAL_SYNC_POINT_SERIAL_NUMBER_NOT_SET Die Initial Syncpoint Serial Number ist nicht gesetzt, obwohl CCR im Kontext ist.
34	NO_MORE_CONTENTION_LOSER_ASSOCIATIONS Der Associationsaufbau von außen wird abgelehnt, weil in der lokalen Anwendung alle Contention Loser Associationen bereits aufgebaut sind.
35	NO_MORE_CONTENTION_WINNER_ASSOCIATIONS Der Associationsaufbau von außen wird abgelehnt, weil in der lokalen Anwendung alle Contention Winner Associationen bereits aufgebaut sind.

XPRJCT	Bedeutung
36	CCR_BUT_NO_PARTNER_AET Der Partner hat keinen Application Entity Title angegeben, obwohl CCR im Kontext ist.
37	CCR_BUT_NO_OWN_AET In der lokalen Anwendung ist kein Application Entity Title angegeben, obwohl CCR im Kontext ist.
38	RESPONDING_APT_TOO_LONG Der in der Association confirmation angegebene Application Process Title ist länger, als von UTM unterstützt.
39	RESPONDING_AEQ_TOO_LONG Der in der Association confirmation angegebene Application Entity Qualifier ist länger, als von UTM unterstützt.
40	ASS_ESTABLISHMENT_TIMEOUT Der von der lokalen Anwendung begonnene Associationaufbau kann nicht in der vorgegebenen Zeit vollendet werden.
41	PARTNER_IS_IN_QUIET_STATE Der Associationsaufbau wird abgelehnt, weil der Partner in der lokalen Anwendung auf Quiet gesetzt ist.
42	NO_SPACE_FOR_RCH Der PutElement Aufruf zum Abspeichern des Recovery Context Handle liefert einen schlechten Returnwert.
43	REMOTE_AET_2_BIG Der Application Entity Title des Partners ist länger, als von UTM unterstützt.
44	REMOTE_AET_CHANGED Beim Aufbau von parallelen Associationen zu einem Partner wird vom Partner nicht derselbe Application Entity Title, wie für die erste aufgebaute Association zu diesem Partner, geliefert.
45	NO_SPACE_FOR_REMOTE_AET Der PutElement Aufruf zum Abspeichern des Application Entity Title des Partners liefert einen schlechten Returnwert.
46	PARTNER_HAS_STATUS_OFF Der Aufbau der Association wird abgelehnt, weil der Partner in der lokalen UTM-Anwendung gesperrt ist (es ist STATUS=OFF gesetzt).



<b>XPNDIA</b>	<b>Bedeutung</b>
0	NO_REASON_GIVEN
1	NO_COMMON_ACSE_VERSION Der Partner lehnt den Associationaufbauwunsch ab, weil es keine gemeinsame ACSE Version gibt.
2	APPL_CONXTX_NAM_NOT_SUPPORTD Der Partner lehnt den Associationaufbauwunsch ab, weil er den Application Context Name nicht unterstützt.
3	CALLING_AP_TITLE_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil beim Partner der Absender nicht richtig generiert ist (z.B. falscher N-SEL).  oder (nur bei heterogener Kopplung): Der Partner lehnt den Associationaufbauwunsch ab, weil er den Calling Application Process Title nicht kennt.
4	CALLING_AE_QUALI_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Calling Application Entity Qualifier nicht kennt.
5	CALLING_AP_INVOC_ID_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Calling Application Process Invocation Identifier nicht kennt.
6	CALLING_AE_INVOC_ID_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Calling Application Entity Invocation Identifier nicht kennt.
7	CALLED_AP_TITLE_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Called Application Process Title nicht kennt.
8	CALLED_AE_QUALI_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Called Application Entity Qualifier nicht kennt.
9	CALLED_AP_INVOC_ID_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Called Application Process Invocation Identifier nicht kennt.
10	CALLED_AE_INVOC_ID_NOT_RECON Der Partner lehnt den Associationaufbauwunsch ab, weil er den Called Application Entity Invocation Identifier nicht kennt.
11	PERMANENT_FAILURE Der Partner baut die Association ab, weil ein permanenter Fehler aufgetreten ist.
12	BEGIN_TRANSACTION_REJECT Der Partner baut die Association ab, weil er den Beginn einer Transaktion ablehnt.

<b>XPNDIA</b>	<b>Bedeutung</b>
13	TRANSIENT_FAILURE Der Partner baut die Association ab, weil ein vorübergehender Fehler aufgetreten ist.
14	PROTOCOL_ERROR Der Partner baut die Association ab, weil ein Protokollfehler aufgetreten ist.
15	UNRECOGNIZED_PDU Die Association wird von außen mit P-ABORT abgebaut, weil die Presentation Schicht eine unbekannte Presentation PDU erhalten hat.
16	UNEXPECTED_PDU Die Association wird von außen mit P-ABORT abgebaut, weil die Presentation Schicht eine unerwartete Presentation PDU erhalten hat.
17	UNEXPECTED_SESSION_SERVICE_PRIMITIVE Die Association wird von außen mit P-ABORT abgebaut, weil die Session Schicht eine unerwartete Session Service Primitive erhalten hat.
18	UNRECOGNIZED_PDU_PARAMETER Die Association wird von außen mit P-ABORT abgebaut, weil die Presentation Schicht einen unbekanntes PDU Parameter erhalten hat.
19	UNEXPECTED_PPDU_PARAMETER Die Association wird von außen mit P-ABORT abgebaut, weil die Presentation Schicht einen unerwarteten PPDU Parameter erhalten hat.
20	INVALID_PPDU_PARAMETER_VALUE Die Association wird von außen mit P-ABORT abgebaut, weil die Presentation Schicht einen ungültigen PPDU Parameterwert erhalten hat.
21	RELEASE_NORMAL Die Association wird vom Partner mit release abgebaut. Der Partner gibt als Grund release normal an.
22	RELEASE_URGENT Die Association wird vom Partner mit release abgebaut. Der Partner gibt als Grund release urgent an.
23	RELEASE_USER_DEFINED Die Association wird vom Partner mit release abgebaut. Der Partner gibt als Grund user defined an.
24	IDLE_TIMEOUT_ABORT Die Association wird von der lokalen Anwendung abgebaut, weil die Association in der mit IDLETIME generierten Zeit nicht benutzt worden ist.
25	ABORT_BY_ASS_INDICATION Die Association wurde von der lokalen UTM-Anwendung abgebaut, weil ein Association-Aufbauwunsch empfangen wurde und keine passende nicht aufgebaute Association verfügbar war.
26	ABORT_BY_ADM_OR_SHUT Die Association wurde von der lokalen Anwendung administrativ oder wegen Shutdown der Anwendung abgebaut.

### 5.3 Meldungen des Generierungstools KDCDEF

- K400** KDCDEF &DEFTYP &VERS wurde gestartet.
- K401** Bitte Steueranweisungen eingeben:
- K402** Syntaxfehler: &SYNERR.
- K403** Unzulaessiger Wert des Operanden &PARAM15.
- K404** DMS-Fehler &ERROR fuer Datei &FNAM.
- K405** Fehler &ERROR.
- K406** Unzulaessiger Wert des Operanden &PARAM15. &PARVAL8 wurde uebernommen.
- K407** User "&PARVAL8" konnte nicht in die Cluster-User-Datei eingetragen werden.
- K408** &STMTNAME-Name &UTMNAME darf nicht mit "KDC" beginnen.
- K409** LTERM-Parameter ist bei PTERM-Anweisung mit PTYPE=APPLI, SOCKET oder UPIC-R Pflicht.
- K410** Warnung: Das LTERM ist einem PTERM mit PTYPE=APPLI, SOCKET oder UPIC-R zugeordnet und ein ungeltiger USER oder USAGE=O ist angegeben.
- K411** Der Pflichtoperand &PARAM15 wurde in keiner &PARAM15 Anweisung angegeben.
- K412** DESTADM &UTMNAME ist weder als Asynchron-TAC noch als LTERM definiert. Es wurde kein DESTADM uebernommen.
- K413** &STMTNAME &OPERAN32 &STRING64 mehrfach definiert.
- K414** LSES-Anweisung fuer LPAP &UTMNAME fehlt.
- K415** Unerlaubte Angabe im MAX-Operanden DPUTLIMIT1 oder DPUTLIMIT2. Standardwert wurde angenommen.
- K416** &PARAM30 ist mehrfach definiert.
- K417** &PARAM10 aber Typ ist nicht "&PARAM30".
- K418** CID-Parameter ist Pflicht fuer ein PTERM das von einer Druckersteuerstation kontrolliert wird.
- K419** Ein Druckerbuendel wurde definiert aber der Typ ist nicht "OUTPUT".
- K420** Ein LTERM mit USAGE=D darf keinem PTERM mit USAGE=O zugeordnet sein.
- K421** &PARAM30 und &PARAM50 schliessen sich gegenseitig aus.
- K422** Der Pflichtoperand &PARAM10 wurde nicht angegeben.

- K423** Warnung: Anzahl der Pagepoolseiten (PGPOOL) nicht groesser als Anzahl der Seiten des LPUT-Puffers (LPUTBUF).
- K424** Ungueltiger Schluesselwert (KEY): &KEYVALUE. Zulaessig sind:  $0 \leq \text{KEY} \leq \text{MAX KEYVALUE}$ .
- K425** &STMTNAME-Name &PARAM32 nicht definiert.
- K426** Steueranweisung &STMTNAME fehlt.
- K427** Event-Exit &EXITNAME bereits definiert.
- K428** Ungueltiger CARD-Parameter: &CARDINFO oder POSITION = 0.
- K429** Maximale Anzahl von Datenbank-Typen ueberschritten.
- K430** Zu viele &STMTNAME-Steueranweisungen.
- K431** User "&PARVAL8" aus der Cluster-User-Datei fehlt in der KDCFILE.
- K432** Es wurde kein &STMTNAME mit &PARAM32 gefunden.
- K433** Name der Filebase zu lang. Der Generierungslauf wird abgebrochen.
- K434** Minimalgroesse fuer Wiederanlaufpuffer (Operand RECBUF) ist 1024: RECBUF = 1024 angenommen.
- K435** Der Event-Exit &PARAM8 muss einem ASSEMBLER-Programm zugeordnet werden.
- K436** MAX-Anweisung fehlt. Der Generierungslauf wird abgebrochen.
- K437** Fehler in KDCDEF; Grund: &TERMREAS.
- TERMREAS bezeichnet die Ursache, die zur abnormalen Beendigung des Dienstprogrammes führte.
- Zur Klärung des Fehlers dient auch die Beschreibung der UTM Meldung K060.
- Maßnahme: Fehlerbehebung laut Maßnahme zu K060.
- In den anderen Fällen liegt ein Fehler in KDCDEF vor. Unterlagen sicherstellen und PM schreiben.
- K438** Mehr LSES-Anweisungen als CON-Anweisungen fuer LPAP &UTMNAME angegeben.
- K439** Im Parameter KDCFILE wurde ein Dateiname mit Catalog-ID angegeben.
- K440** Operand &PARAM8 in &STMTNAME-Anweisung mit &PARAM10 nicht erlaubt.
- K441** SHARE-Tabelle &UTMNAME ist leer.
- K442** Operand &PARAM8 nur sinnvoll wenn &STMTNAME -Anweisung einem &PARVAL8-Teilprogramm zugeordnet.

- K443** ULS-Steueranweisungen nur sinnvoll, wenn die Anwendung mit USER-Anweisungen generiert wurde.
- K444** OPTION-Steueranweisung nur wirksam, wenn von SYSDTA gelesen wird.
- K445** &PARAM30 wird nicht mehr unterstuetzt.
- K446** Der Operand wurde ignoriert.
- K447** Die Steueranweisung wurde ignoriert.
- K448** Warnungen bei der KDCFILE-Generierung; KAA-Groesse: &KAASIZE K
- K449** Es ist mindestens ein Fehler aufgetreten. Der Generierungslauf wurde abgebrochen.
- K450** KDCFILE erzeugt; KAA-Groesse: &KAASIZE K
- K451** Datei &FNAM erzeugt.
- K452** Zu viele &PARAM10-Namen. Der Generierungslauf wird abgebrochen.
- K453** Von APPLNAME abweichender BCAMAPPL-NAME ist nur bei einer &STMTNAME-Anweisung mit PTYPE=APPLI, SOCKET oder UPIC-R erlaubt.
- K454** Laenge des LTERM-Parameters &UTMNAME mit NUMBER-Parameter unvertraeglich.
- K455** Ein dem Parameter STACK zugeordneter TAC darf nicht mit TYPE=A oder CALL=NEXT generiert werden.
- K456** Einem PTERM darf als LTERM kein Gruppen-LTERM zugewiesen werden.
- K457** Warnung: &PARAM10 groesser als &PARAM10A
- K458** LTERM &PARAM8 ist LTERM &PARAM8A als CTERM zugeordnet. Die Angabe "USAGE = OUTPUT" wirkt nicht.
- K459** Bei GROUP darf kein LTERM angegeben werden das selbst ein Gruppen-LTERM ist.
- K460** Datei &FNAM wurde bearbeitet.
- K461** &STMTTP30 &STMTNM30 nicht von &UTMVAR unterstuetzt.
- K462** In Verbindung mit der lokalen Partneranwendung &PARAM8 darf nur eine OSICON-Verbindungsgruppe aktiv sein.
- K463** &PARAM30 auf &PARAM30gesetzt.
- K464** Laenge von Session-Namen + Laenge von Verbindungen > 8.
- K465** &STMTNAME-Steueranweisung obligatorisch, wenn &PARAM30 generiert ist.
- K466** Parameter &PARAM30 ist zwingend, wenn &PARAM30 generiert ist.

- K467** Fehler: User "&PARAM8" ist mehrfach in der Cluster-User-Datei enthalten
- K468** Passwort-Parameter passt nicht zur Sicherheitsstufe.
- K469** Parameter &PARAM32 nur erlaubt bei &PARAM32
- K470** Kein Benutzer kann KDCSHUT aufrufen.
- K471** &OPERAND&PARAM10 bereits gesetzt.
- K472** &PARAM32 zu lang.  
Ist der Principal-Parameter zu lang, so wird der User ohne einen Principal-Eintrag generiert.
- K473** Parameter &PARAM10 mehr als einmal fuer &PARAM30 benutzt.
- K474** Parameter &PARAM30 unzuessaessig, wenn &PARAM30 generiert ist.
- K475** Parameter &PARAM30 unnoetig, wenn &PARAM30 generiert ist.
- K476** In &STMTNAME-Anweisungen wurde &OPERAND mehr als einmal fuer PRONAM=&STRING64 und BCAMAPPL=&PARAM8 angegeben.
- K477** Parameter &PARAM30 unzuessaessig, wenn ein &PARAM30 angegeben ist.
- K478** Anweisung &STMTNAME unzuessaessig, wenn &PARAM30 angegeben wurde.
- K479** &STMTNAME &PARAM50 nicht verwendet.
- K480** Zu viele &STMTNAME Steueranweisungen fuer diesen &PARAM30.
- K481** Das primary LTERM einer Gruppe darf nicht gleichzeitig Slave eines Verbindungsbaendels sein.
- K482** EXIT und TAC muessen im gleichen LOAD-MODULE sein, wenn die EXIT-Routine sich auf ein LOAD-MODULE mit LOAD-MODE=ONCALL bezieht.
- K483** &PARAM32 darf sich nicht auf ein &STMTNAME mit &PARAM32 beziehen.
- K484** &PARAM32 mit &PARAM32 nicht generiert.
- K485** Die angegebene Anweisung ist zu lang.
- K486** Es wurden unterschiedliche Bibliotheken fuer denselben Datenbanktyp angegeben.
- K487** Das Master LTERM eines Verbindungsbaendels darf nicht einem PTERM zugewiesen werden.
- K488** Alle Slave LTERMs eines Verbindungsbaendels muessen PTERMs mit identischem PTYPE (APPLI oder SOCKET) zugewiesen werden.
- K489** Das Master LTERM eines Verbindungsbaendels darf nicht seinerseits Slave LTERM sein.

- K490** Ein Master LTERM muss mit QAMSG=YES und RESTART=YES generiert werden. Die Parameter wurden korrigiert.
- K491** Die Angabe &OPERAND&PARAM32&PARAM20 wird in der Folgeversion nicht mehr unterstuetzt.
- K492** Hinweis: die Angabe von mehr als einer ACCESS-POINT Anweisung kann zu Problemen fuehren.
- K493** Das Erzeugen der TNS-Eingabedatei ist misslungen.
- K494** Der Operand &PARAM30 muss angegeben werden, wenn die Steueranweisung &STMTNAME mehr als einmal benutzt wird.
- K495** Der Standardmeldungsmodul wurde in keiner MESSAGE- Steueranweisung angegeben, oder er wurde durch keine MAX-Steueranweisung definiert.
- K496** Die Datei &FNAM ist zerstort oder keine KDCFILE.
- K497** KDCDEF &VER1 / &OST1 kann KDCFILE &VER2 / &OST2 nicht lesen.
- K498** Es sind keine &STMTNAME Steueranweisungen zu erzeugen.
- K499** &PARAM30 in &STMTNAME Steueranweisung und &PARAM30 passen nicht zusammen.
- K500** Es wurde(n) keine Datei(en) erzeugt.
- K501** Ungueltiges Zeichen "&CHAR1" in der Zeichenkette: &STRING64
- K502** Anzahl freier Eintraege fuer &PARAM10:&PARAM11
- K503** Der referenzierte ACCESS-POINT &PARAM8 muss einen APPLICATION-ENTITY-QUALIFIER enthalten, da das referenzierte OSI-LPAP &PARAM8 einen APPLICATION-CONTEXT mit der ABSTRACT-SYNTAX CCR enthaelt.
- K504** Das OSI-CON &PARAM8 referenziert denselben OSI-LPAP aber einen anderen ACCESS-POINT.
- K505** Der Operand &PARAM32 muss angegeben werden, wenn der referenzierte APPLICATION-CONTEXT die ABSTRACT-SYNTAX CCR enthaelt.
- K506** Es kann kein APPLICATION-CONTEXT angegeben werden, der die ABSTRACT-SYNTAX CCR enthaelt, da in der UTMD-Anweisung kein APPLICATION-PROCESS-TITLE spezifiziert wurde.
- K507** Es wurden zuviele &PARAM32 generiert.
- K508** Fehler beim Erzeugen der RSA-Keys.
- K509** Das OSI-CON &PARAM8 referenziert denselben LOCAL-ACCESS-POINT und ein OSI-LPAP mit identischen APPLICATION-ENTITY-QUALIFIER und APPLICATION-PROCESS-TITLE.

- K510** Der Operand &PARAM32 benoetigt mindestens &PARAM8 Parameter.
- K511** Es sind hoechstens &PARAM11 gleichzeitige gekellerte Vorgaenge moeglich.
- K512** Es sind hoechstens &PARAM11 gleichzeitige Anmelde- und/oder gekellerte Vorgaenge moeglich.
- K513** KDCDEF erzeugt jetzt neue RSA-Keys. Abhaengig von der Schluessellaenge und der Systemkonfiguration kann dieser Prozess einen laengeren Zeitraum in Anspruch nehmen.Bitte warten ...
- K514** Erzeugung der RSA-Keys abgeschlossen
- K515** Warnung: Die Anwendung benoetigt zum korrekten Ablauf RSA-Keys. Es wurden jedoch keine RSA-Keys generiert.

Wenn Objekte mit Encryption-Level generiert sind, dann benoetigt die Anwendung fuer den korrekten Ablauf RSA-Schlüssel. Sind in der Anwendung keine RSA-Schlüssel verfügbar, dann kann die Anwendung nur mit Einschränkungen betrieben werden: TACs mit Encryption-Level können nicht aufgerufen werden und zu PTERMs oder TPOOLS mit Encryption-Level kann keine Verbindung aufgebaut werden.

Drei Maßnahmen sind möglich:

- Vor dem Start der Anwendung können die RSA-Schlüssel per KDCUPD aus einer alten KDCFILE in die neue KDCFILE übertragen werden,
- oder der KDCDEF-Lauf kann mit OPTION GEN-RSA-KEYS=YES wiederholt werden,
- oder die Anwendung kann gestartet werden und die benötigten RSA-Schlüssel können administrativ erzeugt und aktiviert werden (z.B. durch WinAdmin oder WebAdmin).



**K516** Fehler &DIAG1 bei Bearbeitung der Cluster User-Datei &STRING64

Das Insert &DIAG1 beschreibt die Fehlerursache; das Insert &STRING64 enthält den Dateinamen.

Das Insert &DIAG1 kann folgende Werte annehmen:

Wert	Bedeutung
1	OPEN_ERROR Fehler beim Öffnen der Datei.
2	CLOSE_ERROR Fehler beim Schließen der Datei.
3	FILE_ERROR Die Datei ist inkonsistent.
4	LOCK_ERROR Fehler bei Anforderung des Datei-Locks.
5	UNLOCK_ERROR Fehler bei Freigabe des Datei-Locks.
6	USER_NOT_FOUND KCCUFUE gibt beim Update eines User-Eintrags einen schlechten Returncode zurück; der Name des Users ist im zweiten Insert enthalten
7	UNEXPECTED_RETURNCODE Von einer gerufenen Funktion wurde ein nicht erwarteter Returncode zurückgegeben; der Wert des Returncodes ist im zweiten Insert enthalten
8	KDCDEF_RUNNING Die Cluster-User-Datei ist bereits von einem KDCDEF-Lauf gesperrt
9	FILE_IS_EMPTY Die Cluster-User-Datei ist katalogisiert, aber leer
10	Fehler beim Aufruf getLocalHostName Der Fehlercode wird im zweiten Insert bereitgestellt.
11	NO_XCS_GROUP Der Rechner gehört zu keinem XCS-Verbund; die Funktion "distributed lock" steht damit nicht zur Verfügung.

**K517** &PARAM30 nicht unterstuetzt fuer &PARAM30

**K518** Die Datei &FNAM existiert bereits. Der Generierungslauf wird abgebrochen.

**K519** LMS Bibliothekselement erzeugt

Bibliothek: &amp;FNAM

Element: &amp;ELEM

Version: &amp;EVER

Typ: &amp;LMSTYPE

Bedeutung der Inserts:

&amp;FNAM Name der PLAM-Bibliothek, auf die sich die Meldung bezieht.

&amp;ELEM Name des LMS-Elements, auf das sich die Meldung bezieht.

&amp;EVER LMS-Version des LMS-Elements.

&amp;LMSTYPE LMS-Typ des LMS-Elements .

**K520** LMS Fehler (PLA&PLARC, DMS&DMSRC) fuer

Bibliothek: &amp;FNAM

Element: &amp;ELEM

Version: &amp;EVER

Typ: &amp;LMSTYPE

Bedeutung der Inserts:

&amp;PLARC PLAM-Returncode. Der Hilfetext kann mittels /HELP PLA&amp;PLARC erfragt werden.

&amp;DMSRC DMS-Returncode. Der Hilfetext kann mittels /HELP DMS&amp;DMSRC erfragt werden.

&amp;FNAM Name der PLAM-Bibliothek, auf die sich die Meldung bezieht.

&amp;ELEM Name des LMS-Elements, auf das sich die Meldung bezieht.

&amp;EVER LMS-Version des LMS-Elements.

&amp;LMSTYPE LMS-Typ des LMS-Elements .

**K521** MAX PRIVILEGED-LTERM=&UTMNAME ist nicht als Dialog-LTERM definiert.**K522** Hinweis: Es ist kein privilegiertes LTERM generiert (siehe MAX PRIVILEGED-LTERM=)

## 5.4 Meldungen der UTM-Tools KDCCSYSL / KDCPSYSL

- K600** Programm &PRGNMSG &VERS gestartet.
- K601** Programm KDCCSYSL normal beendet - SYSLOG-Datei umgewandelt.
- K602** Programm KDCPSYSL normal beendet - SYSLOG-Datei wurde aufbereitet.
- K603** Programm KDCCSYSL abnormal beendet - SYSLOG-Datei wurde nicht umgewandelt.
- K604** Programm KDCPSYSL abnormal beendet - SYSLOG-Datei wurde nicht aufbereitet.
- K605** Fehler &ERROR.
- K606** Datei mit dem Linknamen &LINK ist keine SYSLOG-Datei.
- K607** DMS-Fehler &DMSE auf der Datei mit dem Linknamen &LINK
- K608** Bitte Namen des Benutzermeldungsmoduls, Leerzeichen oder /EOF eingeben.
- K609** Benutzermeldungsmodul wird verwendet.
- K610** Systemmeldungsmodul wird verwendet.
- K611** Programmfehler in &PRGNMSG; Grund: &TRMA
- K612** Falsche Anzahl von Meldungen im Benutzermeldungsmodul.
- K613** Falsche Meldung &MSGNR im Benutzermeldungsmodul.

## 5.5 Meldungen der UTM-Tools KDCMMOD / KDCMTXT

- K650** Programm &PRGNMSG wurde abnormal beendet.
- K651** Programm Fehler (&TRMA)  
Maßnahme: PM schreiben.
- K652** DMS Fehler &DMSE fuer Datei &FNAM.
- K653** Zeile &LINENR: Pflichtoperand fehlt.
- K654** Zeile &LINENR: Syntax Fehler.
- K655** Zeile &LINENR: Insert &INSMMSG ist verboten fuer Meldung &IDMSG.
- K656** Zeile &LINENR: Konstante &CONMSG wurde nicht definiert.
- K657** Ende von SYSDTA wurde erreicht. END-Steueranweisungen wurden generiert.
- K658** \* \* \* \* \* Steueranweisung wurde ignoriert. \* \* \* \* \*
- K659** Datei &FNAM ist keine erlaubte Meldungsdefinitionsdatei.
- K660** Programm &PRGNMSG &VERS wurde gestartet.
- K661** Programm &PRGNMSG wurde normal beendet.
- K662** Zeile &LINENR: Funktionseinheit &FUMSG wurde nicht definiert.
- K663** Zeile &LINENR: Sprache &LANGMSG ist nicht definiert fuer Funktionseinheit &FUMSG.
- K664** Zeile &LINENR: Meldung &IDMSG ist verboten fuer Funktionseinheit &FUMSG.
- K665** Zeile &LINENR: GEN-Steueranweisung bereits eingegeben.
- K666** Zeile &LINENR: Meldungsziel &DESTMSG ist fuer Meldung &IDMSG verboten.
- K667** Zeile &LINENR: Meldungsziel &DESTMSG wird fuer Meldung &IDMSG gefordert.
- K668** Zeile &LINENR: GEN-Steueranweisung fehlt.
- K669** Zeile &LINENR: Funktionseinheit &FUMSG darf nicht modifiziert werden.
- K670** Quelldatei &FNAM fuer Meldungsmodul wurde erzeugt.
- K671** Quelldatei fuer Meldungsmodul wurde nicht erzeugt.
- K672** Zeile &LINENR: Konstante &CONMSG wurde bereits definiert.
- K673** Falsche Version &VERS der Meldungsdefinitionsdatei &FNAM.

**K686** Zeile &LINENR: Laenge &MSGLTH des Texts fuer Meldung &IDMSG ist laenger als &MSGMAXL.

Der Text der Meldung ist einschließlich der Inserts länger als 512 Zeichen. Die Dienstprogramme KDCMMOD und KDCMTXT können den Text nicht verarbeiten. KDCMMOD erzeugt keine Source für ein neues Meldungsmodul. KDCMTXT ergänzt die Meldungsdefinitionsdatei nicht.

**K687** Zeile &LINENR: Warnung - Laenge &MSGLTH des Texts fuer Meldung &IDMSG (SYSLINE) ist laenger als &MSGMAXL.

Der Text der Meldung (&IDMSG) mit dem Ziel SYSLINE ist einschließlich Inserts länger als 80 Zeichen. Die Dienstprogramme KDCMMOD und KDCMTXT akzeptieren den Meldungstext. Bei der späteren Ausgabe der Meldung in die Systemzeile gibt UTM nur die ersten 80 Zeichen des Meldungstextes aus.

**K688** Zeile &LINENR: Meldungsnummern > 999 sind verboten.

Es wurde eine Meldungsnummer größer 999 angegeben. Die Dienstprogramme KDCMMOD und KDCMTXT lehnen dies ab. KDCMMOD erzeugt keine Source für ein neues Meldungsmodul. KDCMTXT ergänzt die Meldungsdefinitionsdatei nicht.

**K690** Text fuer Meldung &IDMSG in Sprache &LANGMSG und Funktionseinheit &FUMSG wurde nicht definiert.

**K691** Meldungsdefinitionsdatei &FNAM wurde erzeugt.

**K692** Meldungsdefinitionsdatei wurde nicht erzeugt.

**K693** Meldungsdefinitionsdatei &FNAM wurde veraendert.

**K694** Meldungsdefinitionsdatei wurde nicht veraendert.

**K695** Zeile &LINENR: Insert &INSMSG wurde nicht definiert.

**K696** Zeile &LINENR: FU/Konstante/Insert &CONMSG wurde bereits definiert.

**K697** Zeile &LINENR: Falscher Wert fuer Quell-Laenge wurde durch Default-Wert ersetzt.

**K698** Zeile &LINENR: Bereich bereits fuer andere Funktionseinheit genutzt.

## 5.6 Meldungen des UTM-Tools KDCDUMP

Die Meldungen von KDCDUMP sind ausschließlich auf Englisch verfügbar.

**K700** KDCDUMP &VERS started.

**K701** Difference between the number of existing UTM tables and the number of UTM tables which are known by KDCDUMP.  
UTM area(s): &UTMA1&UTMA2&UTMA3&UTMA4&UTMA5&UTMA6&UTMA7

**K702** Abnormal end

**K703** Term application reason &TRMA

**K704** Table entry index not valid.

**K705** UTM dump &FNAM or part of it read in memory.

**K706** This type of preparation is not allowed for table &TABNAM.

**K707** Error on calling EDT. EDT call : &EDTINT Return code: &EDTRET

Diese Meldung wird ausgegeben, wenn EDT einen Fehler meldet. Das Insert EDTINT zeigt an, welche Unterprogrammchnittstelle des EDT den Fehler gemeldet hat (INF=IEDTINF bzw. CMD = IEDTCMD).

Das Insert EDTRET enthält den Returncode des EDT in der folgenden Reihenfolge: Subreturncode 1 (1 Byte), Subreturncode 2 (1 Byte), Mainreturncode (2 Byte). Erklärungen der Returncodes finden Sie in der EDT-Dokumentation.

**K708** EDT version not supported by KDCDUMP. EDT version: &EDTVER

**K709** Only one entry of a slot table can be prepared.

**K710** Error on command

**K711** No UTM dump in memory.

**K712** Requested address X"&SADDR" not in present UTM dump or selected domain.

**K713** Requested address X"&SADDR" is present in table &TABNAM, but not within one table entry.

**K714** Error on link &LNAM: DMS return code &DMSE.

**K716** Open error on file &FNAM. DMS return code : &DMSE.

**K717** Request memory error

**K718** File &FNAM does not contain a UTM dump.

**K719** KDCDUMP &VER1 cannot prepare a UTM dump &VER3.

**K720** Only the directory of the UTM dump exists.

- K721** &UTMA1 is not present in UTM-dump or in selected domain.
- K722** &NUM1 bourse cycle with the criterias CREF=&CREF and ANNO=&ANNO1 is found.
- K723** Write error on file &FNAM. DMS return code: &DMSE.
- K724** Command not allowed at present time.
- K725** Table index is too low or too high.
- K726** Address is too low or too high.
- K727** FSTAT: Error on file &FNAM. DMS return code: &DMSE.
- K728** Normal end
- K729** Output file &FNAM is written.
- K730** Read error on file &FNAM. DMS return code: &DMSE.
- K731** The name &DEFTYPE is not a valid UTM type.
- K732** UTM dump does not contain UTM type &DEFTYPE.
- K733** Table name &TABNAM does not exist.
- K734** The UTM dump &FNAM could not be written completely. Last written UTM area is &UTMA1.

Diese Meldung besagt, dass der UTM-Dumperzeuger den UTM-Dump nicht vollständig schreiben konnte. Es stand u.U. zu wenig Platz auf der Kennung zur Verfügung. Die UTM-Bereiche werden in folgender Reihenfolge abgezogen:

CAA, SLOT, CACHE, MGP, XAPTP-GLOBAL, KTA, XAPTP-LOCAL, STACK, ROOT, REP, User File, Journal File 1, Journal File 2, Buffer Segments, GSSB File, Lock File, CFG File, ULS File

Bei einem PENDER-Dump werden nur ROOT und REP abgezogen.

Die einzelnen Tabellen zu den UTM-Bereichen lassen sich durch die KDCDUMP-Anweisung HELP TABLE-NAMES ([Seite 72](#)) ausgeben, und zwar in der Reihenfolge des Speicherabzugs.

Die XAPTP-Teile bilden hier einen Block.

- K735** Table &TABNAM does not exist in UTM dump or in selected domain.
- K736** Index out of range - index is set to lowest or highest allowed table index.
- K737** End-index lower than start-index - end-index is given the value of the start-index.
- K738** No REP file is contained in UTM dump.
- K739** No REP records ( PM-NO: &PMNO ) found.

- K740** Abbreviation of table name ambiguous with regard to &TABNAM &TABNAM1 &TABNAM2 &TABNAM3 &TABNAM4 &TABNAM5.
- K741** Information not available. Table name: &TABNAM  
Die Meldung besagt, dass ein UTM-Bereich oder eine Tabelle vom UTM-Dump-Erzeuger nicht abgezogen werden konnte. Der Grund ist, dass der UTM-Bereich oder die UTM-Tabelle nicht vollständig zugreifbar waren.
- K742** The KDCDUMP version on the hardware &HW1 with &OS1 as operating system cannot read a UTM dump written on the hardware &HW2 with the operating system &OS2.
- K743** For the table &TABNAM no displacement is possible.
- K744** Offset lower than 0 or higher than the no. of bytes for one table entry.
- K745** No symbolic preparation for this table &TABNAM possible.
- K746** Error on symbolic preparation.
- K747** Command has no result.
- K753** The name &FIRES is not a UTM resource of UTM type &DEFTYPE.
- K754** No further entry with this UTM type in this dictionary table.
- K755** Input error: No further command can be read.
- K756** False value for DB operand. Only &DBMAX database(s) are generated.
- K757** The version of &UTMA1 (&VER1) is not compatible with the version of KDCDUMP (&VER2).
- K758** No summary was written.
- K759** Input string is longer than the permitted length (256 char).
- K760** No entry with the name &FIRES found.
- K761** No further entry with the name &FIRES found.
- K770** Error detect while checking compression of file &FNAM. Error code: &ERRC.
- K771** Error detect while creating temporary file for &FNAM. Error code: &ERRC.
- K772** Error detect while decompressing file &FNAM. Error code: &ERRC.
- K773** Requested information can not be found in the selected domain.
- K774** Requested domain for command SFIND not read from dump file.  
Empfehlung: Domain einlesen mit FILE=<dumpfile>, DOMAIN=<domain> und SFIND-Befehl wiederholen.
- K780** No dump file was closed.



- K781** Compressed dump file could not be uncompressed.
- K782** No hit found.
- K783** Value of HITS is neither ALL nor between 0 and 32767.
- K784** Size of address greater than 32 bit is forbidden.

## 5.7 Meldungen des UTM-Tools KDCUPD

Das UTM-Tool KDCUPD gibt sowohl Meldungen des Transaktionsmonitors als auch eigene Meldungen aus. Bei allen Meldungen von KDCUPD, bei denen der Text mit einem „\*“ (Stern) beginnt, konnten die Daten nicht übertragen werden.

**K800** KDCUPD &VERS gestartet.

**K801** Bitte Steueranweisungen eingeben

(Diese Meldung wird **nur auf SYSOUT** ausgegeben!)

**K802** Steueranweisung KDCFILE/CHECK: Parameter &UPDCMD ist nicht angegeben.

**K803** Basisnamen fuer alte und neue &PARAM17 muessen unterschiedlich sein.

**K804** Uebertragung von UTM &UPDVERS nach UTM &UPDVERS nicht unterstuetzt.

**K805** Konsistenz-Pruefung in Ordnung fuer Filebase &FBASUPD.

**K806** Alle gewuenschten Daten wurden uebertragen.

**K807** Gewuenschte Daten wurden teilweise uebertragen.

**K808** Keine Daten aus &FNAM zu transferieren.

Die bei Parameter OLD angegebene Datei stammt nur aus einer KDCDEF-Generierung. Ein KDCUPD-Lauf ist daher nicht sinnvoll!

**K809** Steueranweisung KDCFILE/CHECK: Basisname &FBASUPD ist zu lang.

Maßnahme: Datei umbenennen.

**K810** Kombination der Kommandos CHECK und KDCFILE ist nicht erlaubt.

**K811** Steueranweisung KDCFILE: Filebase muss ohne CATID angegeben werden, wenn der CATID-Parameter benutzt wird.

**K812** Abnormale Beendigung von KDCUPD.

**K813** Normale Beendigung von KDCUPD.

**K814** &TRANSFER - Transfer unterschiedlicher Versionen nicht unterstuetzt.

**K815** Steueranweisung CLUSTER-FILEBASE OLD fuer UTM Version &UTMVER nicht erlaubt

**K816** Hinweis: Datenbank Generierung geaendert, offene Vorgaenge werden nicht uebertragen

**K851** &UPDTYP-Daten uebertragen. KCRN = &UKCRN,&PARAM4 = &UKCLA.

**K852** &PARAM3-Daten uebertragen. KCRN = &UKCRN,&PARAM4 = &UKCLT, KCLA = &UKCLA.

**K853** Datenbank-Generierung. Nummer: &DBCOUNT FILEOLD: &DBOLD FILENEW:  
&DBNEW

Die Generierung von FILEOLD und FILENEW unterscheidet sich unerlaubt bei der Datenbankgenerierung, die Meldung kommt so oft wie die maximale Anzahl der Datenbanken in OLD/NEW definiert ist.

**K854** Warnung: Die Komplexitätsstufe fuer USER &USER hat sich geaendert. Passwort kann ungueltig werden.

Die Komplexitätsstufe des Passworts für den User &USER ist in der neuen KDCFILE höher als in der alten KDCFILE, siehe USER-Anweisung in KDCDEF-Generierung. Wenn das übertragene Passwort diese Bedingung nicht erfüllt, kann sich der Benutzer nicht anmelden.

Maßnahme: Der Administrator muss dann ein neues Passwort vergeben.

**K855** \* &UPDTYP-Daten nicht uebertragen. KCRN = &UKCRN, &PARAM4 = &UKCLA,  
KCRCCC = &RCCC, KCRCDC = &RCDC.

**K856** \* &PARAM3-Daten nicht uebertragen. KCRN = &UKCRN, &PARAM4 = &UKCLT,  
KCLA = &UKCLA, KCRCCC = &RCCC, KCRCDC = &RCDC.

**K857** \* Load Module &PROG nicht gefunden. Aktuelle Version &PVER nicht uebertragen.

**K858** Aktuelle Version &PVER von Load Module &PROG uebertragen.

## 5.8 Standard-Meldungsdefinitionsdatei

### 5.8.1 Konstanten der Standard-Meldungsdefinitionsdatei

Konstantenname	Konstantenwert	Bemerkung
DAR	12	DARK LAYOUT
DEL	07	DELETE
EM3	13	EMPHASIZED LAYOUT
EPA	08	END PROTECTED AREA
NL	15	NEW LINE
NOR	1E	NORMAL LAYOUT
NP	0C	NEW PAGE
SPA	36	START PROTECTED AREA

## 5.8.2 Inserts in Meldungen

Die Spalte „Länge“ enthält die Ausgabelänge der einzelnen Inserts in Byte, d.h. das Insert belegt bei der Ausgabe des Meldungstexts soviele Zeichen, wie in der Spalte „Länge“ angegeben. Die Länge der Inserts ist insbesondere für die Erstellung von Meldungstexten mit KDCMMOD und KDCMTXT von Bedeutung.

Die Abkürzungen in der Spalte 'Daten-Typ' bedeuten dabei folgendes:

Char abdruckbare Zeichen  
 Int numerisches Feld  
 Hexa hexadezimale Information

Insert-Name	Datentyp	Länge	Bedeutung
AAID	Hexa	128	FIRST 64 BYTE OF ATOMIC ACTION IDENTIFIER
AAIS	Int	4	ATOMIC ACTION IDENTIFIER SIZE
ACPNT	Char	8	ACCESS-POINT-NAME
ACTION	Char	6	SYSTEM ACTION
ADTC	Char	8	ADMINISTRATION TAC
AGUS	Char	8	JOB-SUBMITTING USER
AMOD	Char	1	APPLICATION MODE
ANNO	Hexa	64	ANNO RECEIVED
APPL	Char	8	APPLICATION NAME
ATAC1	Char	8	ASYNCHRONOUS TAC
ATAC2	Char	10	NUMBER OF UNPROCESSED ASYNCHRONOUS TACS
ATTR	Char	11	ATTRIBUT OF LOAD-MODULE/PROGRAM
ATYP	Char	1	APPLICATION TYPE (STANDALONE/CLUSTER)
BCAP	Char	8	BCAM APPLICATION NAME
BCMOPCD	Hexa	8	BCMM-OPCODE
BCMRTCD	Hexa	8	BCMM-RETURNCODE
BMD1	Char	8	BIT MODE OF SYSTEM
BMD2	Char	8	BIT MODE OF SYSTEM
CAMODE	Hexa	2	CONXTXT MACRO: ADDRESS MODE
CBRC	Hexa	8	VTSUCB RETURN CODE
CCC	Hexa	2	CONXTXT MACRO: CONDITION CODE IN PCR FORMAT
CCSN	Char	8	CCSNAME
CDSPL	Char	19	CONTEXT MODULE + DISPLACEMENT
CHAIN	Char	3	CHAINED MESSAGE INFORMATION

Insert-Name	Datentyp	Länge	Bedeutung
CID	Char	8	PRINTER CONTROL ID
CILC	Hexa	2	CONTXT MACRO: INSTRUCTION LENGTH IN PCR FORMAT
CIW	Hexa	2	CONTEXT INTERRUPT WEIGHT
CLSIGT	Int	2	CLUSTER COMMUNICATION SIGNAL TYPE
CMD	Char	8	COMMAND NAME
CNTR	Char	6	NUMBER OF LPUT RECORDS
CON	Char	8	CONNECTION NAME
COND	Char	3	CONDITION
CONU	Char	10	NUMBER OF CONNECTED USERS
COTM	Int	10	ELAPSED CONNECTION TIME IN SECONDS
CPC	Hexa	8	CONTEXT PROGRAM COUNTER
CPM	Hexa	2	CONTXT MACRO: PROGRAM MASK IN PCR FORMAT
CPTM	Int	10	CPU TIME SINCE SIGN-ON IN MILLISECONDS
CPUBEGIN	Hexa	8	CPU TIME AT TAC START IN MILLISECONDS
CPUCLNT	Hexa	8	CPU TIME USED OF THIS CLIENT
CPUEND	Hexa	8	CPU TIME AT TAC END IN MILLISECONDS
CPUREAS	Char	1	INTERNAL REASON
CPUTEXT	Char	8	TEXT: OVERFLOW OR NEGATIV
CPUUSED	Hexa	8	CPU TIME USED OF THIS TAC IN MILLISECONDS
CRYP	Char	1	CRYPTO PACKAGE
CR0	Hexa	8	CONTXT MACRO: REGISTER R0
CR1	Hexa	8	CONTXT MACRO: REGISTER R1
CR10	Hexa	8	CONTXT MACRO: REGISTER R10
CR11	Hexa	8	CONTXT MACRO: REGISTER R11
CR12	Hexa	8	CONTXT MACRO: REGISTER R12
CR13	Hexa	8	CONTXT MACRO: REGISTER R13
CR14	Hexa	8	CONTXT MACRO: REGISTER R14
CR15	Hexa	8	CONTXT MACRO: REGISTER R15
CR2	Hexa	8	CONTXT MACRO: REGISTER R2
CR3	Hexa	8	CONTXT MACRO: REGISTER R3
CR4	Hexa	8	CONTXT MACRO: REGISTER R4
CR5	Hexa	8	CONTXT MACRO: REGISTER R5
CR6	Hexa	8	CONTXT MACRO: REGISTER R6

Insert-Name	Datentyp	Länge	Bedeutung
CR7	Hexa	8	CONXTT MACRO: REGISTER R7
CR8	Hexa	8	CONXTT MACRO: REGISTER R8
CR9	Hexa	8	CONXTT MACRO: REGISTER R9
CTYP	Char	4	TYPE OF PROGRAM EXCHANGE
DBCALL	Char	12	FUNCTION-CALL OF IUTMDB INTERFACE
DBCON	Char	8	DATABASE CONNECTION MODULE
DBTRAC	Hexa	8	DB TRACE INFO
DBV1	Char	8	VERSION OF DB CONNECTION MODULE
DBV2	Char	8	VERSION OF KDCDB MACRO
DEFVER	Char	5	VERSION NUMBER IN KAA
DEST	Char	8	DESTINATION OF ASYNCHRONOUS MSG
DEVC	Hexa	2	DEVICE TYPE
DIA1	Int	11	DIAGNOSTIC INFORMATION
DIA2	Int	11	DIAGNOSTIC INFORMATION
DIA3	Int	11	DIAGNOSTIC INFORMATION
DIA5	Char	80	INTERNAL DIAGNOSTIC INFORMATION
DLDATE	Char	3	DAY OF KDCS CALL PADM DL/DA
DLTIME	Char	8	TIME OF KDCS CALL PADM DL /DA
DMSE	Char	4	DMS ERROR CODE
DMSRC	Hexa	4	DMS RETURN CODE
DPID	Char	8	ASYNCHRONOUS MESSAGE ID
DTM2	Char	18	TIME STAMP 2
DTTM	Char	18	TIME STAMP
EBSR	Char	4	ACTUAL BRACKET STATE
EBSS	Char	4	SAVED BRACKET STATE
ELEM	Char	64	ELEMENT NAME
ENCPW	Hexa	32	ENCRYPTED PASSWORD
ERCD1	Hexa	4	ERROR CODE (IUTMHLL)
ERCD2	Hexa	4	INFO RETURN CODE (IUTMHLL)
ERCD3	Char	4	ERROR CODE (IUTMDB)
ERCD4	Char	4	ERROR CODE (IUTMFORM)
ERCD5	Char	4	INFO RETURN CODE (IUTMFORM)
ERCD6	Char	4	ERROR CODE (ROOT)

Insert-Name	Datentyp	Länge	Bedeutung
ERPRT	Char	1	PRINT ERROR CODE
ERRCODE	Char	16	RETURN CODE OF FAULTY FUNCTION
ERRNAME	Char	8	NAME OF FAULTY FUNCTION
ESQR	Hexa	10	ACTUAL REQUEST SEQUENCE NUMBER
ESQS	Hexa	8	SAVED SEQUENCE NUMBER
ESRR	Hexa	10	ACTUAL RESPONSE SEQUENCE NUMBER
EVER	Char	24	ELEMENT VERSION
EXIT	Char	10	CURRENT ACTIVE EXIT
FBASUPD	Char	42	FILE BASE NAME KDCUPD
FIL1A	Hexa	2	APPLICATION STATE
FIL1B	Hexa	2	BCAM REQUEST OR ANNO TYPE / UTM ANNO TYPE
FIL2A	Hexa	2	LTERM STATE
FIL2B	Hexa	8	BCAM INFOWORD
FIL3	Hexa	4	PTERM STATE
FMH7	Char	80	ERROR RECOVERY PROCEDURE MESSAGE
FMTN	Char	8	FORMAT NAME
FNAM	Char	54	FILE NAME
FNKT	Char	6	FUNCTION
FNOD	Char	1	FIRST NODE (Y/N) IN CLUSTER APPLICATION
FORM	Char	8	FORMAT NAME (FOR K015 ONLY)
FORMATID	Int	1	FORMATID OF THE XID
GBLNBR	Int	11	NUMBER OF LOCKED GSSB
GLOBALSG	Char	1	CLUSTER GLOBAL SIGNON/SIGNOFF
GNDATE	Char	3	GENERATION DATE ASYNCHRONOUS MESSAGE
GNTIME	Char	8	GENERATION TIME ASYNCHRONOUS MESSAGE
GNUSER	Char	8	USER NAME OF ASYNCHRON. MESSAGE GENERATION
GTRID	Hexa	128	FIRST 64 BYTE OF GLOBAL TRANSACTION ID
HITR	Char	3	CACHE HIT RATE
HSTACK	Int	2	HEIGHT OF STACK
HST1	Char	64	HOST NAME
HST2	Char	64	HOST NAME
HST3	Char	64	HOST NAME
IDEFRC	Hexa	16	RETURN CODE OF INVERSE KDCDEF



