

SEPs sesam Administratorhandbuch

SEP AG 83629 Weyarn

17. August 2007

©2007 by SEP AG. Alle Rechte vorbehalten.

Jegliche Form der Reproduktion oder Verbreitung des Inhalts oder eines Teils des Inhalts dieses Handbuchs bedarf der vorherigen schriftlichen Zustimmung durch die Firma SEP AG. Bei der Erstellung dieser Publikation wurde hinsichtlich der Brauchbarkeit und Richtigkeit der enthaltenen Informationen größtmögliche Sorgfalt aufgewendet. Bitte berücksichtigen Sie, dass in dieser Publikation dennoch Irrtümer vorliegen können. Die SEP AG kann keine Garantien hinsichtlich der Richtigkeit der Erläuterungen oder bezüglich der Verwendung dieser Informationen geben.

Hinweis

Dieses Administrator-Handbuch baut auf das **SEPs**esam Benutzerhandbuch auf, d.h. die grundlegende Beschreibung der Funktionsweise ist dort zu finden und wird hier nicht wiederholt.

Dieses Handbuch soll dem Administrator als Anleitung für die Installation der Software, des Startaufrufs, der Konfiguration sowie der Aufrechterhaltung eines reibungslosen Ablaufs von **SEPs**esam dienen.

Dieses Dokument wird mit CVS verwaltet und hat folgende CVS-Id:

`$Id: admin_handbuch.tex,v 1.32 2007/01/25 08:41:44 sesam Exp $`

Inhaltsverzeichnis

1	Einführung	5
1.1	Datenmanagement	5
1.1.1	Datenverfügbarkeit	5
1.1.2	Datensicherheit	5
1.1.3	Datensicherung	6
1.1.4	Datenrücksicherung	6
1.1.5	Disaster Recovery	6
1.2	Objekte von SEPs sesam	6
2	Installation	9
2.1	Verzeichnisstruktur	9
2.2	sm_setup	10
2.3	Installation auf UNIX und LINUX	11
2.3.1	UNIX Backup Server	11
2.3.1.1	Software-Voraussetzungen	11
2.3.1.2	Installation	11
2.3.2	UNIX Klient	13
2.3.3	UNIX Tape Server	13
2.3.4	UNIX Oberfläche (GUI)	13
2.3.5	Deinstallation Server, Tapeserver, Klient, GUI-Klient	14
2.3.6	SEPs sesam Umgebung auf UNIX setzen	15
2.4	Installation auf Windows	17
2.4.1	Grundlegende Vorgehensweise	18
2.4.2	Windows Backup Server	18
2.4.3	Windows Klient	18
2.4.4	Windows Tape Server	18
2.4.5	Windows Oberfläche (GUI)	18
2.5	SEPs sesam Klient: Unterstützte Betriebssysteme	20
2.6	SEPs sesam Klient: Unterstützte Online Module	21

3	Die Programme des SEPsesam	23
3.1	Benutzeroberfläche GUI (Graphical User Interface)	23
3.2	SesamBackupClient - SBC	23
3.2.1	SBC Fehlercodes	24
3.2.2	Optionale Erweiterungen des SBC	28
3.3	Die Programme des Servers	29
3.3.1	Die Kernprogramme	29
3.3.1.1	sm_backup	29
3.3.1.2	sm_drive	30
3.3.1.3	sm_info	31
3.3.1.4	sm_init	31
3.3.1.5	sm_loader	32
3.3.1.6	sm_recover	33
3.3.1.7	sm_restore	33
3.3.1.8	sm_robot	34
3.3.1.9	sm_sms_interface	35
3.3.1.10	sm_sms_watch	38
3.3.1.11	sm_startup	38
3.3.2	SEPuler (SEPsesam Scheduler)	38
3.3.3	Queue Manager	39
3.3.3.1	Queue Manager Konzept	41
3.3.3.2	Queue je Laufwerkgruppe	41
3.3.3.3	Bandendekontrolle je Laufwerk	42
3.3.3.4	Abarbeitung der Jobs	42
3.3.3.5	que_init für Laufwerkgruppen	43
3.3.3.6	Übergabe Medienpool und optionale Laufwerknummer	43
3.3.3.7	Übergabe Laufwerknummer an auszuführendes Programm	44
3.3.3.8	Verhalten falls gewünschtes Laufwerk blockiert ist	44
3.3.3.9	Laufwerke sperren	44
3.3.3.10	Monitoring Laufwerkgruppen Queues	45
3.3.3.11	Jobs ohne Medienpool Zuordnung	45
3.3.3.12	Logging der Queue Manager Programme	46
3.3.4	Die Datenbank	46
3.3.5	Sesam Multi Stream - SMS	47
3.3.6	Datenübertragung mit SesamTransferProtocol - STP	48

3.4	Kommunikation Server - Klient	51
3.4.1	Kommandoausführung am Klienten - CTRL-Dienst	51
3.4.2	Kommandoausführung am Klienten - SSH(secure shell)	52
3.5	Prozesse und Datenfluss	53
3.6	Globale Variablen (GLBV)	55
3.7	SesamLoaderUtility SLU	56
3.7.1	sm_slu scan	56
3.7.2	sm_slu inquiry	57
3.7.3	sm_slu mode sense	58
3.7.4	sm_slu read element status	58
3.7.5	sm_slu read element status with label	59
3.7.6	sm_slu move medium	59
3.7.7	sm_slu unload for tapes	59
3.7.8	sm_slu load	59
3.7.9	sm_slu rewind	59
3.7.10	sm_slu allow removal	59
3.7.11	sm_slu prevent removal	60
3.7.12	sm_slu reset	60
4	Protokolle und Listings	61
4.1	Statusprotokoll	61
4.2	Tagesprotokoll	62
4.3	Sicherungsprotokoll	62
4.4	Liste der gesicherten Dateien	63
4.5	Segment Dateien	63
4.6	Analyseprotokolle - Logging	63
4.6.1	Logging der Kernprogramme	63
4.6.2	sm_db_main Logging	64
4.6.3	SMS/STPD Logging	66
5	Administration	67
5.1	Server starten und beenden	67
5.2	Reboot Konfiguration (Autostart nach Reboot)	67
5.3	Funktionsprüfung	68
5.4	GUI-Server Administration	69
5.4.1	GUI-Server Kommandos	70

5.5	Zugriffseinschränkung der Befehlstermine	71
5.5.1	Problembeschreibung	71
5.5.2	Befehlseinschränkung auf UNIX-Systemen	72
5.5.3	Befehlseinschränkung auf WNT-Systemen	72
5.6	Konfiguration einer Firewall	73
5.7	Konfiguration Master Server	77
5.8	Lizenzierung des SEPs-esam-Server	78
6	Desaster Recovery	79
6.1	Desaster Recovery Konzept	79
6.2	Desaster-Schnittstelle	79
6.3	Desaster Recovery ohne Metadaten	81
7	System Recovery	85
7.1	System Recovery für UNIX, LINUX	85
7.2	System Recovery für Windows	85
7.2.1	Vorgehen im Desaster Fall	86
8	Sondermodule	89
8.1	Online-Backup/Restore Domino-Server	89
8.1.1	Systemvoraussetzungen	89
8.1.2	Full Backup	90
8.1.3	Full Restore	90
8.2	Online-Backup/Restore MS SQL Server	90
8.2.1	Systemvoraussetzungen	90
8.2.2	Beschreibung MS SQL	90
8.2.3	Full Backup	91
8.2.4	Full Restore	91
8.3	Online-Backup Exchange-Server 5.5	91
8.3.1	Systemvoraussetzungen	91
8.3.2	Beschreibung EXCHANGE	92
8.3.3	Full Backup	92
8.3.4	Information Store Backup	93
8.3.5	Full Restore	93

9	Problemanalyse und -behebung	95
9.1	Probleme mit der Oberfläche (GUI)	95
9.2	Fehlermeldungen der Kernsoftware	97
9.2.1	BACKUP - Probleme mit Sicherungen	98
9.2.2	BASICS - grundlegende Fehler	99
9.2.3	BREAK - unerwartetes Abbrechen	100
9.2.4	CONFDR1 - Konfiguration Laufwerke	100
9.2.5	CONFLOA - Konfiguration Lader	101
9.2.6	DATABAS - Probleme mit Datenbank	102
9.2.7	DBFPACK - Datenbank-File-Parker	103
9.2.8	DRIVES - Probleme mit Laufwerken	104
9.2.9	HOSTS - Probleme mit Rechnern	105
9.2.10	LOADERS - Probleme mit Ladern	106
9.2.11	MEDIA - Probleme mit Medien	106
9.2.12	RESTORE - Rücksicherungsprobleme	107
9.2.13	SBC_COM - Meldungen des Sesam Backup Clients	109
9.2.14	SEPULER - Meldungen des SEP_sesam Schedulers	110
9.2.15	SONSTIGE	111

Kapitel 1

Einführung

1.1 Datenmanagement

Der Begriff Datenmanagement wird von fünf wesentlichen Begriffen geprägt:

Datenverfügbarkeit
Datensicherheit
Datensicherung
Datenrücksicherung
Disaster Recovery

Das erklärte Ziel der Firma **SEP AG** ist alle fünf Komponenten des Datenmanagements in den Sicherungskonzepten der Kunden zu integrieren und umzusetzen.

1.1.1 Datenverfügbarkeit

Die Zielvorstellung eines jeden Datenmanagements ist die ständige Verfügbarkeit unternehmensrelevanter Informationen. In einem Umfeld, in dem die geforderten Reaktionszeiten immer kürzer werden, kann ein Datenausfall zu schwerwiegenden Problemen führen, die meist mit hohen Kosten verbunden sind. Die heutzutage anfallenden enormen Datenmengen benötigen sehr viel Zeit für Pflege und Unterhalt. Deshalb soll der dafür benötigte Zeitaufwand durch eine Optimierung der Speicherprozesse minimiert werden. Der Anwender soll schnell und zu jeder Zeit auf die erforderlichen Daten zugreifen können.

1.1.2 Datensicherheit

Durch Verwendung sicherheitsrelevanter Techniken wie z.B. Verschlüsselung, Unterstützung von Firewalls und Secure Shell kann dem Datenmißbrauch vorgebeugt werden. Die dauernde Veränderung von Daten zu dokumentieren und diese ausfallsicher zu speichern stellt eine komplexe Aufgabe dar. Rationalisierung und Automatisierung bedeutet Kosteneffizienz. Konzeptionelle Datensicherheit (Ausfallsicherheit) wie zuverlässige Softwarekomponenten gehören zu den tragenden Systemelementen in der Informationstechnologie.

1.1.3 Datensicherung

Nur eine lückenlose Sicherung sämtlicher unternehmenswichtiger Daten garantiert einen möglichen Schutz vor Datenverlust. Hierbei kommt es darauf an, dass die angewendeten Systeme skalierbar, heterogen, wirtschaftlich und bedienerfreundlich ausgelegt sind.

1.1.4 Datenrücksicherung

Eine Datenrücksicherung soll im allgemeinen die Herstellung von ursprünglichen Datenbeständen gewährleisten.

1.1.5 Disaster Recovery

Im Disaster-Recovery-Fall sollen bei möglichen Totalausfällen von Unternehmensdaten die Bestände so schnell wie möglich wieder verfügbar gemacht werden. Brandschäden, Anschläge oder Naturkatastrophen sind nur einige Beispiele für einen Totalausfall.

1.2 Objekte von SEPsesam

Um die Konfiguration des Systems einfach und effektiv zu gestalten, wurde ein objektorientierter Ansatz gewählt, der sich um folgende 6 Grundobjekte aufbaut:

- **Laufwerk**
Als **Laufwerk** (drive) werden Geräte zur Speicherung von Daten bezeichnet, die entweder am **SEPsesam** Rechner selbst (local drive) oder an anderen Rechnern, die über das Netz erreichbar sind, angeschlossen sind (remote drive). In der Regel handelt es sich dabei um Bandgeräte, in die **Medien** eingelegt werden können. Eine Besonderheit stellen DISK-Drives dar. Dabei handelt es sich um Verzeichnisstrukturen auf Platten, in die Sicherungsdaten abgelegt werden.
- **Lader**
Laufwerke können in **Lader** eingebaut sein, die mittels Greifvorrichtungen **Medien** aus einem Magazin in das **Laufwerk** und zurück transportieren. Diese ermöglichen den vollautomatischen Betrieb des Sicherungssystems über Wochenenden, ganze Nächte etc.
- **Klient**
Klienten sind Rechner im Netz, deren Daten gesichert werden sollen.
- **Medium**
Medien sind auswechselbare Datenträger, auf denen die gesicherten Daten abgelegt und für Notfälle vorgehalten werden.
- **Auftrag**
Das Objekt **Auftrag** stellt die Beschreibung einer Sicherung oder Rücksicherung dar. Je nach Typ ist die Datenquelle oder das Ziel festzulegen (Verzeichnis am Klienten). Standard-Sicherungstyp ist die Pfad-Sicherung.

- **Termin**

Die **Zeitsteuerung** von **SEPsam** wird durch **Termine** geregelt, d.h. sämtliche Aktionen werden in einen zentralen Terminkalender eingetragen und ein Programm - der sogen. **SEPuler** - sorgt dafür, dass diese zur gewünschten Zeit ausgeführt werden.

Kapitel 2

Installation

2.1 Verzeichnisstruktur

Beim Entpacken der **SEPs**sesam Software wird folgende Verzeichnisstruktur aufgebaut (Installationsverzeichnis **IV**) :

IV/skel	Template-Dateien inkl. Rohdatenbank für die Erstinstallation
IV/bin/sesam	Sesam Server-Komponenten
IV/bin/sms	sms- und stpd-Komponenten
IV/bin/gui	Sesam Server Gui-Komponenten (GraficalUserInterface)
IV/bin/db	Sesam Server Datenbank-Komponenten

Beim Setup wird folgende Verzeichnisstruktur aufgebaut, wobei das Installationsverzeichnis IV nicht dem obigen Verzeichnis entsprechen muss:

IV/var/ini	Initialisierungsdateien
IV/var/log	Log- bzw. Trace-Hauptverzeichnis
IV/var/tmp	Kurzlebige temporäre Dateien
IV/var/work	Mittellebige temporäre Dateien
/etc/sesam2000.ini bzw. HKEY_LOCAL_MACHINE/ SOFTWARE/SEP Elektronik GmbH/sesam	Verweise auf Pfade der wichtigsten ini-Files

Im **/var/ini** werden folgende Initialisierungsdateien abgelegt:

debug.ini	Logging Konfiguration sesam Kernel
openrda.ini	sesam db (open Access)
sm.ini	sesam Konfiguration (Pfade etc.)
sms.ini	SMS Server Konfiguration (Pfade etc.)
stpd.ini	STPD Server Konfiguration (Pfade etc.)

Nach der Installation kann der Zugriff auf die Verzeichnisse IV/skel und IV/bin auf Read-Only gesetzt werden.

2.2 sm_setup

Beschreibung

Programm zur Komponenteninstallation bzw. -deinstallation.

Außerdem bietet es die Möglichkeit, Dateien zu erstellen oder zu ändern, Ändern und Auslesen der Windows Registry und eine Sammelkopierfunktion.