Insert-Name	Datentyp	Länge	Bedeutung
IDX1	Char	4	CLUSTER NODE INDEX
IDX2	Char	4	CLUSTER NODE INDEX
IDX3	Char	4	CLUSTER NODE INDEX
IMPVER	Char	5	VERSION NUMBER IN KAA OF KDCFILE TO IMPORT
IMSG1	Char	10	NUMBER OF TERMINAL INPUT MESSAGES
IMSG2	Hexa	64	FIRST PART OF INPUT MESSAGE
INF1	Char	65	ADDITIONAL INFORMATION
INF2	Char	65	ADDITIONAL INFORMATION
INSTNUM	Int	2	RM-INSTANCE NUMBER
INTTAID	Hexa	130	INTERNAL TRANSACTION ID
IOMS	Int	11	DURATION OF IO IN MILLISECONDS
IOPG	Int	11	NUMBER PAGES OF IO
IPADDR	Char	39	IPV4 (123.456.789.012) OR IPV6 (1234:5678:9ABC:DEF0:1234:5678:9ABC:DEF0) ADDRESS
KRACT	Int	1	ACTION CODE OF KERBEROS INTERFACE KRBGSS
KRLH	Int	4	LENGTH OF KERBEROS PRINCIPAL
KRPR	Char	100	KERBEROS CLIENT OR SERVER PRINCIPAL
KRRC	Hexa	8	RETURN CODE OF KERBEROS INTERFACE KRBGSS
LMSTYPE	Char	8	LMS TYPE
LPAP	Char	8	LPAP NAME
LSES	Char	8	LSES NAME
LTAC	Char	8	TAC OR LTAC
LTACINDX	Hexa	8	LTAC INDEX
LTHGTRID	Int	2	LENGTH OF GLOBAL TRANSACTION ID
LTRM	Char	8	LTERM NAME
LWRT	Char	5	NUMBER OF USLOG FILE WRITES
MOD	Char	7	MODULE NAME
MSG	Char	80	MESSAGE TEXT
MSG2	Char	100	MESSAGE TEXT
MSTACK	Int	2	MAXIMUM STACK HEIGHT
MTYPE	Char	4	MESSAGE TYPE
MXLT	Char	8	MUX LTERM
MXPR	Char	8	MUX PROCESSOR

Insert-Name	Datentyp	Länge	Bedeutung
MXPT	Char	8	MUX PTERM
MPX1	Char	4	MUX PROTOCOL VERSION (LOWER BOUNDARY)
MPX2	Char	4	MUX PROTOCOL VERSION (UPPER BOUNDARY)
NCVST	Char	1	NEW CONVERSATION STATE
NMSG	Int	11	NUMBER OF MESSAGES
NNM1	Char	8	CLUSTER NODE NAME
NNM2	Char	8	CLUSTER NODE NAME
NNM3	Char	8	CLUSTER NODE NAME
NTAST	Char	1	NEW TRANSACTION STATE
NUMDAYS	Char	2	NUMBER DAYS PASSWORD VALID
NUMMSGs	Int	11	NUMBER OF WAITING OUTPUT MESSAGES
OBJ1	Char	10	OBJECT NAME
OBJ2	Char	10	OBJECT NAME
OBJ3	Char	54	OBJECT OR FILENAME
OCVST	Char	1	OLD CONVERSATION STATE
OMSG1	Char	10	NUMBER OF TERMINAL OUTPUT MESSAGES
OMSG2	Char	74	BROADCAST MESSAGE
OMSG3	Hexa	64	FIRST PART OF OUTPUT MESSAGE
OPCD1	Char	4	OPCODE
OPCD2	Char	35	OPCODE (IUTMHLL)
OPCD3	Char	5	OPCODE (IUTMFORM)
OSLPAP	Char	8	OSI-LPAP NAME
OST1	Char	24	TYPE OF OPERATING SYSTEM
OST2	Char	24	TYPE OF OPERATING SYSTEM
OTAST	Char	1	OLD TRANSACTION STATE
PALTRM	Char	8	LTERM NAME PRINT ADMIN STATION
PAS1	Char	35	SPACE FOR PASSWORD
PAS2	Char	35	SPACE FOR PASSWORD
PAS3	Char	35	SPACE FOR PASSWORD
PGPOOL	Char	16	(NODE/CLUSTER) PAGEPOOL
PGS1	Int	11	NUMBER OF UTM PAGES
PGS2	Int	11	NUMBER OF UTM PAGES
PHAXAPTP	Char	14	INIT or START/RESTART of XAP-TP

Insert-Name	Datentyp	Länge	Bedeutung
PID	Int	11	BS2000 TASK ID (ITN)
PLARC	Char	4	PLAM RETURN CODE
PRCN	Char	200	PROCEDURE/SCRIPT/COMMAND-FILE NAME
PRNM	Char	64	PROCESSOR NAME
PROG	Char	32	PROGRAM OR LOAD MODULE NAME
PSQN	Hexa	8	SAVED PET SEQUENCE NUMBER
PTCID	Char	27	PTC IDENTIFICATION
PTRM	Char	8	PTERM NAME
PVER	Char	24	PROGRAM VERSION
RBCAUSER	Char	14	CAUSER OF ROLLBACK
RCCC	Char	3	KCRC
RCCC2	Char	4	STARTUP ERROR CODE
RCDC	Char	4	KCRC
RCF1A	Char	4	KCRC
RCF1B	Char	3	RETURN CODE 1
RCF1C	Char	4	RETURN CODE 1
RCF2A	Char	4	SECONDARY FHS/VTSU RET CODE
RCF2B	Char	4	RETURN CODE 2
RCHX	Hexa	8	RETURN CODE IN HEX-FORM
RCVDANNO	Hexa	8	FIRST 4 BYTES OF RECEIVED ANNO
RXCXPTP	Int	3	RETURN CODE XAP-TP STARTFUNCTIONS
REA1	Hexa	2	REASON
REA2	Char	2	REASON
REA3	Char	136	ERROR MESSAGE
REA4	Char	1	DIAGNOSTIC INFORMATION
REA6	Hexa	2	DIAGNOSTIC INFORMATION (DISCONNECT USER REASON)
REA7	Int	2	DIAGNOSTIC INFORMATION (REJECT USER REASON)
REST	Char	1	RESTART INDICATOR OF LTERM
RMSTAT	Char	8	CONNECTION STATUS OF AN RM
ROOTTRMR	Int	11	ROOT-Termination-Reason
RQM	Int	11	REQUESTED NUMBER OF BYTES
RSES	Char	8	RSES NAME

Insert-Name	Datentyp	Länge	Bedeutung
RSLT	Char	1	RESULT
RSOANNO	Hexa	64	RSO ANNO
RSOMSG	Char	7	RSO ERROR MESSAGE
RSOOPC	Hexa	2	RSO ACTION
RSORC	Hexa	8	RSO RETURNCODE
RSO2RC	Hexa	8	RSO ASYN RETURNCODE
RSPTC	Char	1	RESET-PTC (Y/N) FOR NODE RECOVERY
RTAANZ	Int	2	NUMBER OF RECOVERED TRANSACTIONS
RTCD	Hexa	8	RETURN CODE
SATRC	Hexa	8	SAT RETURNCODE
SESSCNTR	Int	3	SESSION COUNTER OF ACTUAL SERVICE
SGNT	Char	19	SIGNON TIME YYYY-MM-DD HH:MM:SS
SNLT	Char	8	MUX SESSION LTERM NAME
SNPR	Char	8	MUX SESSION PROCESSOR NAME
SNPT	Char	8	MUX SESSION PTERM NAME
SRFG	Hexa	8	SAVED SESSION STATE
SSINCID	Hexa	8	SSO CALL CID
SSLENNAM	Hexa	4	SSO NAME LENGTH
SSORC	Hexa	2	SSO RETURNCODE
SSOUTCID	Hexa	8	SSO RETURN CID
SSTYPNAM	Hexa	8	SSO NAME TYPE
STATE	Char	1	STATE
STA2	Char	1	STATE
STDHEAD	Hexa	16	BS2000 STANDARDHEADER
STDRC	Hexa	8	BS2000 STANDARDHEADER-RETURNCODE
STMT	Char	11	STATEMENT OF KDCDEF
STRTPAR1	Char	20	STRTPAR1 LTH = 20
STRTPAR2	Char	10	STRTPAR2 LTH = 8
STSK	Char	1	SYSTEM TASK (Y/N)
STS1	Hexa	4	STSN-REQ SEQUENCE NUMBER RCV-CNT
STS2	Hexa	4	STSN-REQ SEQUENCE NUMBER SEND-CNT
STS3	Hexa	4	STSN-RSP SEQUENCE NUMBER SLU-PLU
STS4	Hexa	4	STSN-RSP SEQUENCE NUMBER PLU-SLU

Insert-Name	Datentyp	Länge	Bedeutung
SUFF	Char	5	FILE SUFFIX
SWNR	Int	11	NUMBER OF JOURNAL SWITCHES
SYN	Char	50	SYNTAX ERROR
SYSD	Hexa	4	SYSTEM SENSE DATA
SYSFILE	Char	6	SYSOUT/SYSLST
SYST	Char	4	SYSTEM
TAC	Char	8	TRANSACTION CODE
TACINDX	Hexa	8	TAC INDEX
TACNTR	Int	5	TA CNTR OF ACTUAL SERVICE
TACTYPE	Char	1	TAC TYPE
TASK	Char	4	TSN OF UTM TASK
TCPCL	Char	18	SOCKET FUNCTION
TCPMS	Int	11	DURATION OF SOCKET FUNCTION IN MILLISECONDS
TCPRC	Hexa	8	SOCKET DIAGNOSTIC WORD
TCVG	Char	8	CONVERSATION TAC
TERM	Char	1	TERMINATION TYPE
TEXT32	Char	32	STANDARD-TEXTPUFFER
TPRIO	Int	3	EXTERNAL TASK-PRIORITY
TRMA	Char	6	TERM APPLICATION REASON
TRMTYP	Char	4	ROOT-TERM-TYP STEP OR PROG
TSNPID	Char	10	TSN (BS2000) / PID (UNIX/WIN)
TTYP	Hexa	2	TERMINAL TYPE
UERCODE	Char	8	ERROR CODE
UERINFO	Char	8	ERROR INFORMATION
UKCHSTA	Int	5	HEIGHT OF STACK
UKCLM	Int	10	LENGTH OF KCLM USED BY KDCUPD
UKCOP	Char	4	OPCODE OF KDCS CALL USED BY KDCUPD
UKCRN	Char	8	REFERENCE NAME USED BY KDCUPD
ULLNBR	Int	11	NUMBER OF LOCKED ULS
UPCPROT	Hexa	16	UPIC PROTOCOLL
UPCREAS	Hexa	2	UPIC ERROR REASON
UPCSTAT	Hexa	4	USRTNSR UPIC STATE
UPDERR	Char	5	UPD ERROR CODE

Insert-Name	Datentyp	Länge	Bedeutung
UPDMODUL	Char	8	UPD MODULE READxxxx/WRITExxxx
UPDPRO	Int	3	PERCENT USED PAGES IN NEW FILE
UPDTYP	Char	6	TYPE OF KCRN FOR UPDATE (LTERM, TAC, LPAP)
UPPENC2	Hexa	4	UPIC ENCRYPTION PTRMDYN INFO
UPVENC1	Hexa	4	UPIC ENCRYPTION VGTDYN INFO
USER	Char	8	USER/LSES/OSI-ASS NAME
USRTYPE	Char	10	USER-TYP: CLIENT/CONNECTION
USSD	Hexa	4	USER SENSE DATA
USTYPPTC	Char	1	TYPE OF USER IN PTC
UTMDEVT	Char	7	UTM-D EVENT
VERS	Char	8	UTM VERSION
VER1	Char	6	VERSION NUMBER
VER2	Char	6	VERSION NUMBER
VGCNTR	Int	11	SERVICE COUNTER OF ACTUAL SERVICE
VTRC	Hexa	8	VTSU OR ASECO RETURN CODE
WLEV	Char	1	WARN LEVEL OF PAGE POOL
WTBF	Char	3	CACHE WAITS FOR BUFFER
XACALL	Char	12	FUNCTION-CALL OF XA-CAE INTERFACE
XADBC1	Char	8	TEXT FOR DB-XA-CALLS
XADBC2	Char	8	TEXT FOR DB-XA-CALLS
XAFLAG	Char	8	FLAGS FOR XA-CALLS
XASPEC	Char	12	VERSION OF XA-SPECIFICATION
XATXT	Char	16	READABLE XA-RETURNCODE
XCS1	Char	8	NAME OF XCS GROUP
XCS2	Char	8	NAME OF XCS GROUP
XPAPDU	Char	20	OSI-TP APDU TYPE
XPASST	Char	20	ASSOCIATION STATE
XPBCAM	Hexa	8	BCAM INFOWORD
XPCCLS	Int	11	CMX ERROR CLASS
XPCORR	Int	11	MESSAGE CORRELATOR NUMBER
XPCPSEL	Char	16	OSI-TP P-SEL OF PARTNER (CHAR)
XPCRES	Int	4	OSI-TP NEGATIVE CONFIRMATION RESULT
XPCSSEL	Char	16	OSI-TP S-SEL OF PARTNER (CHAR)

<b>Insert-Name</b>	<b>Datentyp</b>	<b>Länge</b>	<b>Bedeutung</b>
XPCTYPE	Int	11	CMX ERROR TYPE
XPCVAL	Int	11	CMX ERROR VALUE
XPERR	Int	11	OSI-TP ERROR CODE
XPEVT	Char	10	XAPTP EVENT
XPFSMN	Char	10	OSI-TP FSM NAME
XPFUNC	Char	20	CALLED OSI-TP FUNCTION
XPHPSEL	Hexa	32	OSI-TP P-SEL OF PARTNER (HEX)
XPHSSEL	Hexa	32	OSI-TP S-SEL OF PARTNER (HEX)
XPINI	Int	11	OSI-TP INITIATOR
XPLNK	Int	11	OSI-TP LINK
XPLPSEL	Int	2	OSI-TP LENGTH P-SEL OF PARTNER
XPLSSEL	Int	2	OSI-TP LENGTH S-SEL OF PARTNER
XPLTH	Int	11	OSI-TP INVALID LENGTH
XPNDIA	Int	4	OSI-TP NEGATIVE DIAGNOSTICS
XPNSEL	Char	8	OSI-TP N-SEL OF PARTNER
XPOSAS	Int	8	OSI-TP ASSOCIATION REFERENCE
XPPDU	Int	11	OSI-TP PDU TYPE
XPPTYP	Int	11	OSI-TP PRIMITIVE TYPE
XPRET	Int	11	OSI-TP RETURN CODE
XPRJCT	Int	4	OSI-TP ASSOCIATION REASON FOR REJECT
XPSRC	Int	4	OSI-TP RESULT SOURCE FROM PARTNER
XPTRFAIL	Int	11	OSI-TP WRITE TRACE FAILURE REASON
XPTSEL	Char	8	OSI-TP T-SEL OF PARTNER
XP0OBID	Int	11	OSI-TP OBJECT IDENTIFIER 0
XP1BOOL	Char	5	OSI-TP CCR V2 NOT AVAILABLE
XP1DIA	Int	11	OSI-TP DIAGNOSTIC INFORMATION 1
XP1INFO	Int	11	OSI-TP ADDITIONAL INFORMATION 1
XP1OBID	Int	11	OSI-TP OBJECT IDENTIFIER 1
XP2BOOL	Char	5	OSI-TP PROTOCOL VERSION INCOMPATIBILITY
XP2DIA	Int	11	OSI-TP DIAGNOSTIC INFORMATION 2
XP2INFO	Int	11	OSI-TP ADDITIONAL INFORMATION 2
XP2OBID	Int	11	OSI-TP OBJECT IDENTIFIER 2
XP3BOOL	Char	5	OSI-TP CONTENTION WINNER ASSIGNMENT REJECTED

<b>Insert-Name</b>	<b>Datentyp</b>	<b>Länge</b>	<b>Bedeutung</b>
XP3DIA	Int	11	OSI-TP DIAGNOSTIC INFORMATION 3
XP3INFO	Char	40	OSI-TP ADDITIONAL INFORMATION 3
XP3OBID	Int	11	OSI-TP OBJECT IDENTIFIER 3
XP4BOOL	Char	5	OSI-TP BID MANDATORY REJECTED
XP4OBID	Int	11	OSI-TP OBJECT IDENTIFIER 4
XP5BOOL	Char	5	OSI-TP NO REASON GIVEN
XP5OBID	Int	11	OSI-TP OBJECT IDENTIFIER 5
XP6OBID	Int	11	OSI-TP OBJECT IDENTIFIER 6
XP7OBID	Int	11	OSI-TP OBJECT IDENTIFIER 7
XP8OBID	Int	11	OSI-TP OBJECT IDENTIFIER 8
XP9OBID	Int	11	OSI-TP OBJECT IDENTIFIER 9



### 5.8.3 Ziele der UTM-Meldungen

Die folgende Tabelle zeigt für jede Meldung, welche Angaben möglich sind. Es sind nur die K-Meldungen berücksichtigt, die vom Anwender verändert werden können.

Die Angaben in den Spalten der Tabelle bedeuten auf die einzelnen Meldungen bezogen:

- R (Required) Das Meldungsziel ist der Meldung fest zugeordnet; es kann nicht verändert werden.
- + Das Meldungsziel ist für diese Meldung erlaubt.
- D (Default) Das Meldungsziel ist für diese Meldung angegeben, es kann ihr entzogen werden.
- Das Meldungsziel ist für diese Meldung nicht erlaubt.

in der Spalte COMPRESS:

- Y (Yes) Überflüssige Leerzeichen werden aus der Meldung entfernt.
- N (No) Überflüssige Leerzeichen verbleiben in der Meldung.

in der Spalte EXTEND:

- Y (Yes) Die Nachricht kann im EXTENDED LINE MODE ausgegeben werden.
- N (No) Die Nachricht kann nicht im EXTENDED LINE MODE ausgegeben werden.

in der Spalte BELL:

- Y (Yes) Mit der Meldung wird ein akustisches Signal ausgegeben.
- N (No) Mit der Meldung kann kein akustisches Signal ausgegeben werden.

Damit Sie Fehler, die beim Start einer UTM-Anwendung oder einer Folgetask auftreten, leichter diagnostizieren können, werden alle K-Meldungen von openUTM, die in der Startphase auftreten, auf SYSLIST und SYSOUT ausgegeben, unabhängig davon, welche Meldungsziele für diese Meldungen festgelegt wurden.

Die Funktionseinheit ist für den Anwender modifizierbar.

MSG-Id	Inserts	S	S	P	S	M	S	S	C	U	C	E	B
		t	S	a	S	S	S	S	o	S	o	x	e
		a	L	r	L	T	O	L	n	R	p	t	i
		t	n	n	O	A	S	S	e	D	r	e	d
		i	E	e									
		o											
K001 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , APPL	R	-	+	+	+	+	+	+	-	Y	N	N
K002 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , APPL	R	-	-	+	+	+	+	+	+	Y	N	N
K003	PTRM , PRNM , BCAP , LTRM , CMD	R	D	-	+	+	+	+	+	+	N	N	N
K004 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER , REA7	R	+	-	+	+	+	+	+	+	N	N	N
K005 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER	R	+	-	+	+	+	+	+	+	N	N	N
K006 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER	R	+	-	+	+	+	+	+	+	N	N	N
K007 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER , REA7	R	+	-	+	+	+	+	+	+	N	N	N
K008 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER , SGNT	R	-	-	+	+	+	+	+	-	Y	N	N
K009	PTRM , PRNM , BCAP , LTRM , USER , TAC , RCDC	R	D	D	+	+	+	+	+	+	N	N	N
K010	PTRM , PRNM , BCAP , LTRM , USER , TAC	R	D	D	+	+	+	+	+	+	N	N	N
K011	PTRM , PRNM , BCAP , LTRM , USER , ATAC1	R	+	+	+	+	+	+	+	+	N	N	N
K012	NUMMSGs	R	R	-	-	-	-	-	-	-	N	N	N
K013	PTRM , PRNM , BCAP , LTRM , CMD	R	D	-	+	+	+	+	+	+	N	N	N
K014	PTRM , PRNM , BCAP , LTRM , USER	R	+	-	+	+	+	+	+	+	N	N	N
K015	PTRM , PRNM , BCAP , LTRM , USER , TAC , FORM , RCDC , RCF2A	R	+	-	+	+	+	+	+	+	N	N	N
K016	PTRM , PRNM , BCAP , LTRM , USER	R	D	+	+	+	+	+	+	+	N	N	N
K017	PTRM , PRNM , BCAP , LTRM , USER , TCVG , RCCC , RCDC , RCF2A , TAC	R	-	D	D	+	+	+	+	+	Y	N	N
K018	PTRM , PRNM , BCAP , LTRM , APPL	R	+	-	+	+	+	+	+	+	N	N	N
K019	PTRM , PRNM , BCAP , LTRM , APPL	R	D	-	+	+	+	+	+	+	N	N	N
K020	PTRM , PRNM , BCAP , LTRM , USER	R	D	-	+	+	+	+	+	+	N	N	N
K021	PTRM , PRNM , BCAP , LTRM	R	D	D	+	+	+	+	+	+	N	N	N
K022	PTRM , PRNM , BCAP , LTRM	R	-	+	+	+	+	+	+	+	Y	N	N
K023	OMSG2	R	D	+	-	-	-	-	-	-	N	N	N
K024	PTRM , PRNM , BCAP , LTRM , USER	R	+	D	+	+	+	+	+	+	N	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L O S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K025	PTRM , PRNM , BCAP , LTRM	R	-	D	D	+	+	+	+	+	Y	N	N
K026	PTRM , PRNM , BCAP , LTRM , USER	R	+	+	+	+	+	+	+	+	N	N	N
K027	PTRM , PRNM , BCAP , LTRM	R	+	D	+	+	+	+	+	+	N	N	N
K029	PTRM , PRNM , BCAP , LTRM , USER	R	-	-	+	+	+	+	+	-	N	N	N
K030	PTRM , PRNM , BCAP , LTRM , USER	R	+	-	+	+	+	+	+	+	N	N	N
K031 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER	R	+	-	+	+	+	+	+	+	N	N	N
K032 <sup>1</sup>	CON , PRNM , BCAP , LPAP , USER , RCF1B , RCF2B	-	-	-	D	+	D	+	+	+	N	N	N
K033	PTRM , PRNM , BCAP , LTRM , USER , REST , GLOBALSG	-	-	-	D	+	+	+	+	+	N	N	N
K034		R	R	+	-	-	-	-	-	-	N	N	N
K035		R	R	+	-	-	-	-	-	-	N	N	N
K036	PTRM , PRNM , BCAP , LTRM , RSLT , REA1	-	-	-	D	+	+	+	+	+	N	N	N
K037	HST1 , HST2 , RSLT , RCHX	-	-	-	D	+	+	+	+	+	Y	N	N
K038	SYN	-	-	-	-	-	R	+	-	-	Y	N	N
K039	STRTPAR1 , STRTPAR2	-	-	-	-	-	R	+	-	-	Y	N	N
K040	WLEV , PGPOOL	-	-	-	D	+	R	+	+	+	Y	N	N
K041	WLEV , PGPOOL	-	-	-	D	+	R	+	+	+	Y	N	N
K042		-	-	-	-	-	R	+	-	-	Y	N	N
K043	DMSE , FNAM	-	-	-	D	+	R	D	+	+	Y	N	N
K044		-	-	-	-	-	R	+	-	-	Y	N	N
K045	PTRM , PRNM , BCAP , LTRM , PALTRM , CID	-	-	-	-	+	-	-	-	+	N	N	N
K046	PTRM , PRNM , BCAP , LTRM , PALTRM , CID , DPID , ERPRT , IMSG2 , GNDATE , GNTIME	-	-	-	+	+	+	+	+	+	N	N	N
K047	OPCD1 , RTCD	-	-	-	-	-	R	+	-	-	Y	N	N
K048		-	-	-	-	-	R	+	-	-	Y	N	N
K049	RCCC2	-	-	-	-	-	D	D	-	-	Y	N	N
K050	APPL , VERS , AMOD , TERM , OST1 , BMD1 , ATYP , FNOD	-	-	-	D	+	R	+	+	+	Y	N	N

MSG-Id	Inserts	S	S	P	S	M	S	S	C	U	C	E	B
		t	S	a	S	S	S	C	U	C	E	B	
		a	t	r	L	G	O	O	S	R	p	x	
		t	I	n	O	A	T	S	E	r	r	t	
		i	n	e	C	O	S	O	D	e	e	e	
		o	n	r	T	T	L	L	E	S	s	n	
		n											
K051	APPL , VERS , AMOD , TERM , OST1 , BMD1 , ATYP , FNOD	-	-	-	D	+	R	+	+	+	Y	N	N
K052	TASK , APPL , PID , STSK	-	-	-	D	+	R	+	+	+	Y	N	N
K053	CNTR	-	-	-	D	+	R	+	+	+	Y	N	N
K054		-	-	-	D	+	R	+	+	+	Y	N	N
K055	ATAC1 , RCCC , RCDC , USER , LTRM	-	-	-	D	+	R	+	+	+	Y	N	N
K056	TASK , PID	-	-	-	D	+	D	+	+	+	Y	N	N
K057		-	-	-	D	-	D	+	+	-	Y	N	N
K058	TASK , PID	-	-	-	+	+	D	+	+	+	Y	N	N
K059		-	-	-	+	-	D	+	+	-	Y	N	N
K060	TRMA	-	-	-	D	-	R	D	+	-	Y	N	N
K061	FNAM	-	-	-	D	-	R	D	+	-	Y	N	N
K062		-	-	-	D	-	R	D	+	-	Y	N	N
K063	PTRM , PRNM , BCAP , LTRM , FMTN , RCF1A , RCF2A	-	-	-	D	+	R	+	+	+	Y	N	N
K064	PTRM , PRNM , BCAP , LTRM , DEVC , FIL1A , FIL2A , FIL3 , VTRC , IMSG2 , REA1 , CBRC	-	-	-	D	+	+	+	+	+	Y	N	N
K065	PTRM , PRNM , BCAP , LTRM , FIL1B , FIL2B	-	-	-	D	+	+	+	+	+	N	N	N
K066		-	-	-	-	-	R	+	-	-	Y	N	N
K067	MOD , ERCD1 , ERCD2 , OPCD2	-	-	-	-	-	R	+	-	-	Y	N	N
K068	DBCON , DBV1 , DBV2	-	-	-	-	-	R	+	-	-	Y	N	N
K069	PTRM , PRNM , BCAP , LTRM , COTM , REA4 , REA6	-	-	-	D	+	+	+	+	+	N	N	N
K070	PTRM , PRNM , BCAP , LTRM , USER , COTM , CPTM , GLOBALSG	-	-	-	D	+	+	+	+	+	N	N	N
K071	OPCD1 , ERCD3 , DBTRAC , DBCON	-	-	-	-	-	R	+	-	-	Y	N	N
K072	STMT	-	-	-	-	-	D	D	-	-	Y	N	N
K073	ATTR , STMT , PROG	-	-	-	-	-	D	D	-	-	Y	N	N
K074	CTYP , PROG , PVER	-	-	-	D	+	R	+	+	+	Y	N	N
K075	CTYP , PROG , PVER , TASK , PID	-	-	-	D	+	D	+	+	+	Y	N	N

MSG-Id	Inserts	S	S	P	S	M	S	S	C	U	C	E	B
		t	S	a	S	S	S	C	U	C	E	B	
		a	t	r	L	G	T	O	S	R	D	x	t
		t	I	n	O	A	O	I	E	E	p	t	e
		i	n	e	C	U	S	S	S	S	r	e	n
		o	n	r	T	T	T	T	T	T	e	s	d
		n											
K076	RCCC , RCDC , ADTC , USER , LTRM	-	-	-	D	+	R	+	+	-	Y	N	N
K077	PTRM , PRNM , BCAP , LTRM , CLSIGT	-	-	-	D	-	D	D	-	-	Y	N	N
K078	ERRNAME , ERRCODE , REA3	-	-	-	-	-	R	D	-	-	Y	N	N
K079	REA2	-	-	-	D	+	+	+	+	+	Y	N	N
K080		-	-	-	D	+	+	+	+	+	Y	N	N
K081	IMSG1 , OMSG1 , CONU , ATAC2 , LWRT , HITR , WTBF	-	-	-	D	+	+	+	+	+	N	N	N
K082	FNAM	-	-	-	-	-	R	D	-	-	Y	N	N
K083	FNAM	-	-	-	-	-	R	D	-	-	Y	N	N
K084	OBJ1 , VER1 , OBJ2 , VER2 , OST1 , BMD1 , OST2 , BMD2	-	-	-	-	-	R	D	-	-	Y	N	N
K085	FNKT	-	-	-	-	-	R	D	-	-	Y	N	N
K086	PTRM , PRNM , BCAP , LTRM , USER , SYSD , USSD , FMH7 , AGUS	-	-	-	D	+	+	+	+	+	N	N	N
K087	OPCD3 , ERCD4 , ERCD5	-	-	-	-	-	R	+	-	-	Y	N	N
K088	LSES , RSES , LPAP , SRFG , PSQN , ESQS , EBSS , ESQR , ESRR , EBSR	-	-	-	D	+	+	+	+	+	N	N	N
K089	GNDATE , GNTIME , DEST , GNUMER , USER , DLDATE , DLTIME , CHAIN , DPID	-	-	-	D	+	+	+	+	+	N	N	N
K090	DEST , USER , DLDATE , DLTIME	-	-	-	D	+	+	+	+	+	N	N	N
K091	PTRM , PRNM , BCAP , LTRM , USER , REA7	R	+	+	+	+	+	+	+	+	N	N	N
K092	PTRM , PRNM , BCAP , LTRM , USER , PAS1 , PAS2 , PAS3	R	-	-	+	+	+	+	+	-	N	Y	N
K093	PTRM , PRNM , BCAP , LTRM , USER , HSTACK , MSTACK	R	D	+	+	+	+	+	+	+	Y	N	N
K094	PTRM , PRNM , BCAP , LTRM , USER , RCF1B , REA4	-	-	-	D	+	+	+	+	+	N	N	N
K095		D	R	-	-	-	-	-	-	-	N	N	N
K096		R	R	-	-	-	-	-	-	-	N	N	N
K097 <sup>1</sup>	PTRM , PRNM , BCAP , LTRM , USER	R	-	-	+	+	+	+	+	+	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C T	S Y S O U T	S Y S L O S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e n
K098	PTRM , PRNM , BCAP , LTRM , USER , RCF1C , RCF2B	R	D	-	+	+	+	+	+	+	N	N	N
K099	MSG	-	-	-	-	R	+	-	-	Y	N	N	
K101	PTRM , PRNM , BCAP , LTRM , USER , REA1 , DEST	R	D	+	D	+	D	+	+	+	Y	N	N
K102	CIW , CPC , CILC , CPM , CCC , CR0 , CR1 , CR2 , CR3 , CR4 , CDSPL , CAMODE	-	-	-	-	R	+	+	-	N	N	N	
K103	CR5 , CR6 , CR7 , CR8 , CR9 , CR10 , CR11 , CR12 , CR13 , CR14 , CR15	-	-	-	-	R	+	+	-	N	N	N	
K104	UTMDEVT , LSES , LPAP , AGUS , OCVST , OTAST , ACTION , NCVST , NTA ST , SESSCNTR , VGCNTR , TACNTR , RCV DANNO , TACINDX , LTACINDX	-	-	-	D	+	+	+	+	+	N	N	N
K105	LSES , LPAP , AGUS , SYST	-	-	-	D	+	+	+	+	+	N	N	N
K106	PTRM , PRNM , BCAP , LTRM , USER , DEVC , FIL1A , FIL2A , FIL3 , VTRC , CBRC , OMSG3 , FMTN , CCSN	R	+	-	D	+	+	+	+	+	Y	N	N
K107	TTYP	-	-	-	D	-	R	+	+	+	Y	N	N
K108	PTRM , PRNM , BCAP , LTRM , USER , KRRC , KRLH , KRPR , KRACT	-	-	-	D	+	+	+	+	+	Y	N	N
K109	PTRM , PRNM , BCAP , LTRM , USER , KRRC , KRLH , KRPR , KRACT	R	-	-	D	+	+	+	+	+	Y	N	N
K110	PTRM , PRNM , BCAP , LTRM , USER , KRRC , KRLH , KRPR	R	-	-	D	+	+	+	+	+	Y	N	N
K111	PTRM , PRNM , BCAP , LTRM , USER	R	-	-	+	+	+	+	+	+	Y	N	N
K112	PTRM , PRNM , BCAP , LTRM , USER	+	R	-	-	-	-	-	-	-	Y	N	N
K115	PTRM , PRNM , BCAP , LTRM , SNPT , SNPR , SNLT , CCC , REA1 , ANNO	-	-	-	R	+	+	+	+	+	N	N	N
K116	PTRM , PRNM , BCAP , LTRM , SNPT , SNPR , SNLT , USER , REA1	-	-	-	+	+	+	+	+	+	N	N	N
K117	PTRM , PRNM , BCAP , LTRM , SNPT , SNPR , SNLT , USER , REA1	-	-	-	+	+	+	+	+	+	N	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K118	FORMATID, LTHGTRID, GTRID, AAIS, AAID	-	-	-	D	+	+	+	+	+	Y	N	N
K119	OSLPAP, USER, TAC, DIA1, DIA2, DIA3, SESSCNTR, VGCNTR, TACNTR, AAIS, AAID	-	-	-	D	+	+	+	+	+	Y	N	N
K120	PTRM, PRNM, BCAP, LTRM, USER	R	+	-	D	+	+	+	+	+	N	N	N
K122	PTRM, PRNM, BCAP, LTRM, USER, NUMDAYS	+	R	-	-	-	-	-	-	-	Y	N	N
K123	LTRM, TAC, USER	R	+	-	D	+	+	+	+	+	N	N	N
K124	RCXAPTP, PHAXAPTP	-	-	-	D	-	D	D	+	-	Y	N	N
K125	PTRM, PRNM, BCAP, LTRM, USER	R	+	-	D	+	+	+	+	+	Y	N	N
K126	SATRC	-	-	-	D	+	D	D	+	+	N	N	N
K127	ERCD6	-	-	-	-	-	R	+	-	-	Y	N	N
K128	CON, PRNM, BCAP, LPAP, LSES, REA1, RCDC, TAC	-	-	-	D	+	+	+	+	+	N	N	N
K129	CMD	-	-	-	-	-	R	+	-	-	Y	N	N
K130	TPRIO, TASK	-	-	-	D	+	+	+	+	+	Y	N	N
K132	OBJ1, OBJ2	-	-	-	-	-	R	+	-	-	Y	N	N
K133	EXIT	-	-	-	-	-	R	+	-	-	Y	N	N
K134	DEST, NMSG, DPID, GNDATE, GNTIME	-	-	-	D	+	+	+	+	+	Y	N	N
K135	PTRM, PRNM, BCAP, LTRM, UPCREAS, UPCSTAT, UPCPROT, UPVENC1, UPPENC2	-	-	-	D	+	+	+	+	+	N	N	N
K136	FNAM	-	-	-	-	-	D	+	+	+	Y	N	N
K137	FNAM	-	-	-	+	+	D	+	+	+	Y	N	N
K138	FNAM	-	-	-	-	+	D	+	+	+	Y	N	N
K139	FNAM	-	-	-	+	+	D	+	D	+	Y	N	N
K140	PTRM, PRNM, BCAP, LTRM, MXP1, MXP2	-	-	-	D	+	+	+	+	+	Y	N	N
K141	PTRM, PRNM, BCAP, LTRM, MXP1	-	-	-	D	+	+	+	+	+	Y	N	N
K142	PTRM, PRNM, BCAP, LTRM, MXPT, MXPR, MXLT	-	-	-	D	+	+	+	+	+	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L O S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K143	PTRM , PRNM , BCAP , LTRM , STS1 , STS2 , STS3 , STS4	-	-	-	D	+	+	+	+	+	Y	N	N
K144	PTRM , PRNM , BCAP , LTRM , DEVC , FIL1A , FIL2A , FIL3 , VTRC , CBRC , OMSG3 , FMTN , CCSN	-	-	-	D	+	R	+	+	+	Y	N	N
K145	PTRM , PRNM , BCAP , LTRM , USER	R	+	+	+	+	+	+	+	+	N	N	N
K146	BCMOPCD , BCMRTCD , STDHEAD , TASK , BCAP	-	-	-	D	+	+	+	+	+	Y	N	N
K147	PTRM , PRNM , BCAP , LTRM , USER , REA7 , USRTYPE	-	-	-	D	+	D	+	+	+	Y	N	N
K149	DIA5	-	-	-	-	-	R	+	+	-	Y	N	N
K150	PTRM , PRNM , BCAP , LTRM , RSOANNO , RSOOPC , RSOMSG , RSORC , RSO2RC , STDHEAD	-	-	-	D	+	+	+	+	+	Y	N	N
K151	IDEFRC , DMSE , FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K152	COND , MTYPE , OSLPAP , USER , LTAC , AAIS , AAID , SESSCNTR , VGCNTR , TACNTR	-	-	-	D	+	+	+	+	+	Y	N	N
K153	PTRM , PRNM , BCAP , LTRM , USER	-	-	-	-	-	+	+	-	-	N	N	N
K154	PTRM , PRNM , BCAP , LTRM , TCPCL , TCPRC	-	-	-	D	+	D	+	+	+	N	N	N
K155	PTRM , PRNM , BCAP , LTRM , USER , PAS1 , PAS2 , PAS3	R	-	-	+	+	+	+	+	-	N	Y	N
K156	RSLT	-	-	-	D	+	+	R	+	+	N	N	N
K157	PTRM , PRNM , BCAP , ANNO , STDRC , SSORC , STDHEAD , SSINCID , SSOUTCID , SSTYPNAM , SLENAM	-	-	-	D	+	+	+	+	+	Y	N	N
K158	PTRM , PRNM , BCAP , LTRM , USER , CPUTEXT , CPUBEGIN , CPUEND , CPUUSED , CPUCLNT , CPUREAS	-	-	-	D	+	+	+	+	+	Y	N	N
K159	USER , ENCPW	-	-	-	+	+	+	+	+	-	Y	N	N



MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K160	PTRM , PRNM , BCAP , LTRM , USER , TCVG , TAC , TACNTR , RBCAUSER, RCCC , RCDC , TASK , SESSCNTR, VGCNTR , LTHGTRID, GTRID , PID	-	-	-	D	+	+	+	+	+	Y	N	N
K161	TASK , PID	-	-	-	+	+	+	+	+	-	N	N	N
K162	TASK , IOPG , IOMS , PID	-	-	-	D	-	R	D	+	-	Y	N	N
K163	TASK , IOPG , IOMS , PID	-	-	-	D	+	R	D	+	-	Y	N	N
K164	FNAM	-	-	-	+	+	D	+	+	+	N	N	N
K165	FNAM	-	-	-	+	+	+	D	+	+	N	N	N
K166	FNAM , RCHX , SYSFILE	-	-	-	R	+	D	D	+	+	N	N	N
K167		-	-	-	+	+	D	D	+	+	Y	N	N
K169	ACTION , IDX1 , HST1 , STATE , IDX2 , HST2 , IDX3 , HST3 , NNM1 , NNM2 , NNM3	-	-	-	R	-	R	D	+	-	Y	N	N
K170	DTTM , STATE	-	-	-	R	-	R	D	+	-	Y	N	N
K171	HST1 , XCS1 , XCS2	-	-	-	R	-	R	D	+	-	Y	N	N
K174	DIA1	-	-	-	R	-	R	D	+	-	Y	N	N
K175	FNAM	-	-	-	R	-	R	D	+	-	Y	N	N
K176	PRCN , MSG2 , RCHX	-	-	-	R	+	R	D	+	+	Y	N	N
K178	ACTION , STATE , STA2 , PGS1 , PGS2 , SWNR	-	-	-	R	-	R	D	+	-	Y	N	N
K179		-	-	-	R	+	D	D	+	+	Y	N	N
K180		-	-	-	R	+	D	D	+	+	Y	N	N
K181	FNAM	-	-	-	R	+	D	D	+	+	Y	N	N
K182	FNAM	-	-	-	R	+	D	D	+	+	Y	N	N
K183	FNAM	-	-	-	R	+	D	D	+	+	Y	N	N
K184		-	-	-	R	+	D	D	+	+	Y	N	N
K185		-	-	-	R	+	D	D	+	+	Y	N	N
K186		-	-	-	R	+	D	D	+	+	Y	N	N
K187	FNAM	-	-	-	R	+	D	D	+	+	Y	N	N
K188	OBJ1 , DTTM , OBJ2 , DTM2	-	-	-	R	-	R	D	+	-	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K189	PTRM , PRNM , BCAP , LTRM , USER , HST1	R	+	-	D	+	+	+	+	+	Y	N	N
K190	DIA1 , INF1 , INF2 , SUFF	-	-	-	R	-	R	D	+	-	Y	N	N
K191	SUFF , RQM	-	-	-	D	-	D	D	+	-	Y	N	N
K192	NNM1 , HST1 , RSPTC	-	-	-	-	-	D	D	-	-	Y	N	N
K193	PTCID , USER , LPAP , LSES , USTYPPTC	-	-	-	D	-	D	D	+	-	Y	N	N
K194	GBLNBR , ULLNBR	-	-	-	D	-	D	D	+	-	Y	N	N
K195	MSG2	+	+	+	+	+	+	+	+	+	N	N	N
K196	MSG2	+	+	+	+	+	+	+	+	+	N	N	N
K197	MSG2	+	+	+	+	+	+	+	+	+	N	N	N
K198	MSG2	+	+	+	+	+	+	+	+	+	N	N	N
K199	TRMTYP , ROOTTRMR	-	-	-	-	-	R	D	-	-	Y	N	N
K200	MSG2	-	-	-	+	+	+	+	+	+	N	N	N
K201	TSNPID , XASPEC	-	-	-	-	-	R	D	-	-	Y	N	N
K202	TSNPID , INSTNUM , TEXT32 , RMSTAT	-	-	-	-	-	R	D	-	-	Y	N	N
K203	TSNPID , TEXT32 , INSTNUM , RTAANZ	-	-	-	-	-	R	D	-	-	Y	N	N
K204	TSNPID , INTTAID , XATXT	-	-	-	-	-	R	D	-	-	Y	N	N
K205	TSNPID , INTTAID , XATXT	-	-	-	-	-	R	D	-	-	Y	N	N
K206	TSNPID , INTTAID , XATXT	-	-	-	-	-	R	D	-	-	Y	N	N
K207	TSNPID , INTTAID , INSTNUM	-	-	-	-	-	R	D	-	-	Y	N	N
K210	TSNPID , XATXT , TEXT32 , INSTNUM	-	-	-	-	-	R	D	-	-	Y	N	N
K211	TSNPID , XATXT , TEXT32 , INSTNUM	-	-	-	-	-	R	D	-	-	Y	N	N
K212	TSNPID , XATXT , XAFLAG , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N
K213	TSNPID , XATXT , XAFLAG , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N
K214	TSNPID , XATXT , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N
K215	TSNPID , XATXT , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N
K216	TSNPID , XATXT , TEXT32 , INSTNUM	-	-	-	-	-	R	D	-	-	Y	N	N
K217	TSNPID , XATXT , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N
K218	TSNPID , XATXT , INTTAID	-	-	-	-	-	R	D	-	-	Y	N	N

MSG-Id	Inserts	S	S	P	S	M	S	S	C	U	C	E	B
		t	S	a	S	S	S	C	U	C	E	X	e
		a											
		t											
		i											
		n											
		o											
K220	TSNPID , TEXT32	-	-	-	-	-	R	D	-	-	Y	N	N
K221	TSNPID , TEXT32	-	-	-	-	-	R	D	-	-	Y	N	N
K222	TSNPID , XASPEC , TEXT32	-	-	-	-	-	R	D	-	-	Y	N	N
K223	TSNPID	-	-	-	-	-	R	D	-	-	Y	N	N
K224	TSNPID , XACALL , XATXT , TEXT32 , INSTNUM	-	-	-	-	-	R	D	-	-	Y	N	N
K225	TSNPID , XADBC1 , XADBC2	-	-	-	-	-	R	D	-	-	Y	N	N
K230	TSNPID , TEXT32	-	-	-	-	-	R	D	-	-	Y	N	N
K231	TSNPID	-	-	-	-	-	R	D	-	-	Y	N	N
K232	TSNPID	-	-	-	-	-	R	D	-	-	Y	N	N
K233	TSNPID , INSTNUM , XACALL , DBCALL , XATXT , LTHGTRID , GTRID	-	-	-	-	-	R	D	-	-	Y	N	N
K234	FNAM , ELEM , EVER , LMSTYPE , PLARC , DMSRC	-	-	-	D	+	D	D	+	+	Y	N	N
K235	TCPCL , PRNM , TCPRC , TCPMS , IPADDR	-	-	-	D	+	D	+	+	+	Y	N	N
K236	TCPCL , IPADDR , TCPRC , TCPMS , PRNM	-	-	-	D	+	D	+	+	+	Y	N	N
K237	TSNPID , DBCALL	-	-	-	-	-	R	D	-	-	Y	N	N
K238	TSNPID , DBCALL	-	-	-	-	-	R	D	-	-	Y	N	N
K239	LPAP , DPID , GUSER , LTRM , GNDATE , GNTIME	-	-	-	D	+	+	+	+	+	Y	N	N
K251	IMPVER , FBASUPD , DEFVER	-	-	-	D	+	D	D	+	+	Y	N	N
K252	UPDERR	-	-	-	D	+	D	D	+	+	Y	N	N
K255	DMSE , FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K256	FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K257		-	-	-	D	+	D	D	+	+	Y	N	N
K258	FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K260	DEFVER , FBASUPD	-	-	-	D	+	D	D	+	+	Y	N	N
K261	FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K262	FNAM	-	-	-	D	+	D	D	+	+	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L O S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
K263	FNAM	-	-	-	D	+	D	D	+	+	Y	N	N
K269	OBJ1 , OST1 , BMD1 , OBJ3 , OST2 , BMD2	-	-	-	D	+	D	D	+	+	Y	N	N
K273	TRMA , UPDMODUL	-	-	-	D	+	D	D	+	+	Y	N	N
K274		-	-	-	D	+	D	D	+	+	Y	N	N
K277		-	-	-	D	+	D	D	+	+	Y	N	N
K278		-	-	-	D	+	D	D	+	+	Y	N	N
K279		-	-	-	D	+	D	D	+	+	Y	N	N
K300	UPDPRO , PGPOOL	-	-	-	D	+	D	D	+	+	Y	N	N
K303	UKCOP , UKCRN , UPDTYP , UKCLM	-	-	-	D	+	D	D	+	+	N	N	N
K304	USER , TACTYPE , UKCHSTA	-	-	-	D	+	D	D	+	+	N	N	N
K305	UPDTYP , PGS1 , PGS2 , UKCRN , PGPOOL	-	-	-	D	+	D	D	+	+	Y	N	N
K306	UPDTYP , PGS1 , PGPOOL	-	-	-	D	+	D	D	+	+	Y	N	N
K310	UPDTYP , UKCRN	-	-	-	D	+	D	D	+	+	N	N	N
K311	UPDTYP , UKCRN	-	-	-	D	+	D	D	+	+	N	N	N
K314		-	-	-	D	+	D	D	+	+	N	N	N
K317	UKCOP , UKCRN , UPDTYP , UKCLM , RCCC , RCDC , LTRM , USER	-	-	-	D	+	D	D	+	+	N	N	N
K318	UPDTYP , UKCRN , LTRM , USER	-	-	-	D	+	D	D	+	+	N	N	N
K320	USER , TACTYPE , UERCODE , UERINFO , RCDC	-	-	-	D	+	D	D	+	+	N	N	N
P001	XPFUNC , ACPNT , XPRET , XPERR , XP1INFO , XP2INFO , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P002	XPFUNC , ACPNT , OSLPAP , XPRET , XPERR , XP1INFO , XP2INFO , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P003	ACPNT , XPRJCT , XPLTH	-	-	-	D	+	+	+	+	+	Y	N	N
P004	ACPNT , OSLPAP , XPRJCT	-	-	-	D	+	+	+	+	+	Y	N	N
P005	ACPNT , XPNSEL , XPTSEL , XPLSSEL , XPCSSEL , XPHSSEL , XPLPSEL , XPCPSEL , XPHPSEL	-	-	-	D	+	+	+	+	+	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S O U T	S Y S L S T	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
P006	ACPNT , OSLPAP , XP0OBID , XP1OBID , XP2OBID , XP3OBID , XP4OBID , XP5OBID , XP6OBID , XP7OBID , XP8OBID , XP9OBID	-	-	-	D	+	+	+	+	+	Y	N	N
P007	ACPNT , OSLPAP , XPRET , XPERR , XP1INFO , XP2INFO , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P008	ACPNT , OSLPAP , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P009	ACPNT , OSLPAP , XPRJCT , XPLTH , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P010	ACPNT , OSLPAP , XPNSEL , XPTSEL , XPLSSEL , XPCSSEL , XPHSSEL , XPLPSEL , XPCPSEL , XPHPSEL , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P011	ACPNT , OSLPAP , XP0OBID , XP1OBID , XP2OBID , XP3OBID , XP4OBID , XP5OBID , XP6OBID , XP7OBID , XP8OBID , XP9OBID , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P012	XPCTYPE , XPCCLS , XPCVAL , XPBCAM , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P013	ACPNT , OSLPAP , XPCRES , XPSRC , XPNDIA , XP1BOOL , XP2BOOL , XP3BOOL , XP4BOOL , XP5BOOL , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P014	XPFUNC , ACPNT , OSLPAP , XPRET , XPERR , XP1INFO , XP2INFO , XPOSAS , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P015	XPFUNC , ACPNT , OSLPAP , XPLNK , XPSRC , XPNDIA , XPINI , XP1INFO , XP2INFO , XPOSAS , XPCORR	-	-	-	D	+	+	+	+	+	Y	N	N
P016	ACPNT , OSLPAP , XPLNK , XPNDIA , XPOSAS	-	-	-	D	+	+	+	+	+	Y	N	N
P017	XPPDU , XP1DIA , XP2DIA , XP3DIA	-	-	-	+	+	+	+	+	+	Y	N	N
P018	ACPNT , OSLPAP , XPPTYP , XPFSMN	-	-	-	D	+	+	+	+	+	Y	N	N
P019	ACPNT , OSLPAP , XPAPDU , XP3INFO	-	-	-	D	+	+	+	+	+	Y	N	N
P020	XPTRFAIL	-	-	-	D	+	+	+	+	+	Y	N	N

MSG-Id	Inserts	S t a t i o n	S Y S L I N E	P a r t n e r	S Y S L O G	M S G T A C	S Y S L O G	S Y S L O G	C o n s o l e	U S E R D E S T	C o m p r e s s	E x t e n d	B e i e
P021	XPEVT , ACPNT , OSLPAP , XPOSAS , XPASST	-	-	-	D	+	+	+	+	+	Y	N	N

<sup>1</sup> Wird in einer UTM-Anwendung mit einem Anmelde-Vorgang gearbeitet, dann werden diese Meldungen nicht erzeugt, d.h. sie werden auch nicht an die Meldungsziele MSGTAC oder SYSLOG ausgegeben

---

## 6 Anhang

### 6.1 KDCS-Returncodes in KCRCCC

Folgende Kategorien sind zu unterscheiden:

Code	Kategorie	Maßnahmen
000	ohne Fehler	Operation wurde fehlerfrei ausgeführt.
01Z - 09Z	Bemerkungen	Operation wurde ausgeführt.
10Z - 19Z	Warnung bzw. leichte Fehler	Nach Ausführung geeigneter Maßnahmen kann der Teilprogrammablauf sinnvoll fortgesetzt werden. Die Operation wurde nicht ausgeführt.
20Z - 39Z	Sonderfunktionen	Eine KDCS-Sonderfunktion wird signalisiert. Sonst wie Warnungen.
40Z - 69Z	Fehler	Die Operation wurde nicht ausgeführt. Der Programmablauf wird meistens nicht mehr sinnvoll weiterarbeiten können. Kommunikation mit der Dialogstation ist noch möglich.
70Z - 99Z	Schwerwiegende Fehler	Eine Fortsetzung des Programmablaufs ist nicht möglich. openUTM setzt die Transaktion zurück und bricht den Vorgang ab. openUTM erzeugt eine System-spezifische Fehler-Nachricht und gibt sie bei einem Dialog-Vorgang auf die Dialogstation aus.

Beachten Sie bitte folgende Punkte:

- Eine genauere Erklärung der Returncodes in KCRCCC finden Sie im openUTM-Handbuch „Anwendungen programmieren mit KDCS“ bei jedem einzelnen KDCS-Aufruf.
- Treten mehrere Fehler gleichzeitig auf, so werden die Fehler der jeweils höchsten Kategorie angezeigt. Wurde eine Operation nicht ausgeführt (Returncode > 09Z), so wird der Inhalt in <parm2> nicht verändert.
- Die Auswertung eines UTM-Dumps ist im [Kapitel „Der UTM-Dump“ auf Seite 55](#) beschrieben.

Folgende Returncodes sind definiert (Angabe der Feldnamen: COBOL-Name/C-Name):

Code	Bedeutung
000	Operation erfolgreich ausgeführt.
01Z	Längenkonflikt bezüglich KCLA/kcla bzw. KCLKBPRG/kclcapa.
02Z	Längenkonflikt bezüglich KCLPAB/kclspa.
03Z	Name in KCMF/kcfn ist ungültig.
04Z	Name in KCRN/kcrn wechselt.
05Z	Eingabeformatierung wurde nicht mit dem in KCMF/kcfn angegebenen Formatkennzeichen durchgeführt. Zeilenmodus: 1.Zeichen von KCMF/kcfn ungleich Space.
06Z	Zeitangabe wechselt innerhalb der Nachricht (Aufruf DPUT)
07Z	Längenkonflikt bezüglich KCLI/kcli. Bei DADM MA: Es konnten nicht alle Nachrichten der Dead Letter Queue verschoben werden, weil der taskspezifische Pufferbereich für Wiederanlaufinformation zu klein generiert wurde. Bitte DADM-Aufruf wiederholen.
08Z	Beim Lesen mit Warten im Aufruf DGET: Es liegt zurzeit keine Nachricht vor.
09Z	Bei INFO CD: Der Kerberos-Dialog hat einen Fehler geliefert oder die Kerberos-Information wird verkürzt zurückgegeben.
10Z	Nachricht wurde schon vollständig gelesen
11Z	Beim Lesen ohne Warten im Aufruf DGET: Es liegt keine Nachricht vor.
12Z	von der angegebenen Vorgangs-Identifikation liegen keine Nachrichten (mehr) vor, oder es existiert kein Vorgangsstapel mit der angegebenen Nummer.
14Z	Name in KCRN/kcrn nicht gefunden.
16Z	die Operation ist nicht zulässig und wurde nicht ausgeführt.
19Z	Funktionstaste bzw. Sonderfunktion nicht generiert.
20Z bis 39Z	KDCS-Sonderfunktionen (Kurznachrichten).
40Z	Operation kann vom System nicht durchgeführt werden (UTM-Generierungsfehler, Systemfehler, Deadlock, lang dauernde Sperren).
41Z	Operation an dieser Stelle nicht erlaubt.
42Z	Operationsmodifikation ungültig.
43Z	Längenangabe in KCLM/kclm, KCLI/kcli, KCLA/kcla oder KCWTIME/kcvertime ungültig.
44Z	Name in KCRN/kcrn ungültig.
45Z	Formatkennzeichen KCMF/kcfn oder Erzeugungszeitpunkt (DGET) ungültig.
46Z	Name in KCLT/kclt oder KCPA/kcpa oder KCLANGID/kclangid, KCTERRID/kcterrid, KCCSNAME/kccsname oder KCQMODE/kcqmode ist ungültig.



Code	Bedeutung
47Z	Speicherbereich <parm2> fehlt, die Bereichsadresse ist ungültig, oder der Bereich ist in der angegebenen Länge nicht lesbar/beschreibbar.
48Z	ungültige Schnittstellenversion.
49Z	Nicht verwendete Parameter sind ungleich binär Null.
51Z	Reihenfolge beim DPUT-Aufruf nicht eingehalten.
52Z	Nachrichtenziel in KCRN ist nicht erlaubt (Aufruf DPUT).
53Z	Wert in KCDPID/kcdpid bzw. KCGTM/kcgtm ist ungültig (Aufruf DGET).
54Z	Wert in KCNORPLY/kcnoreply ist ungültig (bei CTRL PR oder PE: Wert ungleich Y oder binär 0).
55Z	Name in KCPI/kcpi ist ungültig (Aufruf APRO).
56Z	Wert von KCMOD oder Zeitangabe in KCTAG/kcday,...,KCSEK/kccec ist ungültig (Aufruf DADM, DPUT).
57Z	Wert in KCPOS/kcpos ist ungültig (Aufruf MCOM).
58Z	Wert in KCOF/kcof ungültig (Aufruf APRO) oder der Wert in KCNEG/kcneg ist ungültig (Aufruf MCOM).
70Z	Operation kann vom System nicht durchgeführt werden (UTM-Generierungsfehler, Systemfehler).
71Z	Operation an dieser Stelle nicht erlaubt, z.B. noch kein INIT.
72Z	Operationsmodifikation ungültig.
73Z	Längenangabe in KCLA/kcla, KCLM/kclm oder KCLI/kcli ungültig.
74Z	Name in KCRN/kcrn ungültig.
75Z	Formatkennzeichen KCMF/kcfn ungültig.
77Z	Speicherbereich <parm2> fehlt, die Bereichsadresse ist ungültig, oder der Bereich ist in der angegebenen Länge nicht lesbar/beschreibbar.
78Z	FORMAT-Exit meldet Fehler (Aufruf MGET).
79Z	Operationscode nicht interpretierbar *.
80Z	UTM-Generierungsfehler vor Programmstart.
81Z	Ziel am PEND in Widerspruch mit Ziel am MPUT.
82Z	PEND-Variante widerspricht dem Operanden KCOM/kcom oder KCRN/kcrn im MPUT-Aufruf.
83Z	Vor einem PEND KP,RE,FI,ER,FR wurde in einem Dialog-Programm kein MPUT gegeben oder vor einem PEND KP,RE wurde in einem Asynchron-Programm kein MPUT gegeben oder vor einem PGWT PR wurde ein MPUT gegeben.
84Z	PEND fehlt.
86Z	Ein Nachrichtenkomplex wurde nicht abgeschlossen oder nach einem APRO AM wurde kein FPUT/DPUT für den adressierten Auftragnehmer- Vorgang gegeben.

<b>Code</b>	<b>Bedeutung</b>
87Z	Operationsmodifikation ist unzulässig.
88Z	Ungültige Schnittstellenversion.
89Z	Nicht verwendete Parameter sind ungleich binär Null.

\* Der Returncode 79Z (Operationscode nicht interpretierbar) ist keiner speziellen Operation zuzuordnen.

## 6.2 Interner Returncode KCRCDC

Der inkompatible KDCS-Returncode KCRCDC besteht aus 4 abdruckbaren Zeichen und enthält genauere Angaben zur Fehlerursache als der kompatible Returncode KCRCCC.

Der inkompatible Returncode kann in folgenden Fällen gesetzt sein:

- wenn ein KDCS-Aufruf den kompatiblen Returncode 40Z liefert.
- wenn openUTM einen Vorgang mit PEND ER abnormal beendet ( $KCRCCC \geq 70Z$ )
- wenn eine Transaktion implizit durch openUTM zurückgesetzt wird.

Der Returncode KCRCDC hat folgenden Aufbau:

sm ##

Das erste Zeichen (s) gibt an, welcher Teil von openUTM den Fehler gemeldet hat:

A	Administration
F	Formatierungskomponente
K	UTM-Systemcode oder ROOT-Code
U	UPIC (Server-Teil)

Bei allen Returncodes, die mit dem Zeichen K beginnen, bezeichnet der zweite Buchstabe (m) das openUTM-Modul, das den Fehler meldet (siehe folgende Liste).

3	KCSPEND (Operationen PEND, PGWT und RSET)
6	KCSMPUT (Operation MPUT)
7	KCSFPUT (Operationen FPUT und DPUT)
8	KCSSSB (Operationen SGET, SPUT, SREL, PTDA, GTDA und UNLK)
9	KCSLPUT (Operation LPUT)
A	KCSWAIT
B	KCSDGET (Operation DGET)
C	KCDCTRL (Operation CTRL)
D	KCDAPRO (UTM-D-Operationen APRO, MPUT, MGET, ...)
E	KCSPADM (Operation PADM)
F	KCSINFO (Operation INFO)
G	KCSDADM (Operation DADM)
H	KCSSIGN (Operation SIGN)
I	KCSISLP (interne Operationen)
K	allgemeiner DC-Code
L	KCSSTRT
M	KCSTRHD (interne Operationen)
N	KCSDLO
Q	KCSCRO (Operation QCRE)
R	KDCRTMM (ROOT-Code)
S	KCDSEFU (UTM-D Servicefunktionen)
T	KCDOSTM (OSI-TP Transaktionsbeendigung)

U	KDCRTDB (Bearbeitung von DB-Prozessen)
V	KCSTUTM (Operation MGET, MPUT für UPIC- und Socket-Clients)
W	KCDOSSF (OSI-TP Spezialfunktionen)
Y	KCDSYPM (LU6.1/OSI TP Transaktionsbeendigung)

Die folgende Tabelle beschreibt die Fehlerursachen und ggf. die Maßnahmen, die zur Fehlerbehebung notwendig sind.

Bei TESTMODE=OFF wird in den mit <sup>(1)</sup> gekennzeichneten Fällen der PENDER-Dump unterdrückt. Gleichzeitig unterbleibt das Nachladen, es sei denn für den aktuellen TAC sind PGWT-Aufrufe erlaubt und der KDCS-Aufruf, bei dem der Fehler auftrat, war kein PEND-Aufruf.

Code	Modul	Fehlerursache und ggf. Maßnahme
A010	KCSSADM	User ist kein SAT-Administrator.
A011	KCSSADM	TAC hat keine SAT-Berechtigung.
A012	KCSSADM	Subsystem SAT nicht verfügbar.
A013	KCSSADM	SAT-Version ist nicht kompatibel.
A015	KCSSADM	Ein Inverser KDCDEF läuft oder soll gestartet werden.
A016	KCSSADM	Die globale Cluster-Administration ist gesperrt.
A100	KCSADMI	Beim Aufruf der Programmschnittstelle der Administration ist die für den Parameterbereich angegebene Adresse entweder gar nicht oder nicht in der Länge des Parameterbereichs zugreifbar oder nicht auf Wortgrenze ausgerichtet. Maßnahme: Überprüfung der Adresse, die im Teilprogramm für den Parameterbereich angegeben wurde.
A101	KCSADMI	Beim Aufruf der Programmschnittstelle der Administration wurde intern ein ungültiger Returncode zurückgeliefert. Systemfehler. Maßnahme: Bitte Systemdienst informieren. Zur Diagnose wird der PENDER-Dump benötigt.
ABTR	KCSENDE	Der Code wird in den UTM-Meldungen K017 oder K055 mitgeteilt. Er tritt dann auf, wenn nach einer abnormalen Task Beendigung (vgl. ABNORMAL TASK TERMINATION im Teilnehmer- oder Batch-Betrieb) ein Vorgang vom UTM-Systemcode abnormal beendet wurde, weil die UTM-STXIT-Behandlung nicht gelaufen ist (STXIT ausgeschaltet oder STXIT-Code überschrieben). Der Ablauf der Anwendung wird dadurch i.a. nicht beeinträchtigt.
FH01	KDCRTFH	Ein Fehler bei der Formatierung ist aufgetreten. Siehe zusätzlichen Returncode von FHSCON bzw. FHS im Feld KCRMF. <sup>1</sup>
FH02	KDCRTFH	Warnung: Hardware-Funktion nicht vorhanden. <sup>1</sup>
FH04	KDCRTFH	Schwerer Fehler beim Initialisieren des Formatierungssystems FHS. UTM bricht den Start der Anwendung ab.

Code	Modul	Fehlerursache und ggf. Maßnahme
FR00	KDCRTFH	Bei der FORMAT-Exit Funktion "CHECK_FOR_DOWNLOAD" wurde ein Fehler entdeckt.
FR01	KDCRTFH	- Es wurde ein Formatkennzeichen mit "-" angegeben, aber die Anwendung ist ohne FORMAT-Exit generiert - es wurde ein "*" oder "+" Format angegeben, aber die UTM-Generierung enthält keine FORMSYS-Anweisung. <sup>1</sup>
FR02	KDCRTFH	KDCFOR nicht erlaubt für Teilformat. <sup>1</sup>
FR98	KDCRTFH	'-' Format, d.h. benutzereigene Formatierung, die Länge der Eingabenachricht wurde vom FORMAT-EXIT unzulässig gesetzt (kleiner 0). Der Returncode wird von KDCRTFH gesetzt.
FR99	KDCRTFH	'-' Format, d.h. benutzereigene Formatierung, die Länge des Restartbereichs wurde vom FORMAT-EXIT unzulässig verändert. Der Returncode wird von KDCRTFH gesetzt.
FRxx	KDCRTFH	Fehler bei '-' Format, d.h. benutzereigene Formatierung: xx ist der Returncode, der im Feld ....MRCF im benutzereigenen Kontrollbereich MDCBL eingetragen ist, er stammt vom FORMAT-Exit. Die Werte X'01', X'02', X'03', X'04', X'08', X'10', X'98' und X'99' sind für den Formatexit verboten, weil diese Rückkehrcodes für UTM und FHS reserviert sind.
K300	KCSPEND	Ungültige Operationsmodifikation beim PEND-Aufruf.
K301	KCSPEND	(Siehe KA00) Der Puffer für die Eingabenachricht ist zu klein. Maßnahme: Generierung mit KDCDEF, Pufferlänge mit MAX TRMSGLTH=length größer definieren. Es wird kein PEND ER Dump erzeugt.
K302	KCSPEND	Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
K303	KCSPEND	Ungültige Operationsmodifikation beim PGWT-Aufruf.
K304	KCSPEND	Das DB-System verlangt einen CLOSE DB-Aufruf vor PEND RE/PI, der aber nicht aufgerufen wurde.
K305	KCSPEND	Es wurde ein PEND KP aufgerufen bzw. ein PEND PA/PR mit TASK-Wechsel, obwohl das DB-System, bei dem eine Transaktion geöffnet wurde, einen PEND KP nicht erlaubt.
K306	KCSPEND	Die Transaktion musste zurückgesetzt und der Vorgang abnormal beendet werden, weil das DB-System die DB-Transaktion beim Schließen zurückgesetzt hatte.
K307	KCSPEND	PEND PS ist nur im ersten Teil des Anmeldevorgangs erlaubt.
K308	KCSPEND	Vor einem PEND KP oder PEND PA/PR mit TASK-Wechsel wurde die DB-Transaktion geschlossen.

Code	Modul	Fehlerursache und ggf. Maßnahme
K309	KCSPEND	Der in KCRN angegebene TAC ist nicht als Folge-TAC erlaubt.
K310	KCSPEND	Fehler beim Speichern der MPUT-Nachricht beim PEND PA/PR mit TASK-Wechsel.
K311	KCSPEND	Bei einem PEND RE/FI wird ein vorläufiges Transaktionsende (PTC) erreicht, aber das DB-System, bei dem eine Transaktion offen ist erlaubt den Transaktionsstatus nicht.
K312	KCSPEND	Eine formatierte Nachricht kann nicht ausgegeben werden, da sie zu lang ist. Maßnahme: evtl. Format einfacher gestalten.
K313	KCSPEND	In einem Teilprogrammablauf wurde ein SIGN OF/OB gegeben, aber die Ausgabenachricht ist für einen Auftragnehmer bestimmt oder der Programmablauf wurde mit einer nicht erlaubten PEND-Variante beendet.
K314	KCSPEND	Ein Teilprogrammablauf in einem Anmeldevorgang wurde nach einem erfolgreichen SIGN ON nicht mit PEND PS beendet.
K315	KCSPEND	Bei Vorgangskellerung wurde vor PEND FI ein MPUT PM mit KCLM/kclm > 0 gegeben. Die letzte Ausgabe vom Vorgängervorgang war eine LINEMODE-Nachricht und kann nur unverändert ausgegeben werden.
K316	KCSPEND	Es wurde PEND RS im Teilprogramm aufgerufen oder es ist eine Situation (z.B. Verbindungsverlust) eingetreten, die das Rücksetzen der Transaktion durch UTM erfordert, aber PEND RS ist nicht erlaubt, weil kein Vorgangswiederanlauf möglich ist. Dies ist der Fall, wenn - der Vorgang durch einen UPIC-Client oder einen OSI TP Auftraggeber, der nicht die Functional Unit Commit ausgewählt hat, gestartet wurde, und auf der Verbindung/Association kein Benutzer mit Wiederanlaufeigenschaft angemeldet ist und auch kein lokaler Vorgangswiederanlauf möglich ist, weil der letzte Sicherungs-Punkt nicht mit PEND SP oder PEND FC (nur UPIC) gesetzt wurde - oder der letzte Sicherungspunkt mit PGWT CM gesetzt wurde. Es wird kein PENDER-Dump geschrieben.
K317	KCSPEND	Die aktuell zu sichernde Größe des Vorgangs-Memory (SQL) ist größer als der durch VGMSIZE begrenzte zulässige maximale Wert.
K318	KCSPEND	PGWT ist in einem Teilprogramm aufgerufen worden, für dessen TAC - keine TAC-Klasse generiert ist oder - falls TAC-Klassen generiert sind: für die TAC-Klasse ist nicht PGWT=YES generiert oder - falls TAC-PRIORITIES generiert ist: der TAC ist nicht mit PGWT=YES generiert Maßnahme: Generierung mit KDCDEF korrigieren.
K319	KCSPEND	Anzahl der Prozesse ist nicht ausreichend für die Nutzung von PGWT oder aus einem UTM-System-Prozess wurde PGWT aufgerufen (nur von einem generierten privilegierten LTERM aus möglich). Maßnahme: Anzahl der Prozesse erhöhen.

Code	Modul	Fehlerursache und ggf. Maßnahme
K320	KCSPEND	Nach einem RSET in einer verteilten Transaktion wurde die Transaktion nicht zurückgesetzt, obwohl UTMD RSET = GLOBAL generiert wurde.
K321	KCSPEND	Der Editprofilname der gesicherten Nachricht des gekellerten Vorgangs ist nicht generiert.
K322	KCSPEND	Die Operationsmodifikation FC ist nicht in Asynchron- oder Server-Vorgängen erlaubt.
K323	KCSPEND	Die Operationsmodifikation FC ist nicht im Anmeldevorgang erlaubt, wenn ein Vorgangswiederanlauf notwendig ist.
K324	KCSPEND	KCRN ist nicht mit Leerzeichen belegt bei PEND RS oder PEND FR.
K325	KCSPEND	Die ausgewählte Operationsmodifikation ist nicht im Anmeldevorgang oder MSGTAC-Programm erlaubt oder der Resource Manager des XA-Anschlusses fordert beim xa_end ein Zurücksetzen der Transaktion im Anmeldevorgang oder MSGTAC-Programm.
K326	KCSPEND	Unerlaubter Wechsel zwischen Dialog- und Asynchron-Transaktionscode oder Folge-TAC bei PEND FC ist kein Vorgangs-TAC oder Folge-TAC bei PEND PA/PR, PS, KP, RE oder SP ist kein Folge-TAC
K327	KCSPEND	Die Operationsmodifikation FC ist im Anmeldevorgang nicht erlaubt, wenn der letzte Anmeldeversuch nicht erfolgreich war.
K328	KCSPEND	Der Anmeldevorgang kann nicht normal beendet werden, wenn die Gültigkeitsdauer des Passworts abgelaufen ist, aber das Passwort wurde nicht geändert wurde.
K329	KCSPEND	Der Anmeldevorgang kann nicht normal beendet werden, da das mit KDCUPD übertragene Passwort die geforderte Komplexitätsstufe nicht erfüllt oder zu kurz ist und nicht mit SIGN CP geändert wurde.
K330	KCSPEND	Ein SIGN OB im Anmeldevorgang für Terminal ist nur in Kombination mit einem MPUT NT/NE erlaubt.
K331	KCSPEND	Wird ein Anmeldevorgang mit anschließendem Vorgangswiederanlauf ohne Abmeldung des Benutzers beendet, so ist kein MPUT NT/NE erlaubt.
K332	KCSPEND	Wird der Anmeldevorgang mit anschließendem Vorgangswiederanlauf mit PEND FI ohne vorhergehenden MPUT beendet, so beendet openUTM den offenen Vorgang.
K333	KCSPEND	Es soll auf eine DGET-Nachricht gewartet werden; das Folgeteilprogramm liegt aber in keiner TAC-Klasse. Maßnahme: Folge-TAC mit TAC-Klasse generieren.
K334	KCSPEND	Es soll auf eine DGET-Nachricht gewartet werden; es sind nur PEND PA/PR/RS/ER/FR und PGWT RB erlaubt.
K335	KCSPEND	Der in PGWT wartende Vorgang wird abnormal beendet, da festgestellt wurde, dass nicht mehr genügend Prozesse aktiv sind, um den Vorgang normal fortzusetzen. Es wird kein PENDER-Dump geschrieben.

Code	Modul	Fehlerursache und ggf. Maßnahme
K336	KCSPEND	Vor PEND KP/RE/FI/FR/ER oder PGWT KP wurde kein MPUT aufgerufen, obwohl er erforderlich ist.
K337	KCSPEND	Vor PGWT PR wurde ein MPUT aufgerufen.
K338	KCSPEND	Die Transaktion wurde zurückgesetzt, weil die Anwendung beendet wird.
K339	KCSPEND	Das MSGTAC-Teilprogramm wurde beendet ohne mit FGET eine Meldung gelesen zu haben.
K340	KCSPEND	Beim Zurücksetzen einer Transaktion hat die Datenbank einen Fehler gemeldet, der das Nachladen des Anwendungsprogramms notwendig macht.
K341	KCSPEND	Nur in UTM-Cluster-Anwendungen: In der aktuellen Transaktion wurde ein User kreiert oder gelöscht. Die User-Datei konnte nicht gesperrt werden, da gerade durch einen KDCDEF-Lauf auf die User-Datei zugegriffen wird.
K342	KCSPEND	Nur in UTM-Cluster-Anwendungen: In der aktuellen Transaktion wurde ein User kreiert oder gelöscht. Die User-Datei konnte nicht gesperrt werden, da gerade ein anderer Prozess der selben Knotenanwendung oder eine andere Knotenanwendung die User-Datei gesperrt hat.
K343	KCSPEND	Nur in UTM-Cluster-Anwendungen: PTC-Status abgelehnt, weil der Vorgang durch die Administration gesperrt ist.
K344	KCSPEND	Nur in UTM-Cluster-Anwendungen: Der CR konnte nicht geschrieben werden, weil das globale ADM-Lock gesetzt ist.
K345	KCSPEND	Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
K346	KCSPEND	Das Beenden eines offenen Vorgangs durch den Anmelde-Vorgang ist nicht erlaubt, wenn eine Transaktion des offenen Vorgangs im Zustand "vorläufiges Transaktionsende (PTC)" ist. Der Anmelde-Vorgang wird abnormal beendet.
K347	KCSPEND	Nur in UTM-Cluster-Anwendungen: Die Transaktion muss in den Status "vorläufiges Transaktionsende (PTC)" gehen, aber beim Schreiben des Status in die User-Datei kam es zu einem Fehler.
K348	KCSPEND	Der Cluster Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF; Cluster Pagepool mit CLUSTER PGPOOL=(number,...) größer einrichten.
K349	KCSPEND	Die Verbindung zum Partner wurde abgebaut.
K350	KCSPEND	Ungültige Operationsmodifikation nach einer abnormalen Beendigung eines OSI TP Dialogs, auf dem die Functional Unit Commit ausgewählt wurde, mit CTRL AB.
K351	KCSPEND	Die Transaktion ist zum Rücksetzen markiert, es wurde aber ein PEND/PGWT-Aufruf zum Vorsetzen einer Transaktion verwendet.
K360	KCSPEND	Die Transaktion wurde mit PGWT RB zurückgesetzt, weil KCSPEND vom KDCROOT mit PGWT RB aufgerufen wurde.



Code	Modul	Fehlerursache und ggf. Maßnahme
K361	KCSPEND	Die Transaktion wurde mit PEND RS zurückgesetzt, weil KCSPEND vom KDCROOT mit PEND RS aufgerufen wurde.
K362	KCSPEND	Die Transaktion wurde zurückgesetzt und der Vorgang mit PEND ER abnormal beendet, weil KCSPEND vom KDCROOT mit PEND ER aufgerufen wurde.
K363	KCSPEND	Die Transaktion wurde mit RSET zurückgesetzt, weil KCSPEND vom KDCROOT mit RSET aufgerufen wurde.
K601	KCSMPUT	Puffer für Dialognachrichten zu klein generiert. Maßnahme: UTM-Generierung ändern, Puffer mit MAX NB=length größer wählen. <sup>1</sup>
K602	KCSMPUT	Der MPUT-Aufruf verlangt eine Format-Ausgabe, es wurde aber keine Formatierung generiert. <sup>1</sup>
K603	KCSMPUT	MPUT mit Bildschirmfunktion KCCARD, das Terminal hat jedoch keinen Ausweisleser. Maßnahme: UTM-Generierung oder Teilprogramm ändern. <sup>1</sup>
K604	KCSMPUT	MPUT mit Bildschirmfunktion KCCARD u. Benutzererkennung hat Operanden CARD=(position,string): Ausweisleser kann nicht für KDCSIGN-Prüfung und Dateneingabe zugleich verwendet werden. Maßnahme: UTM-Generierung oder Teilprogramm ändern.
K605	KCSMPUT	MPUT mit Bildschirmfunktion KCCARD und Ausgabe im Formatmodus. Maßnahme: Teilprogramm ändern.
K606	KCSMPUT	Ursache ist ein MPUT-Aufruf mit KCDF ungleich 0 und eine der folgenden Bedingungen - Folge-Teilnachricht im Formatmodus - KCMF = Name eines #-Formats - KCMF = Name eines Editprofils. Maßnahme: Teilprogramm ändern
K607	KCSMPUT	Die MPUT-Nachricht (angegeben durch Adresse und Länge) überdeckt teilweise einen ROOT-internen Pufferbereich, den MPUT-Buffer. Maßnahme: Teilprogramm ändern (Länge des MPUT zu groß?)
K608	KCSMPUT	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
K609	KCSMPUT	Bei einem Folgenachrichtenteil wurde ein anderer Editprofil-Name angegeben als bei dem ersten Nachrichtenteil. Maßnahme: Teilprogramm ändern
K610	KCSMPUT	Die Operationsmodifikation ES des MPUT-Aufrufs ist nur für UPIC- und SOCKET-Partner erlaubt.
K611	KCSMPUT	Beim Aufruf MPUT ES ist das Feld KCRN nicht mit Leerzeichen belegt.

Code	Modul	Fehlerursache und ggf. Maßnahme
K612	KCSMPUT	Im MSGTAC-Teilprogramm wurde MPUT aufgerufen
K613	KCSMPUT	Operationsmodifikation ungültig
K614	KCSMPUT	MPUT PM im Asynchronvorgang
K615	KCSMPUT	MPUT PM im ersten Teil des Anmeldevorgangs
K616	KCSMPUT	MPUT PM mit KCLM/kclm ungleich 0 im Anmeldevorgang
K617	KCSMPUT	MPUT PM, aber der Vorgang ist weder eingeschoben noch ein Anmeldevorgang.
K618	KCSMPUT	MPUT PM im Anmeldevorgang für einen UPIC-Verbindungs-User.
K701	KCSFPUT	UTM lehnt eine asynchrone Nachricht ab, weil im Pagepool bereits Level 2 erreicht ist. Maßnahme: Siehe K345
K702	KCSFPUT	In KCRN wurde der Name eines Dialog-TACs angegeben. Maßnahme: UTM-Generierung oder Teilprogramm ändern
K703	KCSFPUT	(Siehe K602) Der MPUT-Aufruf verlangt eine Format-Ausgabe, es wurde aber keine Formatierung generiert. <sup>1</sup>
K704	KCSFPUT	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
K705	KCSFPUT	Eine asynchrone Nachricht soll mit FPUT an einen LTERM- oder (OSI-)LPAP-Partner geschickt werden, für die in der KDCDEF-Generierung LTERM ...,QAMSG=N angegeben ist und zu der keine Verbindung besteht. UTM nimmt die Nachricht deshalb nicht an. Maßnahme: Verbindung herstellen
K706	KCSFPUT	Eine asynchrone Linemode-Nachricht soll mit FPUT/DPUT an ein Terminal geschickt werden. Die Nachricht ist länger als der Pufferbereich, der bei der Generierung im Parameter MAX NB festgelegt wurde. UTM nimmt die Nachricht nicht an. Maßnahme: Angabe bei MAX NB vergrößern oder Nachrichtenteile verkürzen
K707	KCSFPUT	Die bei einem DPUT-Aufruf angegebene Zielzeit verletzt die durch die Generierung festgelegten Grenzwerte (MAX DPUTLIMIT1= , DPUTLIMIT2= ) Maßnahme: Zeitangabe beim DPUT-Aufruf oder UTM-Generierung ändern
K708	KCSFPUT	Eine asynchrone Nachricht soll mit FPUT/DPUT an einen LTERM- oder (OSI-)LPAP-Partner, einen Asynchron-TAC, eine TAC-Queue, eine User-Queue, eine temporäre Queue oder ein (OSI-)LPAP geschickt werden. An diesem Ziel liegen bereits so viele asynchrone Nachrichten vor, wie bei der Generierung als Maximalwert festgelegt wurde (Parameter QLEV der KDCDEF-Anweisungen LTERM, TAC, USER, QUEUE, LPAP oder OSI-LPAP).

Code	Modul	Fehlerursache und ggf. Maßnahme
K709	KCSFPUT	Bei einem Folgenachrichtenteil wurde ein anderer Editprofil-Name angegeben als bei dem ersten Nachrichtenteil. Maßnahme: Teilprogramm ändern
K710	KCSFPUT	Ursache ist ein FPUT/DPUT-Aufruf mit KCDF ungl. 0 und eine der folgenden Bedingungen - KCMF/kcfn = Name eines #-Formats - KCMF/kcfn = Name eines Editprofils Maßnahme: Teilprogramm ändern
K711	KCSFPUT	In einem DPUT-Aufruf enthält KCRN ein neues Ziel, obwohl die vorangehende DPUT Nachricht an ein anderes Ziel noch nicht abgeschlossen ist.
K712	KCSFPUT	In KCRN wurde der Name eines UTM-D-Partners, einer MUX-Verbindung, oder eines UPIC-Partners angegeben. Maßnahme: UTM-Generierung oder Teilprogramm ändern
K713	KCSFPUT	Ursache ist ein FPUT/DPUT-Aufruf an einen Ausweisleser. Maßnahme: Teilprogramm ändern
K714	KCSFPUT	Bei einem FPUT wurde in KCRN der Name eines Message-Komplexes angegeben. Maßnahme: Teilprogramm ändern
K715	KCSFPUT	In KCRN wurde der Name eines Message-Komplexes angegeben; dieser Name passt jedoch nicht zu dem momentan eröffneten Message-Komplex Maßnahme: Teilprogramm ändern
K716	KCSFPUT	In KCRN wurde der Name eines Message-Komplexes angegeben; z.Zt. ist jedoch kein Message-Komplex eröffnet. Maßnahme: Programm korrigieren
K717	KCSFPUT	In KCRN wurde der Name eines TACs angegeben; dies ist unvereinbar mit der Angabe in KCOM. Maßnahme: Teilprogramm ändern
K718	KCSFPUT	Ein Asynchronvorgang, der unter einem gelöschten LTERM abläuft, hat versucht einen FPUT oder DPUT abzusetzen.
K719	KCSFPUT	Ein Asynchronvorgang, der unter einem gelöschten USER abläuft, hat versucht einen FPUT oder DPUT abzusetzen.
K720	KCSFPUT	Bei DPUT-Aufruf an eine USER-Queue: Es gibt keinen USER mit dem in KCRN angegebenen Namen oder der USER wurde gelöscht.
K721	KCSFPUT	Bei DPUT-Aufruf an eine USER-Queue: Der KSET des Benutzers und die Schreib-ACL der USER-Queue haben keinen gemeinsamen Schlüssel
K722	KCSFPUT	Zeitgesteuerte DPUT-Aufrufe für USER-Queues sind nicht möglich. (KCMOD ungleich Leerzeichen)
K723	KCSFPUT	Bei DPUT-Aufruf an ein QUEUE-Objekt: Es gibt kein QUEUE-Objekt mit dem in KCRN angegebenen Namen.

Code	Modul	Fehlerursache und ggf. Maßnahme
K724	KCSFPUT	Bei DPUT-Aufruf an eine USER-Queue: Der KSET des LTERMs und die Schreib-ACL der USER-Queue haben keinen gemeinsamen Schlüssel
K725	KCSFPUT	Zeitgesteuerte DPUT-Aufrufe für QUEUE-Objekte sind nicht möglich. (KCMOD ungleich Leerzeichen)
K726	KCSFPUT	Ungültiger Wert in KCQTYP.
K727	KCSFPUT	Die aktuelle RSO-Version unterstützt diese Funktion nicht.
K728	KCSFPUT	Das Nachrichtenziel ist kein RSO-Drucker.
K729	KCSFPUT	Das Nachrichtenziel ist kein LTERM.
K730	KCSFPUT	Eine Nachricht, für die bereits ein Nachrichtenteil geschrieben wurde, darf nicht mit einem FPUT RP bzw. DPUT RP Aufruf erweitert werden.
K731	KCSFPUT	An einen TAC, der mit CALL=NEXT generiert ist, darf keine Asynchronnachricht geschickt werden.
K732	KCSFPUT	An KDCMSGLT darf keine Asynchronnachricht geschickt werden.
K733	KCSFPUT	An ein implizit für die interne Cluster-Kommunikation generiertes LTERM darf keine Asynchronnachricht geschickt werden.
K801	KCSSSB	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
K802	KCSSSB	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
K804	KCSSSB	Es wurden keine GSSBs generiert oder es wurden mehr GSSBs erzeugt als per UTM-Generierung erlaubt.
K805	KCSSSB	Es wurden mit SPUT mehr LSSBs erzeugt als bei der UTM-Generierung angegeben wurde.
K810	KCSSSB	Beim Zugriff auf einen GSSB, TLS oder ULS: Der Speicherbereich kann nach Ablauf einer Wartezeit von der Transaktion nicht belegt werden. Maßnahme: Wartezeit größer machen, dazu in KDCDEF-Generierung Parameter RESWAIT=(time1,...) erhöhen.
K811	KCSSSB	Beim Zugriff auf einen GSSB, TLS oder ULS: Der Speicherbereich ist derzeit von einer anderen Transaktion "unbestimmt lange" gesperrt, d.h. sie hat den Speicherbereich gesperrt und danach einen PEND KP oder PGWT KP Aufruf abgesetzt.
K812	KCSSSB	Beim Zugriff auf einen ULS: Die Benutzerkennung, auf deren ULS zugegriffen werden soll, wird gelöscht.
K813	KCSSSB	Die Anwendung wird beendet.

Code	Modul	Fehlerursache und ggf. Maßnahme
K820	KCSSSB	Das Warten auf einen globalen sekundären Speicherbereich würde zu einem Deadlock führen.
K822	KCSSSB	(Siehe K348) Der Cluster Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF; Cluster Pagepool mit CLUSTER PGPOOL=(number,...) größer einrichten.
K823	KCSSSB	Nur in UTM-Cluster-Anwendungen: Bei der Anforderung der Datei-Sperre auf die Verwaltungsdatei für GSSB oder ULS kam es zu einem Timeout.
K824	KCSSSB	Beim Zugriff auf einen GSSB, TLS oder ULS: Der Speicherbereich ist momentan gesperrt und die Task kann nicht auf die Freigabe der Sperre warten, weil bereits zuviele Tasks in einem Wartezustand sind. Maßnahme: - Mehr Tasks starten. - Die Anzahl der Tasks für die TACs begrenzen, die auf GSSB oder ULS zugreifen, und mehr Tasks als diese Grenze starten. Die Informationen, wie Sie UTM-Services, die GSSB bzw. ULS Speicherbereiche verwenden, auf die Tasks einer UTM-Cluster-Anwendung verteilen, finden Sie im openUTM-Handbuch "Einsatz von openUTM-Anwendungen"
K825	KCSSSB	Nur in UTM-Cluster-Anwendungen: Beim Zugriff auf einen GSSB oder ULS: Die Warteschlange an dem angeforderten Bereich hat bereits die maximale Länge erreicht; der Vorgang kann z.Zt. nicht auf diesen Speicherbereich warten.
K826	KCSSSB	Nur in UTM-Cluster-Anwendungen: Beim Zugriff auf einen GSSB oder ULS: Bei der Anforderung eines internen Betriebsmittels kam es zu einem transienten Fehler; ein Zugriff auf den angeforderten Bereich ist z.Zt. nicht möglich.
K827	KCSSSB	Nur in UTM-Cluster-Anwendungen: Beim Zugriff auf einen GSSB oder ULS: Der Bereich ist momentan durch einen ausgefallenen Knoten gesperrt; ein Warten auf diesen Bereich ist z.Zt. nicht sinnvoll.
K901	KCSLPUT	UTM lehnt LPUT-Aufruf ab, weil im Pagepool bereits Level 2 erreicht ist. Maßnahme: Teilprogramm ändern oder per UTM-Generierung Pagepool vergrößern.
K902	KCSLPUT	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>

Code	Modul	Fehlerursache und ggf. Maßnahme
K903	KCSLPUT	UTM lehnt LPUT-Aufruf ab, da beim letzten Schreibvorgang auf die Benutzer-Protokolldatei(en) ein DMS-Fehler aufgetreten ist. Der DMS-Fehler ist mit der K043-Meldung ausgegeben worden. Die im Pagepool gepufferten Sätze bleiben erhalten. Maßnahme: K043-Meldung und DMS-Fehlercode auswerten, Benutzer-Protokolldatei(en) restaurieren oder neu anlegen, anschließend mit dem Administrationskommando KDCLOG fortfahren (oder über entsprechende Programmschnittstelle). Die LPUT-Sätze im Pagepool werden dann ausgegeben. Die Sperre für den LPUT-Aufruf wird aufgehoben.
KA00	KCSWAIT	Der Puffer für die Eingabenachricht ist zu klein. Maßnahme: Generierung mit KDCDEF, Pufferlänge mit MAX TRMSGLTH=length größer definieren. Es wird kein PEND ER Dump erzeugt.
KA01	KCSWAIT	Ein schwerwiegender Fehler in der Kommunikation zwischen Auftraggeber und Auftragnehmer (UTM-D) ist aufgetreten. Die Kommunikation kann nicht weitergeführt werden, UTM bricht den Auftragnehmer-Vorgang mit PEND ER ab. Mögliche Ursache: - PEND ER durch den Auftraggeber. - Timeout auf der Verbindung zum Auftraggeber Diagnosehilfen: - UTM-D-Fehlernachricht K086 auswerten - bei PEND ER im AG-Vorgang Ursache im Dump suchen.
KA02	KCSWAIT	Der Pagepool in KDCFILE ist voll, deshalb konnten der im PGWT wartenden Task die Nachrichten nicht übergeben werden. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten.
KA03	KCSWAIT	UTM hat eine unverschlüsselte Nachricht empfangen, obwohl eine verschlüsselte Nachricht erwartet wurde.
KA04	KCSWAIT	Die Verbindung zum Partner wurde abgebaut.
KA05	KCSWAIT	Eine Transaktion im Zustand PTC wurde auf Anforderung der Administration zurückgesetzt.
KB01	KCSDGET	Im ersten Teil des Anmeldevorgangs sind DGET-Aufrufe nicht erlaubt.
KB02	KCSDGET	Es muss bereits auf eine DGET-Nachricht gewartet werden; daher sind keine weiteren DGET-Aufrufe erlaubt.
KB03	KCSDGET	Ungültiger Wert von KCQTYP.
KB04	KCSDGET	Zu dem in KCRN angegebenen Namen vom Typ KCQTYP existiert kein Objekt bzw. dieses Objekt wurde gelöscht.
KB05	KCSDGET	Nachrichten an Dialog- oder Asynchron-TACs können nicht mittels DGET-Aufruf gelesen werden.
KB06	KCSDGET	Die Lese-ACL der TAC- bzw. USER-Queue und der KSET des LTERMs des Benutzers haben keinen gemeinsamen Schlüssel.

Code	Modul	Fehlerursache und ggf. Maßnahme
KB07	KCSDGET	Die Lese-ACL der TAC- bzw. USER-Queue und der KSET des Benutzers haben keinen gemeinsamen Schlüssel.
KB08	KCSDGET	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
KB10	KCSDGET	Negative Wartezeit beim Aufruf DGET FT/BF oder: Wartezeit ungleich 0 beim Aufruf DGET NT/BN/PF/PN.
KB11	KCSDGET	Für das MSGTAC-Programm sind DGET-Aufrufe mit Warten nicht erlaubt.
KB12	KCSDGET	Beim Aufruf DGET NT/BN/PN passen Name oder Typ der angegebenen Queue nicht zum vorherigen DGET-Aufruf des aktuellen Teilprogrammlaufes.
KB13	KCSDGET	Es wurde versucht, von einer USER Queue zu lesen, obwohl die Anwendung ohne Benutzer generiert wurde.
KB14	KCSDGET	Es wurde versucht, von einer TAC Queue mit STATUS= HALT oder KEEP zu lesen.
KB15	KCSDGET	Es wurde versucht, von einer TAC Queue mit ADMIN=Y zu lesen; der Benutzer ist aber kein Administrator.
KB16	KCSDGET	Operationsmodifikation in KCOM ungültig.
KB17	KCSDGET	Bei DGET BF/PF: KCDPID enthält keine gültige DPUT-ID.
KB18	KCSDGET	Bei DGET FT/NT: KCMF/kcfn enthält keine Leerzeichen.
KB19	KCSDGET	Bei DGET NT/BN/PN (next): - KCOM passt nicht zum vorigen DGET-Aufruf oder - es wurde in diesem Programmlauf noch kein DGET FT/BF/PF (first) gegeben oder - es wurde dazwischen ein PGWT aufgerufen.
KB20	KCSDGET	Bei DGET NT/BN/PN: Seit dem letzten DGET-Aufruf wurde die DGET-Queue gelöscht und neu erzeugt.
KB21	KCSDGET	Bei DGET BN/PN (next): Seit dem letzten DGET-Aufruf wurde der in KCRN und KCQTYP angegebene USER bzw. die Temporäre Queue gelöscht und neu erzeugt.
KB22	KCSDGET	Bei DGET BN: Es existiert keine Nachricht mit der in KCGTM angegebenen Erzeugungszeit oder diese wurde inzwischen verarbeitet.
KB23	KCSDGET	Bei DGET PF: Es existiert keine Nachricht mit der in KCGTM angegebenen Erzeugungszeit oder diese wurde inzwischen verarbeitet.
KB24	KCSDGET	Bei DGET BF/PF: KCDPID passt nicht zu den Angaben in KCRN und KCQTYP.
KB25	KCSDGET	Im Anmeldevorgang sind DGET-Aufrufe mit Warten nicht erlaubt.





Code	Modul	Fehlerursache und ggf. Maßnahme
KC24	KCDCTRL	Der Aufruf CTRL PE ist nicht erlaubt, weil der Transaktionsstatus P ist.
KC25	KCDCTRL	Der nicht verwendete Teil des erweiterten Parameterbereich (EXTENT) ist nicht mit binär Null versorgt.
KC26	KCDCTRL	Das Feld KCNORPLY hat weder den Wert Y noch binär Null.
KD00	KCDAPRO	KCRN enthält keine gültige Vorgangsidentifikation (MGET, MPUT, FPUT, DPUT )
KD01	KCDAPRO	LTAC ist nicht definiert
KD02	KCDAPRO	LTAC darf nicht benutzt werden, LTAC ist gesperrt
KD03	KCDAPRO	Für das Schloss des LTACs hat der Benutzer keinen Schlüssel an seinem Schlüsselbund
KD04	KCDAPRO	Es wurde ein APRO DM aufgerufen, aber der LTAC ist als Asynchron LTAC generiert, bzw. APRO AM und LTAC als Dialog LTAC generiert
KD05	KCDAPRO	Es wurden mit APRO mehr Auftragnehmervorgänge erzeugt als bei der Generierung angegeben wurde. (KDCDEF Generierung: UTMD MAXJR= )
KD06	KCDAPRO	MPUT an AN-Vorgang und KCDF ungleich binär 0
KD08	KCDAPRO	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
KD09	KCDAPRO	Es wurde ein APRO DM aufgerufen, aber zur entfernten Anwendung wurden die logischen Verbindungen mit dem Administrationskommando KDCLPAP ACT=QUIET abgebaut.
KD10	KCDAPRO	Es wurde ein APRO DM aufgerufen, aber zur entfernten Anwendung ist keine logische Verbindung aufgebaut.
KD11	KCDAPRO	Es wurde ein APRO DM aufgerufen, aber die entfernte Anwendung ist als "contention winner" generiert und die Wartezeit für Sessionbelegung ist 0 (nur bei LU6.1).
KD12	KCDAPRO	Nach einem MPUT NE/HM an einen AN-Vorgang wurde ein weiterer MPUT an den AN-Vorgang gegeben.
KD13	KCDAPRO	Bei einem FPUT (DPUT mit KCMOD=' ') an einen Auftragnehmer-Vorgang besteht zur entfernten Anwendung keine Verbindung und die Wartezeit für die Belegung einer Session bzw. Association ist 0.
KD14	KCDAPRO	Ein asynchroner Vorgang in einer anderen Anwendung soll mit APRO adressiert werden. Für diese Anwendung liegen bereits so viele asynchrone Nachrichten vor, wie bei der Generierung als Maximalwert festgelegt wurde. (LPAP QLEV= bzw. OSI-LPAP QLEV=)
KD15	KCDAPRO	Bei einem MPUT EM oder MPUT HM ist das Ziel in KCRN kein OSI TP Kommunikationspartner.
KD16	KCDAPRO	Vor einem MPUT HM wurde kein MPUT NT gegeben.