Kommandoreferenz

Die Reihenfolge der Parameter muss eingehalten werden.

Das Installationsverzeichnis für die variablen Daten unter IV/var kann mit dem optionalen Argument `var=vardir` verändert werden.

Die folgenden Aufrufe müssen im `{IV}/bin/sesam` ausgeführt werden.

Installation eines Servers `sm_setup make_srv [var=vardir]`

Hinweis:

Bei einer Server Installation, wird auch ein Klient und ein Tapeserver angelegt.

Installation eines Tape Servers `sm_setup make_node {SesamServer} [var=vardir]`

Installation einer graphischen Oberfläche `sm_setup make_gui {sesamserver} [var=vardir]`

Installation eines Klienten `sm_setup make_client {sesamserver} [var=vardir]`

Deinstalliert eines SEPsesam Produkts

`sm_setup delete`

Änderung der SEPsesam Konfiguration:

GUI Zugriff über einen anderen Rechner ermöglichen

`sm_setup allow_gui {gui} {user}`

GUI einen anderen Server zuordnen (Ausführung am GUI Klienten)

`sm_setup set_gui {sesamserver}`

Reboot-Script für Klienten oder Server setzen

`sm_setup set_reboot client—server—set_reboot_script`

Einem Server den Zugriff auf den Klienten zulassen (Eingabe am Klienten)

`sm_setup set_client {sesamserver}`

Beispiel

`sm_setup allow_gui pc21 admin`

Verwendung von sm_setup für Installation und Update:

Voraussetzungen: eigenes Verzeichnis mit dem gepackten Kit, `sm_setup` und ggf. dem `unzip` - Tool
`sm_setup` ohne Parameter aufrufen

2.3 Installation auf UNIX und LINUX

2.3.1 UNIX Backup Server

2.3.1.1 Software-Voraussetzungen

- Der Server muß den Klienten auflösen können und umgekehrt
- Klient und Tapeserver benötigen keine JRE (Java Runtime Environment)
- SUN Java Runtime Environment (JRE) 1.4.2 oder höher
- Acrobat Reader größer/gleich 4.0
- Funktionierende Namensauflösung IP Namen und Adresse (korrekte DNS, oder entsprechende Eintragungen in der /etc/hosts)

Unterstützte Betriebssysteme:

Betriebssystem: Version / Service Pack:

- Linux i386/x86_64/ppc
- Solaris 8/9/10
- FreeBSD ab 5.x
- Tru64 ab 4.0x

Freigegebene Lader/Bandlaufwerke:

Alle Lader mit SCSI-Schnittstelle und alle DAT, LTO, AIT, SUPER DLT und DLT- Laufwerke werden unterstützt.

2.3.1.2 Installation

1. Einloggen unter User „root“
2. CD-ROM-Laufwerk vom Betriebssystem des Servers zum Lesen vorbereiten.
3. Die SUN JRE ist notwendig, da sie eine JDBC-ODBC Bridge enthält, die vom SEPsesam GUI RMI Server verwendet wird. Sollte eine SUN-JRE als Standard Java nicht vorinstalliert sein, kann diese auch von der **SEPsesam** CD innerhalb des jeweiligen Ordners mit dem verwendeten Betriebssystem aus dem Unterverzeichnis „utils“ geladen werden.

Die dort befindlichen Dateien (diese enthalten im Namen „jre“), liegen entweder als Shell - Script (*.sh), als executable oder RPM - Package (*.rpm) vor.

Shell - Scripts werden mit `sh -x jre*.sh` ausgeführt.

Executables können von der Command Line aus direkt ausgeführt werden.

RPM - Packages werden mit `rpm -ivh *.rpm` gestartet.

Achtung: Zu bestimmen ist die Lokation, an die die JRE installiert wurde. Falls keine andere Java - Installation im System aktiv ist, empfiehlt es sich durch einen Link, diese global im System verfügbar zu machen.

Beispiel

Man definiert das soeben installierte Java mit

```
ln -sf /usr/lib/java2/jre/bin/java /usr/bin/java
```

zum Standard-Java Interpreter oder setzt die PATH-Variable vor der Installation so, dass diese Version der JRE für die Installation von **SEPs** verwendet wird.

4. Installation des Servers/Tapeservers/Klienten unter Linux/Unix mit Hilfe von sm_setup:

```
cd {CDROM - Pfad}/{Ordner mit Betriebssystem}
```

```
./sm_setup
```

Bei Schwierigkeiten mit der Installation entnimmt man aus dem Ordner “utils“ das beigelegte unzip - Tool und macht es für das sm_setup verfügbar (Pfadvariable erweitern oder in den gleichen Ordner legen)

Danach folgt man den Anweisungen von sm_setup.

Anschließend startet die Benutzeroberfläche mit

```
{IV}/bin/gui/sesam_gui.sh
```

Hinweise

- Es sollte sichergestellt sein, dass vor dem Start der **SEPs** Benutzeroberfläche (GUI) die richtige JRE verwendet wird.
- Der SEPs Server Zustand läßt sich im {IV}/bin/sesam überprüfen mit:
`./sm_main status`
und bei Nichtaktivität wird der SEPs Server mit
`./sm_startup`
initiiert.
- Stop des **SEPs** Servers mit
`{IV}/bin/sesam/sm_shutdown`

Stop des **SEPs** Tapeservers mit
`{IV}/bin/sesam/sm_main stop`

Stop des **SEPs** Klienten mit
`{IV}/bin/sesam/sm_main stop`

2.3.2 UNIX Klient

Der Server muß den Klienten auflösen können und umgekehrt.

Prüfen kann man dies mit ping, nslookup und sm_ctrlc. Sollten diese Befehle Fehler aufzeigen, so sind diese vor der Einbindung des Rechners in den **SEPs**sesam Server zu klären. Die Funktionsfähigkeit der **SEPs**sesam-Software ist grundlegend abhängig von einer korrekten Netzwerkinstallation.

Installation mit

```
sm_setup make_client {SesamServer}(siehe 2.2)
```

2.3.3 UNIX Tape Server

Ein Tape Server ist ein Client mit Erweiterungen für den Anschluss der RemoteDrives bzw. RemoteLoader (siehe 2.3.2).

```
sm_setup make_node {SesamServer}(siehe 2.2)
```

2.3.4 UNIX Oberfläche (GUI)

Voraussetzung: Java Runtime Environment (JRE) 1.4.2 oder höher

Der Zugriff auf die **SEPs**sesam-CD für das Betriebssystem muß ermöglicht werden.

Man wechselt nun der CD in das Verzeichnis, das dem jeweiligen UNIX - Betriebssystem entspricht. Nur, wenn dort ein Server - Kit liegt, ist auch eine GUI - Installation möglich. Die Dialogführung des sm_setup bietet die Möglichkeit der GUI-Installation an.

```
sm_setup make_gui {SesamServer}(siehe 2.2)
```

Die Zugriffsberechtigung des GUI-Rechners auf den **SEPs**sesam Server ist serverseitig in der Datei **{IV}/var/ini/sm_java.policy** festgelegt.

Ein entsprechender Eintrag kann **serverseitig** durch folgenden Aufruf erstellt werden:

```
sm_setup allow_gui -c {gui-client} -u {user} -m {admin|operator|restore}
```

Beispiel sm_java.policy - Unix

```
// GENERATED sm_java.policy BY sm_setup $Id: admin_handbuch.tex,v 1.32 2007/01/25 08:41:44 sesa
```

```
// DO NOT CHANGE FORMAT DURING EDITING LINES
```

```
grant
{
```

```
// ACCESS
// permission java.security.AllPermission;

// Allow GUI Server to refresh Policy - do not comment
permission java.security.SecurityPermission "getPolicy";

// NET
permission java.net.SocketPermission "*:11301", "connect,accept,resolve";
permission java.net.SocketPermission "*:11401", "connect,accept,resolve";
permission java.net.SocketPermission "*:21", "connect,accept,resolve";
permission java.net.SocketPermission "*:25", "connect,accept,resolve";
permission java.net.SocketPermission "127.0.0.1:*", "connect,accept,resolve";
permission java.net.SocketPermission "abas:*", "connect,accept,resolve";
permission java.net.SocketPermission "localhost:*", "connect,accept,resolve";
permission java.net.SocketPermission "nimbus:*", "connect,accept,resolve";

// ETC
permission java.util.PropertyPermission "*", "read,write";
permission java.util.PropertyPermission "file.encoding", "read";
permission java.util.PropertyPermission "java.rmi.server.hostname", "read";
permission java.util.PropertyPermission "user.dir", "read";
permission java.util.PropertyPermission "user.name", "read";
permission java.lang.RuntimePermission "accessClassInPackage.sun.jdbc.odbc";
permission java.lang.RuntimePermission "loadLibrary.JdbcOdbc";
permission java.sql.SQLPermission "setLog";

// FILE
permission java.io.FilePermission "/etc/sesam2000.ini", "read";
permission java.io.FilePermission "/var/opt/sesam/var/-", "read,write";
permission java.io.FilePermission "<<ALL FILES>>", "execute";
permission java.io.FilePermission "JdbcOdbcSecurityCheck", "write";

// SEP
permission de.sep.sesam.gui.server.AdminPermission "root@abas";
permission de.sep.sesam.gui.server.RestorePermission "rs@nimbus";

// MISC
};
```

2.3.5 Deinstallation Server, Tapeserver, Klient, GUI-Klient

```
cd {IV}/bin/sesam
```

```
sm_setup delete
```

2.3.6 SEPsesam Umgebung auf UNIX setzen

Für die Ausführung der SEPsesam Programme sind Umgebungsvariablen (PATH, LD_LIBRARY_PATH, OPENRDA.INI, ...) zu setzen. Diese sollen mit dem vorhandenen Profile IV/var/ini/sesam2000.ini gesetzt werden.

Vorgehensweise

Am Server auf Terminal-Ebene wechseln und

cat /etc/sesam2000.ini (Befehl) eingeben

Unter SM.INI den Pfad des **IV**ses ausschneiden und mit

cd {IV}/var/ini (Befehl)

in das ini-Verzeichnis wechseln

Die in sesam2000.profile hinterlegten Einstellungen mit

./sesam2000.profile (Befehl, wichtig ist ein Blank zwischen beiden Punkten) übernehmen

Beispiel für das Profile:

```
# GENERATED sesam.profile
#
# sesam2000 environment variables
sesam2000=/opt/sesam/var/ini
sesam_db=/opt/sesam/var/db
OA_ROOT=$sesam_db
OPENRDA_INI=/opt/sesam/var/ini/openrda.ini
SESAM_ROOT=/opt/sesam/bin/sms
LANG=de_DE.ISO8859-1
PATH=$PATH:$sesam2000:/opt/sesam/bin/sesam:/opt/sesam/bin/sms:/opt/sesam/bin/gui:/opt/
sesam/bin/db:$OA_ROOT/schema/sesam/sql
export sesam2000 sesam_db OA_ROOT OPENRDA_INI SESAM_ROOT LANG PATH
echo sesam2000 directory = $sesam2000
#
# environment for sesam database
# ./opt/sesam/bin/db/setenv2.sh
#
# sesam2000 COMPUTERTNAME (necessary for local client!)
COMPUTERTNAME='uname -n | cut -f1 -d\.'
```

```
#NPX_PLUGIN_PATH="$adir"/Browsers/$pconfig
#export NPX_PLUGIN_PATH
#PATH=$PATH:"$adir"/bin
#export PATH
#
#DELIMITER FOR damisql
SEP='/'
export SEP
#
# some alias
alias work="cd $sesam2000"
work_dir='dirname $sesam2000"/work'
alias cat_info='cat /opt/sesam/var/work/info/info.dat'
alias cat_status="ls -t /opt/sesam/var/prot/*.status | head -1 | xargs cat"
alias cat_prot="ls -t /opt/sesam/var/prot/*.prt | head -1 | xargs cat"
#
# EXTENSION PSQL
if type psql } /dev/null 2;&1
then
alias sql="psql -d sesam_db -p 11201 -c"
else
alias sql="damisql -d sesam_db -q"
fi
#
# EXTENSION SET CDPATH
CDPATH=$CDPATH:/opt/sesam:/opt/sesam/var:/opt/sesam/var/log:/opt/sesam/var/work:
/opt/sesam/bin
#
# EXTENSION SET LD_LIBRARY_PATH
LD_LIBRARY_PATH=/opt/sesam/bin/sesam
export LD_LIBRARY_PATH
#
Für die Ausführung der Sesam Kern Module ist auf UNIX das setzen der Umgebungsvariablen
LD_LIBRARY_PATH auf das [IV SEPsesam]/sesam/bin Verzeichnis ( Siehe Beispiel!) unerlässlich.
# EXTENSION SET ODBCINI
ODBCINI=/opt/sesam/bin/sesam
export ODBCINI
#
```

Wurde das Profile nicht übernommen so erscheint bei Aufruf eines SEPs sesam Programmes u.U. folgende Fehlermeldung:

```
UNIX TRU64: /sbin/loader: Fatal Error: cannot map libsm.so
Linux: error while loading shared libraries; libsm.so: cannot load shared object file: No such file or
directory
```

oder eine ähnliche Meldung.

2.4 Installation auf Windows

Betriebssystem:

- Windows 2003
- Windows 2000
- Windows XP
- Windows NT 4.0 (ab ServicePack 6)

Hardware Voraussetzungen:

- Prozessor: Intel/AMD
- Taktrate: ab 500 MHz
- Hauptspeicher: ab 128 MB (Vorteilhaft für die Geschwindigkeit ist ein größerer Hauptspeicher)

Software-Voraussetzungen:

- Funktionierende Namensauflösung (korrekte DNS) für IP Namen und IP Adresse (reverse lookup)

Hinweis

Ist eine Installation über Kommandoaufruf ohne Interaktion gewünscht, so kann sie bei der SEP AG eine Installations-CD, in der die Installationsdateien als *.zip - Dateien vorliegen, bestellt. Mit Kommandoaufruf `sm_setup` im Verzeichnis der *.zip Dateien (siehe 2.2) und mit Zugriff auf ein geeignetes unzip Tool (Auf der CD befindet sich unter `utils unzip.exe`) ist die Installation dann per Kommandoaufruf ohne weitere Eingaben möglich.

Da diese Installationsart unter Windows unüblich ist, liegen auf der Installations-CD drei selbstentpackende Archive. Sie beinhalten einen Installshield, der die im Archiv enthaltenen SEPsesam Programme zur Installation anbietet.

- `win32_cli{VersionNr}.exe` = Installation eines Windows - Klienten
- `win32_gui{VersionNr}.exe` = Installation der Benutzeroberfläche (GUI) für einen im Netzwerk befindlichen **SEPsesam**-Server auf einem Windows - PC
- `win32_srv{VersionNr}.exe` = Installation des **SEPsesam**-Servers auf einem Windows-PC (da dieses Archiv die Software für GUI, Klient und Tapeserver mit beinhaltet, ist damit auch die Installation einer GUI, eines Klienten oder eines Tapeservers möglich)

Folgen sie der Führung des Installationsklienten! Der Installationsklient führt durch die jeweils nötigen Schritte.