Code	Modul	Fehlerursache und ggf. Maßnahme
KD17	KCDAPRO	Der RTAC wurde für einen OSI TP-Partner generiert.
KD18	KCDAPRO	Eine ausgewählte Functional Unit wird von dieser UTM Version nicht unterstützt.
KD22	KCDAPRO	Es wurde die Functional Unit Commit ausgewählt, aber für den Partner ist die abstrakte Syntax CCR nicht generiert.
KD23	KCDAPRO	Mischbetrieb von LU6.1 und OSI TP innerhalb einer verteilten Transaktion
KD24	KCDAPRO	(Siehe KD23) Mischbetrieb von LU6.1 und OSI TP innerhalb einer verteilten Transaktion
KD25	KCDAPRO	Es wird bei einer verteilten Transaktion mit OSI TP mehr als ein ACCESS-POINT benutzt
KD26	KCDAPRO	Es wird bei einem OSI TP-Partner eine nicht generierte abstrakte Syntax bei MPUT/FPUT/DPUT in KCMF/kcfn angegeben. Die Syntaxnamen "CCR" und "OSITP" sind nicht zulässig
KD27	KCDAPRO	KCLM/kclm muss 32 sein.
KD28	KCDAPRO	KCLM/kclm muss Null sein.
KD29	KCDAPRO	KCLM/kclm muss gleich der Länge der Datenstruktur wie in COPY KCAPROC resp. Include kcapro.h sein.
KD30	KCDAPRO	Ungültige Werte in APRO Datenbereich
KD31	KCDAPRO	Bei KCFUCOM = 'N' muss KCFUCHN Leerzeichen enthalten
KD32	KCDAPRO	Es wurde Security Type Same oder Program ausgewählt, aber für den Partner ist die abstrakte Syntax UTMSEC nicht generiert.
KD33	KCDAPRO	Bei Security Type Program wurde eine ungültige Länge für Userid oder Password angegeben.
KD34	KCDAPRO	Fehler beim Encodieren der Security Daten
KD35	KCDAPRO	KCFUHS = 'Y' ist nur bei Dialogpartner erlaubt
KD36	KCDAPRO	Nach einem CTRL PR oder PE wurde ein MPUT HM an den gleichen Partner gegeben.
KD37	KCDAPRO	Nach einem CTRL AB wurde ein MPUT an den gleichen Partner gegeben.
KD38	KCDAPRO	MPUT an Auftraggeber, aber KCSEND = NO.
KD39	KCDAPRO	Die nicht benutzten Felder für die Security-Funktion bei KCSECTYP ungleich 'P' sind nicht gelöscht. (CHAR-Felder auf Blank, numerische Felder auf 0).
KD40	KCDAPRO	Dem OSI-LPAP beim APRO-Aufruf ist keine aktive Verbindung zugeordnet.
KD41	KCDAPRO	Das Master-LPAP ist gesperrt
KD42	KCDAPRO	(Siehe K348) Der Cluster Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF; Cluster Pagepool mit CLUSTER PGPOOL=(number,...) größer einrichten.

Code	Modul	Fehlerursache und ggf. Maßnahme
KE01	KCSPADM	Berechtigung für Aufruf fehlt: Der Benutzer ist kein Administrator und das Terminal ist keine Druckersteuerstation bzw. stimmt nicht mit der im KDCS-Parameterbereich angegebenen (KCLT/kclt) Datenstation überein.
KE02	KCSPADM	Der Druckersteuerstation ist kein Drucker (PTERM) zugeordnet. UTM-Generierung überprüfen!
KE03	KCSPADM	Ein PADM-Aufruf zur Druckbestätigung oder zur Druckwiederholung (KCOM=OK/PR) erfolgte, obwohl keine Druckausgabe für den angegebenen Drucker zu bestätigen war.
KE04	KCSPADM	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
KE05	KCSPADM	Ein Drucker soll einem anderen LTERM zugeordnet werden (KCOM=CA), ist aber mit der Anwendung verbunden.
KE06	KCSPADM	Verbindung zu einem Drucker soll aufgebaut werden, der Drucker ist aber gesperrt.
KE07	KCSPADM	Ein Inverser KDCDEF läuft oder soll gestartet werden.
KE08	KCSPADM	Die Cluster-globale Administration ist zur Zeit nicht möglich, da Knoten-Anwendungen mit unterschiedlicher UTM-Generierung laufen.
KF01	KCSINFO	Der Benutzer ist ohne Ausweiskarte generiert und für den LTERM-Client wurde kein Kerberos-Dialog durchgeführt (INFO CD).
KF02	KCSINFO	Die Information ist nicht mehr verfügbar, z.B. wegen Verbindungsverlust (INFO CD).
KF03	KCSINFO	Der Kerberos-Dialog hatte einen Fehler geliefert, dennoch wird möglicherweise eine Kerberos-Information übergeben (INFO CD)
KF04	KCSINFO	die Kerberos-Information ist nicht vollständig, weil sie länger ist wie der bei MAX PRINCIPAL-LTH generierte Wert (INFO CD).
KG01	KCSDADM	Berechtigung für Aufruf fehlt: Der Benutzer ist kein Administrator und das Terminal ist keine Druckersteuerstation bzw. stimmt nicht mit dem im KDCS-Parameterbereich angegebenen (KCLT/kclt) Terminal überein.
KG02	KCSDADM	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
KG03	KCSDADM	Ein DADM-Aufruf zum Löschen aller freilaufenden Nachrichten (KCOM=DA) erfolgte, obwohl für das angegebene Ziel Nachrichten in Bearbeitung waren.
KG04	KCSDADM	Es wurde versucht eine in Bearbeitung befindliche Nachricht zu administrieren.

Code	Modul	Fehlerursache und ggf. Maßnahme
KG05	KCSDADM	Nach einem Aufruf zum Löschen von Nachrichten (KCOM=DL/DA) erfolgt ein weiterer Löschauftrag bzw. ein Auftrag zum Umketten einer Nachricht (KCOM=DL/DA/CS).
KG06	KCSDADM	Es wurde versucht eine zeitgesteuerte Nachricht umzuketten (KCOM=CS), obwohl deren Startzeitpunkt noch nicht erreicht war.
KG07	KCSDADM	Ein Aufruf zum Lesen von Informationen über Nachrichten (KCOM=RQ) erfolgte, obwohl für das angegebene Ziel keine (oder nur in Bearbeitung befindliche) Nachrichten vorhanden waren.
KG08	KCSDADM	KCQTYP ungültig
KG09	KCSDADM	Bei DADM MV mit Leerzeichen in KCLT/kclt ist das ursprüngliche Ziel der Nachricht in der Dead Letter Queue nicht mehr vorhanden. Dieser Nachricht muss ein neues Ziel zugeordnet werden.
KG10	KCSDADM	Bei DADM MV ist das in KCLT/kclt angegebene Ziel nicht mehr gültig.
KG11	KCSDADM	Bei DADM MV passt der Typ des in KCLT/kclt angegebenen Ziels nicht zum ursprünglichen Ziel der Nachricht.
KH01	KCSSIGN	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
KH02	KCSSIGN	Bei SIGN CP: Das neue Passwort genügt nicht der geforderten Komplexitätsstufe
KH03	KCSSIGN	Bei SIGN CP: Das neue Passwort ist zu kurz
KH04	KCSSIGN	Bei SIGN CP: Das neue Passwort ist identisch dem alten Passwort oder mit einem Passwort aus der Passwort-Historie
KH05	KCSSIGN	Bei SIGN CL: das angegebene Sprachkennzeichen ist ungültig
KH06	KCSSIGN	Bei SIGN CL: das angegebene Territorialkennzeichen ist ungültig
KH07	KCSSIGN	Bei SIGN CL: der angegebene Zeichensatzname ist ungültig
KH08	KCSSIGN	Ein Inverser KDCDEF läuft oder soll gestartet werden.
KH09	KCSSIGN	Bei SIGN CP: Das Passwort darf noch nicht geändert werden, da die minimale Gültigkeitsdauer noch nicht überschritten ist.
KH10	KCSSIGN	Bei SIGN CP: Für einen Benutzer, der mit Kerberos-Authentisierung generiert ist, darf kein Passwort vergeben bzw. geändert werden.
KH11	KCSSIGN	Der Vorgang wurde abnormal beendet, weil beim Abmelden des Benutzers die Pagelist für die Cluster Vorgangs-Daten nicht geschrieben werden konnte, weil der Cluster Pagepool voll war.

Code	Modul	Fehlerursache und ggf. Maßnahme
KH12	KCSSIGN	Das Fortsetzen eines offenen Cluster-Vorgangs ist nicht möglich, da durch eine Neugenerierung der Knoten-KDCFILE die Reihenfolge der TAC-Anweisungen geändert wurde oder die Eigenschaften des Folge-TAC geändert wurden oder der Folge-TAC oder der Vorgangs-TAC nicht mehr existieren.
KI01	KCSISLP	VTSU erkennt, dass eine Dialog-Ausgabenachricht mit Format-Exit fehlerhaft aufbereitet wurde.
KI02	KCSISLP	VTSU meldet unzulässige Edit-Optionen bei Dialog-Ausgabenachricht.
KI03	KCSISLP	VTSU erkennt, dass eine Dialog-Ausgabenachricht mit einem ungültigen oder von dem Terminal nicht unterstützten Zeichensatz gesendet werden sollte.
KJ01	KCCUSF	Der Vorgang wurde abnormal beendet, weil er durch eine andere Knoten-Anwendung zum abnormalen Beenden markiert wurde. Dies kann administrativ erfolgt sein oder dadurch, dass sich der User trotz eines an diese Knoten-Anwendung gebundenen Vorgangs an eine andere Knoten-Anwendung angemeldet hat.
KK01	KCSSVCS	Der zweite Parameter des KDCS-Aufrufs fehlt.
KK02	KCSSVCS	Die Adressvalidierung des zweiten Parameters des KDCS-Aufrufs ist fehlgeschlagen.
KK03	KCSSVCS	Der KDCS-Aufruf erfolgte aus einem Teilprogramm eines Asynchron-Vorgangs.
KK04	KCSSVCS	Der KDCS-Operationscode ist unbekannt.
KK07	KCSSVCS	Der KDCS-Operationscode APRO ist nicht erlaubt.
KK08	KCSSVCS	Der Aufruf ist in UTM-Cluster-Anwendungen nicht erlaubt.
KL00	KCSSTRT	Beim Warmstart der Anwendung wurde keine Auftragnehmer-Session gefunden, die im PTC-Zustand ist.
KL01	KCSSTRT	Beim Warmstart der Anwendung wurde für den OSI TP Vorgang kein gültiger NODE-Tabelleneintrag gefunden.
KL02	KCSSTRT	Der Benutzer, der diesen Vorgang gestartet hat, wurde gelöscht.
KL03	KCSSTRT	Beim Warmstart wurde festgestellt, dass der Vorgang nicht fortgesetzt werden kann. OSI TP Auftragnehmer-Vorgänge werden ggfs. erst nach dem Association-Aufbau zum OSI TP Partner und anschließender OSI TP Recovery beendet.
KL04	KCSSTRT	PEND ER veranlasst durch KCSSTRT.
KL05	KCSSTRT	PEND ER veranlasst durch KCSSTRT.
KL06	KCSSTRT	PEND ER veranlasst durch KCSSTRT.
KL07	KCSSTRT	PEND ER veranlasst durch KCSSTRT.
KL08	KCSSTRT	PEND ER veranlasst durch KCSSTRT.
KM01	KCSTRHD	TAC ist nicht definiert. <sup>1</sup>

Code	Modul	Fehlerursache und ggf. Maßnahme
KM02	KCSTRHD	TAC darf nicht benutzt werden, der TAC ist gesperrt. <sup>1</sup>
KM03	KCSTRHD	Für das Schloss des TACs gibt es keinen Schlüssel im Schlüsselbund der LTERM- oder (OSI-)LPAP-Partner. <sup>1</sup>
KM04	KCSTRHD	Für das Schloss des TACs hat der Benutzer keinen Schlüssel an seinem Schlüsselbund. <sup>1</sup>
KM05	KCSTRHD	Der TAC ist als Administrations-TAC generiert, aber der Benutzer ist kein Administrator. <sup>1</sup>
KM07	KCSTRHD	Für den Transaktionscode ist das zugehörige Teilprogramm nicht eingebunden oder konnte nicht geladen werden. <sup>1</sup>
KM08	KCSTRHD	Der Transaktionscode ist vollständig gesperrt. Er darf weder in einem PEND-Aufruf als Folge-TAC angegeben werden, noch können neue Aufträge für diesen TAC gegeben werden. <sup>1</sup>
KM09	KCSTRHD	Der Transaktionscode ist durch einen Encryptionlevel geschützt und der Aufrufer unterstützt diesen Encryptionlevel nicht oder die Eingabenachricht wurde nicht dem Level entsprechend verschlüsselt.
KM99	KCSTRHD	Ein TAC, der mit XOPEN-API generiert ist, darf nur von einem Auftraggeber verwendet werden, mit dem über das OSI TP, LU6.1- oder das UPIC-Protokoll kommuniziert wird.
KN01	KCSDLO	Der Benutzer, der den Vorgang gestartet hat, wurde gelöscht.
KQ01	KCSCRO	Beim Aufruf QCRE WN beginnt der Name in KCRN mit einer Ziffer.
KQ02	KCSCRO	Beim Aufruf QCRE NN wurde KCRN nicht mit Leerzeichen versorgt.
KQ03	KCSCRO	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>
KQ04	KCSCRO	Die bei der UTM-Generierung mit der QUEUE-Anweisung reservierten Tabellen für QUEUE-Objekte sind aufgebraucht. Maßnahme: Nicht mehr benötigte QUEUE-Objekte löschen oder neu generieren.
KQ05	KCSCRO	Beim Aufruf QCRE WN enthält KCRN ein ungültiges Zeichen oder ist nicht mit Leerzeichen aufgefüllt.
KR01	KDCRTMM	Es fehlt ein Dialog-Teilprogramm (nicht gebunden), mit dem ein Vorgang fortgesetzt werden sollte. <sup>1</sup>
KR02	KDCRTMM	Es fehlt ein Asynchron-Teilprogramm (nicht gebunden) , für das eine Nachricht auf Verarbeitung wartet. <sup>1</sup>
KR04	KDCRTMM	Ein Teilprogrammmlauf hat vor dem letzten KDCS-Aufruf über das Ende des KB hinausgeschrieben. Maßnahme: Teilprogramm oder UTM-Generierung ändern.

Code	Modul	Fehlerursache und ggf. Maßnahme
KR05	KDCRTMM	Ein Teilprogrammlauf hat vor dem letzten KDCS-Aufruf über das Ende des SPAB hinausgeschrieben. Maßnahme: Teilprogramm oder UTM-Generierung ändern.
KR06	KDCRTMM	Ein VORGANG-Exit hat einen KDCS-Aufruf gegeben. Der Vorgang wird mit PEND ER abgebrochen.
KR07	KDCRTMM	Das FORMAT-Exit-Teilprogramm setzt einen KDCS-Aufruf ab. Der Vorgang wird mit PEND ER abgebrochen.
KR08	KDCRTMM	Nicht erlaubter DB-User-Call im Anmeldevorgang. Der Vorgang wird mit PEND ER abgebrochen.
KR09	KDCRTMM	Fehler beim Nachladen eines Teilprogramms. Der Vorgang wird mit PEND ER abgebrochen.
KR10	KDCRTMM	Programm für Vorgangs-Exit fehlt. Der Vorgang wird mit PEND ER abgebrochen.
KS00	KCDSEFU	Vor einer Transaktionsendeanforderung wurden MPUTs an mehrere Partner gegeben.
KS01	KCDSEFU	Bei einer Transaktionsendeanforderung hat mindestens ein Partner einen nicht erlaubten Transaktions- oder Vorgangsstatus.
KS02	KCDSEFU	Bei einer Transaktionsendeanforderung ist mindestens eine Session nicht benutzt worden (zuviele APRO Aufrufe).
KS03	KCDSEFU	Vom Auftraggeber wurde MPUT an eine VGID gegeben, danach wurde die Transaktion mit PEND RE beendet, obwohl nicht alle Auftragnehmer PEND RE/FI gemeldet haben.
KS04	KCDSEFU	Es wurde ein PEND KP aufgerufen, nachdem eine Nachricht an einen LU6.1-Partner abgesetzt wurde, der schon das Transaktionsende eingeleitet hat.
KS05	KCDSEFU	Ist die Session sowohl Auftragnehmer als auch Auftraggeber, darf sie an ihren Auftragnehmer erst dann PEND RE melden, wenn ihr Auftraggeber ebenfalls PEND RE gemeldet hat.
KS06	KCDSEFU	Nach einem APRO AM wurde keine freilaufende Nachricht für den adressierten Vorgang ausgegeben
KS07	KCDSEFU	Es wurde PEND SP gegeben, obwohl es mindestens einen Partner mit offener Transaktion gibt.
KS08	KCDSEFU	Der OSI TP-Client sendet eine Nachricht an den OSI TP-Server, ohne die Transaktion zu beenden, oder fordert den OSI TP-Server auf, die Transaktion zu beenden, obwohl der Server das Transaktionsende bereits eingeleitet hat.
KS09	KCDSEFU	Der OSI TP-Server hat die Transaktion im Widerspruch zur Forderung des OSI TP-Client nicht beendet.
KS10	KCDSEFU	Der OSI TP-Server hat die Transaktion im Widerspruch zur Forderung des OSI TP-Clients mit PEND SP beendet.

Code	Modul	Fehlerursache und ggf. Maßnahme
KS11	KCDSEFU	Der OSI TP-Client sendet eine Nachricht an den OSI TP-Server und beendet die Transaktion, obwohl er das Senderecht zum Transaktionsende nicht besitzt, da er auf einem anderen Dialog selbst OSI TP-Server ist und sein OSI TP-Client das Senderecht zum Transaktionsende nicht abgegeben hat.
KS12	KCDSEFU	Der OSI TP-Server hat die Transaktion im Widerspruch zur Forderung des OSI TP-Clients mit PEND RE beendet.
KS13	KCDSEFU	Der OSI TP-Client beendet den Vorgang mit PEND FI obwohl er noch eine offene Server-Conversation ohne Functional Unit Commit hat.
KS14	KCDSEFU	Der OSI TP-Server hat den Vorgang im Widerspruch zur Forderung des OSI TP-Clients mit PEND FI beendet.
KS15	KCDSEFU	Der OSI TP-Client beendet den Vorgang mit PEND FC, obwohl er noch eine offene Server-Conversation ohne Functional Unit Commit hat.
KS16	KCDSEFU	Der OSI TP-Server sendet eine Nachricht an den OSI TP-Client, obwohl er das Senderecht auf diesem Dialog nicht besitzt.
KS17	KCDSEFU	Der OSI TP-Server beendet den Dialogschritt, sendet aber keine Nachricht an den OSI TP-Client, obwohl er das Senderecht auf diesem Dialog besitzt.
KS18	KCDSEFU	Der OSI TP-Client gibt das Senderecht zum Transaktionsende an mehr als einen OSI TP-Server ab.
KS19	KCDSEFU	Der OSI TP-Client fordert den OSI TP-Server mit CTRL PR oder PE zum Transaktions- oder Vorgangsende auf, sendet eine Nachricht an diesen Partner, leitet dann aber selbst das Transaktionsende ein.
KS20	KCDSEFU	Der lokale Vorgang hat PGWT CM oder PGWT RB aufgerufen, obwohl ein Partner an der verteilten Transaktion beteiligt ist, mit dem über das LU6.1-Protokoll kommuniziert wird.
KS21	KCDSEFU	Der OSI TP-Server hat die Transaktion im Widerspruch zur Forderung des OSI TP-Clients mit PGWT CM beendet.
KS22	KCDSEFU	Der OSI TP-Client beendet den Vorgang mit PEND FI obwohl er eine Server-Conversation mit CTRL PR nur zum Transaktionsende aufgefordert hat.
KS23	KCDSEFU	Der OSI TP-Client beendet den Vorgang mit PEND FC obwohl er eine Server-Conversation mit CTRL PR nur zum Transaktionsende aufgefordert hat.
KS24	KCDSEFU	Der OSI TP-Client fordert eine Server-Conversation, an die noch keine Nachricht gesendet wurde, mit CTRL PR/PE zum Transaktions- bzw. Vorgangsende auf.
KS25	KCDSEFU	Der Session-Wiederanlauf ist fehlgeschlagen.
KT01	KCDOSTM	(Siehe K302) Der taskspezifische Pufferbereich für Wiederanlaufinformation ist zu klein (siehe UTM-Handbuch Anwendungen generieren: Wiederanlaufbereich). Maßnahme: Generierung mit KDCDEF, Pufferbereich mit MAX RECBUF=(...,length) größer definieren. <sup>1</sup>



Code	Modul	Fehlerursache und ggf. Maßnahme
KT02	KCDOSTM	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
KT03	KCDOSTM	(Siehe K306) Die Transaktion musste zurückgesetzt und der Vorgang abnormal beendet werden, weil das DB-System die DB-Transaktion beim Schließen zurückgesetzt hatte.
KT04	KCDOSTM	ein Log-Record passt nicht in den Eingabebuffer. Maßnahme: weniger OSI TP Auftragnehmer in dem Vorgang adressieren. <sup>1</sup>
KT05	KCDOSTM	Die Transaktion muss zurückgesetzt werden, weil die Datenbanktransaktion zurückgesetzt wurde.
KT06	KCDOSTM	Von der Datenbank wurde beim Transaktionsbeenden nach Prepare gemeldet, dass die Transaktion zurückgesetzt wurde.
KT07	KCDOSTM	Von einem OSI TP Server wurde eine heuristische Entscheidung gemeldet (heuristic mixed).
KT08	KCDOSTM	Von einem OSI TP Server wurde eine heuristische Entscheidung gemeldet (heuristic hazard).
KU00	KDCRTDB	Bei den Startparametern fuer Datenbanken wurde ein ungueltiger Prefix angegeben, oder diese Datenbank ist in ROOT nicht generiert (evtl. Anwenderfehler).
KU04	KDCRTDB	Die DB-Transaktion musste zurückgesetzt werden. UTM setzt auch die UTM-Transaktion zurück: Der Teilprogrammmlauf erhält wieder die Kontrolle, wie beim UTM-Aufruf RSET. Im Vorgangs-Exit bei Start des Vorgangs muss UTM den Vorgang mit PEND ER beenden (70Z/KU04). Im Vorgangs-Exit bei Ende des Vorgangs wird ein PENDER-Dump geschrieben.
KU08	KDCRTDB	UTM muss den Vorgang mit PEND ER beenden. Die Meldungen K017 oder K055 enthalten die ersten 4 Bytes der DB-Fehlermeldung.
KU0C	KDCRTDB	Das DB-System (der DBH) ist nicht oder nicht mehr verfügbar. UTM beendet die Anwendung abnormal bzw. bricht den Start der Anwendung ab.
KU10	KDCRTDB	Das DB-System ist als Folge einer DB-Administrator-Aktion nicht mehr verfügbar. UTM beendet die Anwendung abnormal.
KU14	KDCRTDB	Das DB-System ist zur Zeit nicht verfügbar. Nach 5 Sekunden wird ein neuer Versuch zum Verbindungsaufbau gemacht. <sup>1</sup>
KU18	KDCRTDB	Ein eventuell behebbarer Fehler im DB-System liegt vor.
KU1C	KDCRTDB	Ein nicht behebbarer Fehler im DB-System liegt vor. UTM beendet die Anwendung abnormal.

Code	Modul	Fehlerursache und ggf. Maßnahme
KU20	KDCRTDB	Das DB-System hat einen Benutzerfehler entdeckt. Der Fehler tritt beim Start der Anwendung auf, wenn die DB-spezifischen Startparameter geprüft werden. Die DB-Fehlermeldung wird auf SYSOUT ausgegeben.
KU24	KDCRTDB	Aus Sicht des DB-Systems trat ein unerwartetes Verhalten von UTM auf. Die Anwendung wird mit PEND06 beendet.
KU99	KDCRTDB	Bei Multi-DB: Nach einem PETA-Aufruf kann das DB-System den FITA-Aufruf nicht ausführen.
KV01	KCSTUTM	Socketpartner sendet mit USP Version 1.0. Puffer für Dialognachrichten zu klein generiert, bzw. MPUT Länge hat maximal Grösse überschritten. Maßnahme: Generierung ändern, Puffer mit MAX NB=länge größer wählen.
KV02	KCSTUTM	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
KV03	KCSTUTM	(Siehe K348) Der Cluster Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF; Cluster Pagepool mit CLUSTER PGPOOL=(number,...) größer einrichten.
KW01	KCDOSSF	Empfang von TP-HANDSHAKE-IND.
KW02	KCDOSSF	Empfang von TP-BEGIN-DIALOGUE-CNF (negativ).
KW03	KCDOSSF	Empfang von TP-U-ERROR-IND von einem Partner, der nicht das Senderecht hat.
KW04	KCDOSSF	Ablauf eines Timers.
KW05	KCDOSSF	Empfang von TP-U-ABORT-IND oder TP-P-ABORT-IND.
KW06	KCDOSSF	Empfang von TP-ROLLBACK-IND.
KW07	KCDOSSF	Empfang von TP-END-DIALOGUE-IND vom Client wird von UTM nicht unterstützt.
KW08	KCDOSSF	Bei der Functional Unit Unchained Transactions wurde der Dialog mit dem Client nicht mit der 1. Transaktion beendet.
KY00	KCDSYPM	Vom Partner wurde eine negative Quittung erhalten.
KY01	KCDSYPM	Timeout beim Belegen einer Session.
KY02	KCDSYPM	Der Auftragnehmer-Vorgang wurde mit PEND ER beendet.
KY03	KCDSYPM	Der Auftraggeber-Vorgang wurde mit PEND ER beendet.
KY04	KCDSYPM	Mismatch mit dem Partner, und der lokale Vorgang ist im PTC-Zustand.
KY05	KCDSYPM	Mismatch mit dem Partner.
KY06	KCDSYPM	Mismatch mit der Datenbank.
KY07	KCDSYPM	Verbindungsverlust zum Partner.

Code	Modul	Fehlerursache und ggf. Maßnahme
NOTA	KCDRTDB	Transaktion ist beim XA-Resource Manager unbekannt. (Kann z.B. nach Timeouts mit der ORACLE-Datenbank auftreten).
UPCP	KCSUPIC	(Siehe K348) Der Cluster Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF; Cluster Pagepool mit CLUSTER PGPOOL=(number,...) größer einrichten.
UPDE	KCSUPIC	Es wurde Vorgangsabbruch vom UPIC-Partner angefordert.
UPPG	KCSUPIC	(Siehe K345) Der Pagepool in KDCFILE ist voll. Maßnahme: Generierung mit KDCDEF, Pagepool mit MAX PGPOOL=(number,...) größer einrichten. <sup>1</sup>
UPRS	KCSUPIC	Der UPIC Partner hat einen Vorgangswiederanlauf unterdrückt. Dieser Fall tritt dann auf, wenn ein Vorgangswiederanlauf möglich gewesen wäre, aber vom UPIC-Partner nicht angefordert wird. Dann beendet UTM den Vorgang abnormal.
UPXC	KCSUPIC	In einem Teilprogramm mit X/OPEN-API soll die Nachricht an den UPIC-Auftraggeber abgeschlossen werden, obwohl schon eine Nachricht an einen Auftragnehmervorgang abgeschlossen wurde.
UPXM	KCSUPIC	In einem Teilprogramm mit X/OPEN-API soll die Nachricht an den UPIC-Auftraggeber abgeschlossen werden, obwohl schon eine Nachricht an den UPIC-Auftraggeber abgeschlossen wurde.
XTnn	KDCROOT	Es wurde die STXIT-Routine in KDCROOT aufgerufen, nn = Unterbrechungsgewicht des STXIT-Makros. Bei XT80 siehe Fußnote <sup>1</sup>

<sup>1</sup> Bei TESTMODE=OFF werden diese PENDER-Dumps unterdrückt.  
Gleichzeitig unterbleibt das Nachladen, es sei denn für den aktuellen TAC sind PGWT-Aufrufe erlaubt und der KDCS-Aufruf, bei dem der Fehler auftrat, war kein PEND-Aufruf.

## 6.3 STXIT-Ereignisse

Folgende Tabelle zeigt die Zuordnung des Ereigniscodes zu den möglichen STXIT-Ereignissen. Sie ist dem Handbuch „Makroaufrufe an den Ablaufteil“ entnommen.

STXIT-Ereignis-klasse	STXIT-Operand	Unterbrechungsergebnis	Ereigniscode in R3	Einreihung in Warteschlange	max. Schachteltiefe
Programmfehler	PROCHK	unzulässiger SVC	X' 04'	LIFO	127
		unzulässiger Operationscode	X' 58'		
		Datenfehler	X' 60'		
		Exponentenüberlauf	X' 64'		
		Divisionsfehler	X' 68'		
		Mantisse = 0	X' 6C'		
		Exponentenunterlauf	X' 70'		
		Dezimalüberlauf	X' 74'		
		Festpunktüberlauf	X' 78'		
Intervallzeitgeber CPU-Zeit	TIMER	„SETIC-Intervall“ abgelaufen für CPU-Zeit	X' 20'	FIFO	127
Intervallzeitgeber Realzeit	RTIME	„SETIC-Intervall“ abgelaufen für Realzeit	X' A0'	FIFO	127
Ende Programmlaufzeit	RUNOUT	CPU-Zeitgrenze für die Task bzw. das Programm überschritten	X' 80'	FIFO	0
nicht behebbare Programmfehler	ERROR	privilegierter SVC	X' 08'	LIFO	127
		Zugriff auf eine nicht vorhandene Speicherseite	X' 48'		
		privilegierte Operation	X' 54'		
		Adressenfehler	X' 5C'		
		XA-Fehler (falscher Adressierungsmodus)	X' 9C'		
		Realtimer (Condition Error)	X' A4'		
		Ausrichtungsfehler	X' AC'		
		Validierungsfehler	X' B0'		
		Ungültige UNIT-Nummer im Standardheader	X' C4'		

STXIT-Ereignis-klasse	STXIT-Operand	Unterbrechungsergebnis	Ereigniscode in R3	Einreihung in Warteschlange	max. Schachteltiefe
Mitteilung an das Programm	INTR	INFORM-PROGRAM-Kommando	X' 44'	LIFO	127
ESCPBRK	ESCPBRK	BREAK/ESCAPE (über Tasten)	X' 84'	LIFO	127
Programmbeendigung durch asynchrone Ereignisse	ABEND	Systemfehler, Leitungsverlust	X' 88'	LIFO	0
		START-EXECUTABLE-PROGRAM, LOAD-EXECUTABLE-PROGRAM, ABEND, EXIT-JOB, CANCEL-JOB	X' 8C'		
		Adress-Übersetzungsfehler wegen Hardwarefehler	X' 94'		
		Hardwarefehler (CPU)	X' A8'		
		Erzwungenes Entladen eines Subsystems (Systemverw.)	X' B8'		
		nicht behebbarer DMS-Fehler	X'BC'		
		Programmbeendigung durch synchrone Ereignisse	TERM		
		Programmbeendigung durch CMD- / LOGOFF-Makro	X' 98'		
SVC-Unterbrechung	SVC	Aufruf eines angegebenen SVCs	X' 50'	LIFO	127
Hardwarefehler	HWERROR	Ein-/Ausgabefehler bei Data-In-Virtual-Technik	X' 28'	LIFO	0

## 6.4 Codes der CDUMP-Aufrufe von openUTM

In einigen Fehlerfällen, wie z.B. bei fehlerhaften Returncodes der Sprachumgebungen (Language-Connection-Module), von FHS, vom DB-Connection-Modul, oder Fehlern in openUTM, wird ein CDUMP erzeugt. Voraussetzung dafür ist, dass der Anwender die Erzeugung von Dumps nicht per /MODIFY-JOB-OPTIONS ausgeschaltet hat. Auf SYSOUT wird eine Meldung ausgegeben, die ein Insert = 'CODE' besitzt. Anhand dieses Inserts kann die Fehlerursache, die zu diesem CDUMP geführt hat, näher bestimmt werden.

CDUMPs werden unter der Benutzerkennung abgelegt, unter der die UTM-Anwendung gestartet wurde.

Im aufbereiteten Dump, den man mit Hilfe eines Programms zur Dump-Aufbereitung erhält, findet man diesen Returncode zusammen mit der openUTM-Version und der ROOTDATA-Adresse.

Wurde die UTM-Anwendung mit TESTMODE=OFF gestartet, wird der CDUMP mit dem Reason KDCCK02 in Abhängigkeit von dem Returncode KCR CDC unterdrückt (siehe [Seite 371](#)).

Tritt der Dump nach einem STXIT-Ereignis auf, so wird auch die Adresse ausgegeben, an der sich die CONTXT-Area befindet.

Die CONTXT-Area hat folgenden Aufbau:

Inhalte der Register R0 bis R15

program counter	1 byte
instruction length count	1 byte
condition code	1 byte
program mask	1 byte
address mode	1 byte
interrupt weight	4 bytes

**KDCCK..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCCK01	Der SVC UTM lieferte einen fehlerhaften Returncode.	Beendigung der Task mit TERM UNIT=STEP.
KDCCK02 <sup>1</sup>	Nach Rückkehr aus dem UTM-Systemcode erkennt das UTM-Anschlussprogramm ROOT, dass KCRCCC $\geq$ 70Z.	Die betreffende Task läuft auf PEND ER. Bei einer Batch-Task wird das Programm danach entladen, bei einer Dialog-Task bleibt es geladen und läuft weiter.
KDCCK03	Beim Start des Teilprogramms ' INPUT-EXIT' lieferte das Language-Connection-Modul einen fehlerhaften Returncode.	Die betreffende Task läuft auf PEND ER. Bei einer Batch-Task wird das Programm danach entladen, bei einer Dialog-Task bleibt es geladen und läuft weiter.
KDCCK04	Es wurde ein CALL „KDCS“ aus dem START-, SHUT- oder INPUT-Exit aufgerufen.	Die betroffene Task wird mit TERM UNIT=STEP beendet.
KDCCK05	UTM entdeckt Fehler im INPUT-Exit: Ausgabeparameter sind nicht konsistent.	
KDCCK07	Ungültiger Aufruf vom Teilprogramm an das Formatsystem.	Beendigung der Task mit TERM UNIT=STEP
KDCCK08	Mögliche Ursachen: – Start-Fehler – Datenbank wird aufgerufen, ist aber nicht angeschlossen; – Formatierungssystem wird aufgerufen, ist aber nicht generiert.	Beendigung der Task mit TERM UNIT=STEP
KDCCK09	Dieser CDUMP erfolgt nach einem PEND ER (sofern nicht schon zuvor ein CDUMP gezogen wurde)	Eine Batch-Task wird mit TERM UNIT=PROG beendet. Eine Dialog-Task läuft weiter.
KDCCK10	Beim Start eines Teilprogramms liefert das Language-Connection-Modul einen fehlerhaften Returncode	Die betreffende Task läuft auf PEND ER. Bei einer Batch-Task wird das Programm danach entladen, bei einer Dialog-Task bleibt es geladen und läuft weiter.
KDCCK11	Bei der Ende-Behandlung eines Teilprogramms liefert das Language-Connection-Modul einen fehlerhaften Returncode.	Beendigung der Task mit TERM UNIT=STEP

CODE	Fehlerursache	Verhalten von openUTM
KDCCK13	Dieser CDUMP erfolgt, wenn im START-EXIT das entsprechende Language-Connection-Modul einen fehlerhaften Returncode liefert.	Beendigung der Task mit TERM UNIT=STEP
KDCCK14	Beim Austausch eines einzelnen Lademoduls ist ein ungültiger Lademodulindex angegeben.	Beendigung der Task mit TERM UNIT=STEP
KDCCK16	Fehler bei der ILCS-Nachinitialisierung; tritt auf nach dem Laden von STARTUP oder ONCALL Programmen oder nach Programmaustausch.	Beendigung der Task mit TERM UNIT=STEP
KDCCK17	Fehler im Language-Connection-Modul beim Starten des Logging Exits	Die Task läuft weiter.
KDCCK99	Interner Fehler in UTM-Root: Struktur-Fehler in KDCRTMM	Beendigung der Task mit TERM UNIT=STEP

<sup>1</sup> Bei TESTMODE=OFF wird der CDUMP in Abhängigkeit vom Returncode KCRRCDC unterdrückt und das Nachladen unterbleibt.

### KDCRT..

CODE	Fehlerursache	Verhalten von openUTM
KDCRT04	Interner Fehler in UTM-Root: Falscher ACT_INDEX in KDCRTMM	Beendigung der Task mit TERM UNIT=STEP



**KDCLD..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCLD01	interner Fehler in UTM-Root. Ungültiger Lade-Modus eines Lademoduls beim Start der Anwendung	Die betroffene Task wird nach dem Dump mit TERM UNIT=STEP beendet.
KDCLD02	interner Fehler in UTM-Root. Ungültiger Opcode beim Aufruf von KDCRTLD.	Die betroffene Task wird nach dem Dump mit TERM UNIT=STEP beendet.
KDCLD03	interner Fehler in UTM-Root. Ungültiger Lade-Status eines Lademoduls.	Beendigung der Task mit TERM UNIT=STEP
KDCLD04	die Ende-Markierung der Linkage-Section (AREA's) ist überschrieben (evtl. inkonsistente DEF - ROOT).	Beendigung der Task mit TERM UNIT=STEP
KDCLD05	das Lademodul, in dem AREA's geladen sein sollen, ist nicht geladen. Siehe Meldung K078.	Beendigung der Task mit TERM UNIT=STEP
KDCLD99	fehlerhafter Returncode bei BIND / ASHARE-Aufruf und DEQAR (bei Term). Siehe Meldung K078.	Beendigung der Task mit TERM UNIT=STEP

**KDCAF.. , KDCST..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCAF01	Fehler bei der Stackverwaltung mi ILCS in openUTM. Siehe Meldung K078.	Beendigung der Task mit TERM UNIT=STEP
KDCST01	Fehler beim Initialisieren von ILCS oder SPL , oder Ablauf der Wartezeit zum Anschluss an einen Memory Pool. Siehe Meldung K078.	Beendigung der Task mit TERM UNIT=STEP

**KDCDI..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCDI01	Bei der STXIT-Behandlung liefert der STXIT-Ausgang des Language-Connection-Moduls einen fehlerhaften Returncode.	Die betroffene Task läuft auf PEND ER und danach wird die Task mit TERM UNIT=STEP beendet.
KDCDI02	Bei einem SHUT-EXIT-Aufruf liefert das Language-Connection-Modul einen Fehler.	Die Task läuft weiter.
KDCDI03	Bei der ENDE-Behandlung in der STXIT-Routine liefert das Language-Connection-Modul einen fehlerhaften Returncode.	Aufruf des SHUTEXIT, Deaktivierung der Language-Connection-Module und dann Beendigung der Task mit TERM UNIT=STEP.
KDCDI04	Fehler bei der Initialisierung der Language-Connection-Module.	Beendigung der Task mit TERM UNIT=STEP
KDCDI05	Fehler bei der Deaktivierung der Language-Connection-Module.	Die Task läuft weiter.
KDCDI07	Ein STXIT-Ereignis ist beim Start der Anwendung aufgetreten. (KTA-Adresse fehlerhaft)	Beendigung der Task mit TERM UNIT=STEP
KDCDI09	Ein STXIT-Ereignis ist beim Start der Anwendung aufgetreten. (DB-INIT nicht gelaufen)	Beendigung der Task mit TERM UNIT=STEP

**KDCDB..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCDB01	Fehler des DB-Connection-Modul (=DBCON) bei Übergabe der Startparameter.	Beendigung der Task mit TERM UNIT=STEP
KDCDB02	ADMINISTRATION-ERROR oder DBMS-DOWN beim Konnektieren der Datenbank.	Beendigung der Task mit TERM UNIT=STEP
KDCDB03	RECOVERABLE- oder UNRECOVERABLE-ERROR beim Diskonnektieren der Datenbank.	Die Task läuft weiter
KDCDB04	DBMS-DOWN, aber KTA noch vorhanden nach DB-Transaktion 'DELETE DB STATUS'.	Anwendungsabbruch mit PEND03
KDCDB05	DB-Returncode = TA-CHAIN-RSET.	Die Task läuft mit PENDING weiter
KDCDB06	DB-Returncode = DBMS-DOWN.	Anwendungsabbruch mit PEND03
KDCDB07	Der gelieferte DB-Returncode ist bei der DB-Funktion nicht erlaubt (siehe Schnittstelle IUTMDB).	Beendigung der Task mit TERM UNIT=STEP bzw. Anwendungsabbruch mit PEND04 Bei OPCODE=STAT (siehe K071-Meldung). Startabbruch mit Fehlercode 74 oder 82
KDCDB08	UNRECOVERABLE ERROR bei DB-Call	Anwendungsabbruch mit PEND05
KDCDB09	USER CALL in Anmelde-Vorgang, obwohl die UTM-Generierung das nicht erlaubt.	Die Task läuft nach PENDING weiter
KDCDB10	USER CALL im INPUT-Exit, obwohl nicht erlaubt.	Beendigung der Task mit TERM UNIT=STEP
KDCDB11	Unerlaubter USER CALL im START- oder SHUT-Exit.	Die Task läuft weiter
KDCDB12	Während der Beendigung der Task wurde vom DB-System ein Fehler gemeldet (Diagnosedump).	Die Task setzt die Beendigung fort
KDCDB99	Fehler bei der Kommunikation mit dem DB-System (bei Multi-DB): Das DB-System kann nach einem PETA-Aufruf (= vorläufiges Transaktionsende) den FITA-Aufruf nicht ausführen.	Die Task läuft weiter nach PENDING

**KDCFH..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCFH01	Der gelieferte FORM-Returncode ist bei der FORM-Funktion nicht erlaubt	Beendigung mit TERM UNIT=STEP
KDCFH02	Die Initialisierung des Formatierungssystems konnte nicht ordnungsgemäß durchgeführt werden.	Beendigung mit TERM UNIT=STEP
KDCFH03	negativer Returncode vom Formatierungssystem, nicht FORM_ADMINISTRATION_ERR oder FORM_ERR	Die Task läuft weiter.
KDCFH04	Beim Start des Teilprogramms 'FORMAT-EXIT' lieferte das Language-Connection-Modul einen fehlerhaften Returncode.	Die betroffene Task läuft auf PEND ER. Bei einer Batch-Task wird das Programm danach entladen, bei einer Dialog-Task bleibt es geladen und läuft weiter.
KDCFHxx	Fehler beim Initialisieren des Formatierungssystems. xx = vom Formatierungssystem zurückgegebener Returncode	Beendigung mit TERM UNIT=STEP

**KDCMOVE, KCSMOVE**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCMOVE KCSMOVE	Dieser CDUMP entspricht den TRMA-bzw.. KCSMOVE Gründen MOVExx. Er erfolgt, wenn ein TRMA-AUFRUF nicht möglich ist. (z.B. im TU-Teil der Administration)	Beendigung der Task mit TERM UNIT=STEP

**KDCIL60**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Verhalten von openUTM</b>
KDCIL32	Fehler in der Zusammenarbeit mit ILCS	Die Task läuft weiter.
KDCIL33	Fehler in der Zusammenarbeit mit ILCS	Die Task läuft weiter.
KDCIL35	Fehler in der Zusammenarbeit mit ILCS	Die Task läuft weiter.
KDCIL36	Fehler in der Zusammenarbeit mit ILCS	Die Task läuft weiter.
KDCIL60	Dieser CDUMP wird in der ILCS-Fehlerbehandlungsroutine erstellt. Mögliche Ursache: Speichermangel	Beendigung der Task mit UNIT=PROGRAM

**KDCIW.., KDCSTXT, KDCSTRT**

CODE	Fehlerursache	Verhalten von openUTM
KDCIW..	Dieser CDUMP wird von den Language-Connection-Modulen für ASSEMBLER, COBOL bzw. SPL sowie vom Modul KDCRTDI (bei RUNOUT, NORMAL PROGR. TERMINATION UND ABEND) beim Auftreten eines STXIT erzeugt. Insert '..' = Interrupt-Weight.	Batchtask wird mit PEND ER beendet, Dialogtask wird mit TERM UNIT=STEP beendet.
KDCIW88	mögliche Ursache : Die Task wurde vom UTM-Systemcoding aus beendet. Der PC steht dann auf dem Befehl nach dem SVC 134.	
KDCSTRT	Ein STXIT-Ereignis vor oder während der Initialisierung von ILCS. Mögliche Ursache: Installationsfehler.	Beendigung der Task mit TERM UNIT=STEP
KDCSTXT	Ein STXIT-Ereignis ist in Schleife (Anzahl > 3) aufgetreten.	Beendigung der Task mit TERM UNIT=STEP

**KCSDTTM**

CODE	Fehlerursache	Verhalten von openUTM
KCSDTTM	Dieser CDUMP erfolgt, falls nach dem STCK-Befehl zum Holen der System-Zeit der Condition-Code ungleich 0 ist. In diesem Fall ist die gewonnene Uhrzeit für openUTM zur Weiterverarbeitung nicht geeignet. (Tritt nur bei Hardwarefehler auf)	Beendigung der Task mit TERM UNIT=STEP

**KCSEN..**

CODE	Fehlerursache	Maßnahme
KCSEN01 KCSEN02 KCSEN03 KCSEN04 KCSEN05	UTM-interner Fehler	Systemfehler: PM schreiben

**KCSST..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Maßnahme</b>
KCSST01	UTM-interner Systemfehler	Systemfehler: PM schreiben

**KCSWT..**

<b>CODE</b>	<b>Fehlerursache</b>	<b>Maßnahme</b>
KCSWT02	Ungültiges Timer-Announcement	Systemfehler: PM schreiben
KCSWT03	Ungültiges Börsen-Announcement	Systemfehler: PM schreiben

## 6.5 Fehlercodes im UTM-Dump

Die Spalte Grp. in den folgenden Tabellen beschreibt, welcher Ursachengruppe der Dumpcode-Fehlercode angehört. Es gibt folgende Gruppen:

- A Ursache ist ein Anwenderfehler, z.B. ein Fehler beim
  - Generieren und administrieren von UTM-Anwendungen
  - Generieren des Systems (z.B. Aufteilung des Adressraums)
  - Programmieren eines Format-Exits
- U Ursache ist ein Fehler im UTM-Code.
- S Ursache ist ein Fehler in einer anderen Systemkomponente (Software oder Hardware).
- F Es handelt sich um einen Folgedump, ein anderer Task hat den Abbruch der Anwendung veranlasst.
- D Der UTM-Dump wurde für Diagnosezwecke erzeugt. Der UTM-Work-Task läuft weiter.
- M Ursache ist ein Speicherengpass.
- X Ursache ist ein Fehler im XAP-TP-Code (OSI TP).

Mehrfachnennungen sind möglich, z.B. ADS.

Bei allen Fehlern der Gruppe U, X und S und bei allen **nicht** in der folgenden Tabelle aufgelisteten Fehlercodes sollten Sie zur Klärung eine Problemmeldung schreiben. Für die Diagnose sind eine Reihe von Unterlagen nötig. Ausführliche Informationen mit einer Liste der benötigten Unterlagen finden Sie in [Abschnitt „Fehlerunterlagen erstellen“ auf Seite 47](#).

Code	Grp.	Beschreibung
ADMI98	D	Die lokale Data Area ist zu klein.
AHQA00	MX	XAP-TP Baustein. Modul KCOCOTA, Funktion QueueAnno(). Der Makro mGetBufferSpace() lieferte den Returncode LB_NOMEM.
AHSA02	SUX	XAP-TP Baustein. Modul KCOCOTA, Funktion SendAnno(). Der Aufruf des System-Interfaces KCOBRSE lieferte einen schlechten Returncode.
AINF01	S	Nach dem AINF-Aufruf ist der Returncode ungleich X'00'.

Code	Grp.	Beschreibung
ALME01	ASU	Unzulässiger System-Returncode beim Aufruf von KCSALME. Tritt dieser Fehler bei einem Dienstprogramm auf, dann bedeutet er i.a., dass der virtuelle Adressraum für die zu bearbeitende Aufgabe nicht ausreicht. Maßnahme: Erhöhen Sie den Adressraum für die Kennung. Für KDCDUMP ist eine Abschätzung des benötigten Adressraums im Kapitel "Meldungen des Programms KDCDUMP" beschrieben.
ALME02	ASU	UTM will einen Memory Pool für die KAA-Tabellen einrichten, der Memory Pool ist aber bereits vorhanden. Eventuell hängt noch ein Task der 'alten' UTM-Anwendung in der Programm-/Taskbeendigung. Maßnahme: Kontrollieren, ob alle Tasks der 'alten' Anwendung nicht mehr im System sind. Ein Task kann z.B. noch im System sein, weil eine Meldung auf der Konsole nicht beantwortet ist. Anschließend nochmals starten.
ALME03	AMS	Fehler beim Erzeugen oder Konnektieren eines Data Spaces. Nähere Informationen gibt der BS2000-Returncode aus der Diagnosemeldung im SYSOUT- oder SYSLST-Listing.
AREC01	SU	Fehler beim Aufruf \$AREC zum Schreiben eines Satzes durch die BS2000 Accounting Routine
ASA006	SUX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ATTACH" aufgerufen. Schlechter Returncode von "bBuildPAddr".
ASA007	SUX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ATTACH" aufgerufen. Erster Task der Anwendung, aber OSS gibt den Returncode "NOTFIRST".
ASA009	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ATTACH" aufgerufen. OSS liefert beim "attach" Aufruf den Returncode "INVEREF".
ASA010	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ATTACH" aufgerufen. OSS liefert beim "attach" Aufruf einen schlechten Returncode.
ASA011	FSX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_DETACH" aufgerufen. OSS liefert beim "detach" Aufruf den Returncode "ERROR". Die Task ist schon in der Terminierungsphase, nachdem eine andere Task hie Anwendung abnormal beendet hat. OSS liefert einen schlechten Returncode, weil UTM eigentlich noch ein Event bei OSS abholen müsste.
ASA012	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_DETACH" aufgerufen. OSS liefert beim "detach" Aufruf den Returncode "INVAREF".
ASA013	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_DETACH" aufgerufen. OSS liefert beim "detach" Aufruf einen schlechten Returncode.