2.4.1 Grundlegende Vorgehensweise

Um **SEPs** Programme auf Windows zu installieren, führt man folgende Schritte aus:

- CD-ROM in das CD-Laufwerk des Rechners einlegen. Mit Hilfe des Windows Explorers oder eines ähnlichen Programmes die CD-ROM nach dem Verzeichnis win32 durchsuchen und das Verzeichnis öffnen. Hier findet man die selbstentpackenden Archive *.exe und readme*.txt Dateien.

Die readme Dateien sollten vor der Installation unbedingt gelesen werden.

Durch Aufruf der entsprechenden exe Datei wird das Archiv in ein temporäres Archiv entpackt und durch Benutzerauswahl das jeweilige Programm und das **{IV}** ausgewählt. Alle Eingaben werden im Dialog abgefragt.

2.4.2 Windows Backup Server

Bitte auf der CD nt_srv{VersionNr}_{Sprachkenner}.exe starten. (siehe 2.4)

2.4.3 Windows Klient

Auf der CD nt_cli{VersionNr}_{Sprachkenner}.exe starten. (siehe 2.4)

2.4.4 Windows Tape Server

Ein Tape Server ist ein Client mit Erweiterungen für den Anschluss der RemoteDrives bzw. RemoteLoader.

Auf der CD nt_srv{VersionNr}_{Sprachkenner}.exe starten. (siehe 2.4)

2.4.5 Windows Oberfläche (GUI)

Auf der CD nt_gui{VersionNr}_{Sprachkenner}.exe starten. (siehe 2.4)

Hardware Voraussetzungen für GUI:

Hauptspeicher: min. 256 MB

Software-Voraussetzungen:

- Java Runtime Environment (JRE) oder Java Development Kit (JDK) ab 1.4.2
- Internet Explorer ab 5 oder Mozilla Firefox
- Acrobat Reader ab 4.0 für Online-Handbuch

Die Zugriffsberechtigung des GUI-Rechners auf den **SEPs**esam Server ist serverseitig in der Datei `{IV}\var\ini\sm_java.policy` festgelegt.

Ein entsprechender Eintrag kann **serverseitig** durch folgenden Aufruf erstellt werden:

```
sm_setup allow_gui {gui} {user}
```

Beispiel `sm_java.policy` - NT-Version

```
// GENERATED sm_java.policy BY sm_setup $Id: setup.c,v 1.106 2002/11/06
11:11:55 sesam Exp $ FOR VERSION 2.0.1.13

// DO NOT CHANGE FORMAT DURING EDITING LINES

grant
{
  // ACCESS
  //permission java.security.AllPermission;

  // NET
  permission java.net.SocketPermission "*:11401", "connect,accept,resolve";
  permission java.net.SocketPermission "*:11301", "connect,accept,resolve";
  permission java.net.SocketPermission "*:21", "connect,accept,resolve";
  permission java.net.SocketPermission "*:25", "connect,accept,resolve";
  permission java.net.SocketPermission "127.0.0.1:*",
"connect,accept,resolve";
  permission java.net.SocketPermission "softnt6:*",
"connect,accept,resolve";

  // ETC
  permission java.util.PropertyPermission "user.dir", "read";
  permission java.util.PropertyPermission "user.name", "read";
  permission java.util.PropertyPermission "file.encoding", "read";
  permission java.util.PropertyPermission "java.rmi.server.hostname",
"read";
  permission java.util.PropertyPermission "*", "read,write";
  permission java.lang.RuntimePermission
"accessClassInPackage.sun.jdbc.odbc";
  permission java.lang.RuntimePermission "loadLibrary.JdbcOdbc";
  permission de.sep.sesam.gui.server.RestorePermission "*@*";
  permission de.sep.sesam.gui.server.OperatorPermission "*";
  permission java.lang.RuntimePermission "loadLibrary.ICE_JNIRegistry";
  permission java.sql.SQLPermission "setLog";

  // FILE
  permission java.io.FilePermission "D:\\sesam\\var\\-", "read,write";
  permission java.io.FilePermission "JdbcOdbcSecurityCheck", "write";
```

```
permission java.io.FilePermission "<<ALL FILES>>", "execute";  
};
```

Hinweis

Im {**IV**}\bin\gui findet man die Verknüpfung `sesam_gui`. Diese Verknüpfung kann mit der Maus auf den Windows Desktop oder/und auf die Taskleiste oder/und auf den Startbutton gezogen werden. Mit einem Doppelklick auf diese Verknüpfung startet die GUI von **SEPs**`sesam` (falls die Kernprozesse laufen (siehe 3.5)).

Deinstallation:

1. Start-Programme-SEP Aktiengesellschaft-Uninstall
oder
2. Start-Einstellungen-Systemsteuerung, anschließend den Folder „Software“ auswählen und die `SEPs``sesam`-Uninstall-Routine starten.
oder
3. Mit Aufruf
`sm_setup delete`

2.5 SEPs

2.5 SEPs

`SEPs``sesam` Klienten sind für folgende Betriebssysteme verfügbar. Bzgl. UNIX Betriebssysteme gilt, dass generell alle UNIX Systeme unterstützt werden. Weitere Klienten auf Nachfrage.

- OS/2
- PowerMAC_OS
- UNIX
 - AIX ab 4.1
 - BSD
 - Digital UNIX, Tru64 ab 4.0x
 - HP UX Vers. 9,10 u. 11
 - IRIX
 - LINUX Kernel 2.2.x I386 und IA64
 - OSF1UX
 - SCO-UNIX
 - SINIX
 - Solaris Sparc ab Version 2.4

- Solaris auf i386 2.4
- ULTRIX
- VMS ab 7.0
- Windows Systeme (NT, 2000, 2003, XP)

2.6 SEPsam Klient: Unterstützte Online Module

Für die Sicherung folgender Datenbanken bzw. Applikationen stehen spezielle Sicherungsklienten zur Verfügung.

Datenbank bzw. Applikation	auf Betriebssystem
Informix 7.xx, 9,2000	LINUX (SuSE, Red Hat, Debian, weitere auf Nachfrage) UNIX (SUN, HPUX, TRU64, AIX, Reliant UNIX, weitere auf Nachfrage) Windows NT/2000/XP
Lotus Notes	LINUX (SuSE, Red Hat, Debian, weitere auf Nachfrage) Windows NT/2000/XP
MS-Exchange 2000 oder 5.5	Windows NT/2000/XP
MS-SQL Server	Windows NT/2000/XP
Oracle 7, 8.xx und 9i	LINUX (SuSE, Red Hat, Debian, weitere auf Nachfrage) UNIX (SUN, HPUX, TRU64, AIX, Reliant UNIX, weitere auf Nachfrage) Windows NT/2000/XP
SAP R/3 (mit Oracle Datenbank)	LINUX (SuSE, Red Hat, Debian, weitere auf Nachfrage) UNIX (SUN, HPUX, TRU64, AIX, Reliant UNIX, weitere auf Nachfrage) Windows NT/2000/XP
SAP R/3 (mit SAP DB Datenbank)	LINUX (SuSE, Red Hat, Debian, weitere auf Nachfrage) UNIX (SUN, HPUX, TRU64, AIX, Reliant UNIX, weitere auf Nachfrage) Windows NT/2000/XP

Kapitel 3

Die Programme des SEPs sesam

SEPs sesam besteht aus einer Anzahl von C-Programmen und Java-Programmen, die durch ihr Zusammenwirken sowohl auf dem Server wie auf dem Klienten einen reibungslosen Ablauf der Funktion realisieren.

Die Programme findet man unter `{IV}/bin` in den Unterordnern `sesam`, `sms`, `db` und `gui`.

3.1 Benutzeroberfläche GUI (Graphical User Interface)

Die Benutzeroberfläche von **SEPs sesam** ist in Java programmiert und kann auf einem beliebigen Rechner mit Java Runtime Environment (JRE) ab 1.4.2 installiert werden.

Sie kommuniziert mit dem **SEPs sesam**-Kern über die JAVA-Komponenten RMI (RemoteMethodInvocation) und JDBC (JavaDatabaseConnectivity).

Die JRE (JavaRuntimeEnvironment) beinhaltet die Java Virtual Machine (JVM) und dient zur Ausführung der durch die Benutzeroberfläche eingegebenen Befehle.

Am SEPs sesam Server ist der in Java implementierte RMI Server für die Abarbeitung der Aufrufe zuständig. Dieser benutzt eine JDBC-ODBC Bridge für den Zugriff auf die SEPs sesam Datenbank und eine CTRLD Socket Verbindung für den direkten Aufruf von SEPs sesam Programmen.

3.2 SesamBackupClient - SBC

Der SBC ist das eigentliche Kernmodul für die Datensicherung bzw. -rücksicherung.

Das zugehörige Programm `sbc{.exe}` liegt sowohl auf dem **SEPs sesam**-Server als auch auf den Sicherungsklienten.

Der SBC sammelt während eines Backups die Daten auf dem Klienten, bündelt sie in einem Kanal und sendet sie gemäß dem XBSA Standard (X/Open Backup Service API) an den **SEPs sesam**-Transfer-Protocol-Daemon (**STPD**), der sie auf das Ziel-Speichermedium schreibt.

Während eines Restores werden die Daten vom Medium gelesen, über den Kanal an den Klienten gesendet, dort vom SBC aufgelöst und in die Zieldateien verteilt.

Die Daten werden mit einer eigenen ftp-Bibliothek zum bzw. vom jeweiligen STPD-Server übertragen. SBC verwendet auf UNIX-Systemen das Sicherungskommando `cpio` mit „Large File“ Unterstützung, wogegen auf NT-Systemen das NT-spezifische MTF-Format genutzt und die Sicherung bzw. Rücksicherung in eigenen Modulen durchgeführt wird.

Befehl **sbc**

```
sbc -{brgpkh} [-C <control_host>] [-d <device>] [-f <list_source>]
[-F <data_format>] [-i <saveset_info>] [-j <job_name>]
[-l <level>] [-L <control_target>] [-n <segment_number>]
[-o <restore_type>] [-O <STOR/RETR_direct>]
[-R <restore_target>] [-s <saveset_spec>] [-S <storage_node>]
[-t <tape_spec>] [-T <since_time>] [-x <exclude_patterns>]
[-X <exclude list>] [-v <level>]
[<szBackupSource 1> [<szBackupSource 2> | <szBackupSource n>]] |
[<szRestoreSpec 1> [<szRestoreSpec 2> | <szRestoreSpec n>]]
```

-b	Sichert die Daten eines Klienten auf ein Laufwerk eines anderen Rechners
-r	Restauriert Dateien am Klienten
-p	Kopiert eine Datei im ASCII-Modus vom eigenen Rechner zu einem FTP-Server
-g	Kopiert eine Datei im ASCII-Modus von einem FTP-Server auf den eigenen Rechner
-h	Hilfe Ausgabe auf Standard Output und Programm-Abbruch
-d {device}	Quelle oder Ziel auf einem Rechner für Sicherung oder Rücksicherung
-o {restore_type}	Beschreibt verschiedene Restore-Möglichkeiten (r[ename], p[lain], over[write] = Überschreiben, n[ext])
-R {restore_target}	Relocation der Rücksicherungsverzeichnisstruktur
-s {savesetname}	Name des Savesets (Sicherungssatz)
-S {storage_node}	Server oder Storage Node
-v {level}	Setzt den Verbose Level

Beispiel Backup

Dateien werden von einem WindowsNT Rechner auf einen UNIX-Rechner gesichert. Die Dateien aus dem Verzeichnis c:\winnt werden mit erhöhtem Verboselevel 3 in das Verzeichnis work in den Saveset my_file_safesetname_1 kopiert.

```
sbc -b -S miraculix -s my_file_safesetname_1 -v 3 c:\winnt
```

Beispiel Restore:

Die vorher gesicherten Dateien werden in das Verzeichnis c:\temp\restore zurückkopiert.

```
sbc -r -S decunix -s my_file_safesetname_1 -o over -R c:\temp\restore
```

3.2.1 SBC Fehlercodes

Die folgende Liste gibt alle möglichen SBC Fehlermeldungen wieder.

```

/*CLI*/      {1001, E_ERROR, "Option -%c requires an argument."},
/*CLI*/      {1002, E_ERROR, "Unrecognized option: -%c."},
/*CLI*/      {1003, E_ERROR, "Double specified argument: -%c."},
/**/        {1004, E_ERROR, "Invalid function."},
/*CLI*/      {1005, E_ERROR, "Option -%c not available in this mode."},
/*CLI*/      {1006, E_ERROR, "Maximum number of backup files in command line exceeded."},
/*XBSA*/     {1007, E_ERROR, "XBSA Call failed with message: %s"},
/**/        {1008, E_ERROR, "System call failed: %s"},
/*XBSA*/     {1009, E_ERROR, "XBSA Call BSACreateObject failed with message: %s"},
/*XBSA*/     {1010, E_ERROR, "XBSA Call BSAEndData (closing saveset) failed: %s"},
/*FILE*/     {1011, E_ERROR, "Unable to open file [%s] [%s]."},
/**/        {1012, E_ERROR, "Undefined \"HOME\" environment variable."},
/**/        {1013, E_ERROR, "Invalid function."},
/**/        {1014, E_ERROR, "Invalid number of colon separated fields within
savesetsource|savesettarget argument."},
/**/        {1015, E_ERROR, "Cannot get HOME directory."},
/**/        {1016, E_ERROR, "Cannot change directory to %s: %s."},
/*EXIT*/     {1017, E_ERROR, "Backup command call failed with exit value: %i."},
/**/        {1018, E_ERROR, "DATA interface start failed: %s"},
/**/        {1019, E_ERROR, "System call [%s] timed out after [%i] seconds"},
/*EXIT*/     {1020, E_ERROR, "Backup source not specified. Exiting."},
/**/        {1021, E_ERROR, "Cannot read item header from archive: [%s]. Exiting."},
/**/        {1022, E_ERROR, "Unknown item header magic number [%s]. Exiting."},
/**/        {1023, E_ERROR, "Cannot read item header from archive: [%s][%d][%d]. Exiting."},
/**/        {1024, E_ERROR, "Cannot parse item header: [%s]. Exiting."},
/**/        {1025, E_ERROR, "Cannot read item name from archive: [%s][%d][%d].
Exiting."},
/*DIR*/      {1026, E_ERROR, "Cannot create temporary directory [%s]."},
/**/        {1027, E_ERROR, "Cannot read from remote archive: [%s]."},
/**/        {1028, E_ERROR, "Termination signal received!"},
/**/        {1029, E_ERROR, "Failed to kill DATA process [%d]:[%s]."},
/**/        {1030, E_ERROR, "Archive padding failed: [%s]."},
/**/        {1031, E_ERROR, "Cannot write to remote archive: [%s]"},
/**/        {1032, E_ERROR, "Cannot read item [%s]:[%s]. Padding remaining [%u]
bytes..."},
/**/        {1033, E_ERROR, "Cannot write to remote archive. Exiting."},
/**/        {1034, E_ERROR, "Unable to open GEN interface: [%s]"},
/**/        {1035, E_ERROR, "Unknown item type: [%d]. Skipping remaining bytes [%d]"},
/**/        {1036, E_ERROR, "Unknown item type: [%d]."},
/**/        {1037, E_ERROR, "Internal error. eHdrFmt unknown"},
/*CLI*/      {1038, E_ERROR, "Invalid argument [%s] for switch -%c."},
/*XBSA*/     {1039, E_ERROR, "Failed STP call [%s] [%s]."},
/*CLI*/      {1040, E_ERROR, "Invalid host specification for option -%c."},
/*CLI*/      {1041, E_ERROR, "Invalid argument for -%c. Encryption password too long."},
/*CLI*/      {1042, E_ERROR, "Invalid usage. Try \"sbc -h\" for help screen."},
/*CLI*/      {1043, E_ERROR, "Option -%c is required for this operation."},
/*CLI*/      {1044, E_ERROR, "Invalid time specification for since backup."},
/*REGISTRY*/ {1045, E_ERROR, "Error while retrieving user profile images."},

```

```

/*REGISTRY*/ {1046, E_ERROR, "Cannot enable backup/restore privileges."},
/*REGISTRY*/ {1047, E_ERROR, "Cannot disable backup/restore privileges."},
/*REGISTRY*/ {1048, E_ERROR, "Cannot find registry data."},
/*REGISTRY*/ {1049, E_ERROR, "Cannot enable registry privileges to retrieve user
    profile images."},
/*REGISTRY*/ {1050, E_ERROR, "Error while unloading user profile images."},
/*REGISTRY*/ {1051, E_ERROR, "Cannot retrieve user profiles."},
/*REGISTRY*/ {1052, E_ERROR, "Cannot retrieve value for [%s]\\ProfileImagePath"},
/*REGISTRY*/ {1053, E_ERROR, "Unable to open registry key: [%s]"},
/*INI*/ {1054, E_ERROR, "Invalid PHYSICAL_BLK_SIZE or FORMAT_LOGICAL_BLK_SIZE
    in [%s]."},
/*ARCHIVE*/ {1055, E_ERROR, "Initialization of archive failed: [%s]"},
/**/ {1056, E_ERROR, "Error while retrieving items."},
/*ARCHIVE*/ {1057, E_ERROR, "Cannot create local archive [%s]. Error: %s"},
/**/ {1058, E_ERROR, "Error while writing data. Error: %s"},
/*ARCHIVE*/ {1059, E_ERROR, "Unable to open local archive [%s]. Error: %s"},
/**/ {1060, E_ERROR, "Error while reading data. Error: %s"},
/*ARCHIVE*/ {1061, E_ERROR, "Error while unpacking data. %s"},
/**/ {1062, E_ERROR, "Local file [%s] does not exist."},
/**/ {1063, E_ERROR, "Unable to open local file [%s]."},
/**/ {1064, E_ERROR, "Error while processing local file [%s]."},
/*FILE IO*/ {1065, E_ERROR, "Cannot create local file [%s]."},
/*FILE EXCL*/ {1066, E_ERROR, "Error while reading exclude pattern file [%s]:[%s]."},
/*XBSA*/ {1067, E_ERROR, "XBSA Call BSAGetObject failed with message: %s"},
/*CLI*/ {1068, E_ERROR, "-%c: Specified argument too long."},
/*CLI*/ {1069, E_ERROR, "-%c: Invalid argument - must be numeric."},
/*CLI*/ {1070, E_ERROR, "Mandatory argument not specified."},
/*CLI*/ {1071, E_ERROR, "-%c: Invalid time specified [use: hhmm]."},
/*CLI*/ {1072, E_ERROR, "Unable to open include list file. Argument: -%c"},
/*CLI*/ {1073, E_ERROR, "-%c: Invalid verbose level specified [max. 3 allowed]."},
/*COM*/ {1074, E_ERROR, "SBC_COM_INTERFACE call failed: %s"},
/*COM*/ {1075, E_ERROR, "Error checking the include list specs: %s"},
/*COM*/ {1076, E_ERROR, "Error allocating include list specs. buffer: %s"},
/*COM*/ {1077, E_ERROR, "Answer buffer empty."},
/*COM*/ {1078, E_ERROR, "SESAM server answer processing failed ! %s"},
/*ITEM*/ {1079, E_ERROR, "Archive could not be resynchronized. Exiting."},
/*COM*/ {1080, E_ERROR, "-%c: Invalid operation initiator specified !"},
/*CLI*/ {1081, E_ERROR, "-%c: Invalid argument - max. value exceeded!"},
/*CLI*/ {1082, E_ERROR, "-%c: Operation unspecified [open|openjob|closejob|close]!"},
/*CLI*/ {1083, E_ERROR, "-%c: Operation [openjob|closejob|close] requires the
    session ID argument!"},
/*CLI*/ {1084, E_ERROR, "-%c: Operation [open] doesn't need the session ID
    argument!"},
/*COM*/ {1085, E_ERROR, "CWD command assembling failed!"},
/*COM*/ {1086, E_ERROR, "RETR command assembling failed!"},
/*MSEX*/ {1087, E_ERROR, "Invalid backup source [%s]."},
/*MSEX*/ {1088, E_ERROR, "MS Exchange server not specified."},
/*MSEX*/ {1089, E_ERROR, "MS Exchange server path not available. [%s]."},

```

```

/*MSEX*/      {1090, E_ERROR, "Cannot establish connection to remote Sesam server."},
/*MSEX*/      {1091, E_ERROR, "Unable to open file [%s] : [%d]"},
/*MSEX*/      {1092, E_ERROR, "Backup operation failed."},
/*MSEX*/      {1093, E_ERROR, "Cannot end MS Exchange backup:[%s]"},
/*MEM*/       {1094, E_ERROR, "Unable to allocate memory."},
/*MSEX*/      {1095, E_ERROR, "Failed to read MS Exchange database file. [%s]."},
/*MSEX*/      {1096, E_ERROR, "Failed to purge MS Exchange database log files [%s]."},
/*MSEX*/      {1097, E_ERROR, "Cannot retrieve a file list from MS Exchange server.
                [%s]."},
/*MSEX*/      {1098, E_ERROR, "Cannot save file [%s]."},
/*MSEX*/      {1100, E_ERROR, "Unable to open file [%s]: [%s]."},
/*MSEX*/      {1101, E_ERROR, "Restore operation failed."},
/*MSEX*/      {1102, E_ERROR, "Cannot end MS Exchange restore:[%s]."},
/*MSEX*/      {1103, E_ERROR, "Cannot get local database files paths. [%s]."},
/*MSEX*/      {1104, E_ERROR, "Error reading file header from archive."},
/*MSEX*/      {1105, E_ERROR, "Cannot register MS Exchange restore: [%s]"},
/*MSEX*/      {1106, E_ERROR, "Cannot find MS Exchange target for file [%s]."},
/*MSEX*/      {1107, E_ERROR, "Cannot read file header from remote saveset."},
/*MSEX*/      {1108, E_ERROR, "Invalid file header magic number. [%s]"},
/*MSEX*/      {1109, E_ERROR, "Invalid file size in header."},
/*MSEX*/      {1110, E_ERROR, "Invalid file name size in header"},
/*MSEX*/      {1111, E_ERROR, "Cannot read file name from remote saveset."},
/*MSEX*/      {1112, E_ERROR, "Unable to open file [%s]. Make sure the store service
                has been stopped before restoring"},
/*MSEX*/      {1113, E_ERROR, "Unable to open file [%s] for restore: [%d]"},
/*MSEX*/      {1114, E_ERROR, "Failed to read [%d] bytes from remote archive."},
/*MSEX*/      {1115, E_ERROR, "Failed to write to remote archive [%s]: [%s]"},
/*MSEX*/      {1116, E_ERROR, "Failed to read from remote archive [%s]."},
/*MSEX*/      {1117, E_ERROR, "Cannot close saveset."},
/*sbc_gen.sh*/{1118, E_ERROR, "[%s] not existent"},
/*CLI*/       {1119, E_ERROR, "SESAM server not specified."},
/*sbc_gen.sh*/{1120, E_ERROR, "Backup source could not be found. Exiting."},
/**/         {1121, E_ERROR, "For this operation option [-f @] require [-F pipe].
                Exiting."},
/*KILL*/      {1122, E_ERROR, "Can not kill process group [%d], [%s]. Exiting."},
/*sbc_gen.sh*/{1123, E_ERROR, "sbc_gen.sh exit with error."},
/*FP*/        {1124, E_ERROR, "Error while browsing through process list: [%s]"},
/*KILL*/      {1125, E_ERROR, "Cannot find matching process."},
/*KILL*/      {1126, E_ERROR, "Cannot perform core termination."},
/*CTRLC*/     {1127, E_ERROR, "Cannot detect currently logged user: [%s]"},
/*NETWORK*/   {1128, E_ERROR, "Network communication problem: %s"},
/*NETWORK*/   {1129, E_ERROR, "Not connected."},
/*CTRLC*/     {1130, E_ERROR, "STDIN channel not established: [%s]"},
/*INI*/       {1131, E_ERROR, "Unable to open initialization file: [%s]"},
/*CTRLC*/     {1132, E_ERROR, "STDIN channel broken: [%s]"},
/*CTRLD*/     {1133, E_ERROR, "Denying service. Maximum active connection reached."},
/*CTRLD*/     {1134, E_ERROR, "Cannot start child process: %s"},
/*NETWORK*/   {1135, E_ERROR, "Remote IP address could not be resolved: [%s]"},

```

```

/*CTRLD*/      {1136, E_ERROR, "Host [%s] is not allowed to connect or security
                problem for user: [%s]."},
/*INTF*/       {1137, E_ERROR, "Parameter to long."},
/*STREAM*/     {1138, E_ERROR, "Pipe creation problem: [%s]."},
/*CTRLD*/     {1139, E_ERROR, "CTRLD reply: %s"},
/*PROCESS*/    {1140, E_ERROR, "Unable to start process: [%s]"},
/*PROCESS*/    {1141, E_ERROR, "Unable to start thread: [%s]"},
/*CLI*/        {1142, E_ERROR, "-%c: Mandatory argument not specified."},
/*DLL*/        {1143, E_ERROR, "Problem while loading dynamic link library: [%s]."},
/*DLL*/        {1144, E_ERROR, "Problem while unloading dynamic link library: [%s]."},
/*STPD*/       {1145, E_ERROR, "SESAM server signal received: [%s]"},
/*DB*/         {1146, E_ERROR, "DATABASE client error: [%s]"},
/*Item*/       {1147, E_ERROR, "Unable to access item [%s]: [%s]."},
/*W2KSS*/      {1148, E_ERROR, "W2KSS Error: [%s]."},
/*Encryption*/{1149, E_ERROR, "Encryption password validation failed."},
/*Ex2K*/       {1150, E_ERROR, "Ex2K Error: [%s]."},
/*Encryption*/{1151, E_ERROR, "Encryption/Decryption operation failed."},
/*Encryption*/{1152, E_ERROR, "Writing SSET file failed."},
/*Encryption*/{1153, E_ERROR, "Reading SSET file failed."},
/*CLI*/        {1154, E_ERROR, "Invalid argument for switch -%c."},
/*LAST*/       {9999, E_ERROR, "Unknown message code: [%d]."}

```

3.2.2 Optionale Erweiterungen des SBC

Der Standard ist **Sicherungstyp Pfad**, dessen Quelle eine vollständige Pfadangabe am Klienten ist.

Beispiele:

Für WNT : D:\Program Files

Für UNIXe: /usr/users

Sollen mehrere Sicherungsquellen mit einem Auftrag am Klienten gesichert werden, so sind die jeweiligen Quellen durch Kommata getrennt anzugeben.

Beispiele:

Für WNT : D:\Program Files, D:\MSOffice

Für UNIXe: /usr/users, /usr/homes, /opt/sesam

Zur besseren Lesbarkeit sind Leerzeichen nach dem Komma erlaubt. Diese werden ignoriert.

Lizenzpflichtige Erweiterungen sind

- **Sicherungstyp Exchange Server**

Als Quelle einer Exchange Server Sicherung ist die Bezeichnung der Exchange Datenbank (häufig identisch mit dem WNT Server Namen) anzugeben.

- **Sicherungstyp INFORMIX**

Als Quelle einer INFORMIX Sicherung ist die Bezeichnung der INFORMIX Datenbank anzugeben. Ein INFORMIX Auftrag wird extern vom INFORMIX System aus mit ONBAR gestartet.

- **Sicherungstyp Lotus Notes**

Als Quelle einer Lotus Notes Sicherung ist die Bezeichnung der Lotus Notes Datenbank anzugeben.

- **Sicherungstyp MS SQL Server**

Als Quelle einer MS SQL Server Sicherung ist die Bezeichnung der SQL Server Datenbank anzugeben.

- **Sicherungstyp ORACLE**

Als Quelle einer ORACLE Sicherung ist die ORACLE SID der ORACLE Datenbank anzugeben. Ein ORACLE Auftrag wird extern vom ORACLE System aus mit RMAN gestartet.

- **Sicherungstyp ORACLE7**

Als Quelle einer ORACLE7 Sicherung ist die ORACLE SID der ORACLE Datenbank anzugeben. Das Starten einer ORACLE7 Sicherung erfordert die sesam Klienten Software sbc_ora7. ORACLE7 Aufträge können direkt vom SEPsesam System gestartet werden.

- **Sicherungstyp SAP/R3**

Als Quelle einer SAP/R3 Sicherung ist die SAP/R3 ID der SAP/R3 Datenbank anzugeben. Ein SAP/R3 Auftrag wird extern vom SAP/R3 System aus mit BRBACKINT (brbackup/brarchive/brrestore) gestartet.

SEPsesam ermöglicht die Ausführung einer externen Backup Utility (INFORMIX: ONBAR, ORACLE: RMAN, SAP/R3: BRBACKINT) mit Hilfe eines Befehlstermins.

3.3 Die Programme des Servers

3.3.1 Die Kernprogramme

Unter `{IV}/bin/sesam` liegen die Kernprogramme `sm_*.exe` der **SEPsesam**-Software. Die im Folgenden aufgeführten Programme können direkt von der Kommando-Zeile aus aufgerufen werden. Nicht aufgeführte Programme werden intern von den hier beschriebenen als Unterprogramm-Aufrufe angesprochen. Ein direkter Aufruf ist dabei nicht erforderlich.

Die Befehlssyntax entspricht i. allg. dem getopt-Standard, d.h. `sm_* -{Schalter} {Wert} ...`

Hinweis

Unsachgemäße Benutzung dieser Kommandos kann zu schwerwiegenden Funktionsstörungen führen.

3.3.1.1 sm_backup

Beschreibung

`sm_backup` steuert die Ausführung eines Sicherungsauftrags.