Code	Grp.	Beschreibung
ASA033	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. Der gelieferte Application Context Name ist zu lang.
ASA034	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim "assin" Aufruf den Returncode "ERROR".
ASA035	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim "assin" Aufruf den Returncode "INVREF".
ASA036	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim "assin" Aufruf einen schlechten Returncode.
ASA043	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim positiven "assrs" Aufruf den Returncode "ERROR".
ASA044	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. Schlechter Returncode von "PutElement".
ASA045	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim positiven "assrs" Aufruf einen schlechten Returncode.
ASA046	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim negativen "assrs" Aufruf den Returncode "ERROR".
ASA048	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. OSS liefert beim negativen "assrs" Aufruf einen schlechten Returncode.
ASA049	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_IND" aufgerufen. Schlechter Returncode von "PutElement".
ASA051	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. Der gelieferte Application Context Name ist zu lang.
ASA052	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert beim "asscf" Aufruf den Returncode "ERROR".
ASA053	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert beim "asscf" Aufruf den Returncode "INVREF".

Code	Grp.	Beschreibung
ASA054	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert beim "asscf" Aufruf einen schlechten Returncode.
ASA060	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert beim "aborq" Aufruf den Returncode "ERROR".
ASA062	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert beim "aborq" Aufruf einen schlechten Returncode.
ASA064	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert einen ungültigen diagnostic Wert bei einer negativen association confirmation.
ASA065	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. OSS liefert einen ungültigen "result source" Wert bei einer negativen association confirmation.
ASA066	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF_TIMEOUT" aufgerufen. OSS liefert beim "aborq" Aufruf den Returncode "ERROR".
ASA068	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF_TIMEOUT" aufgerufen. OSS liefert beim "aborq" Aufruf einen schlechten Returncode.
ASA071	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_IND" aufgerufen. OSS liefert beim "aboin" Aufruf den Returncode "ERROR".
ASA072	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_IND" aufgerufen. OSS liefert beim "aboin" Aufruf den Returncode "INVREF".
ASA073	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_IND" aufgerufen. OSS liefert beim "aboin" Aufruf einen schlechten Returncode.
ASA080	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_P_ABORT_IND" aufgerufen. OSS liefert beim "paboin" Aufruf den Returncode "ERROR".

Code	Grp.	Beschreibung
ASA081	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_P_ABORT_IND" aufgerufen. OSS liefert beim "paboin" Aufruf den Returncode "INVREF".
ASA082	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_P_ABORT_IND" aufgerufen. OSS liefert beim "paboin" Aufruf einen schlechten Returncode.
ASA083	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. Schlechter Returncode von "PutElement".
ASA084	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF" aufgerufen. Schlechter Returncode von "PutElement".
ASA085	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ASS_CNF_TIMEOUT" aufgerufen. Schlechter Returncode von "PutElement".
ASA088	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "relin" Aufruf den Returncode "ERROR".
ASA089	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "relin" Aufruf den Returncode "INVREF".
ASA090	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "relin" Aufruf einen schlechten Returncode.
ASA091	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert für "release reason" einen ungültigen Wert.
ASA092	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "aborq" Aufruf den Returncode "ERROR".
ASA094	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "aborq" Aufruf einen schlechten Returncode.

Code	Grp.	Beschreibung
ASA095	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "relrs" Aufruf den Returncode "ERROR".
ASA097	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. OSS liefert beim "relrs" Aufruf einen schlechten Returncode.
ASA099	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_REQ" aufgerufen. OSS liefert für "abort diagnostic" einen ungültigen Wert.
ASA100	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_REQ" aufgerufen. OSS liefert beim "aborq" Aufruf den Returncode "ERROR".
ASA101	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_IND" aufgerufen. Schlechter Returncode von "PutElement".
ASA102	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_REQ" aufgerufen. OSS liefert beim "aborq" Aufruf einen schlechten Returncode.
ASA104	AX	XAP-TP Baustein. Die Funktion "bBuildPAddr" des Moduls "KCOASAM" wurde aufgerufen. Der Presentation Selector eines lokalen Accesspoints oder eines entfernten Partners ist zu lang.
ASA105	AX	XAP-TP Baustein. Die Funktion "bBuildPAddr" des Moduls "KCOASAM" wurde aufgerufen. Der Session Selector eines lokalen Accesspoints oder eines entfernten Partners ist zu lang.
ASA116	SX	XAP-TP Baustein. Die Funktion "BuildRemoteAet" des Moduls "KCOASAM" wurde aufgerufen. Der APT liegt in "form2" vor, der AEQ nicht.
ASA120	SX	XAP-TP Baustein. Die Funktion "BuildRemoteAet" des Moduls "KCOASAM" wurde aufgerufen. Der APT liegt in "form1" vor, der AEQ nicht.
ASA122	SX	XAP-TP Baustein. Die Funktion "BuildRemoteAet" des Moduls "KCOASAM" wurde aufgerufen. Der APT liegt weder in "form1" noch in "form2" vor.
ASA128	M	XAP-TP Baustein. Die Funktion "CopyDefinedContext" des Moduls "KCOASAM" wurde aufgerufen. Schlechter Returncode von "PutElement"

Code	Grp.	Beschreibung
ASA137	M	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_GO_IND" aufgerufen. Die Association ist nicht gesperrt.
ASA139	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_P_ABORT_IND" aufgerufen. Schlechter Returncode von "PutElement".
ASA151	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. Schlechter Returncode von "PutElement".
ASA152	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_REQ" aufgerufen. Schlechter Returncode von "PutElement".
ASA153	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_REQ" aufgerufen. Schlechter Returncode von "SacfSeparator".
ASA155	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_ABORT_IND" aufgerufen. Schlechter Returncode von "SacfSeparator".
ASA156	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_P_ABORT_IND" aufgerufen. Schlechter Returncode von "SacfSeparator".
ASA157	SX	XAP-TP Baustein. Der Modul "KCOASAM" wurde mit dem Operationscode "ASAM_RELEASE_IND" aufgerufen. Schlechter Returncode von "SacfSeparator".
ASIO01	AU	Datei ist im FCB nicht als geöffnet gekennzeichnet beim Aufruf von KCSASIO. Ursache kann auch ein CLOSE ALL im Teilprogrammmlauf sein.
ASIS99	D	Normale Ausführung des Administrationskommandos "KDCSHUT KILL" bzw. des Aufrufs der Programmschnittstelle der Administration mit dem Opcode KC_SHUTDOWN und dem Subopcode1 KC_KILL.
BFMM05	M	UTM-Cluster-Anwendung Es konnte kein weiteres Puffer-Segment angefordert werden. Evtl. ist der virtuelle Adressraum zu klein.
BFMM21	ASU	UTM-Cluster-Anwendung Beim Aufheben einer Dateisperre kam es zu einem Timeout. Maßnahme: Siehe CCFG19
BFMM22	SU	UTM-Cluster-Anwendung Beim Aufheben einer Dateisperre wurde ein schlechter Returncode zurückgegeben.

Code	Grp.	Beschreibung
BRSE01	S	Ungültiger Parameter QUEUE_DISCIPLINE beim Erzeugen einer Börse oder beim Senden oder Empfangen eines Announcement. Ungültiger Parameter LENGTH_OF_ANNO beim Senden eines Announcements. Schlechter Returncode von der Börsenschnittstelle.
BRSE02	S	Fehler beim enqueue der Task an der WORK-Börse (Kammer 1)
BRSE03	S	Fehler beim enqueue der Task an taskspezifischer Börse (Kammer 1)
CACHT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
CACHT2	F	(Siehe CACHT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
CACHT3	F	(Siehe CACHT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
CACHT4	F	(Siehe CACHT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
CACHT5	F	(Siehe CACHT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
CC-...	D	Diagnosedump, der abhängig von einem primären KDCS-Returncode erzeugt wird. Nach dem Präfix CC- folgt der primäre KDCS-Returncode (z.B CC-84Z). Ein- und Ausschalten mit der Funktion Message-Dump.
CCFG07	SU	UTM-Cluster-Anwendung Modul KCCCFG, Entry KCCCGFB Schlechter Returncode von KCSGLHN
CCFG19	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_READ_CLUSTER_FILE Timeout bei Anforderung des Shared Lock. Maßnahme: Siehe Abschnitt "Maßnahme bei Sperren von UTM-Cluster-Dateien" im Anschluss an diese Tabelle.
CCFG20	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_READ_CLUSTER_FILE Schlechter Returncode bei Anforderung des Shared Lock.
CCFG21	ASU	UTM-Cluster-Anwendung Modul KCCCFG Timeout bei Anforderung des Exclusive Lock. Maßnahme: Siehe CCFG19
CCFG22	SU	UTM-Cluster-Anwendung Modul KCCCFG Schlechter Returncode bei Anforderung des Exclusive Lock.

<b>Code</b>	<b>Grp.</b>	<b>Beschreibung</b>
CCFG29	ASU	UTM-Cluster-Anwendung Modul KCCCFG Timeout bei Freigabe des Datei-Lock. Maßnahme: Siehe CCFG19
CCFG30	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_READ_CLUSTER_FILE Schlechter Returncode bei Freigabe des Datei-Lock.
CCFG32	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_REGISTER_COMPLETE Ungültige Version der Cluster-Konfigurationsdatei.
CCFG33	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_REGISTER_COMPLETE Korrumpte Cluster-Konfigurationsdatei.
CCFG34	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_REGISTER_COMPLETE Ungültiger Anwendungsname in der Cluster-Konfigurationsdatei.
CCFG35	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_REGISTER_COMPLETE Die Cluster-Konfigurationsdatei wurde während des Betriebs neu erzeugt.
CCFG41	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_OPEN_CLUSTER_FILE Timeout bei Initialisierung des globalen Lock. Maßnahme: Siehe CCFG19
CCFG42	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_OPEN_CLUSTER_FILE Schlechter Returncode bei Initialisierung des globalen Lock.
CCFG43	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_CLOSE_CLUSTER_FILE Timeout bei Destroy des globalen Lock. Maßnahme: Siehe CCFG19
CCFG44	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_CLOSE_CLUSTER_FILE Schlechter Returncode bei Destroy des globalen Lock.
CCFG45	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_INIT_KAA Schlechter Returncode von KCSGLHN
CCFG49	S	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE Schlechter Returncode beim Lesen der Cluster-Datei.

Code	Grp.	Beschreibung
CCFG50	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE Korrumpte Cluster-Datei.
CCFG51	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE Ungültige Version der Cluster-Konfigurationsdatei.
CCFG52	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE Ungültiger Anwendungsname in der Cluster-Konfigurationsdatei.
CCFG53	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE Die Cluster-Konfigurationsdatei wurde während des Betriebs neu erzeugt.
CCFG55	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode NODE_FAILURE Korrumpte Cluster-Konfigurationsdatei.
CCFG56	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode NODE_FAILURE Ungültige Version der Cluster-Konfigurationsdatei.
CCFG57	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode NODE_FAILURE Ungültiger Anwendungsname in der Cluster-Konfigurationsdatei.
CCFG58	A	UTM-Cluster-Anwendung Modul KCCCFG, Opcode NODE_FAILURE Die Cluster-Konfigurationsdatei wurde während des Betriebs neu erzeugt.
CCFG62	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_RESET_START_SERIALIZATION Timeout bei Freigabe des Lock. Maßnahme: Siehe CCFG19
CCFG63	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_RESET_START_SERIALIZATION Schlechter Returncode bei Freigabe des Lock.
CCFG64	ASU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_RESET_START_SERIALIZATION Timeout bei Destroy des Lock. Maßnahme: Siehe CCFG19
CCFG65	SU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_RESET_START_SERIALIZATION Schlechter Returncode bei Destroy des Lock.



Code	Grp.	Beschreibung
CCFG71	AU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode READ_CLUSTER_FILE oder CHECK_CLUSTER_FILE Der lokale Knoten ist in der Cluster-Konfigurationsdatei als ausgefallen gekennzeichnet
CCFG72	AU	UTM-Cluster-Anwendung Modul KCCCFG, Opcode CCFG_NODE_FAILURE Der lokale Knoten ist in der Cluster-Konfigurationsdatei als ausgefallen gekennzeichnet.
CCKF02	SU	UTM-Cluster-Anwendung Modul KCCCKF Unerwarteter Returncode von KCCFILA beim Öffnen der KDCFILE.
CCKF04	SU	UTM-Cluster-Anwendung Modul KCCCKF Unerwarteter Returncode von KCCFILA beim Schließen der KDCFILE.
CCTR17	AU	UTM-Cluster, Modul KCCCTRL, Operationscode UNREGISTER_NODE Die Knoten-Anwendung ist registriert, aber die Journal-Datei konnte nicht geöffnet werden. Evtl. wurde die Journal-Dateien gelöscht. Dieser Abbruch kann auch nach einem vorhergehenden Start-Fehler auftreten.
CDTN02	M	XAP-TP Baustein. Modul KCOCOHF, Funktion CheckDtnidTnid(). Der Makro mGetBufferSpace() lieferte den Returncode LB_NOMEM.
CFMM05	D	Beim Senden passt die Länge in DCF nicht zu der im Letterheader.
CLREST	D	UTM-Cluster-Anwendung Der Dump wird nur im Warmstart einer Knoten-Anwendung bei eingeschaltetem Testmode erzeugt. Er dient der Diagnose nach eventuellen Fehlern im Cluster Warmstart.
CONS03	A	Die KDCFILE wurde im laufenden Betrieb überschrieben. Mögliche Ursache: KDCDEF-Lauf in laufender Anwendung. Maßnahme: Startprozedur überprüfen
CSND04	MX	XAP-TP Baustein Ungültiger Returncode nach Aufruf von PutElement() zum Anfordern eines dynamischen Puffers für Concatenator Sendedaten.
CSND05	SX	XAP-TP Baustein Ungültiger Returncode nach Aufruf einer OSS Presentation Funktion.
CTPF04	M	XAP-TP Baustein. Der Returncode von PutElement war ungleich DM_OK.
DC....	D	Diagnosedump, der abhängig von einem sekundären KDCS-Returncode erzeugt wird. Nach dem Präfix DC folgt der sekundäre KDCS-Returncode (z.B DCKS17). Ein- und Ausschalten mit der Funktion Message-Dump.

Code	Grp.	Beschreibung
DIAGCL	D	UTM-Cluster-Anwendung Eine Knoten-Anwendung wurde abnormal beendet. Vor der Beendigung wurden, sofern noch möglich, alle anderen Knoten-Anwendungen über das abnormale Ende informiert. Die informierten Knoten-Anwendungen schreiben einen Diagnose-Dump.
DIAGDP	D	Mit dem Administrationskommando "KDCDIAG DUMP=YES" bzw. durch den Aufruf der Programmschnittstelle der Administration mit Opcode KC_CREATE_DUMP wurde ein Diagnosedump erzeugt.
DMCA00	M	XAP-TP Baustein Modul KCOCODM, Funktion ConnectDynMemArea(). Die Funktion ConnectSharedMem() lieferte den Returncode MEM_NOMEM.
EDITER	ADS	Fehler im Zusammenhang mit dem verwendeten Editprofil.
EHHP00	M	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Der Returncode von mGetBufferSize() war ungleich LB_OK.
EHHP01	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_datain() lieferte den Returncode P_ERROR.
EHHP02	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_datain() lieferte den Returncode P_INVREF.
EHHP03	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die oss Funktion p_datain() lieferte einen unbekanntem Returncode.
EHHP04	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_typein() lieferte den Returncode P_ERROR.
EHHP05	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_typein() lieferte den Returncode P_INVREF.
EHHP06	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_typein() lieferte einen unbekanntem Returncode.
EHHP07	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Nach Aufruf der OSS Funktion p_typein() liefert die Funktion CalcUserDataLth() als Länge der User Daten 0.
EHHP08	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synin() liefert den Returncode P_ERROR.

Code	Grp.	Beschreibung
EHHP09	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synin() lieferte den Returncode P_INVREF.
EHHP10	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synin() lieferte einen unbekanntem Returncode.
EHHP12	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HamdlePresEvent(). Nach Aufruf der OSS Funktion p_synin() liefert die Funktion CalcUserDataLth() als Länge der User Daten Null.
EHHP13	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synfc() liefert den Returncode P_ERROR.
EHHP14	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synfc() lieferte den Returncode P_INVREF.
EHHP15	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_synfc() lieferte einen unbekanntem Returncode.
EHHP17	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Nach Aufruf der OSS Funktion p_synfc() lieferte die Funktion CalcUserDataLth() die User Daten Länge Null.
EHHP18	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkgin() lieferte den Returncode P_ERROR.
EHHP19	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkgin() lieferte den Returncode P_INVREF.
EHHP20	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkgin() lieferte einen unbekanntem Returncode.
EHHP21	S	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkgin() lieferte nicht das Token S_T_MINOR.
EHHP22	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkpin() lieferte den Returncode P_ERROR.
EHHP23	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkpin() lieferte den Returncode P_INVREF.

Code	Grp.	Beschreibung
EHHP24	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_tkpin() lieferte einen unbekanntem Returncode.
EHHP25	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die Funktion p_tkpin() lieferte nicht das Token S_T_MINOR.
EHHP26	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Nach Aufruf der OSS Funktion p_tkpin() lieferte die Funktion CalcUserDataLth() als User Daten Länge den Wert Null.
EHHP27	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minin() lieferte den Returncode P_ERROR.
EHHP28	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minin() lieferte den Returncode P_INVREF.
EHHP29	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minin() lieferte einen unbekanntem Returncode.
EHHP30	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Nach Aufruf der OSS Funktion p_minin() lieferte die Funktion CalcUserDataLth() die User Daten Länge Null.
EHHP31	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minfc() lieferte den Returncode P_ERROR.
EHHP32	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minfc() lieferte den Returncode P_INVREF.
EHHP33	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion p_minfc() lieferte einen unbekanntem Returncode.
EHHP34	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Nach Aufruf der OSS Funktion p_minfc() liefert die Funktion CalcUserDataLth() den Wert Null als User Daten Länge.
EHHP35	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die Funktion o_event() lieferte das unerwartete Ereignis P_MAJIN.

Code	Grp.	Beschreibung
EHHP36	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die Funktion o_event() lieferte das unerwartete Ereignis P_MAJCF.
EHHP37	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion HandlePresEvent(). Die OSS Funktion o_event() lieferte einen unbekanntem Ereignistyp.
EHRP01	MX	XAP-TP Baustein. Modul KCOXFEH, Funktion ReloadPresEvent(). Der Makro mGetBufferSize() lieferte einen Returncode ungleich LB_OK.
EHSP00	SX	XAP-TP Baustein. Modul KCOXFEH, Funktion StorePresEvent(). Es soll mittels StorePresEvent() ein Presentation Ereignis im DynMem gespeichert werden. Für die betreffende Association ist jedoch schon ein Event gespeichert (mValLth(g, &pAss->PendingEvt.h) > 0).
EHSP01	M	XAP-TP Baustein. Modul KCOXFEH, Funktion StorePresEvent(). Die Funktion PutElement() lieferte den Returncode DM_NOMEM. Maßnahme: In der KDCDEF-Generierung die Größe der OSI-Scratch-Area erhöhen (Parameter MAX OSI-SCRATCH-AREA).
EKAA11	S	UTM-Cluster-Anwendung Timeout beim Freigeben der Datei-Sperre für die JKAA-Datei. Maßnahme: Siehe CCFG19
EKAA12	S	Schlechter Returncode von KCCGFLK beim Freigeben der Datei-Sperre für die JKAA-Datei.
EKAA13	S	UTM-Cluster-Anwendung Timeout beim Zerstören des Datei-Locks für die JKAA-Datei. Maßnahme: Siehe CCFG19
EKAA14	S	Schlechter Returncode von KCCGFLK beim Zerstören des Datei-Locks für die JKAA-Datei.
ENCERR	DSU	UTM hat eine verschlüsselte Nachricht erwartet, aber eine unverschlüsselte empfangen. Es liegt kein Abbruch der Anwendung vor. Der Dump dient allein der Diagnose.
ENDE01	SU	Der KAA-Pointer im KTA zeigt auf keinen KAA.
ENDE02	AU	Unterbrechung bei gesperrtem KAA. Mögliche Ursache: FORCE-JOB-CANCEL durch Operator.
ENDE03	ASU	Abnormale Task-Beendigung zwischen dem Schreiben eines Confirmatory Record und dem (vorläufigen) Ende der UTM-Transaktion. Mögliche Ursache: Runterfahren der Anlage bei laufender UTM-Anwendung oder RUNOUT-STXIT oder FORCE-JOB-CANCEL in diesem Zeitraum.
ENDE04	SU	Der Cache kann nicht freigegeben werden

Code	Grp.	Beschreibung
ENDE05	SU	Der KAA kann nicht freigegeben werden.
ENDE06	ASU	Fehler beim Öffnen der SYSLOG-Datei. Kann nach abnormaler Beendigung eines UTM-Tasks (z.B. durch CANCEL-JOB/FORCE-JOB-CANCEL) auftreten, wenn der Anwender auf die SYSLOG-Datei zugreift, bevor die Anwendung beendet ist. Dadurch geht maximal 1 PAM-Seite mit SYSLOG-Meldungen verloren. Maßnahme: Startprozedur überprüfen (Zugriff auf SYSLOG).
ENDE07	AS	Es wurde ein Task abnormal beendet (z.B. CANCEL-JOB/FORCE-JOB-CANCEL, TERM-Makro), der zu diesem Zeitpunkt einziger Teilnehmer an einem SLOT-Memory-Pool war. Durch die Beendigung des Tasks geht der Memory-Pool verloren und die Anwendung ist nicht mehr ablauffähig. Maßnahme: abnormale Beendigung von UTM-Tasks vermeiden.
ENDE10	ASU	Eine Task hat sich abnormal beendet und hatte die Startphase schon abgeschlossen (Systemdump, FORCE-JOB-CANCEL etc.) Daraufhin wird auch die Anwendung abnormal beendet.
ENDE11	DSU	Die Anwendung wurde normal beendet, während der Abmeldung der Accesspoints von OSS wurde von OSS ein Returncode ungleich OK geliefert. Wenn der Testmode eingeschaltet ist, wird ein UTM Diagnosedump gezogen.
ENDE12	ASU	Eine Task hat sich beendet (normal oder abnormal). Die verbliebenen Tasks reichen nicht aus, damit die Anwendung weiterarbeiten kann (z.B. weil sich Tasks im PGWT befinden).
ENDE14	F	Eine Task hat sich beendet (normal oder abnormal) und die Task hat noch nicht KCSTRMA aufgerufen. Der Status der Anwendung ist jedoch TERM_APPL. Um zu erreichen, dass bei abnormalem Anwendungsende alle Tasks einen UTM-Dump erzeugen, wurde KCSTRMA mit ENDE14 aufgerufen.
ENDE15	ADU	Eine Task hat sich in TPR abnormal beendet (Systemdump, FORCE-JOB-CANCEL etc.). Da es sich um eine Folge-Task der Anwendung in der Start-Phase handelt, wird nicht die ganze Anwendung abnormal beendet, sondern nur ein Diagnose-Dump geschrieben.
ENDE16	AD	Abnormales Anwendungsende durch abnormale Task-Beendigung in TU (TERM-STXIT, CANCEL-JOB, LOGOFF, etc.). Bei jeder abnormalen Task-Beendigung in TU wird ein Diagnose-Dump mit Ursache ENDE16 geschrieben, wenn der Testmodus eingeschaltet ist.
ENDE19	ASU	Es laufen nur noch UTM-System-Prozesse für die Anwendung. Die normalen Prozesse wurden alle beendet. Ursache ist entweder ein Fehler in der Startprozedur oder ein Startfehler beim Nachstarten der Tasks nach PEND ER oder Programmaustausch.

Code	Grp.	Beschreibung
ENDPET	A	Eine UTM-D-Anwendung kann nicht normal beendet werden, weil es noch Vorgänge mit Transaktionsstatus PTC (prepare to commit) gibt oder weil für asynchrone Nachrichten an andere Anwendungen noch keine Quittungen eingetroffen sind. Es wird kein UTM-Dump erzeugt. UTM führt beim nächsten Start der Anwendung einen Warmstart durch.
EVGE00	M	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Der Returncode des Makros mGetBufferSize() war ungleich LB_OK.
EVGE01	S	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Nach dem Aufruf der OSS Funktion o_event() enthielt der Rückgabeparameter <o_echain> einen ungültigen Wert.
EVGE03	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte als Returncode den Wert O_ACSE und es wurden noch nicht alle User Daten beim Aufruf empfangen.
EVGE05	ASX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_ERROR. Dies kann auch passieren wenn "BCAM" beendet wird ("BCEND").
EVGE06	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_INVEREF.
EVGE07	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_TIMEINT.
EVGE08	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_WAKEINT.
EVGE09	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_SYSTEM.
EVGE10	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_LOOK.
EVGE11	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_TRANSPORT.

Code	Grp.	Beschreibung
EVGE12	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte den Returncode O_SESSION.
EVGE13	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion GetOssEvent(). Die OSS Funktion o_event() lieferte einen unbekanntem Returncode.
EVNT03	ASX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_ERROR. Dies kann auch passieren wenn "BCAM" beendet wird ("BCEND"). Der Fehler tritt ebenfalls auf, wenn eine UTM-Anwendung mit OSI-TP im Dialog mit zwei Tasks gestartet und wieder beendet wurde, und anschließend in einem der beiden Tasks nochmals eine UTM-Anwendung mit demselben ACCESS-POINT im Dialog als Folge-Task gestartet wird.
EVNT04	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_INVEREF.
EVNT05	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_TIMEINT.
EVNT06	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_WAKEINT.
EVNT07	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_SYSTEM.
EVNT08	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_TRANSPORT.
EVNT09	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_SESSION.
EVNT10	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_PRESENTATION.



Code	Grp.	Beschreibung
EVNT11	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte den Returncode O_ACSE.
EVNT12	SX	XAP-TP Baustein. Modul KCOXFEV, Funktion aputm_event(). Die OSS Funktion o_event(O_EVALLOOK) lieferte einen unbekanntem Returncode.
EXPI32	ASU	UTM-Cluster-Anwendung Schlechter Returncode von KCCGFLK beim Sperren der Cluster-GSSB-Datei. Maßnahme: Siehe CCFG19
EXPI34	ASU	UTM-Cluster-Anwendung Schlechter Returncode von KCCGFLK beim Sperren der Cluster-ULS-Datei. Maßnahme: Siehe CCFG19.
EXPI35	ASU	UTM-Cluster-Anwendung Schlechter Returncode von KCCGFLK beim Sperren der Cluster-ULS-Datei. Maßnahme: Siehe CCFG19
EXPI36	ASU	UTM-Cluster-Anwendung Schlechter Returncode von KCCGFLK beim Sperren der GSSB-Datei. Maßnahme: Siehe CCFG19
EXPI51	A	Ein Periodic Write wurde von einem Task aufgerufen, da der Wiederanlaufbereich der Task voll geworden ist. Die Task will eine weitere Seite des Wiederanlaufbereichs schreiben, aber durch den Periodic Write wurden keine Seiten frei. Ursache kann ein zu klein generierter Wiederanlaufbereich sein, so dass die Wiederanlaufinformation einer einzigen Transaktion größer ist als der Wiederanlaufbereich (z.B. durch Transaktion mit Zugriffen auf "sehr viele" Page-Pool Seiten). Maßnahme: in der Generierungsanweisung "MAX RECBUF=(number,...) den Parameter number hochsetzen.
EXPI95	A	Siehe EXPI51, tritt aber in RESET_TA_RTN auf.
FMMM10	A	Eine Eingabenachricht kann nicht abgespeichert werden, da der Pagepool voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) größer machen.
FMSM05	DSU	KCDFMSM wurde mit Operations-code PRSP_BID aufgerufen, obwohl Sessionstate nicht gleich WAIT_OF_BID_RSP.
FMSM12	S	Unzulässiger LSES_WORK_STATE beim Empfang von RTR.
FMTERR	ADS	Die Formatierungsroutine (FHS oder User Exit) meldet an den aufrufenden UTM-Modul (KCSWAIT, KCSISLP) einen Fehler. Es liegt kein Abbruch der Anwendung vor. Der Dump dient der Diagnose des Formatierungsfehlers. Der Dump enthält nur KDCROOT Daten, er wird nur im Testmodus ausgegeben.

Code	Grp.	Beschreibung
FREE01	AX	XAP-TP Baustein. Modul KCOXFFO, Funktion ap_free(). Es sind mehr als APFREE_MAX_TO_REL Speicherbereiche freizugeben.
GFLK04	ASU	Unerwarteter Returncode von KCSDDL Mögliche Ursache: Das Subsystem MSCF wurde beendet.
GFLK22	ASU	Unerwarteter Returncode von KCSDDL Mögliche Ursache: Das Subsystem MSCF wurde beendet.
GFLK31	SU	Schlechter Returncode der Funktion KCSGLHN (Get Local Host Name)
GFLK33	ASU	Routine CheckDLMinfo: Unerwarteter Returncode von KCSDDL Mögliche Ursache: Das Subsystem MSCF wurde beendet.
GFLK42	ASU	Der Distributed Lock Manager lieferte mehrfach den Returncode RETRY LATER zurück. Ursache kann z.B. ein nicht gestartetes Subsystem sein. Siehe auch letzte K190-Meldung.
GFLK43	ASU	(Siehe GFLK42) Der Distributed Lock Manager lieferte mehrfach den Returncode RETRY LATER zurück. Ursache kann z.B. ein nicht gestartetes Subsystem sein. Siehe auch letzte K190-Meldung.
GFLKT1	F	UTM-Cluster-Anwendung Lock Hierarchie nach Term Application verletzt (Folge-Dump).
GFLKT2	F	(Siehe GFLKT1) UTM-Cluster-Anwendung Lock Hierarchie nach Term Application verletzt (Folge-Dump).
GMDT11	SU	UTM-Cluster-Anwendung Modul KCCGMDT, Opcode UPDATE_AND_UNLOCK_GSSB. Unerwarteter Returncode von KCCGFLK (lock GSSB Datei). Maßnahme: Siehe CCFG19
INPERR	AD	Fehler im INPUT-Exit wurde erkannt.
IODxxx	ASU	ASIO Return Code: Ein DMS-Fehler ist aufgetreten und konnte nicht korrigiert werden, Dxxx = DMS-Fehlercode. Hinweis: IOD991 vom Lesemodul des KDCUPD kann daran liegen, dass LPAP- oder OSI-LPAP-Nachrichten der Dead Letter Queue übertragen werden sollen und die alte KDCFILE oder die alten Pagepool-Dateien schreibgeschützt sind. Maßnahme: Schreibschutz fuer alte KDCFILE oder die alten Pagepool-Dateien aufheben und KDCUPD wiederholen.
ISLP00	A	Das Subsystem VTSU ist nicht gestartet.
ISLP01	S	Die maximale SCB-Länge ist größer als von UTM erwartet. Mögliche Ursache: eine neue Version des Subsystem VTSU ist gestartet, die größere SCB-Längen benötigt.

Code	Grp.	Beschreibung
ISLP02	ASU	VTSU-Fehler beim Initialisieren eines dynamischen SCBs. Mögliche Ursache: Subsystem VTSU ist nicht gestartet oder schon wieder deaktiviert.
ISLP03	ASU	VTSU-Fehler beim Analysieren der ersten 8 Zeichen der Nachricht. Mögliche Ursache: VTSU kennt Funktionstaste nicht.
ISLP04	SU	VTSU-Fehler beim Analysieren des UTM-Kommandos.
ISLP05	ASU	VTSU-Fehler beim Aufbereiten einer Dialogeingabenachricht. Mögliche Ursache: wie bei ISLP03.
ISLP06	SU	Der VTSU erkennt eine fehlerhafte Ausgabenachricht.
ISLP09	AU	Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: In KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
ISLP11	AU	(Siehe ISLP09) Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: In KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
ISLP12	AU	(Siehe ISLP09) Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: In KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
ISLP13	AU	Der Puffer für die asynchrone Ausgabenachricht ist zu klein. Maßnahme: In KDCDEF-Generierung MAX TRMSGLEN=<length> vergrößern.
ISLP42	ASU	Fehler beim VTSU-Aufruf zum Abfragen der Länge eines SCB (Kontrollblock für VTSU) beim Start eines UTM-Tasks. Mögliche Ursachen: - Ein Terminal-Typ wird von VTSU nicht unterstützt. - Das Subsystem VTSU ist nicht gestartet.
ISLPT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
ISLPT4	F	(Siehe ISLPT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
JFC011	ASU	UTM-Cluster-Anwendung Timeout bei KCCGFLK-Aufruf in KCCJFCT mit Opcode CLOSE_FILES. Maßnahme: Siehe CCFG19
JFC012	ASU	UTM-Cluster-Anwendung Fehler bei KCCGFLK-Aufruf in KCCJFCT mit Opcode CLOSE_FILES. Maßnahme: Siehe CCFG19
JFC016	SU	UTM-Cluster-Anwendung Fehler bei 1. readControlPage-Aufruf in KCCJFCT mit Opcode CHECK_UNPROC_ENTRIES.
JFC018	SU	UTM-Cluster-Anwendung Fehler bei 2. readControlPage-Aufruf in KCCJFCT mit Opcode CHECK_UNPROC_ENTRIES.

Code	Grp.	Beschreibung
JFC055	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT mit Opcode SET_NO-DE_PROCESSING_STATE.
JFC058	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode SET_NO-DE_PROCESSING_STATE.
JFC067	SU	UTM-Cluster-Anwendung Fehler beim Lesen der Verwaltungsseite in KCCJFCT mit Opcode SET_NO-DE_CR_SEQNR.
JFC071	SU	UTM-Cluster-Anwendung Fehler beim ersten readControlPage-Aufruf in KCCJFCT mit Opcode GET_NO-DE_CR_SEQNR.
JFC072	SU	UTM-Cluster-Anwendung Fehler beim zweiten readControlPage-Aufruf in KCCJFCT mit Opcode GET_NO-DE_CR_SEQNR.
JFC076	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT mit Opcode RESET_NO-DE_CR_SEQNR.
JFC082	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode RESET_DY-NADM_LOCK.
JFC083	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT mit Opcode RESET_DY-NADM_LOCK.
JFC400	ASU	UTM-Cluster-Anwendung Timeout bei KCCGFLK-Aufruf(Lock) in KCCJFCT, interne Funktion setFileLock. Maßnahme: Siehe CCFG19
JFC401	ASU	UTM-Cluster-Anwendung Fehler bei KCCGFLK-Aufruf(Lock) in KCCJFCT, interne Funktion setFileLock. Maßnahme: Siehe CCFG19
JFC402	SU	UTM-Cluster-Anwendung Fehler bei KCCGFLK-Aufruf(Lock) in KCCJFCT, interne Funktion setFileLock.
JFC404	SU	UTM-Cluster-Anwendung Timeout bei KCCGFLK-Aufruf(Unlock) in KCCJFCT, interne Funktion release-FileLock.
JFC405	ASU	UTM-Cluster-Anwendung Fehler bei KCCGFLK-Aufruf(Unlock) in KCCJFCT, interne Funktion release-FileLock. Maßnahme: Siehe CCFG19

Code	Grp.	Beschreibung
JFC420	SU	UTM-Cluster-Anwendung Fehler bei KCCFILA-Aufruf(Write) 1. Journaldatei-Verwaltungsseite in KCCJFCT, interne Funktion initControlPages.
JFC421	SU	UTM-Cluster-Anwendung Fehler bei KCCFILA-Aufruf(Write) in KCCJFCT, interne Funktion initControlPages.
JFC430	SU	UTM-Cluster-Anwendung Fehler beim ersten readControlPage-Aufruf in KCCJFCT, interne Funktion switchFiles.
JFC431	SU	UTM-Cluster-Anwendung Fehler beim zweiten readControlPage-Aufruf in KCCJFCT, interne Funktion switchFiles.
JFC460	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT, interne Funktion checkControlPages.
JFC501	SU	UTM-Cluster-Anwendung Fehler bei KCCFILA-Aufruf in KCCJFCT, interne Funktion readPages.
JFC507	SU	UTM-Cluster-Anwendung Fehler bei KCCFILA-Aufruf in KCCJFCT, interne Funktion writePages.
JFC512	SU	UTM-Cluster-Anwendung Fehler bei KCCFILA-Aufruf in KCCJFCT, interne Funktion writeControlPage.
JFC550	SU	UTM-Cluster-Anwendung Fehler beim Lesen einer Verwaltungsseite in KCCJFCT mit Opcode GET_JOURNAL_FILE_INFO.
JFC561	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT mit Opcode SET_COPY_STATE.
JFC570	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT mit Opcode SET_KAA_INFO.
JFC572	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode SET_KAA_INFO.
JFC580	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT mit Opcode GET_KAA_INFO.
JFC581	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode GET_KAA_INFO.
JFC583	SU	UTM-Cluster-Anwendung Fehler beim 3. readControlPage-Aufruf in KCCJFCT mit Opcode GET_KAA_INFO.

Code	Grp.	Beschreibung
JFC584	SU	UTM-Cluster-Anwendung Fehler beim 4. readControlPage-Aufruf in KCCJFCT mit Opcode GET_KAA_INFO.
JFC590	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT mit Opcode SET_GLOBAL_ADM_LOCK.
JFC591	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode SET_GLOBAL_ADM_LOCK.
JFC594	SU	UTM-Cluster-Anwendung Fehler beim 3. readControlPage-Aufruf in KCCJFCT mit Opcode GET_GLOBAL_ADM_LOCK.
JFC595	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT mit Opcode RESET_GLOBAL_ADM_LOCK.
JFC596	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode RESET_GLOBAL_ADM_LOCK.
JFC598	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT mit Opcode GET_GLOBAL_ADM_LOCK.
JFC599	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT mit Opcode GET_GLOBAL_ADM_LOCK.
JFC611	SU	UTM-Cluster-Anwendung Fehler beim 2. readControlPage-Aufruf in KCCJFCT, Routine checkAll-Processed.
JFC613	SU	UTM-Cluster-Anwendung Fehler beim 1. readControlPage-Aufruf in KCCJFCT, Routine checkAll-Processed.
JFC626	SU	UTM-Cluster-Anwendung Fehler beim readControlPage-Aufruf in KCCJFCT, Routine checkAndRepair.
LATCT1	F	Beim Anfordern der Sperre steht der Status der Applikation auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
LCACT1	F	Beim Anfordern der Sperre steht der Status der Applikation auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
LKAAT1	F	Beim Anfordern der Sperre steht der Status der Applikation auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).

Code	Grp.	Beschreibung
LKLC26	FU	Bei Sperren eines Eintrags wird erkannt, dass der Vorgang bereits in einer Warteschlangenverkettung eingetragen ist. Systemfehler oder Folge-Dump, wenn die Anwendung gerade abnormal beendet wird (durch Fehler oder KDCSHUT KILL). Der Fall Folge-Dump ist daran zu erkennen, dass die Ursache des Abbruchs in Meldung K060 nicht LKLC26 ist, und bereits ein weiterer Dump erzeugt wurde.
LKLC42	AU	Börsen-Wartezeit abgelaufen. Maßnahme: KDCDEF-Generierung ändern, Parameter RESWAIT = (...time2) in MAX-Anweisung erhöhen. Die Ursache könnte auch ein Hardware-Fehler sein, der eine Operateur-Meldung zur Folge hat, die beantwortet werden muss, die aber nicht oder zu spät beantwortet wurde. Man sollte dazu das Konsolprotokoll überprüfen.
LKLC51	AU	Siehe LKLC42.
LKLC64	AU	Siehe LKLC42.
LKLCT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump)
LKLCT2	F	Siehe LKLCT1
LKLCT3	F	Siehe LKLCT1
LKLCT4	F	Siehe LKLCT1
LKMTC0	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump vor Aufruf von KCSBRSE).
LKMTC1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump nach Aufruf von KCSBRSE).
LKMTKP	D	Dieser Diagnosedump wird im Fall von TESTMODE=ON erzeugt, wenn beim unlock das PEND KP Flag im Lockfeld nicht zurückgesetzt ist.
LPCMT1	F	Beim Anfordern der Sperre steht der Status der Applikation auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
LWRT02	SU	Fehler beim Öffnen des Benutzerlogfiles <filebase>.USLA (Kopie A).
LWRT03	SU	Fehler beim Öffnen des Benutzerlogfiles <filebase>.USLB (Kopie B).
LWRT04	ASU	Fehler beim SETL-Aufruf für Benutzerlogfile <filebase>.USLA. Evtl. wurden Schreibmanipulationen an der aktuellen Dateigeneration vorgenommen oder die Base geändert. Zum Umschalten während laufender Anwendung KDCLOG-Kommando benutzen. Maßnahme: Falls der LWRT04 beim nächsten Anwendungsstart wieder auftritt: Die Benutzerprotokolldatei, d.h. die gesamte FGG muss gelöscht und wieder neu erzeugt werden.
LWRT05	ASU	Fehler beim SETL-Aufruf für Benutzerlogfile <filebase>.USLB Grund kann gleich sein wie bei LWRT04.
LWRT06	SU	Fehler beim SETL-Aufruf für Benutzerlogfile <filebase>.USLB beim Rückpositionieren auf Dateianfang nach einem nicht erfolgreich abgeschlossenen Schreibauftrag (z.B. wegen Plattenspeicherengpass).

Code	Grp.	Beschreibung
LWRT07	SU	Fehler beim SETL-Aufruf für Benutzerlogfile <filebase>.USLA beim Rückpositionieren auf Dateianfang nach einem nicht erfolgreich abgeschlossenen Schreibauftrag (z.B. wegen Plattenspeicherengpass). Fehler beim PUT-Aufruf zum Schreiben eines LPUT-Records auf Benutzerlogfile <filebase>.USLA
LWRT08	SU	Fehler beim PUT-Aufruf zum Schreiben eines LPUT-Records auf Benutzerlogfile <filebase>.USLA
LWRT09	SU	Fehler beim PUT-Aufruf zum Schreiben eines LPUT-Records auf Benutzerlogfile <filebase>.USLB
LWRT10	SU	Fehler beim CLOSE-Aufruf für Benutzerlogfile <filebase>.USLB
LWRT11	SU	Fehler beim CLOSE-Aufruf für Benutzerlogfile <filebase>.USLA
LWRT17	A	KCSLWRT fordert über KCSALME einen Puffer zum Schreiben auf das Benutzerlogfile an, der Puffer (Klasse 6 Speicher) kann nicht bereitgestellt werden. Maßnahme: Speicherbedarf und BS2000-Generierung überprüfen.
LWRT19	SU	fehlerhafter RELSE-Aufruf für Kopie A des Benutzerlogfiles.
LWRT20	SU	Siehe LWRT19, jedoch für Kopie B.
MACF02	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MACF03	M	XAP-TP Baustein Der Returncode von SetTimer() war ungleich TI_OK.
MACF04	M	XAP-TP Baustein Der Returncode von GetLogRecord war ungleich MACF_OK.
ME....	D	Diagnosedump, der abhängig von einer bestimmten UTM-Meldung erzeugt wird. Nach dem Präfix ME folgt die Meldungsnummer der UTM-Meldung (z.B. MEK135). Ein- und Ausschalten mit der Funktion Message-Dump.
MFCR04	M	XAP-TP Baustein Der Returncode von GetLogRecord war MACF_NO_MEM.
MFCR07	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR08	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR09	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR10	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR11	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.



Code	Grp.	Beschreibung
MFCR16	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR17	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR18	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR19	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR20	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR21	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFCR24	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFDM03	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFDM04	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFDM05	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFDM06	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM05	AX	XAP-TP Baustein Beim TP_RECOVER_REQ ist kein freier Dialogtabelleneintrag für einen Transaktionszweig vorhanden. Mögliche Ursache: die Anzahl der Associations im vorherigen Anwendungslauf war größer als die Anzahl Associations im aktuellen Anwendungslauf.
MFRM06	AX	(Siehe MFRM05) XAP-TP Baustein Beim TP_RECOVER_REQ ist kein freier Dialogtabelleneintrag für einen Transaktionszweig vorhanden. Mögliche Ursache: die Anzahl der Associations im vorherigen Anwendungslauf war größer als die Anzahl Associations im aktuellen Anwendungslauf.
MFRM07	AX	(Siehe MFRM05) XAP-TP Baustein Beim TP_RECOVER_REQ ist kein freier Dialogtabelleneintrag für einen Transaktionszweig vorhanden. Mögliche Ursache: die Anzahl der Associations im vorherigen Anwendungslauf war größer als die Anzahl Associations im aktuellen Anwendungslauf.

Code	Grp.	Beschreibung
MFRM08	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFRM09	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM10	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM11	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM12	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM13	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM14	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM15	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM16	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM17	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM18	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM19	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM21	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM24	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFRM25	AX	XAP-TP Baustein. Kein freier Tabelleneintrag für einen Log-Damage-Record vorhanden. Maßnahme: Log-Damage-Records mit TP_UPDATE_LOG_DAMAGE_REQ löschen oder nMaxLogDamRec auf einen größeren Wert setzen.
MFT102	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFT103	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT104	M	XAP-TP Baustein Der Returncode von ChangeDescriptor war ungleich DM_OK.

Code	Grp.	Beschreibung
MFT105	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT106	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT107	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT108	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFT109	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT110	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT111	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT113	M	XAP-TP Baustein Der Returncode von ChangeDescriptor war ungleich DM_OK.
MFT114	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT115	M	XAP-TP Baustein Der Returncode von ChangeDescriptor war ungleich DM_OK.
MFT119	M	XAP-TP Baustein Der Returncode von GetLogRecord() war ungleich MACF_OK.
MFT120	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT121	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT122	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT123	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT124	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT126	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT127	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT128	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.

Code	Grp.	Beschreibung
MFT129	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT130	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT131	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT132	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT133	M	XAP-TP Baustein Der Returncode von CopyElement war ungleich DM_OK.
MFT134	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT135	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT138	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT139	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT141	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFT142	M	XAP-TP Baustein Die Funktion CopyElement hat einen Returncode ungleich DM_OK geliefert.
MFT147	M	XAP-TP Baustein Die Funktion PutElement hat einen Returncode ungleich DM_OK geliefert.
MFT151	M	XAP-TP Baustein Die Funktion PutElement hat einen Returncode ungleich DM_OK geliefert.
MFTP03	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFTP04	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.
MFTP05	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFTP06	M	XAP-TP Baustein Der Returncode von PutElement war ungleich DM_OK.
MFTP07	M	XAP-TP Baustein Der Returncode von SetTimer war ungleich TI_OK.
MFTP10	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSize() war ungleich LB_OK.

Code	Grp.	Beschreibung
MFTP11	M	XAP-TP Baustein Der Returncode von RequestBuffer() war ungleich LB_OK.
MFTP12	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP15	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP16	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP17	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP18	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP19	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP20	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MFTP24	M	XAP-TP Baustein Der Returncode vom Makro mGetBufferSpace() war ungleich LB_OK.
MOVE03	AU	Ungültige Überlappung von Ziel und Quellbereich in KCSCMOVE.
MSCFER	DSU	Diagnosedump nach Fehler beim Aufruf von MSCF beim Anmelden oder Abmelden einer Task bei MSCF.
MSG000	DU	KCSCRMS wurde mit einer ungültigen Message-Id aufgerufen. Es wird ein Diagnose-Dump geschrieben und die Meldung "K000 MESSAGE NOT DEFINED" ausgegeben.
MUX001	DS	Falsche Längenangabe im Protokoll-Header. Mögliche Fehlerursache: Partner ist kein MUX-Partner.
MUX003	DSU	Unerwarteter Returncode von KCCLKLC. Eventuell Protokollfehler der Partneranwendung.
MUX004	DS	Falscher Operationscode im Protokoll-Header. Mögliche Fehlerursache: Partner ist kein MUX-Partner.
NDRCVY	D	Der Dump wird nur am Ende einer Knoten-Recovery bei eingeschaltetem Testmode erzeugt. Er dient der Diagnose nach eventuellen Fehlern nach einer Knoten-Recovery.
NMTE00	M	XAP-TP Baustein. Modul KCOCOHF, Funktion NewMemTabEntry(). Die Funktion RequestBuffer() lieferte den Returncode LB_NOMEM.
NMTE02	M	XAP-TP Baustein. Modul KCOCOHF, Funktion NewMemTabEntry(). Der Makro mGetBufferSpace() lieferte den Returncode LB_NOMEM.

Code	Grp.	Beschreibung
NQATT1	F	Der Status der Applikation steht auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
NQCAT1	F	(Siehe NQATT1) Der Status der Applikation steht auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
NQKAT1	F	(Siehe NQATT1) Der Status der Applikation steht auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
NQPCT1	F	(Siehe NQATT1) Der Status der Applikation steht auf TERM APPLICATION. Der Task beendet sich dadurch ebenfalls (= Folge-Dump).
NRDBER	A	Cluster-Anwendung. Die Knoten-Recovery muss wegen Fehler bei der Datenbank Recovery abnormal beendet werden. Es wird kein UTM-Dump erzeugt.
ODIA00	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssDiagInfo(). Die OSS Funktion o_error() lieferte den Returncode O_ERROR.
ODIA01	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssDiagInfo(). Die OSS Funktion o_error() lieferte den Returncode O_INVEREF.
ODIA02	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssDiagInfo(). Die OSS Funktion o_error() lieferte einen ungültigen Returncode.
OREA00	SX	XAP-TP Baustein. Die OSS Funktion o_reason() lieferte den Returncode O_ERROR.
OREA01	SX	XAP-TP Baustein. Die OSS Funktion o_reason() lieferte den Returncode O_INVEREF.
OREA02	SX	XAP-TP Baustein. Die OSS Funktion o_reason() lieferte einen unbekanntenen Returncode.
OSAFT2	F	Returncode APEXT_TERMAPPL von XAP-TP.
OSGO01	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssGo(). Die OSS-Funktion o_go() lieferte den Returncode O_INVCREF.
OSGO02	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssGo(). Die OSS-Funktion o_go() lieferte den Returncode O_ERROR.
OSGO03	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssGo(). Die OSS-Funktion o_go() lieferte einen unerwarteten Returncode.
OSTM01	AS	Schwerwiegender Fehler in Zusammenarbeit mit dem DB-System.