Abfolge der Unterfunktionen:

```

GET_BACKUP_TYP
GET_NODE_PARAMS
GET_TASK_PARAMS
CHECK_LICENSE
CHECK_SEMAPHORES
CHECK_MEDIUM      -> sm_init falls kein Medium verfügbar
CHECK_PERMIT      -> Alivetest am Klienten
PRE               -> PRE-Schnittstelle
CHECK_SMS_WATCH   -> Performance- und EOM-Beobachtung
EXECUTE_BACKUP    -> CTRLC -> SBC -> STPC -> STPD -> SMS
POST              -> POST-Schnittstelle
RESULT_ANALYSE

```

Schalter	Beschreibung	Default
-j task	Auftragsname	TEST_BACKUP
-l char	Backuptyp (C F D I) (Copy Full Diff Incr)	C
-m pool	Medienpool in dem gesichert werden soll	Voreinstellung
-d drive	sesam-interne Laufwerknummer	1.Laufwerk des Medienpools
-T time	Startzeit "hh:mm dd-mm-yyyy"	
-W time	Endezeit "hh:mm dd-mm-yyyy"	
-S tapeserver	Rechner auf dessen Laufwerk die Daten fließen sollen	SEPsam-Server
-p num	Priorität	1
-s saveset	Name des Sicherungssatzes	Zeitstempel
-v num	0 1 2 3 = verbose level	1

3.3.1.2 sm_drive

Kernprogramm für den sog. Drivehandler in **GUI:Komponenten-Laufwerke-Laufwerksaktion**.

Beschreibung

Modul zur Abwicklung aller laufwerkspezifischen Anforderungen (local und remote).

Syntax: sm_drive {Funktion} {Laufwerknummer} [Label]

Funktion	Beschreibung
init	initialisiert Medium im Laufwerk mit Label
info	gibt Label und MOUNT-Zustand des Mediums aus
mount	setzt Medium in MOUNT-Zustand
dismount	löscht MOUNT-Zustand

Im MOUNT-Zustand wird das Label bei einer Infoanfrage nicht vom Medium gelesen, was ein Rückspulen zum Medienbeginn erforderlich machen würde, sondern aus der Datenbank.

MOUNT ist also ein rein softwaretechnischer Zustand, der zeitraubende Spulvorgänge verhindert. Dies macht es zwingend notwendig, die Laufwerke des **SEPs** nur über diesen Drivehandler zu bedienen. Im MOUNT ist es durchaus möglich, Bandkassetten aus einem Drive zu nehmen. Problematik: **SEPs** arbeitet danach ggf. auf einem falschen Medium. Erst wenn vor dem Entladen der Kassette ein Dismount angestossen wurde, bleibt die Laufwerksinformation konsistent.

Rückgabewerte auf STDOUT:

STATUS=SUCCESS|ERROR MSG={gewünschte Ausgabe | Fehlermeldung}

3.3.1.3 sm_info

Frontend Kernprogramm für den **GUI:Anzeige laufende Meldungen**.

Beschreibung

Schreibt und liest Meldungen in bzw. aus dem SEPs Online Informationsbereich („Anzeige laufende Meldungen“).

Syntax: sm_info {P/S/M/C} [{Drive Nummer}] [{Text}]

Lesen aus dem Informationsbereich sm_info P [{Drive Nummer}]

Status Meldung schreiben sm_info S {Drive Nummer} {Text}

Meldung schreiben sm_info M {Drive Nummer} {Text}

Lizenz prüfen und ausgeben sm_info C

3.3.1.4 sm_init

Kernprogramm für den **GUI:Aktivitäten-Sofortstart-Medientermin**, bzw. terminierten Medientermin.

Beschreibung

Initialisiert ein Medium in einem Laufwerk, sodass es danach für Sicherungen bereitsteht. Dabei kann ein fixes Label oder ein Medienpool vorgegeben werden. Bei Angabe eines Pools wird das Label nach der **GET_OLDEST**-Strategie ausgewählt.

Schalter	Beschreibung
-d drive number	sesam-interne Nummer des Laufwerkes
-m media-pool	Name des Medienpools (unbenutzt, wenn -t gegeben)
-t label	Label mit dem initialisiert werden soll
-L typ	optional Logintyp; default: leer ; eom backup aus interner Steuerung
-i identifier	optionale Kennung für diese Ausführung ; wird intern gesetzt

Output:

exits with 0/1 = ok / error

schreibt GLBV gv_init_msg_'drive number' = tatsächlich initialisiertes Medium

3.3.1.5 sm_loader

Kernprogramm für den sog. Loaderhandler in **GUI:Komponenten-Lader-Laderaktion**;
Liegt eine Schicht über der SesamLoaderUtility SLU (vgl. 3.7).

Beschreibung

Laderprogramm zur Koordination der Laderfunktionen. Führt Umladesequenzen aus, hält Datenbank konsistent, setzt Mount-Zustand, schreibt Protokollmeldungen in das **SEPsam** Tagesprotokoll.

sm_loader import/export/load/unload/show + weitere Schalter

Schalter	Beschreibung
import	holt ein Medium aus einem IO-Port in ein Fach und nimmt die Fachinformation für das Mediumlabel in das Medienarchiv auf, sofern der Mediumlabel über Barcode bestimmt werden kann und das Medium bereits aufgenommen wurde.
export	transportiert ein Medium aus einem Fach in einen IO-Port und entfernt die entsprechende Fachinformation aus dem Medienarchiv
load	lädt ein Medium von einem Magazinfach in ein Laufwerk
unload	entlädt ein Medium aus einem Laufwerk in ein Fach, ggfs. wird das Medium zunächst entladen (offline)
show	zeigt die Befüllung des Laders, benutzt den Barcodeleser falls vorhanden; Ergebnis in <code>{IV}/var/work/showlist_{ln}_{dn}.tmp</code>
-l num	sesam-interne Nummer des Laders
-d num	sesam-interne Nummer des Laufwerks für load, unload, show sesam-interne Nummer des IO-Ports für export, import
-s num/label	Fach (Slot) Nummer oder Mediumlabel (Fach wird über Medienarchiv ermittelt) (obligatorisch für load, export und import; optional für unload und show)
-o no_mount	optional; geladenes Medium nicht in MOUNT-Zustand setzen
-L logfile	Umlenkung der Logging-Ausgabe in spezielle Datei
-v num	Verbose level 0, 1 oder 2
-h	Hilfe Ausgabe der Aufrufchnittstelle

Rückgabewert(e):

GLBV gv_loader_sts_{ln} Ergebnisstatus: +/- (o.k./mislungen)

GLBV gv_loader_msg_{ln} Wert oder Meldung

Beispiel

sm_loader import -l 1 -d 0 -s 0 >> Import Medium im Lader 1 von IO-Port 0 nach Fach 0

sm_loader unload -l 1 -d 2 s 5 >> Unload Medium im Laufwerk 2 des Lader 1 nach Fach 5

3.3.1.6 sm_recover

Beschreibung

SEPs Kommandozeilen Interface zur Analyse eines Mediums und für die Rücksicherung von Daten oder Listings eines Savesets.

Voraussetzung ist ein im Laufwerk geladenes Medium.

Im Benutzerdialog werden nötige Informationen abgefragt.

Mediumwechsel werden ggf. automatisch angestoßen.

Schalter	Beschreibung
-d drive_number	SEPs Laufwerksnummer
[-j SEPs task]	zum Filtern von Savesets für diesen Auftrag
[-o stop nostop disk_info]	stoppt ggf. SEPuler (Default: nostop) LINUX disk info Rücksicherung
[-f file]	Datei mit SEPs Rücksicherungskommando (z.B. Daten der Disaster Schnittstelle) in diesem Fall wird die Rücksicherung unmittelbar ohne zusätzliche Suche auf dem Medium gestartet
[-F file]	Datei wird mit den Daten vom Scan des Mediums gefüllt
[-R target-path]	Rücksicherungs Ziel (ohne abschließendem Pfadtrenner /) im Falle von -o disk_info gilt der Default /mnt/disk

Rückgabewert(e):

- Listing des Savesets auf dem Medium oder
- Rücksicherung des ausgewählten Savesets vom Medium ins Ziel Verzeichnis

Beispiel

sm_recover

3.3.1.7 sm_restore

Beschreibung

Zentrales Rücksicherungsprogramm

- stellt benötigte Medien zur Verfügung
- sortiert die Liste der selektierten Dateien (selektive Rücksicherung) gemäß LIS-Datei
- verwaltet die Abläufe der spezifischen Rücksicherungs-Programme
- steuert Generationsrestore

- protokolliert den Vorgang mit (Datenbank, Protokolldateien)

Parameter	Beschreibung
P1	Zielrechner/Benutzer/Ausführungszähler des Sicherungslaufs oder Rücksicherungsauftrag/Startzeit (Format Startzeit: YYYY-MM-TT hh:mm:ss) zum Neustarten eines Rücksicherungsauftrags (hier keine weiteren Parameter mehr nötig)
P2	Laufwerknummer/Interface
P3	modus = (new over not system image) ist ein '&' angefügt, so wird die Verzeichnisstruktur aufgebaut ansonsten wird 'flach' in das Zielverzeichnis restauriert
P4	SEPsam-Sicherungsauftrag
P5	Datum der Sicherung
P6	Mediumlabel
P7	Vollständigkeits-Flag = 'yes', Gesamt-Rücksicherung soll komplett ausgeführt werden oder Zeitstempel, die Datei {IV}/var/sel/{Zeitstempel}.sel enthält die Liste der selektierten Dateien
P8	Zielverzeichnis oder '//ORIGINAL'

Syntax

sm_restore P1 P2 P3 P4 P5 P6 P7 P8 oder
sm_restore P1

Hinweis

Für selektive Rücksicherungen sind Selektions-Dateien notwendig. Diese werden bei der selektiven Auswahl im Rücksicherungsassistent (GUI) erstellt, oder können per Editor aus den bestehenden LIS-Dateien erstellt werden.

Beispiel

sm_restore sesamx/DK/8 1/LOCAL 'not&' sesam_etc 04.04.2000 S00012 yes „//ORIGINAL“
Inhalt des Rücksicherungsauftrags wird in das Originalverzeichnis zurückgesichert. Vorhandene Original-Dateien werden nicht überschrieben und keine Dateibaum-Struktur wird aufgebaut

sm_restore decuberlin/root/1 1 over decuberlin_te6 08.08.2001 DISK00003 yes “/usr/test7“
Inhalt des Rücksicherungsauftrags wird in ein neues Zielverzeichnis zurückgesichert. Vorhandene Original-Dateien werden überschrieben.

sm_restore “20010802_101680/2001-08-08 14:32:28“
Neustart eines Rücksicherungsauftrags 20010802_101680, der vorher bereits erzeugt worden war.

3.3.1.8 sm_robot

Kernprogramm für den **GUI:Komponenten-Lader-Inhalt-Archivabgleich**

Überprüft im Ladermagazin alle Labels der Medien in den Fächern. Ist das Label bereits im Medienarchiv hinterlegt, werden Lader und Fach angepasst. Ist das Label nicht im Medienarchiv, wird das Medium in das Medienarchiv mit einer Überprüfung des Label-Formats übernommen (bei

Angabe von `-i` (introduce)).

Der Inhalt der Datenbank-Tabelle `media` und `loader_contents` wird aktualisiert und ein Eintrag in das Tagesprotokoll wird erstellt.

Eingabeparameter:

Schalter	Beschreibung
<code>-l num</code>	Lader-Nummer
<code>-d num</code>	Laufwerk-Nummer
<code>-i</code>	Eingangs-Flag (introduce), führt zur Neuaufnahme von neuen bzw. unbekanntem Medien ins Archiv
<code>-t typ</code>	Medien-Typ (nur bei Verwendung von <code>-i</code>)
<code>-m pool</code>	Medienpool (nur bei Verwendung von <code>-i</code>)
<code>-s num</code>	Fächer-Anzahl (Angabe wenn nicht alle Fächer bedient werden) Syntax: <code>xx-yy</code> z.B. 0-9
<code>-c</code>	Flag, wenn das Label des Mediums unabhängig vom erkannten Barcode zusätzlich geprüft werden soll
<code>-v</code>	Verbose-Flag (0/1/2)
<code>-h</code>	Hilfe Ausgabe auf Standard Output und Programm-Abbruch

Beispiel

```
sm_robot -l 2 -d 1 -i -t THXKD -m E -s 2-7 -cv >> Neuaufnahme aller unbekanntem Medien im Slot Bereich 2-7
```

```
sm_robot -l 2 -d 1 >> Archivabgleich
```

3.3.1.9 sm_sms_interface

Der Zugriff auf die SMS-Medien (Bänder, Disks) wird über das `sm_sms_interface` Programm realisiert. Die wichtigsten Aufrufe sind das Bestimmen des Labels (Get Label) und das Initialisieren des Mediums mit einem neuen Mediumlabel (Initialisieren). Im Problemfall können damit Bänder neu gelabelt werden.

Syntax:

```
sm_sms_interface command options
```

Initialisierung eines Mediums

```
sm_sms_interface init
-d {Device}
-B {Block-size} Default: 65536
-t {Label}: {Tape-ID}
-i {Tape Info}
-r {Rewind Flag} Default: R
-T {Timeout} Default: 120
```

Beispiel

```
sm_sms_interface init -d Tape1 -t S00007:7 -i Info -r R -T 60
```

Lesen des Mediumlabels

```
sm_sms_interface getlabel  
-d {Device}  
-r {Rewind Flag} Default: R  
-T {Timeout} Default: 120
```

Beispiel

```
sm_sms_interface getlabel -d Tape1 -r R -T 60
```

Lesen des Bandinhalts

```
sm_sms_interface dir -d {Device}
```

Beispiel

```
sm_sms_interface dir -d Tape1
```

Lesen des Tape Trailers (enthält ggf. Folgeband-Label)

```
sm_sms_interface read_tape_trailer -d {Device}
```

Beispiel

```
sm_sms_interface read_tape_trailer -d Tape1
```

Schreiben des Tape Trailers

```
sm_sms_interface write_tape_trailer -d {Device}  
-t {Label}:{Tape-ID}  
-l {NextTapeLabel}  
-i {TapeTrailerInfo}  
-e {EOMFlag} Default: F = FALSE  
-g {EMRGFlag} Default: F = FALSE
```

Beispiel

```
sm_sms_interface write_tape_trailer -d Tape1 -t S00001:7 -l S00008 -i Info -e T -g T
```

Entladen des Laufwerks

```
sm_sms_interface unload -d {Device}
```

Beispiel

```
sm_sms_interface unload -d Tape1
```

Daten auf Medium schreiben

```
sm_sms_interface backup_file -d {Device}  
-t {Label}:{Tape-ID}  
-f {Source File Name}  
-s {Saveset Name}  
-i {Saveset Info}  
-I {Interpreter} Default: none
```

Beispiel

```
sm_sms_interface backup_file -d Tape1 -t S00007:7 -f x.dat -s x.sav -i Saveset-Info
```

Saveset vom Medium lesen

```
sm_sms_interface restore_file -d {Device}  
-t {Label}:{Tape-ID}
```

-f {Target File Name}
 -s {Saveset Name}
 -i {Saveset Info}

Beispiel

```
sm_sms_interface restore_file -d Tape1 -t S00007:7 -f x.dat -s x.sav -i Saveset-Info
```

Erstellen eines LIS- und SGM-Files aus Saveset vom Medium

```
sm_sms_interface restore_listing -d {Device}  

-t {Label}:{Tape-ID}  

-n {Start Segment} Default: 0  

-b {Number of Blocks} Default: 0  

-s {Saveset Name}  

-i {Saveset Info}  

-I {Interpreter} Default: none
```

Beispiel

```
sm_sms_interface restore_listing -d Tape1 -t S00007:7 -n 15 -b 150 -s x.sav -i Saveset-Info
```

Rücksicherung eines Saveset direkt vom Medium

```
sm_sms_interface restore_emergency -d {Device}  

-t {Label}:{Tape-ID}  

-f {Target Pipe Name}  

-n {Start Segment} Default: 0  

-b {Number of Blocks} Default: 0
```

Beispiel

```
sm_sms_interface restore_emergency -d Tape1 -t S00007:7 -f sesam -n 15 -b 150
```

SMS restarten

```
sm_sms_interface restart_sms -d {Device}
```

Beispiel

```
sm_sms_interface restart_sms -d Tape1
```

Geänderte Konfiguration für laufenden SMS übernehmen

```
sm_sms_interface reread_ini -d {Device}
```

Beispiel

```
sm_sms_interface reread_ini -d Tape1
```

SMS Device Zustand abfragen

```
sm_sms_interface inquire_status -d {Device}
```

Beispiel

```
sm_sms_interface inquire_status -d Tape1
```

Für sämtliche hier dargestellten Funktionen von `sm_sms_interface` gibt es noch die Parameter

-v {n} für tracing Default: 0 (kein trace)
 -L {LoginType} Default: SHARE. Für Abfragen genügt INQUIRE, für Rücksicherungen ist EXCLUSIVE notwendig, im Zustand CHANGING_MEDIA wird nur EOM akzeptiert und für Administrative Zwecke dient ADMIN.

3.3.1.10 sm_sms_watch

Kernprogramm für die Kontrolle des Medienwechsels und für die Ausgabe des Bandzustandes im GUI Infofenster „Anzeige laufende Meldungen“ (siehe 3.3.1.3).

Beschreibung

Überwacht den Laufwerkszustand und übernimmt die Kontrolle beim Mediumwechsel. Kommuniziert im Falle eines „End Of Media“ (EOM) mit SMS zur Übergabe des neuen Bandlabels und mit sm_loader für den Wechsel des Mediums.

Syntax:

sm_sms_watch {Drive Nummer}

Hinweis

Im Falle eines Problems beim Bandwechsel ist das zugehörige sm_sms_watch Logging für die Problemanalyse hilfreich.

Hinweis

Beim Start eines Sicherungsauftrags bzw. Rücksicherungsauftrag wird geprüft ob sm_sms_watch in der zugehörigen Queue läuft und ggf. nachgestartet (siehe 3.3.3.3). Läuft ein sm_sms_watch Prozess bereits, so beendet sich ein nachträglich gestarteter Prozess.

Hinweis

Ohne dem sm_sms_watch Prozess für das jeweilige Laufwerk wird das Erreichen des Bandendes nicht erkannt! Der Mediumwechsel für Sicherung bzw. Rücksicherung wird nicht durchgeführt!

3.3.1.11 sm_startup

Startet die **SEPsam** Software.

- startet die Hintergrundprozesse (Services) (siehe 3.5)
- bereitet Laufwerke und Lader vor
- richtet die Queues ein
- richtet den aktuellen **SEPsam**- Sicherungstag ein

3.3.2 SEPuler (SEPsam Scheduler)

Zum SEPuler gehören die Programme `{IV}/bin/sesam/sm_sepuler{.exe}` und `./sm_sepuler_cal{.exe}`.

`sm_sepuler{.exe}` wird als Hintergrundprozess (Service/Dämon) gestartet (siehe 3.5) und verwaltet Termine in der Datenbanktabelle *terms*.

Ist die Ausführungszeit eines Termins erreicht, so ruft `sm_sepuler{.exe}` das Programm `{IV}/bin/sesam/sm_sepul_event{.exe}` mit den Termin Parametern auf.

In `{{IV}}/var/ini/sm.ini` können folgende Ausführungsparameter für den SEPuler gesetzt werden:

```

...
[SM_SEPULER]           -> Sektion für SEPuler
SLEEP=30               -> Arbeitstakt in Sekunden
EVENT=sm_sepul_event  -> Ereignis-Programm
...

```

Das Programm `{IV}/bin/sesam/sm_sepuler_cal{.exe}` dient zur Bearbeitung des Kalenderblattes, dessen Daten in der Datenbanktabelle `cal_sheets` abgelegt werden.

Das Logging der SEPuler-Programme wird in `{IV}/var/ini/debug.ini` eingestellt (siehe 4.6.1):

```

...
SM_SEPULER=1
SM_SEPULER_CAL=1
SM_SEPUL_EVENT=1
...

```

Es entstehen

```

{IV}/var/log/lgc/sepuler.lgc
{IV}/var/log/lgc/sepul.ev-_{Zeitplanname}-_{Zeitstempel}.lgc

```

3.3.3 Queue Manager

Der Queue Manager ist eine Software, die Queues (= Warteschlangen) erzeugt und verwaltet. In eine Queue können Jobs zur Ausführung mit Startzeitpunkt gestellt werden (`submit`). Der Queue Manager sorgt dann dafür, dass ein Job zur Startzeit aktiviert wird (**executing**), wobei immer nur so viele Jobs `executing` sein können, wie das Joblimit der Queue angibt. Jobs, deren Startzeit noch nicht erreicht ist, sind **holding**, deren Startzeit erreicht ist, die aber wg. dem Joblimit noch nicht laufen dürfen sind **pending**.

Im **SEPs**esam wird der Queuemanager zur Verteilung der Sicherungen, Rücksicherungen, Initialisierungen etc. auf die verfügbaren Laufwerke eingesetzt. Für jede Laufwerksgruppe wird eine gleichnamige Queue, die für jedes Laufwerk eine `executing` Queue besitzt, erzeugt.

Dies gestattet den gezielten Zugriff auf System-Ressourcen gemäß dem der jeweiligen Ressource zugeordneten Grad an paralleler Bearbeitung (Joblimit). Ist ein Laufwerk für maximal 4 gleichzeitige Sicherungen konfiguriert, so wird durch die Einschränkung der zugeordneten Queue auf maximal 4 Aufträge (Joblimit) genau der entsprechende Parallelitätsgrad erzwungen.

Die Zusammenarbeit zwischen SEPuler und Queuemanager geschieht in der Weise, dass der Benutzer im SEPuler Termine für Sicherungen, Rücksicherungen, Initialisierungen etc. festlegt und diese bei Erreichen der Ausführungszeit entsprechend der zu benutzenden Ressource an den Queue Manager übergeben werden. Dieser regelt dann gemäß der Auslastung der Ressource, wann die Aktion tatsächlich zur Ausführung kommt.

Es können also z.B. im Terminkalender 100 Backups auf ein Laufwerk zum gleichen Zeitpunkt vorgesehen sein. Diese werden an den Queue Manager übergeben, d.h. in die Warteschlange

des Laufwerks gestellt. Da das Laufwerk max. 60 SMS-Kanäle haben kann (siehe 3.3.6), werden dann vom Queue Manager 60 Backups sofort aktiviert und die restlichen sukzessive nachgeschoben, sobald ein Kanal wieder frei geworden ist.

Falls das Starzeitfenster der Backups knapp bemessen ist, kann es passieren, dass gegen Ende einige Backups nicht mehr zur Ausführung gelangen, was vom **SEPsam** gemeldet und mitprotokolliert wird.

Zum Queue Manager gehören die Programme

`{IV}/bin/sesam/sm_qm_main{.exe}` und `./sm_qm{.exe}`.

`sm_qm_main{.exe}` ist der Queue Manager-Server und wird als Hintergrundprozess (Dämon) gestartet (siehe 3.5), der auf Anforderungen über den Queue Manager-Client `sm_qm{.exe}` wartet und die Queues verwaltet.

Syntax sm_qm:

Der Aufruf hat die Form: `sm_qm {request} {weitere Argumente}`

Request	Argumente	Beschreibung
block	QueueName[/Ressource]	erstellt Queue, mit Ressource wird eine bestimmte Executing Queue ausgewählt
delete	QueueName	entfernt die Queue
init	QueueName[/Ressource[...]] Joblimit	erstellt Queue mit Joblimit, mit „Ressourcen“ können mehrere Executing Queues festgelegt werden
sho	QueueName all	zeigt die Queue bzw. alle Queues und deren Inhalte und Zustände
sho/name	QueueName all	zeigt die Queue bzw. alle Queues mit Jobnamen
sho/full	QueueName all	zeigt die Queue bzw. alle Queues mit Ausführungsparametern
start	QueueName[/Ressource]	startet die Queue nach stop oder block
submit	QueueName,Programm + Argumente	stellt Programm in die Queue unter einer Entrynummer
job_sho	QueueName Jobname [Count]	zeigt Entrynummer und Status für „count“ Jobname
del_entry	Entrynummer	beendet den über Entrynummer ausgewählten Job (Prozess)
shutdown		beendet den sm_qm_main Prozeß (delete Queues und Jobs)

Beispiel

```
# sm_qm -h
$Id: admin_handbuch.tex,v 1.32 2007/01/25 08:41:44 sesam Exp $
-----
Usage: sm_qm {command} ...      {command} ... as shown below
block   {queue}[/ressource]    blocks queue's holding and pending jobs
delete  {queue}                deletes the queue
init    {queue}[/ressource][...] {job_limit}    creates queue with job limit
                                             every ressource defines an executing queue
```

```

sho      {queue}|all          shows queue or all queues with file names
sho/name {queue}|all          shows queue or all queues with job names
sho/full {queue}|all          shows queue or all queues with parameters
start    {queue}[/{ressource}] restarts queue or ressource
stop     {queue}[/{ressource}] stops queue or ressource and kills running jobs
submit   {queue}[/{allocate}[/{ressource}|all]]
        {starting time}      format: hh:mm
        {starting date}      format: dd-mm-yyyy
        {priority}           starting priority: integer
        {job}                 job name as you like
        {file}                full qualified file name to execute
        [{argument} [...]]   optional job arguments for execution
                             creates a job with entry number in queue
                             optional: allocate ressource (default: unknown)
                             preferred ressource
job_sho  {queue} {job} [{count}] shows entry and status of [count] job
del_entry {entry_number}      deletes the given job
shutdown                               shutdown of sm_qm_main; delete queues & jobs

```

3.3.3.1 Queue Manager Konzept

In SEPsesam wird jeder Sicherung ein Medienpool zugeordnet. Ein Medienpool ist immer eindeutig einer Laufwerkgruppe zugeordnet. Üblicherweise werden mehrere unterschiedliche Medienpools derselben Laufwerkgruppe zugeordnet. z.B. eine Laufwerkgruppe für FULL/DIFF/INCR, Wochenend/Werktags oder ORACLE/Filesystem.

Für die Abarbeitung eines Sicherungsauftrags ohne Laufwerksangabe muss ein geeignetes Laufwerk aus der Laufwerkgruppe ausgewählt werden. Ist kein Laufwerk mit einem Medium des gewünschten Medienpools befüllt, so muss eine Strategie zur Einbringung eines Mediums abgearbeitet werden.

Gemäß Konzept können Medien der Medienpools dynamisch (on demand) angefordert werden.

3.3.3.2 Queue je Laufwerkgruppe

Die Sicherungsaufträge werden zunächst in Laufwerkgruppen-Queues, gemäß der Zuordnung Medienpool zu Laufwerkgruppe, eingebracht. Die Laufwerkgruppen-Queue besitzt eine holding bzw. pending Queue und je Laufwerk eine executing-Queue.

Jeder Job und die executing-Queue besitzen das Attribut „allocate“. Einem Job wird mit diesem Attribut eine Belegung (i. allg. Medienpool) zugeordnet. Eine Executing Queue wird mit dem Attribut des Jobs belegt, sobald ein Job in dieser Queue gestartet wird.

Ein Job kann nur für eine executing-Queue gestartet werden, wenn

- die executing-Queue unbelegt („deallocated“), oder
- die executing-Queue bereits identisch belegt ist (Attribut allocate ist identisch).

Zusätzlich kann für einen Job eine Vorauswahl der executing-Queue, also eine bestimmte executing-Queue (Laufwerk) übergeben werden (Attribut Preferred Ressource).

3.3.3.3 Bandendekontrolle je Laufwerk

Der SEPSAM Prozess `sm_sms_watch` dient zur Überwachung des Bandendes eines Mediums während einer Sicherung oder Rücksicherung. Er prüft den Laufwerkszustand und gibt die Laufwerksinformationen (Throughput, Status, etc.) an das SEPSAM Informationssystem weiter (siehe auch 3.3.1.3).

Dieser Prozess wird in einer eigenen Queue je Laufwerk, nämlich der Queue „`qu_tape_trans{Drive Nummer}`“ abgearbeitet. Damit ist sichergestellt, dass dieser Prozess unabhängig vom Zustand der Laufwerkgruppe das Laufwerk überwacht.

3.3.3.4 Abarbeitung der Jobs

Die Abarbeitung der Sicherungsaufträge erfolgt in den Laufwerkgruppen-Queues. Die Jobs werden gemäß Priorität und FIFO (first-in-first-out) verarbeitet.

Das Programm sucht ein verfügbares Laufwerk, d.h. das erste Laufwerk, welches für den gewünschten Medienpool belegt ist und in dem noch Platz für den Sicherungsauftrag vorhanden ist. Existiert kein entsprechendes Laufwerk, so wird das erste unbelegte Laufwerk verwendet. Sind alle Laufwerke belegt, so wird gemäß FIFO der nächste Job mit gleicher Priorität aber anderem Medienpool untersucht.

Sicherungen mit fester Laufwerkszuordnung (wahlweise) werden im jeweiligen Laufwerk gestartet, sobald dieses dazu geeignet ist (unbenutzt bzw. für gewünschten Medienpool belegt).

Die Abarbeitung wird gestartet, wenn

- ein neuer Job in die Laufwerkgruppen-Queue eingebracht wird (submit),
- ein Job aus der Holding-Queue in die Pending-Queue wechselt (Ablauf der Holding Time)
- ein Platz in einer Executing-Queue frei wird, d.h. ein Job in einer Laufwerks-Queue beendet wird.

Die Abarbeitung der Laufwerkgruppe stoppt, wenn

- alle Jobs mit der aktuell höchsten Priorität abgearbeitet sind,
- kein freier Platz in den Laufwerks-Queues vorhanden ist.

Ein unbelegtes Laufwerk wird durch die Belegung für den jeweiligen Medienpool reserviert.

Der Sicherungsauftrag versucht, ein verfügbares Medium zu finden und zu laden. Ggf. wird ein neues Medium angefordert.

3.3.3.5 que_init für Laufwerkgruppen

Der Aufruf für das Anlegen einer Laufwerkgruppen-Queue erfolgt folgendermaßen:

```
sm_qm init <queue-name>[/<ressource>][/...] <job_limit>
```

Beispiel

```
sm_qm init loader1/1/2/3 5      ## Laufwerkgruppe:   loader1
                                ## Drive Nummern:      1, 2 und 3
                                ## Joblimit je Drive ist 5
```

3.3.3.6 Übergabe Medienpool und optionale Laufwerknummer

Wird ein Job in die Queue submitted, so muss das Attribut allocate mit dem gewünschten Medienpool belegt werden. Ist zusätzlich eine Laufwerksnummer vorgegeben, so muss auch diese übergeben werden.