Code	Grp.	Beschreibung
OSTM05	AU	Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: in KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
OSTM06	AS	Schwerwiegender Fehler in Zusammenarbeit mit dem DB-System.
OSTM07	A	Ein Log-Record kann nicht gesichert werden, da der Pagepool voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) vergrößern.
OSTP01	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssStop(). Die OSS-Funktion o_stop() lieferte den Returncode O_INVCREF.
OSTP02	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssStop(). Die OSS-Funktion o_stop() lieferte den Returncode O_ERROR.
OSTP03	SX	XAP-TP Baustein. Modul KCOCOHF, Funktion OssStop(). Die OSS-Funktion o_stop() lieferte einen unerwarteten Returncode.
PCMM05	AU	Bei KCSPCMM mit dem Operationscode Get Pagechain ist der Parameter NR_PAGES ungültig oder größer als Anzahl aller Pagepoolseiten. Maßnahme: In KDCDEF-Generierung MAX PGPOOL=number erhöhen
PCTR00	M	XAP-TP Baustein. Modul KCOCOHF, Funktion PrepareCtrlReq(). Der Makro mGetBufferSize() lieferte den Returncode LB_NOMEM.
PEND02	A	Für "System PEND ER" kann kein TACB mehr geschrieben werden, da der Pagepool auf KDCFILE voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) vergrößern.
PEND03	AS	Es wurde ein schwerwiegender Fehler vom DB-System gemeldet, siehe DIAGAREA im UTM-Dump.
PEND04	S	Das DB-System lieferte einen unerlaubten Returncode, siehe DIAGAREA im UTM-Dump.
PEND05	AS	(Siehe PEND03) Es wurde ein schwerwiegender Fehler vom DB-System gemeldet, siehe DIAGAREA im UTM-Dump.
PEND07	ASU	STXIT zwischen Schreiben eines Confirmatory Record und dem (vorläufigen) Ende der UTM-Transaktion (KCSEXPI-Aufruf END_TA). Mögliche Ursache: Fehler im DB-Connection Modul. Eine Timer-Stxit während dieser Zeit wird ignoriert.
PEND11	S	Inkonsistente XID

Code	Grp.	Beschreibung
PEND26	A	Im Falle eines "System PEND ER" für einen Socket-Client kann keine K017-Meldung mehr für einen evtl. später notwendigen KDCDISP geschrieben werden, da der Pagepool auf KDCFILE voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) vergrößern.
PEND97	A	(Siehe PEND26) Im Falle eines "System PEND ER" für einen Socket-Client kann keine K017-Meldung mehr für einen evtl. später notwendigen KDCDISP geschrieben werden, da der Pagepool auf KDCFILE voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) vergrößern.
PEND98	A	(Siehe PEND02) Für "System PEND ER" kann kein TACB mehr geschrieben werden, da der Pagepool auf KDCFILE voll ist. Maßnahme: Pagepool vergrößern, dazu in KDCDEF-Generierung MAX PGPOOL=(number,...) vergrößern.
PENDER	ADU	Ursache des Dump ist nicht ein Abbruch der Anwendung, sondern ein PEND ER des Benutzers, oder ein UTM-interner PEND ER nach KCRCCC >= 70Z oder STXIT.
PENDT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump vor Aufruf von KCSBRSE).
PENDT2	F	(Siehe PENDT1) Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump vor Aufruf von KCSBRSE).
PENDT3	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump nach Aufruf durch KDCROOT mit PEND ER)
PLCA00	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ActivateCmxCallback(). Die CMX-Funktion t_callback() lieferte den Returncode T_ERROR.
PLCC00	SX	XAP-TP Baustein Modul KCOXFPL, Funktion CmxCallback(). Die Systemfunktion select() lieferte einen unerwarteten Returncode.
PLCC01	S	XAP-TP Baustein Modul KCOXFPL, Funktion CmxCallback(). Die Systemfunktion select() lieferte einen unbekanntenen Returncode.
PLCD00	SX	XAP-TP Baustein Modul KCOXFPL, Funktion DeactivateCmxCallback(). Die CMX-Funktion t_callback() lieferte den Returncode T_ERROR.



Code	Grp.	Beschreibung
PMIO20	ASU	Integrity IDs der Seite sowohl in Original wie auch in Duplikat File inkonsistent. Mögliche Ursache: Überschreiben der KDCFILE-Dateien im laufenden Betrieb durch Kopieren in Startprozedur. Maßnahme in diesem Fall: Startprozedur korrigieren Wenn dies als Ursache ausscheidet, PM schreiben. Als Diagnoseunterlagen sind erforderlich: UTM-Dump, KDCFILE-Dateien, Startprozedur und Task-/ Prozess-Protokoll. Bei Cluster-Anwendung auch Cluster-Pagepool-Dateien.
PMIO22	ASU	Der Page Type, angegeben in der Cache Control Table, und der Page Type im Header der gelesenen Seite stimmen nicht überein. Mögliche Ursache: Siehe PMIO20 Maßnahme: Siehe PMIO20
PMIO23	ASU	Zu lesende bzw. schreibende Pagepoolseite ist nicht allokiert. Ursache und Maßnahme: Siehe PMIO20
POLL03	MX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Der Returncode des Makros mGetBufferSize() war ungleich LB_OK.
POLL05	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). die OSS-Funktion o_event() meldet ein ACSE Event und es wurden noch nicht alle User Daten beim o_event() empfangen.
POLL07	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_ERROR.
POLL08	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_INVEREF.
POLL09	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die Oss-Funktion o_event() lieferte den Returncode O_TIMEINT.
POLL10	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_WAKEINT.
POLL11	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_SYSTEM.
POLL12	SX	XAP-TP Baustein Modul LCOXFPL, Funktion ap_poll(). Die OSS-Funktion ap_poll() lieferte den Returncode O_LOOK.

Code	Grp.	Beschreibung
POLL13	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_TRANSPORT.
POLL14	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_SESSION.
POLL15	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). Die OSS-Funktion o_event() lieferte den Returncode O_SHUTDOWN.
POLL16	SX	XAP-TP Baustein Modul KCOXFPL, Funktion ap_poll(). die OSS-Funktion o_event() lieferte einen unerwarteten Returncode.
PPMM10	ASU	UTM-Cluster-Anwendung Timeout in Routine lockFile beim Versuch, eine Cluster Pagepool Datei zu sperren. Maßnahme: Siehe CCFG19
PPMM12	SU	UTM-Cluster-Anwendung Timeout in Routine unlockFile beim Versuch, eine Cluster Pagepool Datei zu entsperren. Maßnahme: Siehe CCFG19
PPMM14	SU	UTM-Cluster-Anwendung Schlechter Returncode von KCSBFMM in Routine readControlPage.
PPMM15	ASU	UTM-Cluster-Anwendung Ungültiger Seiten-Header der Verwaltungsseite des Cluster Pagepools in Routine readControlPage.
PPMM41	SU	UTM-Cluster-Anwendung Fehler beim Lesen oder Schreiben der ersten CONS-Seite einer Cluster Pagepool Datei.
PPMM42	SU	UTM-Cluster-Anwendung Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers.
PRIV02	SU	Der Versuch, ein Systemprivileg zu setzen ist fehlgeschlagen.
PRIV03	SU	Der Versuch, ein Systemprivileg rückzusetzen ist fehlgeschlagen.
PWRT03	AMU	KCSPWRT hat (über KCSALME) Speicher verlangt, bekommt ihn aber nicht. Maßnahme: Speicherbedarf und Betriebssystem-Generierung überprüfen.
PWRT06	F	Bei KCSPWRT wurde während des Periodic Write von einem anderen UTM-Task der Anwendung term application aufgerufen (= Folge-Dump).
RALC01	AU	Index ist außerhalb des erlaubten Bereichs beim Aufruf von KCSRALC
RCV009	MX	XAP-TP Baustein. Modul KCOXFRV, Funktion ap_rcv(). Der Returncode der Funktion CopyElement() war ungleich DM_OK.

Code	Grp.	Beschreibung
RCV012	MX	XAP-TP Baustein. Modul KCOXFRV, Funktion ap_rcv(). Inkonsistenz in den booleschen Variablen <bSwitchToNextTtnid> und <bClearTtnid>.
RESTRT	D	Der Dump wird nur im Warmstart bei eingeschaltetem Testmode erzeugt. Er dient der Diagnose nach eventuellen Fehlern im Warmstart.
RQOB00	M	XAP-TP Baustein. Modul KCOCOHF, Funktion ReqOssInBuff(). Die Funktion RequestBuffer() lieferte den Returncode LB_NOMEM.
RVCS03	M	XAP-TP Baustein. Modul KCOXFRV, Funktion CheckSaRetc(). Der von der Funktion SetAttribute() gelieferte Returncode war SA_NOMEM.
SACT14	MX	XAP-TP Baustein Ungültiger Returncode nach Aufruf von PutElement() zum Anfordern eines dynamischen Speicherbereichs für SACF Aktion COPY.
SACT27	M	XAP-TP Baustein Ungültiger Returncode nach Aufruf von PutElement() zum Abspeichern in SACF Aktion FLUSHALL oder FLUSHPAR.
SACT28	M	XAP-TP Baustein. Ungültiger Returncode nach Aufruf von ReAllocElement() oder PutElement() in SACF Aktion QUEUE. Maßnahme: Den Wert des KDCDEF Parameters MAX OSI-SCRATCH-AREA erhöhen.
SAT002	S	Interner SAT-Fehler.
SAT003	A	Subsystem SATCP nicht verfügbar. Bitte den BS2000-Sicherheitsbeauftragten informieren.
SAT004	A	SAT-Funktion zur Zeit nicht ausführbar. Bitte den BS2000-Sicherheitsbeauftragten informieren.
SAT005	A	SAT-Record wurde nicht geschrieben. Bitte den BS2000-Sicherheitsbeauftragten informieren.
SC0005	SU	Die Half-Session ist als PLU generiert. Ein RU-Request ist gekommen, aber der Request-Code ist nicht SDT, STSN oder CLEAR.
SC0006	SU	Die Half-Session ist als PLU generiert. Eine RU-Response ist gekommen, aber der Request-Code ist nicht RQR.
SC0007	SU	#FSM.SESS ist nicht im Zustand "ACTIVE"
SC0008	SU	Die Half-Session ist als SLU generiert. Ein RU-Request ist gekommen, aber der Request-Code ist nicht RQR.
SC0009	SU	Die Half-Session ist als SLU generiert. Eine RU-Response ist gekommen, aber der Request-Code ist nicht SDT,STSN oder CLEAR.

Code	Grp.	Beschreibung
SC0010	SU	Für mindestens eine der FSMs #FSM_DT, #FSM_STSN oder #FSM_RQR liegt ein Protokollfehler vor.
SC0011	SU	Mindestens eine der FSMs #FSM_DT oder #FSM_CNTL_IMMED_EXP setzt einen Sense-Code.
SC0014	SU	FSM_SESS_LU_LU hat einen unerlaubten Returncode zurückgegeben.
SC0019	ADU	Protokollfehler von FSM gemeldet. Maßnahme: In der KDCDEF-Generierung überprüfen, ob beide Partner als VTV-Partner generiert sind.
SC0020	ADU	Der Zustand der PLU-FSM ist nicht erlaubt. Maßnahme: In der KDCDEF-Generierung überprüfen, ob beide Partner als VTV-Partner generiert sind.
SC0021	ADU	Der Zustand der SLU-FSM ist nicht erlaubt. Maßnahme: In der KDCDEF-Generierung überprüfen, ob beide Partner als VTV-Partner generiert sind.
SC0023	DSU	Bei Überprüfung von UNBIND-RSP für FSM_SESS_LU_LU ist ein Fehler gefunden worden.
SC0024	DSU	Beim Schalten von UNBIND-RSP für FSM_SESS_LU_LU ist ein Fehler aufgetreten.
SC0025	DSU	Bei Überprüfung von UNBIND-Request für FSM_SESS_LU_LU ist ein Fehler gefunden worden.
SC0026	DSU	Beim Schalten von UNBIND-Request für FSM_SESS_LU_LU ist ein Fehler aufgetreten (Diagnosedump).
SC0027	DSU	FSM_STSN_SEND meldet S,RQ,STSN.
SC0028	DSU	Unerlaubte STSN-REQ für PLU_TO_SLU.
SC0029	DSU	Unerlaubte STSN-REQ für SLU_TO_PLU.
SC0030	DSU	ACT_SC_RECEIVE konnte keinen Returncode setzen.
SC0032	A	Die Session kann nicht aktiviert werden, da keine freien Connections zur Verfügung stehen. Maßnahme: KDCDEF-Generierung ändern, mehr CON-Anweisungen aufnehmen.
SC0034	DSU	Der Partner hat beim STSN-Response einen Protokollfehler gemacht.
SDCS02	M	XAP-TP Baustein. Modul KCOXFSD, Funktion CheckSaRetc(). Der SetAttribute() Returncode war SA_NOMEM.
SDUI01	M	XAP-TP Baustein. Modul KCOXFSD, Funktion UserDataIn(). Der Makro mGetBufferSize() lieferte den Returncode LB_NOMEM.
SEND05	SU	Schwerer BCAM-Fehler bei der Verarbeitung eines BCAM-SENDLET-Calls. Fehler beim Canceln eines Timers für Output-Terminals.

Code	Grp.	Beschreibung
SENDT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump).
SG-nnn	D	Diagnosedump, der abhängig von einem Returncode KCRSIGN1/2 nach einem SIGN ON Aufruf erzeugt wird. Nach dem Präfix SG- folgt der Returncode (z.B SG-U17). Ein- und Ausschalten mit der Funktion Message-Dump.
SIGND1	DU	Der Aufruf der Kerberos-Schnittstelle liefert einen nicht erwarteten Returncode. Nur bei eingeschaltetem Testmodus.
SIGND2	DU	(Siehe SIGND1) Der Aufruf der Kerberos-Schnittstelle liefert einen nicht erwarteten Returncode. Nur bei eingeschaltetem Testmodus.
SIGND3	DU	(Siehe SIGND1) Der Aufruf der Kerberos-Schnittstelle liefert einen nicht erwarteten Returncode. Nur bei eingeschaltetem Testmodus.
SIGND4	DU	(Siehe SIGND1) Der Aufruf der Kerberos-Schnittstelle liefert einen nicht erwarteten Returncode. Nur bei eingeschaltetem Testmodus.
SLOG05	SU	Beim Versuch, die Message-Buffer zu sperren, hat KCSLKLC einen schlechten Returncode geliefert (eventuell TIMEOUT).
SLOG06	SU	Beim Versuch, die Message-Buffer zu entsperren, hat KCSLKLC einen schlechten Returncode geliefert.
SLOG07	SU	Beim Versuch, die SYSLOG-Datei zu sperren, hat KCSLKLC einen schlechten Returncode geliefert (eventuell TIMEOUT).
SLOG08	SU	Beim Versuch, die SYSLOG-Datei zu entsperren, hat KCSLKLC einen schlechten Returncode geliefert.
SLOG09	SU	Das Schreiben des Message-Buffers in die aktuelle SYSLOG-Datei ist schiefgegangen (eventuell gibt der DMS-Fehlercode in der vorausgegangenen K043-Meldung Aufschluss über die Fehlerursache).
SLOG10	SU	Der Versuch eines Folgetasks, auf die gerade aktuelle SYSLOG-Dateigeneration umzuschalten, ist fehlgeschlagen (eventuell vorausgegangene K043-Meldungen beachten).
SLOG21	ASU	Beim Administrationsaufruf KDCSLOG INFO bzw. beim Aufruf der Programmschnittstelle der Administration mit Opcode KC_SYSLOG und Subopcode1 KC_INFO hat KCSSLOG die Komponente KCSFST aufgerufen, um die Attribute der Dateigenerationsgruppe der SYSLOG zu ermitteln (nur wenn die SYSLOG als FGG angelegt ist). KCSFST lieferte einen DMS-Fehler (bitte auch die vorangegangene K043-Meldung beachten).
SLOT07	AM	In KCSSLOT muss die Adresse eines Slot berechnet werden. Der Task kennt den Memory Pool noch nicht, in dem der Slot liegt. Der Versuch, sich an den Pool anzuschließen, misslingt wegen Adressraummangel. Maßnahme: Speicherbedarf und ADDRESS-SPACE-LIMIT der BS2000-Kennung überprüfen.

Code	Grp.	Beschreibung
SM2001	SU	Aufruf von KCSSM2 mit falschem Opcode
SM2002	SU	Aufruf von KCSSM2 mit Opcode SM2_CONNECT bei KAA.STATE.SM2 = SM2_OFF
SM2003	SU	Aufruf von KCSSM2 mit einem Opcode "Daten schreiben", aber KAA.SM2.RECORD_ADDR = NULL
SM2004	SU	Fehlerhafte Parameter bei DSSM-Aufruf CONNECT_SS
SM2005	SU	Fehlerhafte Parameter bei DSSM-Aufruf DISCONNECT_SS
SM2006	SU	DSSM-Aufruf CONNECT_SS, aber Task ist bereits konnektiert
SM2007	SU	Aufruf von KCSSM2 bei KAA.STATE.SM2 = SM2_NO
SMSG00	AUD	SMSG00 ist ein Diagnose-Dump und führt nicht zum Ende des Anwendungslaufs; ein SMSG00-Dump wird nur gezogen, wenn der Testmode eingeschaltet ist. Ein SMSG-Dump wird erstellt, wenn KCSSMSG eine Meldung an ein Meldungsziel vom Typ MSG-DEST schicken möchte und dabei von KCSFPUT einen schlechten Returncode erhält. Der Returncode kann der Tabelle NKB des UTM-Dumps entnommen werden; auf Distanz X'5C' in dieser Tabelle finden sich der CC- und der DC-Code des DPUT-Aufrufs. Ein möglicher Grund kann eine Sperre der Queue/des TACs sein.
SMSG03	ASU	Die Komponente KCSSLOG lieferte beim Schreiben einer Meldung in die SYSLOG-Datei einen schlechten Returncode. Ursache siehe DMS-Returncode in K043-Meldung
SMSG09	D	SMSG09 ist ein Diagnose-Dump und führt nicht zum Ende des Anwendungslaufs; ein SMSG09-Dump wird nur gezogen, wenn der Testmode eingeschaltet ist. Bitte Systemdienst informieren.
SND005	M	XAP-TP Baustein. Modul KCOXFSD, Funktion ap_snd(). Die Funktion UserDataIn() lieferte den Returncode UDIN_NOMEM.
SND007	MX	XAP-TP Baustein. Modul KCOXFSD, Funktion ap_snd(). Der Returncode der Funktion GetVarLthAttr() war ungleich GA_OK.
SSO002	DSU	Diagnosedump, wird nach schwerwiegendem Fehler beim Aufruf des Subsystem SRPMSSO erzeugt
STnnnn	ADS U	Fehler in der Start-Verarbeitung einer UTM-Task. Dabei ist nnnn die Nummer, die in der UTM-Meldung "K049 Fehler <nnnn> bei Start der Anwendung" die Fehlerursache anzeigt.
STPF10	SX	XAP-TP Baustein Ungültiger Indikator für HeurRep Parameter in TP-HEUR-REP-RI.
STPF11	SX	XAP-TP Baustein Ungültiger Indikator für FuUnits Parameter in TP-BEG-DIAL(dialogue)-RC.

Code	Grp.	Beschreibung
STPF12	SX	XAP-TP Baustein Ungültiger Indikator für Result Parameter in TP-BEG-DIAL(dialogue)-RC.
STPF13	SX	XAP-TP Baustein Ungültiger Indikator für Diagnostic Parameter in TP-BEG-DIAL(dialogue)-RC.
STPF14	SX	XAP-TP Baustein Ungültiger Indikator für user data Parameter in TP-BEG-DIAL(dialogue)-RC.
STPF15	SX	XAP-TP Baustein Ungültiger Indikator für Result Parameter in TP-BEG-DIAL(channel)-RC.
STPF16	SX	XAP-TP Baustein Ungültiger Indikator für Diagnostic Parameter in TP-BEG-DIAL(channel)-RC.
STPF17	SX	XAP-TP Baustein Ungültiger Indikator für HeurRep Parameter in TP-(ABORT-AND)-HEUR-REP-RI.
STPT02	SX	XAP-TP Baustein Ungültiger Wert in p_udt.p_udtnxt.
STPT10	SX	XAP-TP Baustein Ungültiger Indikator für InitTPSUT Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT11	SX	XAP-TP Baustein Ungültiger Indikator für RecTPSUT Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT12	SX	XAP-TP Baustein Ungültiger Indikator für FuUnits Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT13	SX	XAP-TP Baustein Ungültiger Indikator für BegTrans Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT14	SX	XAP-TP Baustein Ungültiger Indikator für Conf Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT15	SX	XAP-TP Baustein Ungültiger Indikator für LastPartId Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT16	SX	XAP-TP Baustein Ungültiger Indikator für user data Parameter in TP-BEG-DIAL(dialogue)-RI.
STPT17	SX	XAP-TP Baustein Ungültiger Indikator für FuUnits Parameter in TP-BEG-DIAL(channel)-RI.
STPT18	SX	XAP-TP Baustein Ungültiger ChanUtil Parameter in TP-BEG-DIAL(channel)-RI.
STPT19	SX	XAP-TP Baustein Ungültiger Indikator für LastPartId Parameter in TP-BEG-DIAL(channel)-RI.
STPT20	SX	XAP-TP Baustein Ungültiger Indikator für CcrTokReq Parameter in TP-BID-RI.
STPT21	SX	XAP-TP Baustein Ungültiger Indikator für LastPartId Parameter in TP-BID-RI.

Code	Grp.	Beschreibung
STPT22	SX	XAP-TP Baustein Ungültiger Indikator für Result Parameter in TP-BID-RC.
STPT23	SX	XAP-TP Baustein Ungültiger Indikator für Conf Parameter in TP-END-DIAL-RI.
STPT24	SX	XAP-TP Baustein Ungültiger Indikator für ConfUrg Parameter in TP-HSK-RI.
STPT25	SX	XAP-TP Baustein Ungültiger Indikator für ConfUrg Parameter in TP-HSK-AND-GRT-CTRL-RI.
STPT26	SX	XAP-TP Baustein Ungültiger Indikator für Type Parameter in TP-DEFER-RI.
STPT27	SX	XAP-TP Baustein Ungültiger Indikator für Reason Parameter in TP-TOKEN-GIVE-RI.
STPT28	SX	XAP-TP Baustein Ungültiger Indikator für Correlator Parameter in TP-TOKEN-GIVE-RI.
STRT01	DU	Beim Austausch des Anwendungsprogramms ist die Anzahl der Prozesse, die noch austauschen müssen, kleiner 0, d.h. der Zähler ist inkonsistent. Bei eingeschaltetem Testmodus wird die Anwendung abnormal beendet, ansonsten wird eine Diagnose-Dump geschrieben.
SYPM01	AU	Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: in KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
SYPM03	AS	Schwerwiegender Fehler in Zusammenarbeit mit dem DB-System (FITA).
SYPM11	AU	Taskspezifischer Puffer für Wiederanlaufinformation ist zu klein. Maßnahme: in KDCDEF-Generierung MAX RECBUF=(...,length) vergrößern.
TADR03	SU	Die sogenannte Application-Reference ist im Announcement ungültig.
TC0040	DSU	Die vom Transportsystem gelieferte Eingabenachrichtenlänge stimmt nicht mit der in der Nachricht enthaltenen Länge überein.
TC0041	DSU	Die vom Transportsystem gelieferte Eingabenachrichtenlänge ist kürzer als der Wert im Data Count Field (DCF) im Transmission Header (TH).
TCPI13	SU	Ungültige Länge der IP-Adresse (TCP_GET_HOST_BY_ADDR).
TCPI14	SU	Ungültige Länge der IP-Adresse (TCP_CONNECTION_REQUEST).
TIEV19	DS	Inkonsistenz zwischen der annoncierten und der tatsächlichen Länge der Daten
TIMR02	SU	Fehler bei \$NCANTI.
TIMR03	SU	Fehler bei \$NCHNGTI.
TINF01	SU	Fehler beim Lesen der Task-Priorität
TINF02	SU	Fehler beim Setzen der Task-Priorität
UMDT12	SU	KCCUMDT Opcode UMDT_LOCK_ULS Schlechter Returncode von KCSBFMM beim Lesen einer ULS-Verwaltungsseite



Code	Grp.	Beschreibung
UMDT13	SU	KCCUMDT Opcode UMDT_LOCK_ULS Schlechter Returncode von KCCLKMT
UMDT15	ASU	UTM-Cluster-Anwendung KCCUMDT Opcode UMDT_UNLOCK_ULS Fehler beim Sperren der ULS-Datei. Maßnahme: Siehe Abschnitt "Maßnahme bei Sperren von UTM-Cluster-Dateien" im Anschluss an diese Tabelle.
UMDT32	SU	UTM-Cluster-Anwendung KCCUMDT Funktion readHdrPage Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers
UMDT33	ASU	UTM-Cluster-Anwendung KCCUMDT Funktion releaseFileLock Timeout beim Entsperren der Cluster-ULS-Datei. Maßnahme: Siehe UMDT15
UMDT34	SU	UTM-Cluster-Anwendung KCCUMDT Funktion releaseFileLock Schlechter Returncode von KCCGFLK beim Entsperren der Cluster-ULS-Datei. Maßnahme: Siehe UMDT15
UMDT42	SU	UTM-Cluster-Anwendung KCCUMDT Funktion addrUlsEntry Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers
UMDT44	SU	UTM-Cluster-Anwendung KCCUMDT Funktion setFileLock Schlechter Returncode von KCCGFLK beim Sperren der Cluster-ULS-Datei. Maßnahme: Siehe UMDT15
UMDT52	ASU	UTM-Cluster-Anwendung KCCUMDT Funktion allocateUlsEntry Schlechter Returncode von KCCUSF beim Sperren der Cluster-USER-Datei. Maßnahme: Siehe UMDT15
UMDT63	ASU	UTM-Cluster-Anwendung CUMDT Entry KCCUGLI Die Cluster-ULS-Datei kann nicht gesperrt werden. Maßnahme Siehe UMDT15
UMDT73	SU	UTM-Cluster-Anwendung KCCUMDT Funktion checkFile Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers
UMDT74	SU	UTM-Cluster-Anwendung KCCUMDT Funktion checkFile Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers

Code	Grp.	Beschreibung
UMDT77	SU	UTM-Cluster-Anwendung KCCUMDT Funktion checkFile Schlechter Returncode von KCSBFMM beim Anfordern eines Puffers
UMES02	DSU	Systemfehler. Bedeutung siehe Meldung K146.
USF002	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode OPEN_FILE Timeout beim Initialisieren des globalen Datei-Locks. Maßnahme: In der KDCDEF-Generierung den Wert des Parameters FILE-LOCK-TIMER-SEC bzw. FILE-LOCK-RETRY der CLUSTER-Anweisung erhöhen.
USF003	SU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode OPEN_FILE Ungültiger Returncode beim Initialisieren des globalen Datei-Locks
USF009	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode CUSF_RESET_PTC_FLAG Fünfmal Timeout beim Sperren der Cluster-User-Datei. Maßnahme: Siehe USF002
USF013	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode CUSF_SIGNOFF_USER_WITH_CONTEXT Fünf Mal Timeout beim Sperren der Cluster-User-Datei. Maßnahme: Siehe USF002
USF018	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode CLOSE_FILE Timeout beim Zerstören des Datei-Locks. Maßnahme: Siehe USF002
USF019	SU	UTM-Cluster-Anwendung Modul KCCUSF, Opcode CLOSE_FILE Ungültiger Returncode von KCCGFLK beim Zerstören des Datei-Locks
USF021	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion setFileLock Ungültiger Returncode von KCCGFLK beim Anfordern des exklusiven Locks
USF023	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion releaseFileLock Timeout bei der Freigabe des Datei-Locks. Maßnahme: Siehe USF002
USF024	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion releaseFileLock Ungültiger Returncode von KCCGFLK bei der Freigabe des Datei-Locks
USF025	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion readPage Timeout bei der Anforderung des shareable Datei-Locks. Maßnahme: Siehe USF002

Code	Grp.	Beschreibung
USF026	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion readPage Ungültiger Returncode von KCCGFLK bei Anforderung des shareable Datei-Locks
USF027	ASU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion readPage Timeout bei der Freigabe des Datei-Locks. Maßnahme: Siehe USF002
USF028	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion readPage Ungültiger Returncode von KCCGFLK bei der Freigabe des Datei-Locks
USF051	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion writePage Fehler beim Schreiben der Cluster-User-Datei
USF052	SU	UTM-Cluster-Anwendung Modul KCCUSF, Funktion readPage Fehler beim Lesen der Cluster-User-Datei
VTSUER	ADS	VTSU-B meldet einen Fehler in Zusammenhang mit dem verwendeten Zeichensatz oder der Verschlüsselung. Es liegt kein Abbruch der Anwendung vor. Der Dump dient der Diagnose des von VTSU-B gemeldeten Fehlers. Der Dump enthält nur KDCROOT und KTA Daten.
WAI102	DU	Diagnose-Dump
WAI112	DU	Beim Umschalten der System-Protokoll-Datei ist die Anzahl der Prozesse, die noch umschalten müssen, kleiner 0, d.h. der Zähler ist inkonsistent. Bei eingeschaltetem Testmodus wird die Anwendung abnormal beendet, ansonsten wird ein Diagnose-Dump geschrieben.
WAI113	DU	Beim Austausch des Anwendungsprogrammms ist die Anzahl der Prozesse, die noch austauschen müssen, kleiner 0, d.h. der Zähler ist inkonsistent. Bei eingeschaltetem Testmodus wird die Anwendung abnormal beendet, ansonsten wird ein Diagnose-Dump geschrieben.
WAI114	DU	Siehe WAI113
WAI120	DU	Beim Verwalten der Diagnose-Trace-Einstellungen ist die Anzahl der Prozesse, die ihre Verwaltung noch aktualisieren müssen, kleiner 0, d.h. der Zähler ist inkonsistent. Bei eingeschaltetem Testmodus wird die Anwendung abnormal beendet, ansonsten wird ein Diagnose-Dump geschrieben.
WAIT01	SU	Schlechter Returncode beim Aufruf von KCSBRSE für die Workbourse im Aktionsblock ENQUEUE_WORKBOURSE von KCSWAIT.
WAIT02	DS	Ungültiger Timer-Announcement Typ (Aktionsblock ENQUEUE_WORKBOURSE).

Code	Grp.	Beschreibung
WAIT03	DS	Ungültiger Typ im Announcement und kein Timer-Announcement (Aktionsblock ENQUEUE_WORKBOURSE).
WAIT55	SU	Ungültiger Code für ein internes Timer-Announcement (Aktionsblock ANALYSE_INTERNAL_TIMER).
WAIT78	ASU	Schwerwiegender Fehler beim Aufruf der Socket Funktionalität von BCAM.
WAITT1	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump vor Aufruf von KCSBRSE).
WAITT2	F	Eine andere Task hat die Anwendung abnormal beendet (= Folge-Dump nach Aufruf von KCSBRSE).
XATT02	F	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_att(). Die Funktion bCheckAndSetState() lieferte einen schlechten Returncode und der Anwendungsstatus war WAITING_DUMP_APPL. Eine andere Task hat bereits den Abbruch der Anwendung veranlasst.
XATT04	MX	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_att(). Der Returncode der Funktion EstablishBuffer() war ungleich LB_OK.
XATT12	MX	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_att(). Der Returncode der Funktion EstablishBuffer() war ungleich LB_OK.
XATT13	MX	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_att(). Der Returncode der Funktion RequestBuffer() war ungleich LB_OK.
XFGA07	SUX	XAP-TP Baustein. Modul KCOXFHF, Funktion GetAttribute(). Beim Lesen des Attributs AP_DTNID in der singletasking Variante lieferte die Funktion AllocUserMem() einen unerwarteten Returncode.
XFGA11	M	XAP-TP Baustein. Modul KCOXFHF, Funktion GetAttribute(). Der Makro mGetBufferSpace() lieferte den Returncode LB_NOMEM.
XFGE01	F	XAP-TP Baustein. Modul KCOXFHF, Funktion bCheckAndGetCallEnv(). Die Funktion bCheckAndSetState() lieferte einen schlechten Returncode und der Anwendungsstatus war WAITING_DUMP_APPL. Eine andere Task hat den Anwendungsabbruch veranlasst.
XFSA07	MX	XAP-TP Baustein. Modul KCOXFHF, Funktion SetAttribute(). Beim Setzen des Attributs AP_DTNID lieferte die Funktion PutElement() einen Returncode ungleich DM_OK.

Code	Grp.	Beschreibung
XFTM01	SX	XAP-TP Baustein. Modul KCOXFHF, Funktion TraceMgmt(). Die OSS Funktion o_tron() lieferte den Returncode O_ERROR.
XFTM02	SX	XAP-TP Baustein. Modul KCOXFHF, Funktion TraceMgmt(). Die OSS Funktion o_tron() lieferte den Returncode O_INVEREF.
XFTM03	SX	XAP-TP Baustein. Modul KCOXFHF, Funktion TraceMgmt(). Die OSS Funktion o_tron() lieferte einen unbekanntem Returncode.
XFTM04	SX	XAP-TP Baustein. Modul KCOXFHF, Funktion TraceMgmt(). Die OSS Funktion o_troff() lieferte den Returncode O_ERROR.
XFTM05	SX	XAP-TP Baustein. Modul KCOXFHF, Funktion TraceMgmt(). Die OSS Funktion o_troff() lieferte einen unbekanntem Returncode.
XINI06	ASX	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_init(). Die OSS Funktion o_create() lieferte den Returncode O_ERROR. Mögliche Ursachen: - Falls in der zugehörigen UTM Meldung P001 das Insert XPFUNC den Wert o_create() und das Insert XPRET den Wert -1 hat, kann der Fehler daran liegen, dass das OSS Subsystem nicht gestartet ist. - Eine OSI-TP-Anwendung wurde erneut im Dialog im gleichen Task gestartet. Von OSS blieben jedoch noch Betriebsmittel von der vorher beendeten Anwendung belegt, was zu diesem Abbruch führt. Tritt speziell dann auf, wenn die Anwendung zuvor im Dialog in zwei Tasks gestartet und wieder beendet wurde. Abhilfe: Nach LOGOFF BUT sind die Betriebsmittel wieder frei.
XINI07	SX	XAP-TP Baustein. Modul KCOXFEX, Funktion apext_init(). Die OSS Funktion o_create() lieferte einen unbekanntem Returncode.

#### *Maßnahmen bei Sperren von UTM-Cluster-Dateien*

Bei allen Fehlern, die beim Anfordern oder beim Freigeben von Sperren auf UTM-Cluster-Dateien aufgetreten sind, wird folgende Umgehungsmaßnahme empfohlen:

- ▶ Erhöhen Sie in der CLUSTER-Anweisung der KDCDEF-Generierung den Wert des Parameters FILE-LOCK-TIMER-SEC oder des Parameters FILE-LOCK-RETRY.



---

# Fachwörter

Fachwörter, die an anderer Stelle erklärt werden, sind mit *kursiver* Schrift ausgezeichnet.

## **Ablaufinvariantes Programm**

siehe *reentrant-fähiges Programm*.

## **Abnormale Beendigung einer UTM-Anwendung**

Beendigung einer *UTM-Anwendung*, bei der die *KDCFILE* nicht mehr aktualisiert wird. Eine abnormale Beendigung wird ausgelöst durch einen schwerwiegenden Fehler, z.B. Rechnerausfall, Fehler in der Systemsoftware. Wird die Anwendung erneut gestartet, führt openUTM einen *Warmstart* durch.

## **Abstrakte Syntax (OSI)**

Eine abstrakte Syntax ist die Menge der formal beschriebenen Datentypen, die zwischen Anwendungen über *OSI TP* ausgetauscht werden sollen. Eine abstrakte Syntax ist unabhängig von der eingesetzten Hardware und der jeweiligen Programmiersprache.

## **Access-List**

Eine Access-List definiert die Berechtigung für den Zugriff auf einen bestimmten *Service*, auf eine bestimmte *TAC-Queue* oder auf eine bestimmte *USER-Queue*. Eine Access-List ist als *Keyset* definiert und enthält einen oder mehrere *Keycodes*, die jeweils eine Rolle in der Anwendung repräsentieren. Benutzer, LTERMs oder (OSI-)LPAPs dürfen nur dann auf den Service oder die *TAC-Queue/USER-Queue* zugreifen, wenn ihnen die entsprechenden Rollen zugeteilt wurden, d.h. wenn ihr *Keyset* und die Access-List mindestens einen gemeinsamen *Keycode* enthalten.

## **Access Point (OSI)**

siehe *Dienstzugriffspunkt*.

## **ACID-Eigenschaften**

Abkürzende Bezeichnung für die grundlegenden Eigenschaften von *Transaktionen*: Atomicity, Consistency, Isolation und Durability.

### **Administration**

Verwaltung und Steuerung einer *UTM-Anwendung* durch einen *Administrator* oder ein *Administrationsprogramm*.

### **Administrations-Journal**

siehe *Cluster-Administrations-Journal*.

### **Administrationskommando**

Kommandos, mit denen der *Administrator* einer *UTM-Anwendung* Administrationsfunktionen für diese Anwendung durchführt. Die Administrationskommandos sind als *Transaktionscodes* realisiert.

### **Administrationsprogramm**

*Teilprogramm*, das Aufrufe der *Programmschnittstelle für die Administration* enthält. Dies kann das Standard-Administrationsprogramm *KDCADM* sein, das mit *openUTM* ausgeliefert wird, oder ein vom Anwender selbst erstelltes Programm.

### **Administrator**

Benutzer mit Administrationsberechtigung.

### **AES**

AES (Advanced Encryption Standard) ist der aktuelle symmetrische Verschlüsselungsstandard, festgelegt vom NIST (National Institute of Standards and Technology), basierend auf dem an der Universität Leuven (B) entwickelten Rijndael-Algorithmus. Wird das AES-Verfahren verwendet, dann erzeugt der UPIC-Client für jede Sitzung einen AES-Schlüssel.

### **Akzeptor (CPI-C)**

Die Kommunikationspartner einer *Conversation* werden *Initiator* und Akzeptor genannt. Der Akzeptor nimmt die vom Initiator eingeleitete *Conversation* mit *Accept\_Conversation* entgegen.

### **Anmelde-Vorgang (KDCS)**

Spezieller *Dialog-Vorgang*, bei dem die Anmeldung eines Benutzers an eine *UTM-Anwendung* durch *Teilprogramme* gesteuert wird.

### **Anschlussprogramm**

siehe *KDCROOT*.

### **Anwendungsinformation**

Sie stellt die Gesamtmenge der von der *UTM-Anwendung* benutzten Daten dar. Dabei handelt es sich um Speicherbereiche und Nachrichten der *UTM-Anwendung*, einschließlich der aktuell auf dem Bildschirm angezeigten Daten.



Arbeitet die UTM-Anwendung koordiniert mit einem Datenbanksystem, so gehören die in der Datenbank gespeicherten Daten ebenfalls zur Anwendungsinformation.

**Anwendungs-Kaltstart**

siehe *Kaltstart*.

**Anwendungsprogramm**

Ein Anwendungsprogramm bildet den Hauptbestandteil einer *UTM-Anwendung*. Es besteht aus der Main Routine *KDCROOT* und den *Teilprogrammen*. Es bearbeitet alle Aufträge, die an eine *UTM-Anwendung* gerichtet werden.

**Anwendungs-Warmstart**

siehe *Warmstart*.

**Apache Axis**

Apache Axis (Apache eXtensible Interaction System) ist eine SOAP-Engine zur Konstruktion von darauf basierenden Web Services und Client-Anwendungen. Es existiert eine Implementierung in C++ und Java.

**Apache Tomcat**

Apache Tomcat stellt eine Umgebung zur Ausführung von Java-Code auf Web-Servern bereit, die im Rahmen des Jakarta-Projekts der Apache Software Foundation entwickelt wird. Es handelt sich um einen in Java geschriebenen Servlet-Container, der mithilfe des JSP-Compilers Jasper auch JavaServer Pages in Servlets übersetzen und ausführen kann. Dazu kommt ein kompletter HTTP-Server.

**Application Context (OSI)**

Der Application Context ist die Menge der Regeln, die für die Kommunikation zwischen zwei Anwendungen gelten sollen. Dazu gehören z.B. die *abstrakten Syntaxen* und die zugeordneten *Transfer-Syntaxen*.

**Application Entity (OSI)**

Eine Application Entity (AE) repräsentiert alle für die Kommunikation relevanten Aspekte einer realen Anwendung. Eine Application Entity wird durch einen global (d.h. weltweit) eindeutigen Namen identifiziert, den *Application Entity Title* (AET). Jede Application Entity repräsentiert genau einen *Application Process*. Ein Application Process kann mehrere Application Entities umfassen.

### **Application Entity Qualifier (OSI)**

Bestandteil des *Application Entity Titles*. Der Application Entity Qualifier identifiziert einen *Dienstzugriffspunkt* innerhalb der Anwendung. Ein Application Entity Qualifier kann unterschiedlich aufgebaut sein. openUTM unterstützt den Typ "Zahl".

### **Application Entity Title (OSI)**

Ein Application Entity Title ist ein global (d.h. weltweit) eindeutiger Name für eine *Application Entity*. Er setzt sich zusammen aus dem *Application Process Title* des jeweiligen *Application Process* und dem *Application Entity Qualifier*.

### **Application Process (OSI)**

Der Application Process repräsentiert im *OSI-Referenzmodell* eine Anwendung. Er wird durch den *Application Process Title* global (d.h. weltweit) eindeutig identifiziert.

### **Application Process Title (OSI)**

Gemäß der OSI-Norm dient der Application Process Title (APT) zur global (d.h. weltweit) eindeutigen Identifizierung von Anwendungen. Er kann unterschiedlich aufgebaut sein. openUTM unterstützt den Typ *Object Identifier*.

### **Application Service Element (OSI)**

Ein Application Service Element (ASE) repräsentiert eine Funktionsgruppe der Anwendungsschicht (Schicht 7) des *OSI-Referenzmodells*.

### **Association (OSI)**

Eine Association ist eine Kommunikationsbeziehung zwischen zwei *Application Entities*. Dem Begriff Association entspricht der *LU6.1-Begriff Session*.

### **Asynchron-Auftrag**

*Auftrag*, der vom Auftraggeber zeitlich entkoppelt durchgeführt wird. Zur Bearbeitung von Asynchron-Aufträgen sind in openUTM *Message Queuing* Funktionen integriert, vgl. *UTM-gesteuerte Queue* und *Service-gesteuerte Queue*. Ein Asynchron-Auftrag wird durch die *Asynchron-Nachricht*, den Empfänger und ggf. den gewünschten Ausführungszeitpunkt beschrieben. Ist der Empfänger ein Terminal, ein Drucker oder eine Transportsystem-Anwendung, so ist der Asynchron-Auftrag ein *Ausgabe-Auftrag*; ist der Empfänger ein Asynchron-Vorgang derselben oder einer fernen Anwendung, so handelt es sich um einen *Hintergrund-Auftrag*. Asynchron-Aufträge können *zeitgesteuerte Aufträge* sein oder auch in einen *Auftrags-Komplex* integriert sein.

### Asynchron-Conversation

CPI-C-Conversation, bei der nur der *Initiator* senden darf. Für den *Akzeptor* muss in der *UTM-Anwendung* ein asynchroner Transaktionscode generiert sein.

### Asynchron-Nachricht

Asynchron-Nachrichten sind Nachrichten, die an eine *Message Queue* gerichtet sind. Sie werden von der lokalen *UTM-Anwendung* zunächst zwischengespeichert und dann unabhängig vom Auftraggeber weiter verarbeitet. Je nach Empfänger unterscheidet man folgende Typen von Asynchron-Nachrichten:

- Bei Asynchron-Nachrichten an eine *UTM-gesteuerte Queue* wird die Weiterverarbeitung komplett durch openUTM gesteuert. Zu diesem Typ gehören Nachrichten, die einen lokalen oder fernen *Asynchron-Vorgang* starten (vgl. auch *Hintergrund-Auftrag*) und Nachrichten, die zur Ausgabe an ein Terminal, einen Drucker oder eine Transportsystem-Anwendung geschickt werden (vgl. auch *Ausgabe-Auftrag*).
- Bei Asynchron-Nachrichten an eine *Service-gesteuerte Queue* wird die Weiterverarbeitung durch einen *Service* der Anwendung gesteuert. Zu diesem Typ gehören Nachrichten an eine *TAC-Queue*, Nachrichten an eine *USER-Queue* und Nachrichten an eine *Temporäre Queue*. Die User-Queue und die Temporäre Queue müssen dabei zur lokalen Anwendung gehören, die TAC-Queue kann sowohl in der lokalen als auch in einer fernen Anwendung liegen.

### Asynchron-Programm

*Teilprogramm*, das von einem *Hintergrund-Auftrag* gestartet wird.

### Asynchron-Vorgang (KDCS)

*Vorgang*, der einen *Hintergrund-Auftrag* bearbeitet. Die Verarbeitung erfolgt entkoppelt vom Auftraggeber. Ein Asynchron-Vorgang kann aus einem oder mehreren Teilprogrammen/Transaktionen bestehen. Er wird über einen asynchronen *Transaktionscode* gestartet.

### Auftrag

Anforderung eines *Services*, der von einer *UTM-Anwendung* zur Verfügung gestellt wird, durch Angabe eines *Transaktionscodes*. Siehe auch: *Ausgabe-Auftrag*, *Dialog-Auftrag*, *Hintergrund-Auftrag*, *Auftrags-Komplex*.

### Auftraggeber-Vorgang

Ein Auftraggeber-Vorgang ist ein *Vorgang*, der zur Bearbeitung eines Auftrags einen Service von einer anderen Server-Anwendung (*Auftragnehmer-Vorgang*) anfordert.

### **Auftragnehmer-Vorgang**

Ein Auftragnehmer-Vorgang ist ein *Vorgang*, der von einem *Auftraggeber-Vorgang* einer anderen Server-Anwendung gestartet wird.

### **Auftrags-Komplex**

Auftrags-Komplexe dienen dazu, *Asynchron-Aufträgen* *Quittungsaufträge* zuzuordnen. Ein Asynchron-Auftrag innerhalb eines Auftrags-Komplexes wird *Basis-Auftrag* genannt.

### **Ausgabe-Auftrag**

Ausgabeaufträge sind *Asynchron-Aufträge*, die die Aufgabe haben, eine Nachricht, z.B. ein Dokument, an einen Drucker, ein Terminal oder eine Transportsystem-Anwendung auszugeben.

Ausgabeaufträge werden ausschließlich von UTM-Systemfunktionen bearbeitet, d.h. für die Bearbeitung müssen keine Teilprogramme erstellt werden.

### **Authentisierung**

siehe *Zugangskontrolle*.

### **Autorisierung**

siehe *Zugriffskontrolle*.

### **Axis**

siehe *Apache Axis*.

### **Basis-Auftrag**

*Asynchron-Auftrag* in einem *Auftrags-Komplex*.

### **Basisformat**

Format, in das der Terminal-Benutzer alle Angaben eintragen kann, die notwendig sind, um einen Vorgang zu starten.

### **Basisname**

Basisname UTM-Anwendung.

Auf BS2000-Systemen ist Basisname das Präfix für die *KDCFILE*, die *Benutzerprotokoll-Datei* USLOG und die *System-Protokolldatei* SYSLOG.

Auf Unix-, Linux- und Windows-Systemen ist Basisname der Name des Verzeichnisses, unter dem die *KDCFILE*, die *Benutzerprotokoll-Datei* USLOG, die *System-Protokolldatei* SYSLOG und weitere Dateien der UTM-Anwendung abgelegt sind.

**Basisname der Knoten-Anwendung**

Dateinamens-Präfix bzw. Verzeichnisname für die *KDCFILE*, *Benutzerprotokoll-Datei* und *Systemprotokoll-Datei* der *Knoten-Anwendung*.

**Basisname der UTM-Cluster-Anwendung**

Dateinamens-Präfix bzw. Verzeichnisname für die *UTM-Cluster-Dateien*.

**Benutzerausgang**

Begriff ersetzt durch *Event-Exit*.

**Benutzerkennung**

Bezeichner für einen Benutzer, der in der *Konfiguration* der *UTM-Anwendung* festgelegt ist (optional mit Passwort zur *Zugangskontrolle*) und dem spezielle Zugriffsrechte (*Zugriffskontrolle*) zugeordnet sind. Ein Terminal-Benutzer muss bei der Anmeldung an die *UTM-Anwendung* diesen Bezeichner (und ggf. das zugeordnete Passwort) angeben. Auf *BS2000-Systemen* ist außerdem eine *Zugangskontrolle* über *Kerberos* möglich. Für andere Clients ist die Angabe der Benutzerkennung optional, siehe auch *Verbindungs-Benutzerkennung*. *UTM-Anwendungen* können auch ohne Benutzerkennungen generiert werden.

**Benutzer-Protokolldatei**

Datei oder Dateigeneration, in die der Benutzer mit dem *KDCS-Aufruf* *LPUT* Sätze variabler Länge schreibt. Jedem Satz werden die Daten aus dem *KB-Kopf* des *KDCS-Kommunikationsbereichs* vorangestellt. Die *Benutzerprotokolldatei* unterliegt der *Transaktionssicherung* von *openUTM*.

**Berechtigungsprüfung**

siehe *Zugangskontrolle*.

**Beweissicherung (BS2000-Systeme)**

Im Betrieb einer *UTM-Anwendung* können zur *Beweissicherung* sicherheitsrelevante *UTM-Ereignisse* von *SAT* protokolliert werden.

**Bildschirm-Wiederanlauf**

Wird ein *Dialog-Vorgang* unterbrochen, gibt *openUTM* beim *Vorgangswiederanlauf* die *Dialog-Nachricht* der letzten abgeschlossenen *Transaktion* erneut auf dem Bildschirm aus, sofern die letzte *Transaktion* eine *Nachricht* auf den Bildschirm ausgegeben hat.

### **Browsen von Asynchron-Nachrichten**

Ein *Vorgang* liest nacheinander die *Asynchron-Nachrichten*, die sich in einer *Service-gesteuerten Queue* befinden. Die Nachrichten werden während des Lesens nicht gesperrt und verbleiben nach dem Lesen in der Queue. Dadurch ist gleichzeitiges Lesen durch unterschiedliche Vorgänge möglich.

### **Bypass-Betrieb (BS2000-Systeme)**

Betriebsart eines Druckers, der lokal an ein Terminal angeschlossen ist. Im Bypass-Betrieb wird eine an den Drucker gerichtete *Asynchron-Nachricht* an das Terminal gesendet und von diesem auf den Drucker umgeleitet, ohne auf dem Bildschirm angezeigt zu werden.

### **Cache-Speicher**

Pufferbereich zur Zwischenspeicherung von Anwenderdaten für alle Prozesse einer *UTM-Anwendung*. Der Cache-Speicher dient zur Optimierung der Zugriffe auf den *Pagepool* und für UTM-Cluster-Anwendungen zusätzlich auf den *Cluster-Pagepool*.

### **CCS-Name (BS2000-Systeme)**

siehe *Coded-Character-Set-Name*.

### **Client**

Clients einer *UTM-Anwendung* können sein:

- Terminals
- UPIC-Client-Programme
- Transportsystem-Anwendungen (z.B. DCAM-, PDN-, CMX-, Socket-Anwendungen oder UTM-Anwendungen, die als *Transportsystem-Anwendung* generiert sind)

Clients werden über LTERM-Partner an die UTM-Anwendung angeschlossen. Hinweis: UTM-Clients mit Trägersystem OpenCPIC werden wie *OSI TP-Partner* behandelt.

### **Client-Seite einer Conversation**

Begriff ersetzt durch *Initiator*.

### **Cluster**

Eine Anzahl von Rechnern, die über ein schnelles Netzwerk verbunden sind und die von außen in vielen Fällen als ein Rechner gesehen werden können. Das Ziel des "Clustering" ist meist die Erhöhung der Rechenkapazität oder der Verfügbarkeit gegenüber einem einzelnen Rechner.

### Cluster-Administrations-Journal

Das Cluster-Administrations-Journal besteht aus:

- zwei Protokolldateien mit Endungen JRN1 und JRN2 für globale Administrationsaktionen,
- der JKAA-Datei, die eine Kopie der KDCS Application Area (KAA) enthält. Aus dieser Kopie werden administrative Änderungen übernommen, die nicht mehr in den beiden Protokolldateien enthalten sind.

Die Administrations-Journal-Dateien dienen dazu, administrative Aktionen, die in einer UTM-Cluster-Anwendung Cluster-weit auf alle Knoten-Anwendungen wirken sollen, an die anderen Knoten-Anwendungen weiterzugeben.

### Cluster-GSSB-Datei

Datei zur Verwaltung von GSSBs in einer *UTM-Cluster-Anwendung*. Die Cluster-GSSB-Datei wird mit dem UTM-Generierungstool *KDCDEF* erstellt.

### Cluster-Konfigurationsdatei

Datei, die die zentralen Konfigurationsdaten einer *UTM-Cluster-Anwendung* enthält. Die Cluster-Konfigurationsdatei wird mit dem UTM-Generierungstool *KDCDEF* erstellt.

### Cluster-Lock-Datei

Datei einer *UTM-Cluster-Anwendung*, die dazu dient, Knoten-übergreifende Sperren auf Anwenderdatenbereiche zu verwalten.

### Cluster-Pagepool

Der Cluster-Pagepool besteht aus einer Verwaltungsdatei und bis zu 10 Dateien, in denen die Cluster-weit verfügbaren Anwenderdaten (Vorgangsdaten inklusive LSSB, GSSB und ULS) einer *UTM-Cluster-Anwendung* gespeichert werden. Der Cluster-Pagepool wird mit dem UTM-Generierungstool *KDCDEF* erstellt.

### Cluster-Startserialisierungs-Datei

Lock-Datei, mit der die Starts einzelner Knoten-Anwendungen serialisiert werden (nur auf Unix-, Linux- und Windows-Systemen).

### Cluster-ULS-Datei

Datei zur Verwaltung von ULS-Bereichen einer *UTM-Cluster-Anwendung*. Die Cluster-ULS-Datei wird mit dem UTM-Generierungstool *KDCDEF* erstellt.

### Cluster-User-Datei

Datei, die die Verwaltungsdaten der Benutzer einer *UTM-Cluster-Anwendung* enthält. Die Cluster-User-Datei wird mit dem UTM-Generierungstool *KDCDEF* erstellt.

### **Coded-Character-Set-Name (BS2000-Systeme)**

Bei Verwendung des Produkts *XHCS* (eXtended Host Code Support) wird jeder verwendete Zeichensatz durch einen Coded-Character-Set-Namen (abgekürzt: "CCS-Name" oder "CCSN") eindeutig identifiziert.

### **Communication Resource Manager**

Communication Resource Manager (CRMs) kontrollieren in verteilten Systemen die Kommunikation zwischen den Anwendungsprogrammen. openUTM stellt CRMs für den internationalen Standard OSI TP, für den Industrie-Standard *LU6.1* und für das openUTM-eigene Protokoll UPIC zur Verfügung.

### **Contention Loser**

Jede Verbindung zwischen zwei Partnern wird von einem der Partner verwaltet. Der Partner, der die Verbindung verwaltet, heißt *Contention Winner*. Der andere Partner ist der *Contention Loser*.

### **Contention Winner**

Der *Contention Winner* einer Verbindung übernimmt die Verwaltung der Verbindung. Aufträge können sowohl vom *Contention Winner* als auch vom *Contention Loser* gestartet werden. Im Konfliktfall, wenn beide Kommunikationspartner gleichzeitig einen Auftrag starten wollen, wird die Verbindung vom Auftrag des *Contention Winner* belegt.

### **Conversation**

Bei CPI-C nennt man die Kommunikation zwischen zwei CPI-C-Anwendungsprogrammen *Conversation*. Die Kommunikationspartner einer *Conversation* werden *Initiator* und *Akzeptor* genannt.

### **Conversation-ID**

Jeder *Conversation* wird von CPI-C lokal eine *Conversation-ID* zugeordnet, d.h. *Initiator* und *Akzeptor* haben jeweils eine eigene *Conversation-ID*. Mit der *Conversation-ID* wird jeder CPI-C-Aufruf innerhalb eines Programms eindeutig einer *Conversation* zugeordnet.

### **CPI-C**

CPI-C (Common Programming Interface for Communication) ist eine von X/Open und dem CIW (CPI-C Implementor's Workshop) normierte Programmschnittstelle für die Programm-Programm-Kommunikation in offenen Netzen. Das in openUTM implementierte CPI-C genügt der CPI-C V2.0 CAE Specification von X/Open. Die Schnittstelle steht in COBOL und C zur Verfügung. CPI-C in openUTM kann über die Protokolle OSI TP, LU6.1, UPIC und mit openUTM-LU6.2 kommunizieren.



### **Cross Coupled System / XCS**

Verbund von BS2000-Rechnern mit *Highly Integrated System Complex Multiple System Control Facility* (HIPLEX® MSCF).

### **Datenraum (BS2000-Systeme)**

Virtueller Adressraum des BS2000, der in seiner gesamten Größe vom Anwender genutzt werden kann.

In einem Datenraum können nur Daten und als Daten abgelegte Programme adressiert werden, es kann kein Programmcode zum Ablauf gebracht werden.

### **Dead Letter Queue**

Die Dead Letter Queue ist eine *TAC-Queue* mit dem festen Namen KDCDLETQ. Sie steht immer zur Verfügung, um Asynchron-Nachrichten an *Transaktions-codes*, TAC-Queues, LPAP- oder OSI-LPAP-Partner zu sichern, die nicht verarbeitet werden konnten.

Die Sicherung von Asynchron-Nachrichten in der Dead Letter Queue kann durch den Parameter DEAD-LETTER-Q der TAC-, LPAP- oder OSI-LPAP-Anweisung für jedes Nachrichtenziel einzeln ein- und ausgeschaltet werden.

### **DES**

DES (Data Encryption Standard) ist eine internationale Norm zur Verschlüsselung von Daten. Bei diesem Verfahren wird ein Schlüssel zum Ver- und Entschlüsseln verwendet. Wird das DES-Verfahren verwendet, dann erzeugt der UPIC-Client für jede Sitzung einen DES-Schlüssel.

### **Dialog-Auftrag**

Auftrag, der einen *Dialog-Vorgang* startet. Der Auftrag kann von einem *Client* oder - bei *Server-Server-Kommunikation* - von einer anderen Anwendung erteilt werden.

### **Dialog-Conversation**

CPI-C-Conversation, bei der sowohl der *Initiator* als auch der *Akzeptor* senden darf. Für den *Akzeptor* muss in der *UTM-Anwendung* ein Dialog-Transaktionscode generiert sein.

### **Dialog-Nachricht**

Nachricht, die eine Antwort erfordert oder selbst eine Antwort auf eine Anfrage ist. Dabei bilden Anfrage und Antwort einen *Dialog-Schritt*.

### **Dialog-Programm**

*Teilprogramm*, das einen *Dialog-Schritt* teilweise oder vollständig bearbeitet.

### **Dialog-Schritt**

Ein Dialog-Schritt beginnt mit dem Empfang einer *Dialog-Nachricht* durch die *UTM-Anwendung*. Er endet mit der Antwort der UTM-Anwendung.

### **Dialog-Terminalprozess (Unix-, Linux- und Windows-Systeme)**

Ein Dialog-Terminalprozess verbindet ein Unix-, Linux- oder Windows-Terminal mit den *Workprozessen* der *UTM-Anwendung*. Dialog-Terminalprozesse werden entweder vom Benutzer durch Eingabe von *utmdtp* oder über die LOGIN-Shell gestartet. Für jedes Terminal, das an eine UTM-Anwendung angeschlossen werden soll, ist ein eigener Dialog-Terminalprozess erforderlich.

### **Dialog-Vorgang**

*Vorgang*, der einen *Auftrag* im Dialog (zeitlich gekoppelt) mit dem Auftraggeber (*Client* oder eine andere Server-Anwendung) bearbeitet. Ein Dialog-Vorgang verarbeitet *Dialog-Nachrichten* vom Auftraggeber und erzeugt Dialog-Nachrichten für diesen. Ein Dialog-Vorgang besteht aus mindestens einer *Transaktion*. Ein Dialog-Vorgang umfasst in der Regel mindestens einen *Dialog-Schritt*. Ausnahme: Bei *Vorgangskettung* können auch mehrere Vorgänge einen Dialog-Schritt bilden.

### **Dienst**

Programm auf Windows-Systemen, das im Hintergrund unabhängig von angemeldeten Benutzern oder Fenstern abläuft.

### **Dienstzugriffspunkt**

Im *OSI-Referenzmodell* stehen einer Schicht am Dienstzugriffspunkt die Leistungen der darunterliegenden Schicht zur Verfügung. Der Dienstzugriffspunkt wird im lokalen System durch einen *Selektor* identifiziert. Bei der Kommunikation bindet sich die *UTM-Anwendung* an einen Dienstzugriffspunkt. Eine Verbindung wird zwischen zwei Dienstzugriffspunkten aufgebaut.

### **Distributed Lock Manager / DLM (BS2000-Systeme)**

Konkurrierende, Rechner-übergreifende Dateizugriffe können über den Distributed Lock Manager synchronisiert werden. DLM ist eine Basisfunktion von HIPLEX<sup>®</sup> MSCF.

### **Distributed Transaction Processing**

X/Open-Architekturmodell für die transaktionsorientierte *verteilte Verarbeitung*.

### **Druckadministration**

Funktionen zur *Drucksteuerung* und Administration von *Ausgabeaufträgen*, die an einen Drucker gerichtet sind.

### Druckerbündel

Mehrere Drucker, die demselben *LTERM-Partner* zugeordnet sind.

### Druckergruppe (Unix- und Linux-Systeme)

Die Unix- oder Linux-Plattform richtet für jeden Drucker standardmäßig eine Druckergruppe ein, die genau diesen Drucker enthält. Darüber hinaus lassen sich mehrere Drucker einer Druckergruppe, aber auch ein Drucker mehreren Druckergruppen zuordnen.

### Druckerprozess (Unix- und Linux-Systeme)

Prozess, der vom *Mainprozess* zur Ausgabe von *Asynchron-Nachrichten* an eine *Druckergruppe* eingerichtet wird. Er existiert, solange die Druckergruppe an die *UTM-Anwendung* angeschlossen ist. Pro angeschlossener Druckergruppe gibt es einen Druckerprozess.

### Druckersteuerstation

Begriff wurde ersetzt durch *Druckersteuer-LTERM*.

### Druckersteuer-LTERM

Über ein Druckersteuer-LTERM kann sich ein *Client* oder ein Terminal-Benutzer an eine *UTM-Anwendung* anschließen. Von dem Client-Programm oder Terminal aus kann dann die *Administration* der Drucker erfolgen, die dem Druckersteuer-LTERM zugeordnet sind. Hierfür ist keine Administrationsberechtigung notwendig.

### Drucksteuerung

openUTM-Funktionen zur Steuerung von Druckausgaben.

### Dynamische Konfiguration

Änderung der *Konfiguration* durch die Administration. Im laufenden Betrieb der Anwendung können UTM-Objekte wie z.B. *Teilprogramme*, *Transaktionscodes*, *Clients*, *LU6.1-Verbindungen*, Drucker oder *Benutzerkennungen* in die Konfiguration aufgenommen, modifiziert oder teilweise auch gelöscht werden. Hierzu können die Administrationsprogramme WinAdmin oder WebAdmin verwendet werden, oder es müssen eigene *Administrationsprogramme* erstellt werden, die die Funktionen der *Programmschnittstelle der Administration* nutzen.

### Einschritt-Transaktion

*Transaktion*, die genau einen *Dialog-Schritt* umfasst.

### Einschritt-Vorgang

*Dialog-Vorgang*, der genau einen *Dialog-Schritt* umfasst.

### **Ereignisgesteuerter Vorgang**

Begriff ersetzt durch *Event-Service*.

### **Event-Exit**

Routine des *Anwendungsprogramms*, das bei bestimmten Ereignissen (z.B. Start eines Prozesses, Ende eines Vorgangs) automatisch gestartet wird. Diese darf - im Gegensatz zu den *Event-Services* - keine KDCS-, CPI-C- und XATMI-Aufrufe enthalten.

### **Event-Funktion**

Oberbegriff für *Event-Exits* und *Event-Services*.

### **Event-Service**

*Vorgang*, der beim Auftreten bestimmter Ereignisse gestartet wird, z.B. bei bestimmten UTM-Meldungen. Die *Teilprogramme* ereignisgesteuerter Vorgänge müssen KDCS-Aufrufe enthalten.

### **Generierung**

siehe *UTM-Generierung*.

### **Globaler Sekundärer Speicherbereich/GSSB**

siehe *Sekundärspeicherbereich*.

### **Hardcopy-Betrieb**

Betriebsart eines Druckers, der lokal an ein Terminal angeschlossen ist. Dabei wird eine Nachricht, die auf dem Bildschirm angezeigt wird, zusätzlich auf dem Drucker abgedruckt.

### **Heterogene Kopplung**

Bei *Server-Server-Kommunikation*: Kopplung einer *UTM-Anwendung* mit einer Nicht-UTM-Anwendung, z.B. einer CICS- oder TUXEDO-Anwendung.

### **Highly Integrated System Complex / HIPLEX®**

Produktfamilie zur Realisierung eines Bedien-, Last- und Verfügbarkeitsverbunds mit mehreren BS2000-Servern.

### **Hintergrund-Auftrag**

Hintergrund-Aufträge sind *Asynchron-Aufträge*, die an einen *Asynchron-Vorgang* der eigenen oder einer fernen Anwendung gerichtet sind. Hintergrund-Aufträge eignen sich besonders für zeitintensive oder zeitunkritische Verarbeitungen, deren Ergebnis keinen direkten Einfluss auf den aktuellen Dialog hat.

## HIPLEX® MSCF

(MSCF = **M**ultiple **S**ystem **C**ontrol **F**acility)

stellt bei HIPLEX® die Infrastruktur sowie Basisfunktionen für verteilte Anwendungen bereit.

## Homogene Kopplung

Bei *Server-Server-Kommunikation*: Kopplung von *UTM-Anwendungen*. Dabei spielt es keine Rolle, ob die Anwendungen auf der gleichen oder auf unterschiedlichen Betriebssystem-Plattformen ablaufen.

## Inbound-Conversation (CPI-C)

siehe *Incoming-Conversation*.

## Incoming-Conversation (CPI-C)

Eine *Conversation*, bei der das lokale CPI-C-Programm *Akzeptor* ist, heißt Incoming-Conversation. In der X/Open-Specification wird für Incoming-Conversation auch das Synonym Inbound-Conversation verwendet.

## Initiale KDCFILE

In einer *UTM-Cluster-Anwendung* die *KDCFILE*, die von *KDCDEF* erzeugt wurde und vor dem Start der Knoten-Anwendungen für jeden Knoten kopiert werden muss.

## Initiator (CPI-C)

Die Kommunikationspartner einer *Conversation* werden Initiator und *Akzeptor* genannt. Der Initiator baut die Conversation mit den CPI-C-Aufrufen *Initialize\_Conversation* und *Allocate* auf.

## Insert

Feld in einem Meldungstext, in das openUTM aktuelle Werte einträgt.

## Inverser KDCDEF

Funktion, die aus den Konfigurationsdaten der *KDCFILE*, die im laufenden Betrieb dynamisch angepasst wurde, Steueranweisungen für einen *KDCDEF*-Lauf erzeugt. Der inverse KDCDEF kann "offline" unter *KDCDEF* oder "online" über die *Programmschnittstelle zur Administration* gestartet werden.

## IUTMDB

Schnittstelle für die koordinierte Zusammenarbeit mit externen Resource Managern auf BS2000-Systemen. Dazu gehören Datenhaltungssysteme (LEASY) und Datenbanksysteme (SESAM/SQL, UDS/SQL).

### **JConnect-Client**

Bezeichnung für Clients auf Basis des Produkts openUTM-JConnect. Die Kommunikation mit der UTM-Anwendung erfolgt über das *UPIC-Protokoll*.

### **JDK**

Java Development Kit

Standard-Entwicklungsumgebung von Oracle Corporation für die Entwicklung von Java-Anwendungen.

### **Kaltstart**

Starten einer *UTM-Anwendung* nach einer *normalen Beendigung* der Anwendung oder nach einer Neugenerierung (vgl. auch *Warmstart*).

### **KDCADM**

Standard-Administrationsprogramm, das zusammen mit openUTM ausgeliefert wird. KDCADM stellt Administrationsfunktionen zur Verfügung, die über Transaktionscodes (*Administrationskommandos*) aufgerufen werden.

### **KDCDEF**

UTM-Tool für die *Generierung* von *UTM-Anwendungen*. KDCDEF erstellt anhand der Konfigurationsinformationen in den KDCDEF-Steueranweisungen die UTM-Objekte *KDCFILE* und die ROOT-Tabellen-Source für die Main Routine *KDCROOT*.

In UTM-Cluster-Anwendungen erstellt KDCDEF zusätzlich die *Cluster-Konfigurationsdatei*, die *Cluster-User-Datei*, den *Cluster-Pagepool*, die *Cluster-GSSB-Datei* und die *Cluster-ULS-Datei*.

### **KDCFILE**

Eine oder mehrere Dateien, die für den Ablauf einer *UTM-Anwendung* notwendige Daten enthalten. Die KDCFILE wird mit dem UTM-Generierungstool *KDCDEF* erstellt. Die KDCFILE enthält unter anderem die *Konfiguration* der Anwendung.

### **KDCROOT**

Main Routine eines *Anwendungsprogramms*, die das Bindeglied zwischen *Teilprogrammen* und UTM-Systemcode bildet. KDCROOT wird zusammen mit den *Teilprogrammen* zum *Anwendungsprogramm* gebunden.

### **KDCS-Parameterbereich**

siehe *Parameterbereich*.

### KDCS-Programmschnittstelle

Universelle UTM-Programmschnittstelle, die den nationalen Standard DIN 66 265 erfüllt und Erweiterungen enthält. Mit KDCS (Kompatible Datenkommunikationsschnittstelle) lassen sich z.B. Dialog-Services erstellen und *Message Queuing* Funktionen nutzen. Außerdem stellt KDCS Aufrufe zur *verteilten Verarbeitung* zur Verfügung.

### Kerberos

Kerberos ist ein standardisiertes Netzwerk-Authentisierungsprotokoll (RFC1510), das auf kryptographischen Verschlüsselungsverfahren basiert, wobei keine Passwörter im Klartext über das Netzwerk gesendet werden.

### Kerberos-Principal

Eigentümer eines Schlüssels.

Kerberos arbeitet mit symmetrischer Verschlüsselung, d.h. alle Schlüssel liegen an zwei Stellen vor, beim Eigentümer eines Schlüssels (Principal) und beim KDC (Key Distribution Center).

### Keycode

Code, der in einer Anwendung eine bestimmte Zugriffsberechtigung oder eine bestimmte Rolle repräsentiert. Mehrere Keycodes werden zu einem *Keyset* zusammengefasst.

### Keyset

Zusammenfassung von einem oder mehrerer *Keycodes* unter einem bestimmten Namen. Ein Keyset definiert Berechtigungen im Rahmen des verwendeten Berechtigungskonzepts (Lock-/Keycode-Konzept oder *Access-List*-Konzept). Ein Keyset kann einer *Benutzerkennung*, einem *LTERM-Partner*, einem (OSI-)LPAP-Partner, einem *Service* oder einer *TAC-Queue* zugeordnet werden.

### Knoten

Einzelner Rechner eines *Clusters*.

### Knoten-Anwendung

*UTM-Anwendung*, die als Teil einer *UTM-Cluster-Anwendung* auf einem einzelnen *Knoten* zum Ablauf kommt.

### Knoten-Recovery

Wenn für eine abnormal beendete Knoten-Anwendung zeitnah kein Warmstart auf ihrem eigenen *Knoten-Rechner* möglich ist, kann man für diesen Knoten auf einem anderen Knoten des UTM-Clusters eine Knoten-Recovery (Wiederherstellung) durchführen. Dadurch können Sperren, die von der ausgefallenen Knoten-Anwendung gehalten werden, freigegeben werden, um die laufende *UTM-Cluster-Anwendung* nicht unnötig zu beeinträchtigen.

### **Knotengebundener Vorgang**

Ein knotengebundener Vorgang eines Benutzers kann nur an der Knoten-Anwendung fortgesetzt werden, an der der Benutzer zuletzt angemeldet war.

Folgende Vorgänge sind immer knotengebunden:

- Vorgänge, die eine Kommunikation mit einem Auftragnehmer über LU6.1 oder OSI TP begonnen haben und bei denen der Auftragnehmervorgang noch nicht beendet wurde
- eingeschobene Vorgänge einer Vorgangskellerung
- Vorgänge, die eine SESAM-Transaktion abgeschlossen haben

Außerdem ist der Vorgang eines Benutzers knotengebunden, solange der Benutzer an einer Knoten-Anwendung angemeldet ist.

### **Kommunikationsbereich/KB (KDCS)**

Transaktionsgesicherter KDCS-*Primärspeicherbereich*, der Vorgangs-spezifische Daten enthält. Der Kommunikationsbereich besteht aus 3 Teilen:

- dem KB-Kopf mit allgemeinen Vorgangsdaten,
- dem KB-Rückgabebereich für Rückgaben nach KDCS-Aufrufen
- dem KB-Programmbereich zur Datenübergabe zwischen UTM-Teilprogrammen innerhalb eines *Vorgangs*.

### **Konfiguration**

Summe aller Eigenschaften einer *UTM-Anwendung*. Die Konfiguration beschreibt:

- Anwendungs- und Betriebsparameter
- die Objekte der Anwendung und die Eigenschaften dieser Objekte. Objekte sind z.B. *Teilprogramme* und *Transaktionscodes*, Kommunikationspartner, Drucker, *Benutzerkennungen*
- definierte Zugriffsschutz- und Zugangsschutzmaßnahmen

Die Konfiguration einer UTM-Anwendung wird bei der UTM-Generierung festgelegt (*statische Konfiguration*) und kann per *Administration* dynamisch (während des Anwendungslaufs) geändert werden (*dynamische Konfiguration*).

Die Konfiguration ist in der *KDCFILE* abgelegt.

### **Logging-Prozess**

Prozess auf Unix-, Linux- und Windows-Systemen, der die Protokollierung von Abrechnungssätzen oder Messdaten steuert.

### **Logische Verbindung**

Zuordnung zweier Kommunikationspartner.



## Log4j

Log4j ist ein Teil des Apache Jakarta Projekts. Log4j bietet Schnittstellen zum Protokollieren von Informationen (Ablauf-Informationen, Trace-Records,...) und zum Konfigurieren der Protokoll-Ausgabe. *WS4UTM* verwendet das Softwareprodukt Log4j für die Trace- und Logging-Funktionalität.

## Lockcode

Code, um einen LTERM-Partner oder einen Transaktionscode vor unberechtigtem Zugriff zu schützen. Damit ist ein Zugriff nur möglich, wenn das *Keyset* des Zugreifenden den passenden *Keycode* enthält (Lock-/Keycode-Konzept).

## Lokaler Sekundärer Speicherbereich/LSSB

siehe *Sekundärspeicherbereich*.

## LPAP-Bündel

LPAP-Bündel ermöglichen die Verteilung von Nachrichten an LPAP-Partner auf mehrere Partner-Anwendungen. Soll eine UTM-Anwendung sehr viele Nachrichten mit einer Partner-Anwendung austauschen, kann es für die Lastverteilung sinnvoll sein, mehrere Instanzen der Partner-Anwendung zu starten und die Nachrichten auf die einzelnen Instanzen zu verteilen. In einem LPAP-Bündel übernimmt *openUTM* die Verteilung der Nachrichten an die Instanzen der Partner-Anwendung. Ein LPAP-Bündel besteht aus einem Master-LPAP und mehreren Slave-LPAPs. Die Slave-LPAPs werden dem Master-LPAP bei der UTM-Generierung zugeordnet. LPAP-Bündel gibt es sowohl für das OSI TP-Protokoll als auch für das LU6.1-Protokoll.

## LPAP-Partner

Für die *verteilte Verarbeitung* über das *LU6.1*-Protokoll muss in der lokalen Anwendung für jede Partner-Anwendung ein LPAP-Partner konfiguriert werden. Der LPAP-Partner spiegelt in der lokalen Anwendung die Partner-Anwendung wider. Bei der Kommunikation wird die Partner-Anwendung nicht über ihren Anwendungsnamen oder ihre Adresse, sondern über den Namen des zugeordneten LPAP-Partners angesprochen.

## LTERM-Bündel

Ein LTERM-Bündel (Verbindungsbündel) besteht aus einem Master-LTERM und mehreren Slave-LTERMs. Mit einem LTERM-Bündel (Verbindungsbündel) verteilen Sie asynchrone Nachrichten an eine logische Partner-Anwendung gleichmäßig auf mehrere parallele Verbindungen.

## LTERM-Gruppe

Eine LTERM-Gruppe besteht aus einem oder mehreren Alias-LTERMs, den Gruppen-LTERMs, und einem Primary-LTERM. In einer LTERM-Gruppe ordnen Sie mehrere LTERMs einer Verbindung zu.

### **LTERM-Partner**

Um *Clients* oder Drucker an eine *UTM-Anwendung* anschließen zu können, müssen in der Anwendung LTERM-Partner konfiguriert werden. Ein Client oder Drucker kann nur angeschlossen werden, wenn ihm ein LTERM-Partner mit entsprechenden Eigenschaften zugeordnet ist. Diese Zuordnung wird i.A. in der *Konfiguration* festgelegt, sie kann aber auch dynamisch über Terminal-Pools erfolgen.

### **LTERM-Pool**

Statt für jeden *Client* eine LTERM- und eine PTERM-Anweisung anzugeben, kann mit der Anweisung TPOOL ein Pool von LTERM-Partnern definiert werden. Schließt sich ein Client über einen LTERM-Pool an, wird ihm dynamisch ein LTERM-Partner aus dem Pool zugeordnet.

### **LU6.1**

Geräteunabhängiges Datenaustauschprotokoll (Industrie-Standard) für die transaktionsgesicherte *Server-Server-Kommunikation*.

### **LU6.1-LPAP-Bündel**

*LPAP-Bündel* für *LU6.1-Partner-Anwendungen*.

### **LU6.1-Partner**

Partner der *UTM-Anwendung*, der mit der UTM-Anwendung über das Protokoll *LU6.1* kommuniziert.

Beispiele für solche Partner sind:

- eine UTM-Anwendung, die über LU6.1 kommuniziert
- eine Anwendung im IBM-Umfeld (z.B. CICS, IMS oder TXSeries), die über LU6.1 kommuniziert

### **Mainprozess (Unix-, Linux- und Windows-Systeme)**

Prozess, der die *UTM-Anwendung* startet. Er startet die *Workprozesse*, die *UTM-System-Prozesse*, *Druckerprozesse*, *Netzprozesse*, *Logging-Prozess* und den *Timerprozess* und überwacht die *UTM-Anwendung*.

### **Main Routine KDCROOT**

siehe *KDCROOT*.

### **Management Unit**

Komponente des *SE Servers*; ermöglicht mit Hilfe des *SE Managers* ein zentrales, web-basiertes Management aller Units eines SE Servers.

### **Meldung / UTM-Meldung**

Meldungen werden vom Transaktionsmonitor openUTM oder von UTM-Tools (wie z.B. *KDCDEF*) an *Meldungsziele* ausgegeben. Eine Meldung besteht aus einer Meldungsnummer und dem Meldungstext, der ggf. *Inserts* mit aktuellen Werten enthält. Je nach Meldungsziel werden entweder die gesamte Meldung oder nur Teile der Meldung (z.B. nur die *Inserts*) ausgegeben.

### **Meldungsdefinitionsdatei**

Die Meldungsdefinitionsdatei wird mit openUTM ausgeliefert und enthält standardmäßig die UTM-Meldungstexte in deutscher und englischer Sprache und die Definitionen der Meldungseigenschaften. Aufbauend auf diese Datei kann der Anwender auch eigene, individuelle Meldungsmodule erzeugen.

### **Meldungsziel**

Ausgabemedium für eine *Meldung*. Mögliche Meldungsziele von Meldungen des Transaktionsmonitors openUTM sind z.B. *Terminals*, *TS-Anwendungen*, der *Event-Service MSGTAC*, die *System-Protokolldatei SYSLOG* oder *TAC-Queues*, *Asynchron-TACs*, *USER-Queues*, *SYSOUT/SYSLST* bzw. *stderr/stdout*. Meldungsziele von Meldungen der UTM-Tools sind *SYSOUT/SYSLST* bzw. *stderr/stdout*.

### **Mehrschritt-Transaktion**

*Transaktion*, die aus mehr als einem *Verarbeitungsschritt* besteht.

### **Mehrschritt-Vorgang (KDCS)**

*Vorgang*, der in mehreren *Dialog-Schritten* ausgeführt wird.

### **Message Queuing**

Message Queuing (MQ) ist eine Form der Kommunikation, bei der die Nachrichten (Messages) nicht unmittelbar, sondern über zwischengeschaltete *Message Queues* ausgetauscht werden. Sender und Empfänger können zeitlich und räumlich entkoppelt ablaufen. Die Übermittlung der Nachricht hängt nicht davon ab, ob gerade eine Netzverbindung besteht oder nicht. Bei openUTM gibt es *UTM-gesteuerte Queues* und *Service-gesteuerte Queues*.

### **Message Queue**

Warteschlange, in der bestimmte Nachrichten transaktionsgesichert bis zur Weiterverarbeitung eingereiht werden. Je nachdem, wer die Weiterverarbeitung kontrolliert, unterscheidet man *Service-gesteuerte Queues* und *UTM-gesteuerte Queues*.

### **MSGTAC**

Spezieller Event-Service, der Meldungen mit dem Meldungsziel MSGTAC per Programm verarbeitet. MSGTAC ist ein Asynchron-Vorgang und wird vom Betreiber der Anwendung erstellt.

### **Multiplex-Verbindung (BS2000-Systeme)**

Spezielle Möglichkeit, die *OMNIS* bietet, um Terminals an eine *UTM-Anwendung* anzuschließen. Eine Multiplex-Verbindung ermöglicht es, dass sich mehrere Terminals eine *Transportverbindung* teilen.

### **Nachrichten-Bereich/NB (KDCS)**

Bei KDCS-Aufrufen: Puffer-Bereich, in dem Nachrichten oder Daten für openUTM oder für das *Teilprogramm* bereitgestellt werden.

### **Network File System/Service / NFS**

Ermöglicht den Zugriff von Unix- und Linux-Rechnern auf Dateisysteme über das Netzwerk.

### **Netzprozess (Unix-, Linux- und Windows-Systeme)**

Prozess einer *UTM-Anwendung* zur Netzanbindung.

### **Netzwerk-Selektor**

Der Netzwerk-Selektor identifiziert im lokalen System einen *Dienstzugriffspunkt* zur Vermittlungsschicht des *OSI-Referenzmodells*.

### **Normale Beendigung einer UTM-Anwendung**

Kontrollierte Beendigung einer *UTM-Anwendung*; das bedeutet u.a., dass die Verwaltungsdaten auf der *KDCFILE* aktualisiert werden. Eine normale Beendigung veranlasst der *Administrator* (z.B. mit KDCSHUT N). Den Start nach einer normalen Beendigung führt openUTM als *Kaltstart* durch.

### **Object Identifier**

Ein Object Identifier ist ein weltweit eindeutiger Bezeichner für Objekte im OSI-Umfeld. Ein Object Identifier besteht aus einer Folge von ganzen Zahlen, die einen Pfad in einer Baumstruktur repräsentiert.

### **Offener Terminalpool**

*Terminalpool*, der nicht auf *Clients* eines Rechners oder eines bestimmten Typs beschränkt ist. An diesen Terminalpool können sich alle Clients anschließen, für die kein Rechner- oder Typ-spezifischer Terminalpool generiert ist.

### **OMNIS (BS2000-Systeme)**

OMNIS ist ein „Session-Manager“ auf einem BS2000-System, der die gleichzeitige Verbindungsaufnahme von einem Terminal zu mehreren Partnern in einem Netzwerk ermöglicht. OMNIS ermöglicht es außerdem, mit *Multiplex-Verbindungen* zu arbeiten.

### **Online-Import**

Als Online-Import wird in einer *UTM-Cluster-Anwendung* das Importieren von Anwendungsdaten aus einer normal beendeten Knoten-Anwendung in eine laufende Knoten-Anwendung bezeichnet.

### **Online-Update**

Als Online-Update wird in einer *UTM-Cluster-Anwendung* die Änderung der Konfiguration der Anwendung oder des Anwendungsprogramms oder der Einsatz einer neuen UTM-Korrekturstufe bei laufender *UTM-Cluster-Anwendung* bezeichnet.

### **OpenCPIC**

Trägersystem für UTM-Clients, die das *OSI TP* Protokoll verwenden.

### **OpenCPIC-Client**

*OSI TP* Partner-Anwendungen mit Trägersystem *OpenCPIC*.

### **openSM2**

Die Produktlinie openSM2 ist eine einheitliche Lösung für das unternehmensweite Performance Management von Server- und Speichersystemen. openSM2 bietet eine Messdatenerfassung, Online-Überwachung und Offline-Auswertung.

### **openUTM-Cluster**

aus der Sicht von UPIC-Clients, **nicht** aus Server-Sicht:  
Zusammenfassung mehrerer Knoten-Anwendungen einer UTM-Cluster-Anwendung zu einer logischen Anwendung, die über einen gemeinsamen Symbolic Destination Name adressiert wird.

### **openUTM-D**

openUTM-D (openUTM-Distributed) ist eine openUTM-Komponente, die *verteilte Verarbeitung* ermöglicht. openUTM-D ist integraler Bestandteil von openUTM.

### **OSI-LPAP-Bündel**

*LPAP-Bündel* für *OSI TP*-Partner-Anwendungen.

### **OSI-LPAP-Partner**

OSI-LPAP-Partner sind die bei openUTM generierten Adressen der *OSI TP-Partner*. Für die *verteilte Verarbeitung* über das Protokoll *OSI TP* muss in der lokalen Anwendung für jede Partner-Anwendung ein OSI-LPAP-Partner konfiguriert werden. Der OSI-LPAP-Partner spiegelt in der lokalen Anwendung die Partner-Anwendung wider. Bei der Kommunikation wird die Partner-Anwendung nicht über ihren Anwendungsnamen oder ihre Adresse, sondern über den Namen des zugeordneten OSI-LPAP-Partners angesprochen.

### **OSI-Referenzmodell**

Das OSI-Referenzmodell stellt einen Rahmen für die Standardisierung der Kommunikation von offenen Systemen dar. ISO, die Internationale Organisation für Standardisierung, hat dieses Modell im internationalen Standard ISO IS7498 beschrieben. Das OSI-Referenzmodell unterteilt die für die Kommunikation von Systemen notwendigen Funktionen in sieben logische Schichten. Diese Schichten haben jeweils klar definierte Schnittstellen zu den benachbarten Schichten.

### **OSI TP**

Von der ISO definiertes Kommunikationsprotokoll für die verteilte Transaktionsverarbeitung. OSI TP steht für Open System Interconnection Transaction Processing.

### **OSI TP-Partner**

Partner der UTM-Anwendung, der mit der UTM-Anwendung über das OSI TP-Protokoll kommuniziert.

Beispiele für solche Partner sind:

- eine UTM-Anwendung, die über OSI TP kommuniziert
- eine Anwendung im IBM-Umfeld (z.B. CICS), die über openUTM-LU62 angeschlossen ist
- ein *OpenCPIC-Client*
- Anwendungen anderer TP-Monitore, die OSI TP unterstützen

### **Outbound-Conversation (CPI-C)**

siehe *Outgoing-Conversation*.

### **Outgoing-Conversation (CPI-C)**

Eine Conversation, bei der das lokale CPI-C-Programm der *Initiator* ist, heißt Outgoing-Conversation. In der X/Open-Specification wird für Outgoing-Conversation auch das Synonym Outbound-Conversation verwendet.

### Pagepool

Teil der *KDCFILE*, in dem Anwenderdaten gespeichert werden.

In einer *stand-alone Anwendung* sind dies z.B. *Dialog-Nachrichten*, Nachrichten an *Message Queues*, *Sekundärspeicherbereiche*.

In einer *UTM-Cluster-Anwendung* sind dies z.B. Nachrichten an *Message Queues*, *TLS*.

### Parameterbereich

Datenstruktur, in der ein *Teilprogramm* bei einem UTM-Aufruf die für diesen Aufruf notwendigen Operanden an openUTM übergibt.

### Partner-Anwendung

Partner einer UTM-Anwendung bei *verteilter Verarbeitung*. Für die verteilte Verarbeitung werden höhere Kommunikationsprotokolle verwendet (*LU6.1*, *OSI TP* oder *LU6.2* über das Gateway openUTM-LU62).

### Postselection (BS2000-Systeme)

Auswahl der protokollierten UTM-Ereignisse aus der SAT-Protokolldatei, die ausgewertet werden sollen. Die Auswahl erfolgt mit Hilfe des Tools SATUT.

### Programmraum (BS2000-Systeme)

In Speicherklassen aufgeteilter virtueller Adressraum des BS2000, in dem sowohl ablauffähige Programme als auch reine Daten adressiert werden.

### Prepare to commit (PTC)

Bestimmter Zustand einer verteilten Transaktion:

Das Transaktionsende der verteilten Transaktion wurde eingeleitet, es wird jedoch noch auf die Bestätigung des Transaktionsendes durch den Partner gewartet.

### Preselection (BS2000-Systeme)

Festlegung der für die *SAT-Beweissicherung* zu protokollierenden UTM-Ereignisse. Die Preselection erfolgt durch die UTM-SAT-Administration. Man unterscheidet Ereignis-spezifische, Benutzer-spezifische und Auftrags-(TAC-)spezifische Preselection.

### Presentation-Selektor

Der Presentation-Selektor identifiziert im lokalen System einen *Dienstzugriffspunkt* zur Darstellungsschicht des *OSI-Referenzmodells*.

### Primärspeicherbereich

Bereich im Arbeitsspeicher, auf den das *KDCS-Teilprogramm* direkt zugreifen kann, z.B. *Standard Primärer Arbeitsbereich*, *Kommunikationsbereich*.

### **Printerprozess (Unix- und Linux-Systeme)**

siehe *Druckerprozess*.

### **Programmschnittstelle zur Administration**

UTM-Programmschnittstelle, mit deren Hilfe der Anwender eigene *Administrationsprogramme* erstellen kann. Die Programmschnittstelle zur Administration bietet u.a. Funktionen zur *dynamischen Konfiguration*, zur Modifikation von Eigenschaften und Anwendungsparametern und zur Abfrage von Informationen zur *Konfiguration* und zur aktuellen Auslastung der Anwendung.

### **Prozess**

In den openUTM-Handbüchern wird der Begriff "Prozess" als Oberbegriff für Prozess (Unix-, Linux- und Windows-Systeme) und Task (BS2000-Systeme) verwendet.

### **Queue**

siehe *Message Queue*

### **Quick Start Kit**

Beispielanwendung, die mit openUTM (Windows-Systeme) ausgeliefert wird.

### **Quittungs-Auftrag**

Bestandteil eines *Auftrags-Komplexes*, worin der Quittungs-Auftrag dem *Basis-Auftrag* zugeordnet ist. Es gibt positive und negative Quittungsaufträge. Bei positivem Ergebnis des *Basis-Auftrags* wird der positive Quittungs-Auftrag wirksam, sonst der negative.

### **Redelivery**

Erneutes Zustellen einer *Asynchron-Nachricht*, nachdem diese nicht ordnungsgemäß verarbeitet werden konnte, z.B. weil die *Transaktion* zurückgesetzt oder der *Asynchron-Vorgang* abnormal beendet wurde. Die Nachricht wird wieder in die Message Queue eingereiht und lässt sich damit erneut lesen und/oder verarbeiten.

### **Reentrant-fähiges Programm**

Programm, dessen Code durch die Ausführung nicht verändert wird. Auf BS2000-Systemen ist dies Voraussetzung dafür, *Shared Code* zu nutzen.

### **Request**

Anforderung einer *Service-Funktion* durch einen *Client* oder einen anderen Server.



### Requestor

In XATMI steht der Begriff Requestor für eine Anwendung, die einen Service aufruft.

### Resource Manager

Resource Manager (RMs) verwalten Datenressourcen. Ein Beispiel für RMs sind Datenbank-Systeme. openUTM stellt aber auch selbst Resource Manager zur Verfügung, z.B. für den Zugriff auf *Message Queues*, lokale Speicherbereiche und Logging-Dateien. Anwendungsprogramme greifen auf RMs über RM-spezifische Schnittstellen zu. Für Datenbank-Systeme ist dies meist SQL, für die openUTM-RMs die Schnittstelle KDCS.

### RFC1006

Von IETF (Internet Engineering Task Force) definiertes Protokoll der TCP/IP-Familie zur Realisierung der ISO-Transportdienste (Transportklasse 0) auf TCP/IP-Basis.

### RSA

Abkürzung für die Erfinder des RSA-Verschlüsselungsverfahrens Rivest, Shamir und Adleman. Bei diesem Verfahren wird ein Schlüsselpaar verwendet, das aus einem öffentlichen und einem privaten Schlüssel besteht. Eine Nachricht wird mit dem öffentlichen Schlüssel verschlüsselt und kann nur mit dem privaten Schlüssel entschlüsselt werden. Das RSA-Schlüsselpaar wird von der UTM-Anwendung erzeugt.

### SAT-Beweissicherung (BS2000-Systeme)

*Beweissicherung* durch die Komponente SAT (Security Audit Trail) des BS2000-Softwareproduktes SECOS.

### SE Manager

Web-basierte Benutzeroberfläche (GUI) für Business Server der SE Serie. Der SE Manager läuft auf der *Management Unit* und ermöglicht die zentrale Bedienung und Verwaltung von Server Units (mit /390-Architektur und/oder x86-Architektur), Application Units (x86-Architektur), Net Unit und der Peripherie.

### SE Server

Ein Business Server der SE Serie von Fujitsu.

### Sekundärspeicherbereich

Transaktionsgesicherter Speicherbereich, auf den das KDCS-*Teilprogramm* mit speziellen Aufrufen zugreifen kann. Lokale Sekundärspeicherbereiche (LSSB) sind einem *Vorgang* zugeordnet, auf globale Sekundärspeicherbereiche

(GSSB) kann von allen Vorgängen einer *UTM-Anwendung* zugegriffen werden. Weitere Sekundärspeicherbereiche sind der *Terminal-spezifische Langzeitspeicher (TLS)* und der *User-spezifische Langzeitspeicher (ULS)*.

### **Selektor**

Ein Selektor identifiziert im lokalen System einen *Zugriffspunkt* auf die Dienste einer Schicht des *OSI-Referenzmodells*. Jeder Selektor ist Bestandteil der Adresse des Zugriffspunktes.

### **Semaphor (Unix-, Linux- und Windows-Systeme)**

Betriebsmittel auf Unix-, Linux- und Windows-Systemen, das zur Steuerung und Synchronisation von Prozessen dient.

### **Server**

Ein Server ist eine *Anwendung*, die *Services* zur Verfügung stellt. Oft bezeichnet man auch den Rechner, auf dem Anwendungen laufen, als Server.

### **Server-Seite einer Conversation (CPI-C)**

Begriff ersetzt durch *Akzeptor*.

### **Server-Server-Kommunikation**

siehe *verteilte Verarbeitung*.

### **Service Access Point**

siehe *Dienstzugriffspunkt*.

### **Service**

Services bearbeiten die *Aufträge*, die an eine Server-Anwendung geschickt werden. Ein Service in einer UTM-Anwendung wird auch *Vorgang* genannt und setzt sich aus einer oder mehreren *Transaktionen* zusammen. Ein Service wird über den *Vorgangs-TAC* aufgerufen. Services können von *Clients* oder anderen Services angefordert werden.

### **Service-gesteuerte Queue**

*Message Queue*, bei der der Abruf und die Weiterverarbeitung der Nachrichten durch *Services* gesteuert werden. Ein Service muss zum Lesen der Nachricht explizit einen KDCS-Aufruf (DGET) absetzen. Service-gesteuerte Queues gibt es bei openUTM in den Varianten *USER-Queue*, *TAC-Queue* und *Temporäre Queue*.

### **Service Routine**

siehe *Teilprogramm*.

**Session**

Kommunikationsbeziehung zweier adressierbarer Einheiten im Netz über das SNA-Protokoll *LU6.1*.

**Session-Selektor**

Der Session-Selektor identifiziert im lokalen System einen *Zugriffspunkt* zu den Diensten der Kommunikationssteuerschicht (Session-Layer) des *OSI-Referenzmodells*.

**Shared Code (BS2000-Systeme)**

Code, der von mehreren Prozessen gemeinsam benutzt werden kann.

**Shared Memory**

Virtueller Speicherbereich, auf den mehrere Prozesse gleichzeitig zugreifen können.

**Shared Objects (Unix-, Linux- und Windows-Systeme)**

Teile des *Anwendungsprogramms* können als Shared Objects erzeugt werden. Diese werden dynamisch zur Anwendung dazugebunden und können im laufenden Betrieb ausgetauscht werden. Shared Objects werden mit der KDCDEF-Anweisung SHARED-OBJECT definiert.

**Sicherungspunkt**

Ende einer *Transaktion*. Zu diesem Zeitpunkt werden alle in der Transaktion vorgenommenen Änderungen der *Anwendungsinformation* gegen Systemausfall gesichert und für andere sichtbar gemacht. Während der Transaktion gesetzte Sperren werden wieder aufgehoben.

**Single System Image**

Unter single system image versteht man die Eigenschaft eines *Clusters*, nach außen hin als ein einziges, in sich geschlossenes System zu erscheinen. Die heterogene Natur des Clusters und die interne Verteilung der Ressourcen im Cluster ist für die Benutzer des Clusters und die Anwendungen, die mit dem Cluster kommunizieren, nicht sichtbar.

**SOA**

SOA (Service-oriented architecture).