Wenn allocate nicht per Commandline übergeben wird, so wird als Standardwert 'unknown' verwendet.

Demnach gilt:

```
sm_qm submit <queue-name>[/<allocate>[/<ressource>]]
                                kreiert einen Job mit Entry Nummer in der
                                angegebenen Queue
                                optional: allocate (Default: 'unknown')
                                preferred Ressource

                                <starting time>      Format hh:mm
                                <starting date>      Format dd-mm-yyyy
                                <priority>           Integer
                                <job-name>           Bezeichner
                                <file>              voll qualifizierter Programmname
                                <par1>...<par8>     optional max. 8 Parameter
```

Beispiel 1: Job oracle_P01 mit Medienpool ORACLE auf Laufwerkgruppe loader1

```
sm_qm submit loader1/ORACLE 12:00 04-04-2001 3 oracle_P01
/opt/sesam/bin/sesam/sm_auto_backup -l F -T "12:00 04-04-2001"
-W "14:00 04-04-2001" -m ORACLE oracle_P01
```

Beispiel 2: Job linux1_full mit Medienpool FULL auf Laufwerkgruppe loader1 mit Drive_num=1

```
sm_qm submit loader1/FULL/1 12:00 04-04-2001 3 oracle_P01
/opt/sesam/bin/sesam/sm_backup -l F -T "12:00 04-04-2001"
-W "14:00 04-04-2001" -m FULL linux1_full
```

3.3.3.7 Übergabe Laufwerksnummer an auszuführendes Programm

Wird ein Job in der Executing Queue gestartet, so benötigt dieser die Laufwerksnummer. Diese Nummer wird deshalb als zusätzlicher Parameter beim Start des jeweiligen Programmes (z.B. sm_backup) an den Kommandoaufruf übergeben (-d drive_number).

Aufruf über: sm_qm submit ... (ohne „-d 2“):

```
sm_backup -j SESAM_BACKUP -m DISK ... -d 2
                #^^^^ von sm_qm_main wird die Ressource
                #      übergeben
```

Ein gerufener sm_backup erhält als Argument das Laufwerk auf dem er tatsächlich gestartet wird. sm_backup trägt beim Start sofort die Laufwerksnummer in die Tabelle results ein.

3.3.3.8 Verhalten falls gewünschtes Laufwerk blockiert ist

Wurde eine Laufwerksnummer übergeben, so wird diese, falls die Ressource geblockt ist (z.B. da das Laufwerk defekt ist), übergangen.

Der Queue Manager wählt stattdessen ein anderes Laufwerk mit der gewünschten Belegung (Pool) aus.

3.3.3.9 Laufwerke sperren

Ist ein Laufwerk, defekt so muss es vorübergehend gesperrt werden. Die Funktion block wird hierzu verwendet.

```
sm_qm block <queue-name>[/<ressource>] # blockiert holding und pending Jobs
                                           # Falls Ressource angegeben ist wird
                                           # die jeweilige executing Queue
                                           # blockiert
```

Beispiel

```
sm_qm block loader1/3          ## sperrt Laufwerk 3 in der Laufwerkgruppe loader1
```

Hinweis

Die Queues für Laufwerksgruppen werden beim Konfigurieren der Laufwerke erneut initialisiert und gestartet. Dies erfolgt bei Änderung der Konfiguration und beim SEPSAM Startup.

3.3.3.10 Monitoring Laufwerkgruppen Queues

Die Funktion sho zeigt bei einer MultiQueue die Attribute allocate, Ressource und den Zustand aller executing Queues.

```
sm_qm sho [/full] [/name] <queue-name>
```

Beispiel

```
sm_qm sho/name loader1
```

```
que_name= loader1      job_limit= 5      status= running
Entry  Job_name           Time    Date              Priority  Allocate  Preferred-Ressource

holding queue

pending queue
32     oracle_P01        12:30   3-4-2001          1         ORACLE    1
30     oracle_C01        12:30   3-4-2001          1         ORACLE

executing queue  Ressource= 1      status= running    allocated= DIFF
2      bfasco_2bnd       10:24   3-4-2001          1         DIFF
31     bfasco_ex1       12:30   3-4-2001          1         DIFF

executing queue  Ressource= 2      status= running    allocated= FULL
4      bfasco_2         10:24   3-4-2001          1         FULL      2
17     bfaaix_2         12:30   3-4-2001          1         FULL      3

executing queue  Ressource= 3      status= blocked    allocated=

executing queue  Ressource= 4      status= running    allocated= INCR
29     bfasinix_2       10:24   3-4-2001          1         INCR      2

executing queue  Ressource= 5      status= running    allocated= RESTORE (hard)
-----
```

3.3.3.11 Jobs ohne Medienpool Zuordnung

Einem Sicherungsjob muss immer ein Medienpool zugeordnet werden. Andere Jobs ohne Medienpool, die ein Laufwerk exklusiv belegen wollen, generieren einen eindeutigen Wert und belegen damit das Laufwerk. Damit ist sichergestellt, dass z.B. während einer Medienneuaufnahme oder einer Rücksicherung das Laufwerk exklusiv belegt wird.

Nicht-Sicherungs-, Rücksicherungs-, bzw. Medieninit-Jobs wie, z.B. sm_start für ein Laufwerk, werden in der Queue qu.all abgearbeitet.

3.3.3.12 Logging der Queue Manager Programme

Das Logging der Queue Manager Programme wird in **IV**/var/ini/debug.ini eingestellt (siehe 4.6.1):

```
...
SM_QM=0
SM_QM_MAIN=1
...
```

Es entstehen

```
{IV}/var/log/lgc/sm_qm.lgc
{IV}/var/log/lgc/sm_qm_main.lgc
```

3.3.4 Die Datenbank

SEPsam hält seine internen Konfigurations- und Laufzeitdaten in einer Datenbank.

Die zugehörigen DBF-Dateien liegen unter **{IV}**/var/db/schema/sesam.

Verwaltet wird die Datenbank über den DB-Server **sm_db_main** (siehe 3.5).

Die Tabellenstruktur der DB wird hier nicht explizit gezeigt, kann aber bei Bedarf von der **SEP AG** als SQL-Datei angefordert werden.

Auf Kommandoebene kann über das Programm

```
{IV}/bin/sesam/sm_db{.exe}
```

auf die DB mit SQL-Anweisungen zugegriffen werden.

Syntax:

```
sm_db "SQL-Anweisung"
```

```
sm_db "SQL-Select-Anweisung" export > Datei
```

=> Inhalt in Form von SQL-Insert-Anweisungen für Reimport

Ein weiteres Programm zur Bearbeitung der DB ist

```
{IV}/bin/sesam/sm_dbf_pack{.exe}.
```

Da gelöschte Datensätze in den DBF-Dateien nicht wirklich gelöscht werden sondern nur als gelöscht markiert werden, nehmen diese Dateien im Laufe der Zeit ständig an Umfang zu. Um dies zu vermeiden, können die DBF-Dateien mit Hilfe von **sm_dbf_pack** "gepackt" werden, d.h. die als gelöscht markierten Datensätze werden tatsächlich gelöscht.

Zusätzlich können DBF-Dateien analysiert werden.

Syntax:

```
sm_dbf_pack [p/P/a/A] {DBF-filename.dbf} {records file name}
```

p	packe alle Sesam-DBF-Dateien	kein weiteres Argument
P	packe das gegebene DBF	1 Argument
a	analysiere das gegebene DBF (alle Bytes nummeriert ausgegeben)	1 Argument
A	analysiere das gegebene DBF (alle Records)	1 Argument

Der automatische Tageswechsel des **SEPs**esam (NEWDAY-Termin) führt u.a. das Packen aller DBFs durch.

Die Ausführung kann durch Änderung in der Datei `{IV}/var/ini/sm.ini` unterdrückt werden.

```
...
[SM_DBF_PACK]           # Sektion für sm_dbf_pack
AUTOSTART=YES           # YES -> sm_dbf_pack beim NEWDAY ausführen
...
```

3.3.5 Sesam Multi Stream - SMS

Für die Aufzeichnung der Sicherungsdaten wird ein spezielles Multi-Stream-Verfahren verwendet. Dieses Verfahren ermöglicht die Übertragung der Sicherungsdaten von mehreren Sicherungsklienten gleichzeitig auf ein Medium. Mit diesem Verfahren können Laufwerke bei entsprechend hoher Datenrate im sogenannten „streaming mode“ betrieben werden. Die Aufzeichnung der Daten erfolgt mit maximaler Aufzeichnungsgeschwindigkeit, Stop- und Repositionierungszeiten entfallen, Netzbelastung sowie Transferzeiten werden auf ein Minimum reduziert.

Zusätzlich werden zu Beginn einer Sicherung und im weiteren Verlauf besondere Markierungen (file marks) auf dem Band hinterlegt, wodurch ein extrem schnelles Positionieren auf die Sicherungsdaten im Falle der Rücksicherung möglich ist. Dies gilt insbesondere für die Rücksicherung von einzelnen Dateien. Durch das verwendete Verfahren wird lediglich auf die letzte Markierung vor Beginn der benötigten Daten positioniert und die entsprechenden Daten über den stpd an den Sicherungsklienten weitergereicht.

Das eingesetzte Verfahren ermöglicht das Schreiben von Sicherungsdaten über Bandgrenzen, sodass die Sicherung beliebig großer Datenmengen möglich ist (siehe 3.3.1.10).

In `{IV}/var/ini/sms.ini` werden die Laufparameter des SMS vorgegeben.

Hinweis

Diese Datei wird durch die SEPs sesam Konfiguration der Laufwerke (`sm_config.drives`) automatisch angepasst, wenn sich die Laufwerke-Konfiguration (GUI) ändert. Die Konfiguration wird auch beim Startup und beim SEPs sesam Tageswechsel erneut durchlaufen.

```
[SMS_Device]
Maximal_Number_Of_Devices=2
Device_Name_0=disk1
Device_Name_1=/dev/nst0

[SMS_Disk]
Directory=/opt/sesam/var/ini/

[SMS_Tape]
Tapemark_Type_0=SETMARK
```

```
Tapemark_Type_1=FILEMARK
Default_Segment_Size=1000
Tape_Online_Timeout=120
Max_Retries_Num=20
Retry_Delay_0n_1106=1
Max_Repeat_0n_1106=20
Repair=TRUE
```

```
[SMS_Server]
Throughput_Interval=5
Alarm_Interval=30
Maximal_Number_Of_Parallel_Streams_0=9
Maximal_Number_Of_Parallel_Streams_1=9
Bin_Directory=/opt/sesam/bin/sms
Ctl_Directory=/opt/sesam/var/work/ctl
```

```
[SMS_Logging]
#Log_Levels=0 1 2 3 4 5 6
# -1=NO_LOG, 0=ERROR, 1=EMERGENCY, 2=WARNING, 3=NOTICE, 4=INFO, 5=DEBUG, 6=TRACE
Log_Levels=0 1 2 3 4
Log_Size=256
Log_Max_Files=10
Log_Details=0 1 2
Log_Directory=/opt/sesam/var/log/sms
Perf_Directory=/opt/sesam/var/work/info
```

Der Zugriff auf die SMS-Medien (Bänder, Disks) wird mit dem STPD Service/Dämon (siehe 3.3.6) oder über das **sm_sms_interface** Programm realisiert (siehe 3.3.1.9).

Loggings sind unter dem Verzeichnis **{IV}/var/log/sms** abgelegt.

3.3.6 Datenübertragung mit SesamTransferProtocol - STP

Der STPD (STPDaemon) **{IV}/bin/sesam/sm_stpd_main{.exe}** dient als spezieller FTP-Server zur Übertragung der Sicherungsdaten vom Klienten zum Aufzeichnungsgerät (Band- oder Plattenlaufwerk) über SMS (siehe 3.3.5).

Der STPC (STPClient) stellt über das Netzwerk eine Datentransferverbindung zum STPD her.

Für die Übertragung wird das FTP-Protokoll verwendet. Dies gewährleistet einen hohen Datendurchsatz und die Verträglichkeit mit allen Netzkomponenten.

Die Daten werden direkt an das Aufzeichnungsgerät übermittelt, eine Zwischenspeicherung entfällt (vgl. deutsches **SEPs** Patent Nr. 4335836 und US-Patent Nr. 5,901,331).

Eine spezielle Analyse der Sicherungsdaten ermöglicht das Erkennen von Fehlern, z.B. ob während der Sicherung eine Datei in ihrer Größe verändert wurde. Zu Beginn der Sicherung wird die Dateigröße in den Sicherungssatz übernommen. Ändert sich während der Sicherung die Dateigröße, so erkennt die Analyse dies und verändert die Dateigröße so, dass die Längenangabe eingehalten wird.

Zusätzlich werden während der Analyse auch der Inhalt des Sicherungssatzes (Listing) sowie die Positionsdaten (Segment-Info) in spezielle Dateien übertragen.

Der Prozess `sm_passwd` wird vom STPD für die Authentifizierung von Systembenutzern verwendet (siehe auch 3.5).

Syntax

Prüfung mit ftp, Default-Port: 11001, siehe `{IV}/var/ini/stpd.ini` STPD_PORT

```
# ftp
ftp> open linux 11001
Connected to linux.sep.de.
220 STP Service Ready. Version: 2.8 R16. PID: 25854. Type HELP for list of commands.
Name (linux:root): sms
331 Password required for sms.
Password:
230 User logged in, proceed.
Remote system type is Linux.
ftp>cd SMS:disk7:INQUIRE
250 CWD Command successful.
ftp> quote site inquire
213-SMS Inquire status begin
  STATUS:SMS_WRT
  OP_COUNT:BLOCKS:0.000000
  AMOUNT:BYTES:91553792.000000
  THROUGHPUT:GB/H:2.504883
213 SMS Inquire status end
ftp> bye
221 Service closing control connection.
```

Hinweis

User muss STPD User sms oder System User sein.

Mit `cd SMS:tape:login-type` wird zum SMS Dataserver verbunden.

tape muss entsprechend des Eintrags in der `{IV}/var/ini/sms.ini` gewählt werden, z.B. Tape0 (WNT), `/dev/nst0` (Linux), `/dev/nrmt0h` (UNIX) oder `disk1` für DISK device.

Als login-type wird „SHARE“ bei gemeinsamer Nutzung für Backup, „EXCLUSIVE“ bei Rücksicherungen und „INQUIRE“ bei Anfragen verwendet.

Bei Fehler gemäß Fehlermeldung reagieren

z.B. Angabe korrigieren
oder Zustand des SMS Dataservers abwarten bzw. ändern
bei 'SMS not in allowed state' ist dieser EXCLUSIVE (Rücksicherung) belegt. Dann kann nur mit login-type „INQUIRE“ eine Verbindung zum Dataserver hergestellt werden.