SOA ist ein Konzept für eine Systemarchitektur, in dem Funktionen in Form von wieder verwendbaren, technisch voneinander unabhängigen und fachlich lose gekoppelten *Services* implementiert werden. Services können unabhängig von zugrunde liegenden Implementierungen über Schnittstellen aufgerufen werden, deren Spezifikationen öffentlich und damit vertrauenswürdig sein können. Service-Interaktion findet über eine dafür vorgesehene Kommunikationsinfrastruktur statt.

### **SOAP**

SOAP (Simple Object Access Protocol) ist ein Protokoll, mit dessen Hilfe Daten zwischen Systemen ausgetauscht und Remote Procedure Calls durchgeführt werden können. SOAP stützt sich auf die Dienste anderer Standards, XML zur Repräsentation der Daten und Internet-Protokolle der Transport- und Anwendungsschicht zur Übertragung der Nachrichten.

### **Socket-Verbindung**

Transportsystem-Verbindung, die die Socket-Schnittstelle verwendet. Die Socket-Schnittstelle ist eine Standard-Programmschnittstelle für die Kommunikation über TCP/IP.

### **Stand-alone Anwendung**

siehe *stand-alone UTM-Anwendung*.

### **Stand-alone UTM-Anwendung**

Herkömmliche *UTM-Anwendung*, die nicht Bestandteil einer *UTM-Cluster-Anwendung* ist.

### **Standard Primärer Arbeitsbereich/SPAB (KDCS)**

Bereich im Arbeitsspeicher, der jedem KDCS-*Teilprogramm* zur Verfügung steht. Sein Inhalt ist zu Beginn des Teilprogrammlaufs undefiniert oder mit einem Füllzeichen vorbelegt.

### **Startformat**

Format, das openUTM am Terminal ausgibt, wenn sich ein Benutzer erfolgreich bei der *UTM-Anwendung* angemeldet hat (ausgenommen nach *Vorgangswiederanlauf* und beim Anmelden über *Anmelde-Vorgang*).

### **Statische Konfiguration**

Festlegen der *Konfiguration* bei der UTM-Generierung mit Hilfe des UTM-Tools *KDCDEF*.

### **SYSLOG-Datei**

siehe *System-Protokolldatei*.

### **System-Protokolldatei**

Datei oder Dateigeneration, in die openUTM während des Laufs einer *UTM-Anwendung* alle UTM-Meldungen protokolliert, für die das *Meldungsziel* SYSLOG definiert ist.

### **TAC**

siehe *Transaktionscode*.

### TAC-Queue

*Message Queue*, die explizit per KDCDEF-Anweisung generiert wird. Eine TAC-Queue ist eine *Service-gesteuerte Queue* und kann unter dem generierten Namen von jedem Service aus angesprochen werden.

### Teilprogramm

UTM-*Services* werden durch ein oder mehrere Teilprogramme realisiert. Die Teilprogramme sind Bestandteile des *Anwendungsprogramms*. Abhängig vom verwendeten API müssen sie KDCS-, XATMI- oder CPIC-Aufrufe enthalten. Sie sind über *Transaktionscodes* ansprechbar. Einem Teilprogramm können mehrere Transaktionscodes zugeordnet werden.

### Temporäre Queue

*Message Queue*, die dynamisch per Programm erzeugt wird und auch wieder per Programm gelöscht werden kann, vgl. *Service-gesteuerte Queue*.

### Terminal-spezifischer Langzeitspeicher/TLS (KDCS)

*Sekundärspeicher*, der einem LTERM-, LPAP- oder OSI-LPAP-Partner zugeordnet ist und über das Anwendungsende hinaus erhalten bleibt.

### Timerprozess (Unix-, Linux- und Windows-Systeme)

*Prozess*, der Aufträge zur Zeitüberwachung von *Workprozessen* entgegennimmt, sie in ein Auftragsbuch einordnet und nach einer im Auftragsbuch festgelegten Zeit den Workprozessen zur Bearbeitung wieder zustellt.

### TNS (Unix-, Linux- und Windows-Systeme)

Abkürzung für den Transport Name Service, der einem Anwendungsnamen einen Transport-Selektor und das Transportsystem zuordnet, über das die Anwendung erreichbar ist.

### Tomcat

siehe *Apache Tomcat*

### Transaktion

Verarbeitungsabschnitt innerhalb eines *Services*, für den die Einhaltung der *ACID-Eigenschaften* garantiert wird. Von den in einer Transaktion beabsichtigten Änderungen der *Anwendungsinformation* werden entweder alle konsistent durchgeführt oder es wird keine durchgeführt (Alles-oder-Nichts Regel). Das Transaktionsende bildet einen *Sicherungspunkt*.

### **Transaktionscode/TAC**

Name, über den ein *Teilprogramm* aufgerufen werden kann. Der Transaktionscode wird dem Teilprogramm bei der *statischen* oder *dynamischen Konfiguration* zugeordnet. Einem Teilprogramm können auch mehrere Transaktionscodes zugeordnet werden.

### **Transaktionsrate**

Anzahl der erfolgreich beendeten *Transaktionen* pro Zeiteinheit.

### **Transfer-Syntax**

Bei *OSI TP* werden die Daten zur Übertragung zwischen zwei Rechnersystemen von der lokalen Darstellung in die Transfer-Syntax umgewandelt. Die Transfer-Syntax beschreibt die Daten in einem neutralen Format, das von allen beteiligten Partnern verstanden wird. Jeder Transfer-Syntax muss ein *Object Identifier* zugeordnet sein.

### **Transport-Selektor**

Der Transport-Selektor identifiziert im lokalen System einen *Dienstzugriffspunkt* zur Transportschicht des *OSI-Referenzmodells*.

### **Transportsystem-Anwendung**

Anwendung, die direkt auf einer Transportsystem-Schnittstelle wie z.B. CMX, DCAM oder Socket aufsetzt. Für den Anschluss von Transportsystem-Anwendungen muss bei der *Konfiguration* als Partnertyp APPLI oder SOCKET angegeben werden. Eine Transportsystem-Anwendung kann nicht in eine *Verteilte Transaktion* eingebunden werden.

### **TS-Anwendung**

siehe *Transportsystem-Anwendung*.

### **Typisierter Puffer (XATMI)**

Puffer für den Austausch von typisierten und strukturierten Daten zwischen Kommunikationspartnern. Durch diese typisierten Puffer ist die Struktur der ausgetauschten Daten den Partnern implizit bekannt.

### **UPIC**

Trägersystem für UTM-Clients. UPIC steht für Universal Programming Interface for Communication. Die Kommunikation mit der UTM-Anwendung erfolgt über das UPIC-Protokoll.

### **UPIC-Client**

Bezeichnung für UTM-Clients mit Trägersystem UPIC und *JConnect-Clients*.

### **UPIC-Protokoll**

Protokoll für die Client-Server-Kommunikation mit *UTM-Anwendungen*. Das UPIC-Protokoll wird von *UPIC-Clients* und von *JConnect-Clients* verwendet.

### **UPIC Analyzer**

Komponente zur Analyse der mit *UPIC Capture* mitgeschnittenen UPIC-Kommunikation. Dieser Schritt dient dazu, den Mitschnitt für das Abspielen mit *UPIC Replay* aufzubereiten.

### **UPIC Capture**

Mitschneiden der Kommunikation zwischen UPIC-Clients und UTM-Anwendungen, um sie zu einem späteren Zeitpunkt abspielen zu können (*UPIC Replay*).

### **UPIC Replay**

Komponente zum Abspielen der mit *UPIC Capture* mitgeschnittenen und mit *UPIC Analyzer* aufbereiteten UPIC-Kommunikation.

### **USER-Queue**

*Message Queue*, die openUTM jeder Benutzerkennung zur Verfügung stellt. Eine USER-Queue zählt zu den *Service-gesteuerten Queues* und ist immer der jeweiligen Benutzerkennung zugeordnet. Der Zugriff von fremden UTM-Benutzern auf die eigene USER-Queue kann eingeschränkt werden.

### **User-spezifischer Langzeitspeicher/ULS**

*Sekundär Speicher*, der einer *Benutzerkennung*, einer *Session* oder einer *Association* zugeordnet ist und über das Anwendungsende hinaus erhalten bleibt.

### **USLOG-Datei**

siehe *Benutzer-Protokolldatei*.

### **UTM-Anwendung**

Eine UTM-Anwendung stellt *Services* zur Verfügung, die Aufträge von *Clients* oder anderen Anwendungen bearbeiten. openUTM übernimmt dabei u.a. die Transaktionssicherung und das Management der Kommunikations- und Systemressourcen. Technisch gesehen ist eine UTM-Anwendung eine Prozessgruppe, die zur Laufzeit eine logische Server-Einheit bildet.

### **UTM-Client**

siehe *Client*.

### UTM-Cluster-Anwendung

*UTM-Anwendung*, die für den Einsatz in einem *Cluster* generiert ist und die man logisch als **eine** Anwendung betrachten kann.

Physikalisch gesehen besteht eine UTM-Cluster-Anwendung aus mehreren, identisch generierten UTM-Anwendungen, die auf den einzelnen *Knoten* laufen.

### UTM-Cluster-Dateien

Oberbegriff für alle Dateien, die für den Ablauf einer UTM-Cluster-Anwendung benötigt werden. Dazu gehören folgende Dateien:

- *Cluster-Konfigurationsdatei*
- *Cluster-User-Datei*
- Dateien des *Cluster-Pagepool*
- *Cluster-GSSB-Datei*
- *Cluster-ULS-Datei*
- Dateien des *Cluster-Administrations-Journals\**
- *Cluster-Lock-Datei\**
- Lock-Datei zur Start-Serialisierung\* (nur bei Unix-, Linux- und Windows-Systemen)

Die mit \* gekennzeichneten Dateien werden beim Start der ersten Knoten-Anwendung angelegt, alle anderen Dateien werden bei der Generierung mit KDCDEF erzeugt.

### UTM-D

siehe *openUTM-D*.

### UTM-Datenstation

Begriff ersetzt durch *LTERM-Partner*.

### UTM-F

UTM-Anwendungen können als UTM-F-Anwendungen (UTM-Fast) generiert werden. Bei UTM-F wird zugunsten der Performance auf Platteneingaben/-ausgaben verzichtet, mit denen bei *UTM-S* die Sicherung von Benutzer- und Transaktionsdaten durchgeführt wird. Gesichert werden lediglich Änderungen der Verwaltungsdaten.

In UTM-Cluster-Anwendungen, die als UTM-F-Anwendung generiert sind (APPLIMODE=FAST), werden Cluster-weit gültige Anwenderdaten auch gesichert. Dabei werden GSSB- und ULS-Daten genauso behandelt wie in UTM-Cluster-Anwendungen, die mit UTM-S generiert sind. Vorgangs-Daten von Benutzern mit RESTART=YES werden jedoch nur beim Abmelden des Benutzers anstatt bei jedem Transaktionsende geschrieben.

### UTM-Generierung

*Statische Konfiguration* einer *UTM-Anwendung* mit dem UTM-Tool *KDCDEF* und Erzeugen des *Anwendungsprogramms*.



### UTM-gesteuerte Queues

Message Queues, bei denen der Abruf und die Weiterverarbeitung der Nachrichten vollständig durch openUTM gesteuert werden. Siehe auch *Asynchron-Auftrag*, *Hintergrund-Auftrag* und *Asynchron-Nachricht*.

### UTM-S

Bei UTM-S-Anwendungen sichert openUTM neben den Verwaltungsdaten auch alle Benutzerdaten über ein Anwendungsende und einen Systemausfall hinaus. Außerdem garantiert UTM-S bei allen Störungen die Sicherheit und Konsistenz der Anwendungsdaten. Im Standardfall werden UTM-Anwendungen als UTM-S-Anwendungen (UTM-Secure) generiert.

### UTM-SAT-Administration (BS2000-Systeme)

Durch die UTM-SAT-Administration wird gesteuert, welche sicherheitsrelevanten UTM-Ereignisse, die im Betrieb der *UTM-Anwendung* auftreten, von *SAT* protokolliert werden sollen. Für die UTM-SAT-Administration wird eine besondere Berechtigung benötigt.

### UTM-Seite

Ist eine Speichereinheit, die entweder 2K, 4K oder 8K umfasst. In *stand-alone UTM-Anwendungen* kann die Größe einer UTM-Seite bei der Generierung der UTM-Anwendung auf 2K, 4K oder 8K gesetzt werden. In einer *UTM-Cluster-Anwendung* ist die Größe einer UTM-Seite immer 4K oder 8K. *Pagepool* und Wiederanlauf-Bereich der KDCFILE sowie *UTM-Cluster-Dateien* werden in Einheiten der Größe einer UTM-Seite unterteilt.

### UTM-System-Prozess

UTM-Prozess, der zusätzlich zu den per Startparameter angegebenen Prozessen gestartet wird und nur ausgewählte Aufträge bearbeitet. UTM-System-Prozesse dienen dazu, eine UTM-Anwendung auch bei sehr hoher Last reaktionsfähig zu halten.

### UTM-Tool

Programm, das zusammen mit openUTM zur Verfügung gestellt und für bestimmte UTM-spezifische Aufgaben benötigt wird (z.B. zum Konfigurieren).

### utmpfad (Unix-, Linux- und Windows-Systeme)

Das Dateiverzeichnis unter dem die Komponenten von openUTM installiert sind, wird in diesem Handbuch als *utmpfad* bezeichnet. Um einen korrekten Ablauf von openUTM zu garantieren, muss die Umgebungsvariable UTMPATH auf den Wert von *utmpfad* gesetzt werden. Auf Unix- und Linux-Systemen müssen Sie UTMPATH vor dem Starten einer UTM-Anwendung setzen. Auf Windows-Systemen wird UTMPATH passend zu der zuletzt installierten UTM-Version gesetzt.

### Verarbeitungsschritt

Ein Verarbeitungsschritt beginnt mit dem Empfangen einer *Dialog-Nachricht*, die von einem *Client* oder einer anderen Server-Anwendung an die *UTM-Anwendung* gesendet wird. Der Verarbeitungsschritt endet entweder mit dem Senden einer Antwort und beendet damit auch den *Dialog-Schritt* oder er endet mit dem Senden einer Dialog-Nachricht an einen Dritten.

### Verbindungs-Benutzerkennung

Benutzerkennung, unter der eine *TS-Anwendung* oder ein *UPIC-Client* direkt nach dem Verbindungsaufbau bei der *UTM-Anwendung* angemeldet wird.

Abhängig von der Generierung des Clients (= LTERM-Partner) gilt:

- Die Verbindungs-Benutzerkennung ist gleich dem USER der LTERM-Anweisung (explizite Verbindungs-Benutzerkennung). Eine explizite Verbindungs-Benutzerkennung muss mit einer USER-Anweisung generiert sein und kann nicht als "echte" *Benutzerkennung* verwendet werden.
- Die Verbindungs-Benutzerkennung ist gleich dem LTERM-Partner (implizite Verbindungs-Benutzerkennung), wenn bei der LTERM-Anweisung kein USER angegeben wurde oder wenn ein LTERM-Pool generiert wurde.

In einer *UTM-Cluster-Anwendung* ist der Vorgang einer Verbindungs-Benutzerkennung (RESTART=YES bei LTERM oder USER) an die Verbindung gebunden und damit Knoten-lokal.

Eine Verbindungs-Benutzerkennung, die mit RESTART=YES generiert ist, kann in jeder *Knoten-Anwendung* einen eigenen Vorgang haben.

### Verbindungsbandel

siehe *LTERM-Bündel*.

### Verschlüsselungsstufe

Die Verschlüsselungsstufe legt fest, ob und inwieweit ein Client Nachrichten und Passwort verschlüsseln muss.

### Verteilte Transaktion

*Transaktion*, die sich über mehr als eine Anwendung erstreckt und in mehreren (Teil)-Transaktionen in verteilten Systemen ausgeführt wird.

### Verteilte Transaktionsverarbeitung

*Verteilte Verarbeitung* mit *verteilten Transaktionen*.

### Verteilte Verarbeitung

Bearbeitung von *Dialog-Aufträgen* durch mehrere Anwendungen oder Übermittlung von *Hintergrundaufträgen* an eine andere Anwendung. Für die verteilte Verarbeitung werden die höheren Kommunikationsprotokolle *LU6.1* und *OSITP* verwendet. Über openUTM-LU62 ist verteilte Verarbeitung auch mit LU6.2 Partnern möglich. Man unterscheidet verteilte Verarbeitung mit *verteilten*

*Transaktionen* (Anwendungs-übergreifende Transaktionssicherung) und verteilte Verarbeitung ohne verteilte Transaktionen (nur lokale Transaktionssicherung). Die verteilte Verarbeitung wird auch Server-Server-Kommunikation genannt.

### **Vorgang (KDCS)**

Ein Vorgang dient zur Bearbeitung eines *Auftrags* in einer *UTM-Anwendung*. Er setzt sich aus einer oder mehreren *Transaktionen* zusammen. Die erste Transaktion wird über den *Vorgangs-TAC* aufgerufen. Es gibt *Dialog-Vorgänge* und *Asynchron-Vorgänge*. openUTM stellt den Teilprogrammen eines Vorgangs gemeinsame Datenbereiche zur Verfügung. Anstelle des Begriffs Vorgang wird häufig auch der allgemeinere Begriff *Service* gebraucht.

### **Vorgangs-Kellerung (KDCS)**

Ein Terminal-Benutzer kann einen laufenden *Dialog-Vorgang* unterbrechen und einen neuen Dialog-Vorgang einschieben. Nach Beendigung des eingeschobenen *Vorgangs* wird der unterbrochene Vorgang fortgesetzt.

### **Vorgangs-Kettung (KDCS)**

Bei Vorgangs-Kettung wird nach Beendigung eines *Dialog-Vorgangs* ohne Angabe einer *Dialog-Nachricht* ein Folgevorgang gestartet.

### **Vorgangs-TAC (KDCS)**

Transaktionscode, mit dem ein *Vorgang* gestartet wird.

### **Vorgangs-Wiederanlauf (KDCS)**

Wird ein Vorgang unterbrochen, z.B. infolge Abmeldens des Terminal-Benutzers oder Beendigung der *UTM-Anwendung*, führt openUTM einen Vorgangs-Wiederanlauf durch. Ein *Asynchron-Vorgang* wird neu gestartet oder beim zuletzt erreichten *Sicherungspunkt* fortgesetzt, ein *Dialog-Vorgang* wird beim zuletzt erreichten Sicherungspunkt fortgesetzt. Für den Terminal-Benutzer wird der Vorgangs-Wiederanlauf eines Dialog-Vorgangs als *Bildschirm-Wiederanlauf* sichtbar, sofern am letzten Sicherungspunkt eine Dialog-Nachricht an den Terminal-Benutzer gesendet wurde.

### **Warmstart**

Start einer *UTM-S-Anwendung* nach einer vorhergehenden abnormalen Beendigung. Dabei wird die *Anwendungsinformation* auf den zuletzt erreichten konsistenten Zustand gesetzt. Unterbrochene *Dialog-Vorgänge* werden dabei auf den zuletzt erreichten *Sicherungspunkt* zurückgesetzt, so dass die Verar-

beitung an dieser Stelle wieder konsistent aufgenommen werden kann (*Vorgangs-Wiederanlauf*). Unterbrochene *Asynchron-Vorgänge* werden zurückgesetzt und neu gestartet oder beim zuletzt erreichten *Sicherungspunkt* fortgesetzt. Bei UTM-F-Anwendungen werden beim Start nach einer vorhergehenden abnormalen Beendigung lediglich die dynamisch geänderten Konfigurationsdaten auf den zuletzt erreichten konsistenten Zustand gesetzt. In UTM-Cluster-Anwendungen werden die globalen Sperren auf GSSB und ULS, die bei der abnormalen Beendigung von dieser Knoten-Anwendung gehalten wurden, aufgehoben. Außerdem werden Benutzer, die zum Zeitpunkt der abnormalen Beendigung an dieser Knoten-Anwendung angemeldet waren, abgemeldet.

### **Web Service**

Anwendung, die auf einem Web-Server läuft und über eine standardisierte und programmatische Schnittstelle (öffentlich) verfügbar ist. Die Web Services-Technologie ermöglicht es, UTM-Teilprogramme für moderne Web-Client-Anwendungen verfügbar zu machen, unabhängig davon, in welcher Programmiersprache sie entwickelt wurden.

### **WebAdmin**

Web-basiertes Tool zur Administration von openUTM-Anwendungen über Web-Browser. WebAdmin enthält neben dem kompletten Funktionsumfang der *Programmschnittstelle zur Administration* noch zusätzliche Funktionen.

### **Wiederanlauf**

siehe *Bildschirm-Wiederanlauf*,  
siehe *Vorgangs-Wiederanlauf*.

### **WinAdmin**

Java-basiertes Tool zur Administration von openUTM-Anwendungen über eine grafische Oberfläche. WinAdmin enthält neben dem kompletten Funktionsumfang der *Programmschnittstelle zur Administration* noch zusätzliche Funktionen.

### **Workload Capture & Replay**

Programmfamilie zur Simulation von Lastsituationen, bestehend aus den Haupt-Komponenten *UPIC Capture*, *UPIC Analyzer* und *Upic Replay* und auf Unix-, Linux- und Windows-Systemen dem Dienstprogramm *kdcsort*. Mit Workload Capture & Replay lassen sich UPIC-Sessions mit UTM-Anwendungen aufzeichnen, analysieren und mit veränderten Lastparametern wieder abspielen.

### **Workprozess (Unix-, Linux- und Windows-Systeme)**

Prozess, in dem die *Services* der *UTM-Anwendung* ablaufen.

## WS4UTM

WS4UTM (**WebServices for openUTM**) ermöglicht es Ihnen, auf komfortable Weise einen Service einer UTM-Anwendung als Web Service zur Verfügung zu stellen.

## XATMI

XATMI (X/Open Application Transaction Manager Interface) ist eine von X/Open standardisierte Programmschnittstelle für die Programm-Programm-Kommunikation in offenen Netzen.

Das in openUTM implementierte XATMI genügt der XATMI CAE Specification von X/Open. Die Schnittstelle steht in COBOL und C zur Verfügung. XATMI in openUTM kann über die Protokolle *OSI TP*, *LU6.1* und *UPIC* kommunizieren.

## XHCS (BS2000-Systeme)

XHCS (Extended Host Code Support) ist ein BS2000-Softwareprodukt für die Unterstützung internationaler Zeichensätze.

## XML

XML (eXtensible Markup Language) ist eine vom W3C (WWW-Konsortium) genormte Metasprache, in der Austauschformate für Daten und zugehörige Informationen definiert werden können.

## Zeitgesteuerter Auftrag

*Auftrag*, der von openUTM bis zu einem definierten Zeitpunkt in einer *Message Queue* zwischengespeichert und dann an den Empfänger weitergeleitet wird. Empfänger kann sein: ein *Asynchron-Vorgang* der selben Anwendung, eine *TAC-Queue*, eine Partner-Anwendung, ein Terminal oder ein Drucker. Zeitgesteuerte Aufträge können nur von *KDCS-Teilprogrammen* erteilt werden.

## Zugangskontrolle

Prüfung durch openUTM, ob eine bestimmte *Benutzererkennung* berechtigt ist, mit der *UTM-Anwendung* zu arbeiten. Die Berechtigungsprüfung entfällt, wenn die UTM-Anwendung ohne Benutzerkennungen generiert wurde.

## Zugriffskontrolle

Prüfung durch openUTM, ob der Kommunikationspartner berechtigt ist, auf ein bestimmtes Objekt der Anwendung zuzugreifen. Die Zugriffsrechte werden als Bestandteil der Konfiguration festgelegt.

## Zugriffspunkt

siehe *Dienstzugriffspunkt*.



---

# Abkürzungen

ACSE	Association Control Service Element
AEQ	Application Entity Qualifier
AES	Advanced Encryption Standard
AET	Application Entity Title
APT	Application Process Title
ASCII	American Standard Code for Information Interchange
ASE	Application Service Element
Axis	Apache eXtensible Interaction System
BCAM	Basic Communication Access Method
BER	Basic Encoding Rules
BLS	Binder-Lader-Starter (BS2000-Systeme)
CCP	Communication Control Program
CCR	Commitment, Concurrency and Recovery
CCS	Codierter Zeichensatz (Coded Character Set)
CCSN	Name des codierten Zeichensatzes (Coded Character Set Name)
CICS	Customer Information Control System (IBM)
CID	Control Identification
CMX	Communication Manager in Unix-, Linux- und Windows-Systemen
COM	Component Object Model
CPI-C	Common Programming Interface for Communication
CRM	Communication Resource Manager
CRTE	Common Runtime Environment (BS2000-Systeme)
DB	Datenbank
DC	Data Communication
DCAM	Data Communication Access Method

DES	Data Encryption Standard
DLM	Distributed Lock Manager (BS2000-Systeme)
DMS	Data Management System
DNS	Domain Name Service
DSS	Datensichtstation (=Terminal)
DTD	Document Type Definition
DTP	Distributed Transaction Processing
DVS	Datenverwaltungssystem
EBCDIC	Extended Binary-Coded Decimal Interchange Code
EJB	Enterprise JavaBeans <sup>TM</sup>
FGG	File Generation Group
FHS	Format Handling System
FT	File Transfer
GSSB	Globaler Sekundärer Speicherbereich
HIPLEX®	Highly Integrated System Complex (BS2000-Systeme)
HLL	High-Level Language
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IFG	Interaktiver Format-Generator
ILCS	Inter Language Communication Services (BS2000-Systeme)
IMS	Information Management System (IBM)
IPC	Inter-Process-Communication
IRV	Internationale Referenzversion
ISO	International Organization for Standardization
Java EE	Java Platform, Enterprise Edition
JCA	Java EE Connector Architecture
JDK	Java Development Kit
KAA	KDCS Application Area
KB	Kommunikationsbereich
KBPROG	KB-Programmbereich
KDCADMI	KDC Administration Interface
KDCS	Kompatible Datenkommunikationsschnittstelle



---

KTA	KDCS Task Area
LAN	Local Area Network
LCF	Local Configuration File
LLM	Link and Load Module (BS2000-Systeme)
LSSB	Lokaler Sekundärer Speicherbereich
LU	Logical Unit
MQ	Message Queuing
MSCF	Multiple System Control Facility (BS2000-Systeme)
NB	Nachrichtenbereich
NEA	Netzwerkarchitektur bei BS2000-Systemen
NFS	Network File System/Service
NLS	Unterstützung der Landessprache (Native Language Support)
OLTP	Online Transaction Processing
OML	Object Modul Library
OSI	Open System Interconnection
OSI TP	Open System Interconnection Transaction Processing
OSS	OSI Session Service
PCMX	Portable Communication Manager
PID	Prozess-Identifikation
PIN	Persönliche Identifikationsnummer
PLU	Primary Logical Unit
PTC	Prepare to commit
RAV	Rechenzentrums-Abrechnungs-Verfahren
RDF	Resource Definition File
RM	Resource Manager
RSA	Encryption-Algorithmus nach Rivest, Shamir, Adleman
RSO	Remote SPOOL Output (BS2000-Systeme)
RTS	Runtime System (Laufzeitsystem)
SAT	Security Audit Trail (BS2000-Systeme)
SECOS	Security Control System
SEM	SE Manager
SGML	Standard Generalized Markup Language

SLU	Secondary Logical Unit
SM2	Software Monitor 2
SNA	Systems Network Architecture
SOA	Service-oriented Architecture
SOAP	Simple Object Access Protocol
SPAB	Standard Primärer Arbeitsbereich
SQL	Structured Query Language
SSB	Sekundärer Speicherbereich
SSO	Single-Sign-On
TAC	Transaktionscode
TCEP	Transport Connection End Point
TCP/IP	Transport Control Protocol / Internet Protocol
TIAM	Terminal Interactive Access Method
TLS	Terminal-spezifischer Langzeitspeicher
TM	Transaction Manager
TNS	Transport Name Service
TP	Transaction Processing (Transaktions-Betrieb)
TPR	Task privileged (privilegierter Funktionszustand des BS2000-Systems)
TPSU	Transaction Protocol Service User
TSAP	Transport Service Access Point
TSN	Task Sequence Number
TU	Task user (nicht privilegierter Funktionszustand des BS2000-Systems)
TX	Transaction Demarcation (X/Open)
UDDI	Universal Description, Discovery and Integration
UDS	Universelles Datenbanksystem
UDT	Unstructured Data Transfer
ULS	User-spezifischer Langzeitspeicher
UPIC	Universal Programming Interface for Communication
USP	UTM-Socket-Protokoll
UTM	Universeller Transaktionsmonitor
UTM-D	UTM-Funktionen für verteilte Verarbeitung („Distributed“)
UTM-F	Schnelle UTM-Variante („Fast“)
UTM-S	UTM-Sicherheitsvariante

UTM-XML	XML-Schnittstelle von openUTM
VGID	Vorgangs-Identifikation
VTSU	Virtual Terminal Support
VTV	Verteilte Transaktionsverarbeitung
VV	Verteilte Verarbeitung
WAN	Wide Area Network
WS4UTM	WebServices for openUTM
WSDD	Web Service Deployment Descriptor
WSDL	Web Services Description Language
XA	X/Open Access Interface (Schnittstelle von X/Open zum Zugriff auf Resource Manager)
XAP	X/OPEN ACSE/Presentation programming interface
XAP-TP	X/OPEN ACSE/Presentation programming interface Transaction Processing extension
XATMI	X/Open Application Transaction Manager Interface
XCS	Cross Coupled System
XHCS	eXtended Host Code Support
XML	eXtensible Markup Language



---

# Literatur

Die Handbücher finden Sie im Internet unter <http://manuals.ts.fujitsu.com>.

## Dokumentation zu openUTM

### **openUTM**

#### **Konzepte und Funktionen**

Benutzerhandbuch

### **openUTM**

#### **Anwendungen programmieren mit KDCS für COBOL, C und C++**

Basishandbuch

### **openUTM**

#### **Anwendungen generieren**

Benutzerhandbuch

### **openUTM**

#### **Einsatz von UTM-Anwendungen auf BS2000-Systemen**

Benutzerhandbuch

### **openUTM**

#### **Einsatz von UTM-Anwendungen auf Unix-, Linux- und Windows-Systemen**

Benutzerhandbuch

### **openUTM**

#### **Anwendungen administrieren**

Benutzerhandbuch

### **openUTM**

#### **Meldungen, Test und Diagnose auf BS2000-Systemen**

Benutzerhandbuch

**openUTM**

**Meldungen, Test und Diagnose auf Unix-, Linux- und Windows-Systemen**

Benutzerhandbuch

**openUTM**

**Anwendungen erstellen mit X/Open-Schnittstellen**

Benutzerhandbuch

**openUTM**

**XML für openUTM**

**openUTM-Client (Unix-Systeme)**

**für Trägersystem OpenCPIC**

**Client-Server-Kommunikation mit openUTM**

Benutzerhandbuch

**openUTM-Client**

**für Trägersystem UPIC**

**Client-Server-Kommunikation mit openUTM**

Benutzerhandbuch

**openUTM WinAdmin**

**Grafischer Administrationsarbeitsplatz für openUTM**

Beschreibung und Online-Hilfe

**openUTM WebAdmin**

**Web-Oberfläche zur Administration von openUTM**

Beschreibung und Online-Hilfe

**openUTM, openUTM-LU62**

**Verteilte Transaktionsverarbeitung**

**zwischen openUTM und CICS-, IMS- und LU6.2-Anwendungen**

Benutzerhandbuch

**openUTM (BS2000)**

**Anwendungen programmieren mit KDCS für Assembler**

Ergänzung zum Basishandbuch

**openUTM (BS2000)**

**Anwendungen programmieren mit KDCS für Fortran**

Ergänzung zum Basishandbuch

**openUTM (BS2000)**  
**Anwendungen programmieren mit KDCS für Pascal-XT**  
Ergänzung zum Basishandbuch

**openUTM (BS2000)**  
**Anwendungen programmieren mit KDCS für PL/I**  
Ergänzung zum Basishandbuch

**WS4UTM (Unix- und Windows-Systeme)**  
**Web-Services für openUTM**

## **Dokumentation zum openSEAS-Produktumfeld**

**BeanConnect**  
Benutzerhandbuch

**openUTM-JConnect**  
**Verbindung von Java-Clients zu openUTM**  
Benutzerdokumentation und Java-Docs

**WebTransactions**  
**Konzepte und Funktionen**

**WebTransactions**  
**Template-Sprache**

**WebTransactions**  
**Anschluss an openUTM-Anwendungen über UPIC**

**WebTransactions**  
**Anschluss an MVS-Anwendungen**

**WebTransactions**  
**Anschluss an OSD-Anwendungen**

## Dokumentation zum BS2000-Umfeld

**AID**  
**Advanced Interactive Debugger**  
**Basishandbuch**  
Benutzerhandbuch

**AID**  
**Advanced Interactive Debugger**  
**Testen von COBOL-Programmen**  
Benutzerhandbuch

**AID**  
**Advanced Interactive Debugger**  
**Testen von C/C++-Programmen**  
Benutzerhandbuch

**BCAM**  
**BCAM Band 1/2**  
Benutzerhandbuch

**BINDER**  
Benutzerhandbuch

**BS2000 OSD/BC**  
**Kommandos Band 1-7**  
Benutzerhandbuch

**BS2000 OSD/BC**  
**Makroaufrufe an den Ablaufteil**  
Benutzerhandbuch

**BS2IDE**  
Eclipse-based Integrated Development Environment for BS2000  
User Guide and Installation Guide  
Webseite: <https://bs2000.ts.fujitsu.com/bs2ide/>

**BLSSERV**  
**Bindelader-Starter in BS2000/OSD**  
Benutzerhandbuch



**DCAM**

**COBOL-Aufrufe**

Benutzerhandbuch

**DCAM**

**Makroaufrufe**

Benutzerhandbuch

**DCAM**

**Programmschnittstellen**

Beschreibung

**FHS**

**Formatierungssystem für openUTM, TIAM, DCAM**

Benutzerhandbuch

**IFG für FHS**

Benutzerhandbuch

**HIPLEX AF**

**Hochverfügbarkeit von Anwendungen in BS2000/OSD**

Produkt-handbuch

**HIPLEX MSCF**

**BS2000-Rechner im Verbund**

Benutzerhandbuch

**IMON**

**Installationsmonitor**

Benutzerhandbuch

**LMS**

**SDF-Format**

Benutzerhandbuch

**MT9750 (MS Windows)**

**9750-Emulation unter Windows**

Produkt-handbuch

**OMNIS/OMNIS-MENU**

**Funktionen und Kommandos**

Benutzerhandbuch

**OMNIS/OMNIS-MENU**

**Administration und Programmierung**

Benutzerhandbuch

**OSS (BS2000)**

**OSI Session Service**

User Guide

**openSM2**

**Software Monitor**

Benutzerhandbuch

**RSO**

**Remote SPOOL Output**

Benutzerhandbuch

**SECOS**

**Security Control System**

Benutzerhandbuch

**SECOS**

**Security Control System**

Tabellenheft

**SESAM/SQL**

**Datenbankbetrieb**

Benutzerhandbuch

**TIAM**

Benutzerhandbuch

**UDS/SQL**

**Datenbankbetrieb**

Benutzerhandbuch

**Unicode im BS2000/OSD**

Übersichtshandbuch

**VTSU**

**Virtual Terminal Support**

Benutzerhandbuch

**XHCS****8-bit-Code- und Unicode-Unterstützung im BS2000/OSD**

Benutzerhandbuch

**Dokumentation zum Umfeld von Unix-, Linux- und Windows-Systemen****CMX V6.0 (Unix-Systeme)****Betrieb und Administration**

Benutzerhandbuch

**CMX V6.0**

CMX-Anwendungen programmieren

Programmierhandbuch

**OSS (UNIX)****OSI Session Service**

User Guide

PRIMECLUSTER<sup>TM</sup>**Konzept (Solaris, Linux)**

Benutzerhandbuch

**openSM2**

Die Dokumentation zu openSM2 wird in Form von ausführlichen Online-Hilfen bereitgestellt, die mit dem Produkt ausgeliefert werden.

## Sonstige Literatur

### **CPI-C**

X/Open CAE Specification  
Distributed Transaction Processing:  
The CPI-C Specification, Version 2  
ISBN 1 85912 135 7

### **Reference Model**

X/Open Guide  
Distributed Transaction Processing:  
Reference Model, Version 2  
ISBN 1 85912 019 9

### **TX**

X/Open CAE Specification  
Distributed Transaction Processing:  
The TX (Transaction Demarcation) Specification  
ISBN 1 85912 094 6

### **XATMI**

X/Open CAE Specification  
Distributed Transaction Processing  
The XATMI Specification  
ISBN 1 85912 130 6

### **XML**

Spezifikation des W3C (www – Konsortium)  
Webseite: <http://www.w3.org/XML>

---

# Stichwörter

!

KDCDUMP-Anweisung 63

!!

KDCDUMP-Anweisung 64

## A

Abnormal beenden

Anwendung 177

Vorgang 42

Accounting Area 85

ADMI-Trace 54

Administration DIAGAREA 84, 104

Administration USERAREA 84, 108

Administrations-Journal 463

Administrations-Journal-Dateien

Meldung K190 271

Administrationskommando

Testmodus einschalten 44

Administrationsprogramm

Message-Dump-Funktion ausschalten 46

Message-Dump-Funktion einschalten 45

Adresse NULL 43

AFIND

KDCDUMP-Anweisung 64

AID

TEST-OPTION 60

Testhilfe 30

Analyse der Register 115

Anmelde-Vorgang

Meldungen 149

Announcements

Trace 48

Anwendung

UTM-Meldung an andere 122

Anwendungsprogramm, austauschen 33

Area Table 83

Aufbereiten

SYSLOG-Datei 146

Aufbereiteter Dump 81

Aufbereitungstool

KDCCSYSL 144

KDCPSYSL 146

Auftragsschalter

KDCDUMP 62

Ausdrucken

Dump-Datei 61

Ausfall Knoten-Anwendung

Meldung K190 279

Ausgabe-Nachrichten

Trace 49

Ausgaben von KDCMMOD 133

Austausch

Anwendungsprogramm 33

von Programmteilen 30

Auswerten

BCAM-Trace 51

## B

BCAM-INFO-Word 182

BCAM-Parameterblöcke 48

BCAM-Trace 48

auswerten 51

ein-/ausschalten 50

Installationshinweise 49

Bedienplatz 120

Beenden

KDCDUMP 67

KDCMMOD 134

Benutzer-spezifische Meldungsziele 120, 123

- Bildschirmfunktionen 91
- Blättern
  - KDCDUMP 64
- Blockgröße
  - Länge der Trace-Sätze 50
- BUFFER-LENGTH
  - Anzahl Blöcke 50
- C**
- C-String 63
- CDUMP-Aufrufe 398
- Cluster-Administrations-Journal 463
- Cluster-GSSB-Datei
  - Meldung K190 276
- Cluster-Konfigurationsdatei
  - Meldung K190 262
- Cluster-Pagepool
  - Meldung K190 272
- Cluster-ULS-Datei
  - Meldung K190 274, 275
- Cluster-User-Datei
  - Meldung K190 264
- CMX-Returncode 301
- Codes der CDUMP-Aufrufe von UTM 398
- Connection-Letters
  - Trace 49
- CONSOLE (Meldungsziel) 120
- Context Area 83, 115
- D**
- Darstellungsmittel 26
- Dateigenerationsgruppe 56
  - SYSLOG 143
- Datensichtstation (Meldungsziel) 120
- Datenstruktur
  - Meldungskopf 148
- Datenverschlüsselung 234
- Datum
  - Meldungen 121
- DB Diagarea 85
- DB Error Code 103
- Db Info Application 85
- DB Info Program Unit 85
- DB Operations Code 102
- DB Summary Info 85
- DB Table 85
- DB User Area 85
- DB-Connection-Modul 398
- Dezimal-Eingabe
  - KDCDUMP 63
- Diagnose
  - SYSLOG 143
  - Unterlagen 47
- Dienstprogramm
  - KDCDUMP 59
  - KDCMMOD 132
  - KDCMTXT 125
- Dokumentation, Wegweiser 10
- DUMP
  - KDCDUMP-Anweisung 66
- Dump
  - aufbereiten 59
  - Tabelle anzeigen 78
- Dump-Beispiel FH01 112
- DUMP-CONTENT
  - Startparameter 58
- Dump-Dateien 56
  - ausdrucken 61
- Dump-Fehlercode 177
  - Grp 262
  - Gruppen 407
- Dump-Fehlercodes 407
- DUMP-USERID
  - Startparameter 57
- E**
- EDT
  - KDCDUMP-Anweisung 67
- Eigener Meldungsmodul 132
- Ein-/Ausschalten
  - BCAM-Trace 50
- Eingabe-Nachrichten
  - Trace 49
- Einträge in der Meldungsdatei 353
- ENCRYPTION\_LEVEL 234
- END
  - KDCDUMP-Anweisung 67
- ENTER-PROC 85

- ETPNDS-TU-LIST 85  
 Event-Exit SHUT beim Testen 33  
 Exit Table 83  
 Extended Line Mode 353
- F**  
 Fehler  
   im INPUT-Exit 43  
 Fehlercode  
   UTM-Dump 177  
 Fehlerdiagnose 40  
 Fehlertexte in der DIAGAREA 94  
 Fehlerunterlagen  
   erstellen 47  
   UTM-Cluster-Anwendung 47  
 FGG  
   KDCDUMP-Anweisung 68  
 FH01 112  
 FHS-Rückkehrcode 112  
 FILE  
   KDCDUMP-Anweisung 70  
 Folgetask  
   beim Testen 31  
 FORM USER Area 84  
 Formatierungsfehler 41, 112  
 FORMUSER Buffer 84
- G**  
 Gestaltung der Meldungs Ausgabe 124  
 Grp  
   Dump-Fehlercode 262
- H**  
 HELP  
   KDCDUMP-Anweisung 72, 76  
 HLL User Area 84
- I**  
 INPUT-Exit 93  
 Inserts 117  
 Installation  
   BCAM-Trace 49  
 Interner Returncode KRCRDC 40, 371  
 IO Buffer 84
- K**  
 K-Meldungen 149, 353  
 K065-Meldung 146  
 K608-Meldung 146  
 KB 83, 109  
 KCMMSGx 144  
 KCRCCC 367  
 KRCRDC 40, 371  
 KDCADMI  
   Trace 54  
 KDCBTRC 51  
 KDCCSYSL 143, 144  
   aufrufen 144  
   Meldungen 147, 331  
   starten 144  
 KDCDEF  
   Meldungen 323  
 KDCDIAG  
   Message-Dump-Funktion ausschalten 46  
   Message-Dump-Funktion einschalten 45  
   Testmodus einschalten 44  
 KDCDUMP 59  
   AFIND 64  
   Anweisungen 62  
   beenden 67  
   blättern 64  
   DUMP 66  
   EDT 67  
   FGG 68  
   FILE 70  
   HELP 72, 76  
   LIST 73  
   Meldungen 334  
   REPFIL 75  
   SH/SYS 77  
   starten 60  
   SYSLST 77  
   Systemkommando 63  
   TABLE 78  
   unterbrechen 77  
 KDCMMOD 132  
   Meldungen 332  
 KDCMSGLT (LTERM-Partner) 123  
 KDCMSGUS (Benutzer) 123

KDCMTXT 125  
  Meldungen 332  
KDCPSYSL 144  
  Meldungen 147  
  starten 146  
KDCROOT im UTM-Dump 83  
KDCS-Returncode 367  
KDCUPD  
  Meldungen 338  
Knoten-Anwendung  
  Meldung K190 279  
Konsole 120

## L

Lademodule  
  im UTM-Dump 87  
Länge  
  Meldung in Systemzeile 140  
Länge der Trace-Sätze  
  zugehörige Blockgröße 50  
Längenbeschränkungen  
  Meldung 140  
Language-Connection-Module 398  
Library Table 83  
Linkname  
  Dumpdatei 70  
  KDCDUMP 60  
  Prefix 68  
Linkprefix  
  Dumpdateien 68  
LIST  
  KDCDUMP-Anweisung 73  
LMOD-Tabelle 87  
Load Contxt Table 83  
Load Module Table 83  
LOG-EXIT-MESSAGE-BUFFER 85  
LSD-Information 30

## M

Meldungen 121, 323  
  andere Sprachen 125  
  ändern 124  
  Anmelde-Vorgang 149  
  Aufbau 148

KDCCSYSL 147, 331  
KDCDUMP 334  
KDCMMOD 332  
KDCMTXT 332  
KDCPSYSL 147  
KDCUPD 338  
  Maximal-Länge 140  
  XAP-TP-Provider 295  
Meldungsdefinitionsdatei 119, 124  
Meldungskopf 121  
  Datenstruktur 148  
Meldungsmodul 119  
Meldungsnummer 117  
Meldungstext 117  
  übersetzen 124  
Meldungsziele 117, 120  
  Benutzer-spezifisch 120, 123  
Memory Pool Table 83  
Message Mod Table 83  
Message-Dump 44  
Message-Dump-Funktion  
  ausschalten 46  
  einschalten 44  
Metasyntax 26  
MPUT Buffer 84  
MSCF 465  
MSGTAC 120  
MSGTAC-Programm 123  
MSGTACT 123

## N

NULL 43

## O

openUTM-Anwendung, siehe UTM-Anwendung

## P

Parallelbetrieb  
  zweier Versionen 29  
Parameter des INPUT-Exits im Dump 93  
PARTNER (Meldungsziel) 120  
PCMX 14  
PMIO22 47  
Prefix Linkname 68



- Program Table 83
- Programmbeendigung
  - beim Testen 31
- Programmschnittstelle Administration
  - Trace 54
- Programmteile
  - austauschen 30
- Protokollierung in KDCDUMP 77
  
- R**
- Rahmendefinitionen 353
- Reduzierung der DUMP-Information 58
- REP-Datei 75
- REPFIL
  - KDCDUMP-Anweisung 75
- RESTART Buffer 84
- Returncode
  - Programmschnittstelle 40
- Returncodes des UTM-SVC 162
- Root gen by user 84
- ROOT Trace 84
- ROOTDATA 84
  
- S**
- SH
  - KDCDUMP-Anweisung 77
- SHUT beim Testen 33
- Socket-Partner
  - UTM-Meldung an 122
- SPAB 84
- Sprache der Meldungen 124
- Stand-alone UTM-Anwendung 8
- Start-Fehlercodes 164
- Starten
  - einer Anwendung 33
  - KDCSYSL 144
  - KDCDUMP 60
  - KDCDUMP über Linknamen 60
  - KDCMMOD 134
  - KDCPSYSL 146
- Startparameter
  - DUMP-CONTENT 58
  - DUMP-USERID 57
  - Testmodus einschalten 44
- STATION (Meldungsziel) 120
- Status
  - DB-Transaktion 102
- STRT-PAR-AREA 85
- STXIT-Ereignisse 396
- STXIT-Verhalten beim Testen 34
- Summary-Information 111
  - FGG-Anweisung 69
  - FILE-Anweisung 70
- SVC 134 162
- SYS
  - KDCDUMP 77
- SYSLINE (Meldungsziel) 120
  - Meldungslänge 140
- SYSLOG
  - Meldungen im virtuellen Speicher 145
- SYSLOG-Datei 143
  - aufbereiten 146
  - in SAM-Format umwandeln 144
- SYSLOG-FGG 143
- SYSLST
  - KDCDUMP-Anweisung 77
  - Meldungsziel 120
- SYSOUT (Meldungsziel) 120
- SYSPRC.UTM.065(SLOG-FGG)
  - SYSLOG-FGG auswerten 144
- SYSPRC.UTM.065(SYSLOG)
  - SYSLOG auswerten 144
- SYSTEM-PEND-ER 94
- System-Protokolldatei 120
- Systemkommando
  - KDCDUMP 63
- Systemmeldungsmodul 119
- Systemzeile
  - der Datensichtstation 120
  - Meldung 140
  
- T**
- Tabelle anzeigen im Dump 78
- Tabellenausschnitt von Dump ausgeben 73
- TABLE
  - KDCDUMP-Anweisung 78
- Table Descriptors 84
- TAM 85

### Task

beim Testen 31

TCB-ENTRY (COB1) 84

### Testen

Anwendung im Dialog 30

### Testmodus

einschalten 44

### Tool

KDCCSYSL 144

KDCPSYSL 146

### Trace 48

KDCADMI 54

TRACEREC 47

TRMA-Reason 177

TSKM 85

## U

Übersetzungslistings 47

Übertragungsfehler bei KDCUPD 338

### Uhrzeit

Meldungen 121

Unterbrechen KDCDUMP 77

### Unterdrücken

UTM-Dump 57

USER-DEST 120, 123

USP-Header 122

UTM 149

UTM Diagarea 83, 88

UTM-Cluster-Anwendung 8

Cluster-Administrations-Journal 463

Fehlerunterlagen 47

UTM-Cluster-Dateien

Meldung K190 278

### UTM-Dump

unterdrücken 57

UTM-Meldung 117

an andere Anwendung 122

an Socket-Anwendung 122

Aufbau 148

UTM-Meldungen 149

UTM-MSG-DATE 121

UTM-Protokolldatei SYSLOG 117, 143

UTM-SVC 162

## V

Variable Inserts 117

Verhalten im Fehlerfall 47

VGM Area 84

## W

Warmstartfehler 47

## X

X-String 63

XA Area 84

XAP-TP-Provider

Meldungen 295

## Z

Zeitangaben-Header 121

### Ziel

UTM-Meldung 120, 353