Backup Test

Mit der folgenden FTP Anweisung kann eine Datei auf das eingelegte Medium übertragen werden.

put Datei Saveset:device:analyzer:tape ID:tapelabel:IP Bufferize

z.B. put test.dat SF123:/dev/nvrmt0h:cpio:4:S00004:32

analyzer none = keine Analyse

cpio = cpio-Format

mtf = mtf-Format

Restore Test

Mit der folgenden FTP Anweisung kann eine Datei vom eingelegte Medium gelesen werden.

get saveset:tape ID:segment Number:tape

z.B. get SF123:4:0:4

In {IV}/var/ini/stpd.ini werden die Laufparameter des STPD vorgegeben.

```
[STPD_Server]
STPD_PORT=11001                # TCP/IP Port
STPD_MAX_CONNECTIONS=64
Bin_Directory=C:\su\bin\sms
Ctl_Directory=d:\su\var\work\ctl
Work_Directory=d:\su\var\work

[STPD_Thread]
STPD_BUFSIZE=4
DATA_TIMEOUT=900
AUTH_USERS=sms,PSL            # autorisierte Benutzer
UPDATE_THROUGHPUT=30
STPD_ACCEPT=TRUE
STPD_ID=THIS_IS_KEY
STPD_USER=root                # _TMP_ ermöglicht Zugang ohne User Prüfung

[STPD_Logging]
Log_Levels=0 1 2              # max. 0 1 2 3 4 5 6
Log_Details=0 1 2
Log_Size=64
Log_Max_Files=10
Log_Directory=d:\su\var\log\sms

[STPD_Inquire]
InquireLog=d:\su\var\work\info\inq.log
```

Inquire_Timeout=10

Hinweis

Die maximale Anzahl der SMS Kanäle ist auf 64 beschränkt, d.h. für Sicherungen können maximal 60 SMS Kanäle verwendet werden. Die restlichen 4 Kanäle werden für administrative Zwecke, Bandwechsel etc. benötigt.

Loggings sind unter dem Verzeichnis `{IV}/var/log/sms` abgelegt.

3.4 Kommunikation Server - Klient

Die Kommandoausführung am Klienten wird gemäß der konfigurierten Accessmethode (GUI) durchgeführt. Zusätzlich können Optionen (z.B. -s für Single Port Option) konfiguriert werden (siehe 5.6). Im folgenden werden die Accessmethoden für den SEPs sesam proprietären CTRL und den SecureShell (SSH) Zugang beschrieben.

3.4.1 Kommandoausführung am Klienten - CTRL-Dienst

Die Kommunikation zwischen den einzelnen Rechnern erfolgt über die **SEPs sesam-CTRL-Dienste**, einer **SEP**-eigenen Entwicklung auf TCP/IP-Basis.

Über einen CTRL-Client `sm_ctrlc` kann an einem anderen Rechner der CTRL-Daemon (= Server) `sm_ctrlld` angesprochen und **SEPs sesam**-spezifische Aktionen gestartet werden.

Der CTRLD wird auf UNIX als Dämon, auf WINDOWS als Service im Hintergrund gestartet. Durch ein proprietäres Protokoll und die am Klienten hinterlegten erlaubten Knoten, Benutzer, Verzeichnisse und Kommandos kann eine Fremdbenutzung vermieden werden. Das CTRLD-Programm muss dazu unter einem Superuser-Account laufen.

Die Zugänge zum CTRLD werden in der Datei `{IV}/var/ini/sm.ini` konfiguriert:

```

...
[CTRLD_Server]                # Sektionsname
CTRLD_Port=11301              # TCP/IP-Port des STPD
Max_Connections=20            # max.Verbindungen
Bin_Directory=/su/bin/sesam   # Verzeichnis des sm_stpd\{.exe\}
Work_Directory=/su/bin/sesam  # Arbeitsverzeichnis; es dürfen nur von/nach
                                hier Dateien kopiert werden
CTRLD_Path=/su/bin/sesam;/su/bin/sms # nur in diesen Verzeichnissen dürfen Pro-
                                gramme gestartet werden

[CTRLD_Logging]              # Sektionsname
Log_Directory=/su/var/log/lgc  # Verzeichnis der Loggingdateien
Log_Level=0                   # 0 = kein Logging; bis max. Level 6 mgl.
Log_Details=0 1 2             # bis max. Level 3 mgl
Log_Size=256                  # max.Stückelungsgrösse der Loggingdateien,

```

```
[CTRLD_Auth]                # Sektionsname
Auth_File=/su/var/ini/sm_ctrlld.auth # Datei, die die zugelassenen User beinhaltet

Einträge in der Autorisierungsdatei sm_ctrlld.auth :
...
softnt6.sep.de root system SYSTEM # Rechnername, Userliste
qswnt2000.sep.de root system SYSTEM # Rechnername, Userliste
...
```

sm_ctrlc

Unter Verwendung der Funktion gethostbyname versucht sm_ctrlc, den angegebenen Hostnamen aufzulösen. Existiert dieser Hostname nicht, wird „-1“ zurückgegeben. Im Erfolgsfall wird zu diesem Hostnamen eine Verbindung aufgebaut. Der Port wird als INET-Port verwendet und der Befehl als Kommando für die Ausführung am Remote Host verwendet.

Die Verbindung wird über einen Socket in der Internet Domain vom Typ SOCK_STREAM (TCP-Verbindung) hergestellt.

Der Aufrufer übergibt seinen STDOUT an das remote Kommando als STDIN. Dieses erzeugt Ausgaben auf STDOUT und STDERR.

Werden Firewalls zwischen SEPSAM Server und Klient verwendet, so kann mit speziellen Optionen (-s, -o port=...) die Portbenutzung eingeschränkt werden (siehe 5.6).

Syntax

```
sm_ctrlc [-p port] [-l user] [-s] [-o port={first}-{last}] {host} {command}
```

Eingabeparameter:

{host} Hostname des Klienten
 {command} auf dem Klienten auszuführender Befehl

optionale Eingabeparameter:

```
-p port          # Listening Port des remote CTRLDs
-l user         # Remote User (siehe sm_ctrlld.auth)
-o port={first}-{last} # Port Range für Firewall Nutzung
-s             # Kommunikation nur über single Port, d.h. über Listening Port
```

Beispiel

```
sm_ctrlc -p 11301 -l root qlinux1.sep.de sbc
```

3.4.2 Kommandoausführung am Klienten - SSH(secure shell)

Sesam kann für die Kommandoausführung auf dem Sicherungsklienten auch das ssh-Protokoll verwenden. Voraussetzung ist, dass am Server ein ssh-Client-Programm und am Sicherungsklienten

ein `sshd` (Service/Dämon) installiert ist und sich der Server ohne Passworteingabe als User `root` für UNIX und `system` für Windows am Klienten anmelden kann.

Dies wird durch Erzeugung eines privaten und eines public Keys am Server und der Übertragung des public Keys zum Sicherungsklienten erreicht. Details entnimmt man der jeweiligen Dokumentation der jeweils eingesetzten SSH Implementierung.

Werden Firewalls zwischen SEPsesam Server und Klient verwendet so kann mit ssh Optionen die Portbenutzung eingeschränkt werden (siehe 5.6). Die Optionen werden im GUI:Klienten-Eigenschaften im Feld „Access Optionen“ eingetragen.

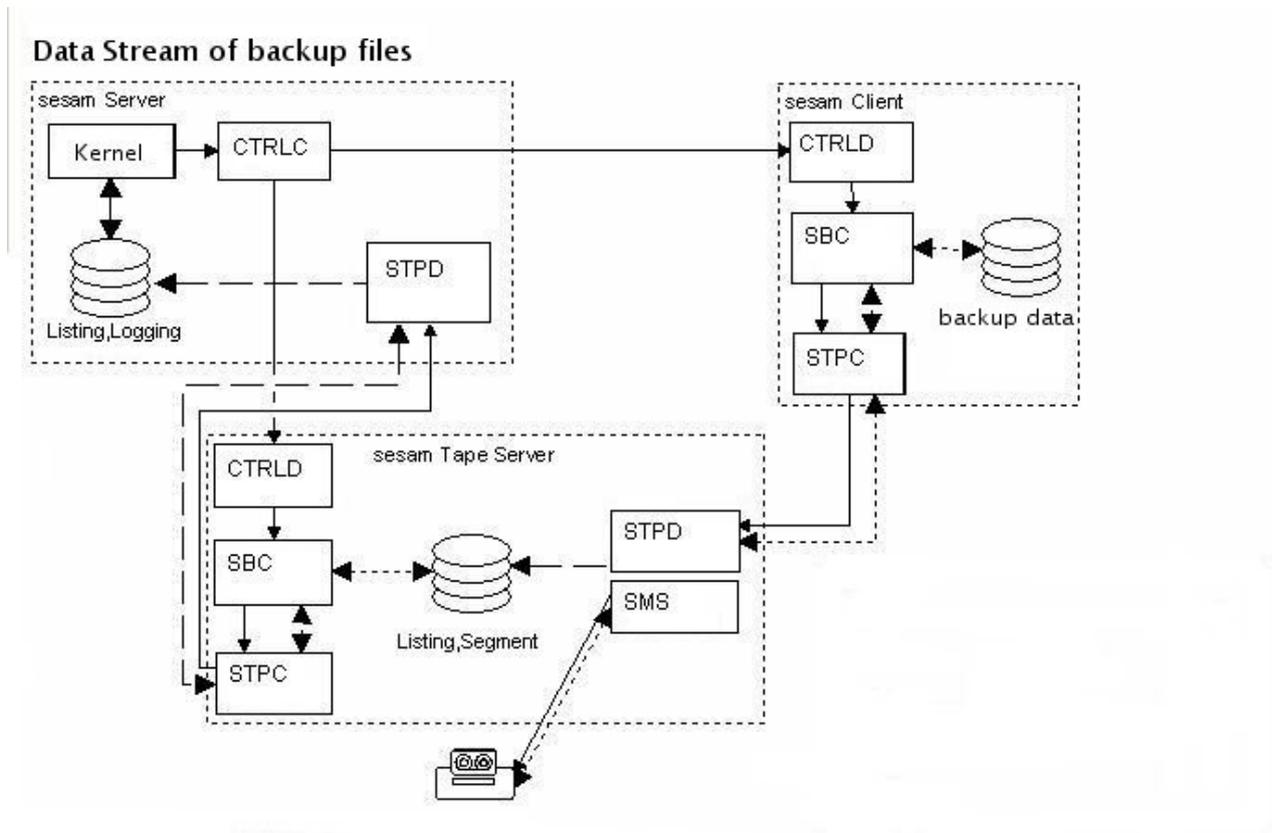
3.5 Prozesse und Datenfluss

SEPsesam erzeugt und benutzt folgende Hintergrundprozesse (Services):

<code>sm_main\{.exe\}</code>	Sesam Hauptprozess
<code>sm_sepuler\{.exe\}</code>	SEPuler
<code>sm_db_main\{.exe\}</code>	Datenbank
<code>sm_qm_main\{.exe\}</code>	Queuemanager
<code>java\{.exe\}</code>	Java für GUI
<code>sm_passd\{.exe\}</code>	Zugangskontrolle
<code>sm_ctrlld_main\{.exe\}</code>	CTRLD-Dämon
<code>sm_sms_main\{.exe\}</code>	SMS Server
<code>sm_data_server\{.exe\}</code>	SMS Datenserver Laufwerk 1
<code>sm_data_server\{.exe\}</code>	SMS Datenserver Laufwerk 2
<code>sm_stpd_main\{.exe\}</code>	SesamTransferProtocolDaemon STPD

Diese Prozesse werden mit `sm_main{.exe}` verwaltet:

```
sm_main install # Einrichten des SEPsesam Services, nur auf Windows
sm_main remove # Entfernen des SEPsesam Services, nur auf Windows
sm_main start   # Starten des SEPsesam Server
sm_main stop    # Stoppen des SEPsesam Server
sm_main restart # Restarten des SEPsesam Server
sm_main reload [main|node|sepuler|db|qm|rmi|sms|stpd|passd|ctrl]
                # Einzel Prozess erneut starten
sm_main status  # Status Abfrage der SEPsesam Server Prozesse
```

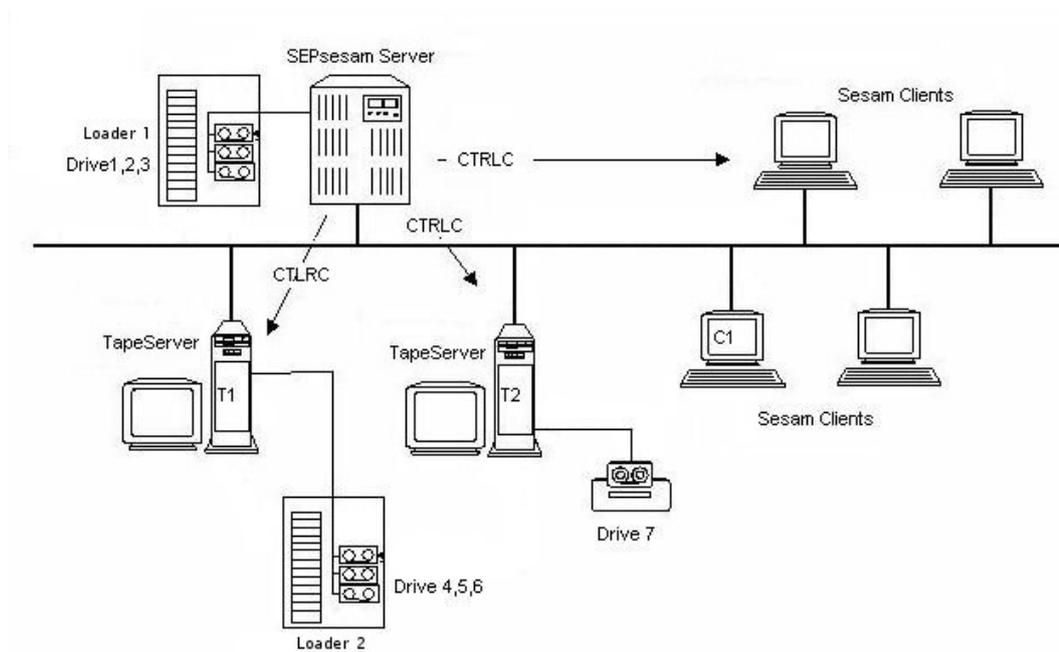


Die Grafik zeigt das Zusammenspiel der einzelnen Funktionseinheiten, die hier in Komponentensicht dargestellt sind, d.h. diese können ggf. auf einem einzigen Rechner kombiniert sein.

So hat der **SEPsam**-Server selbst die Server- und Tapeserverkomponente, ein reiner Sicherungsklient nur die Client-Komponente usw. .

Ebenso können die Datenflüsse beliebig von den Quellrechnern zu Tapeservern gelenkt werden, wobei die oberste Kontrolle immer vom **SEPsam** ausgeübt wird.

3.6. GLOBALE VARIABLEN (GLBV)



Das obige Beispiel zeigt einen **SEPsasam**-Server mit einem lokalen Lader 1, der 3 lokale Laufwerke 1,2,3 bedient.

Desweiteren einen Rechner T1 mit gleichartigem Lader, sowie T2 mit Einzellaufwerk und mehrere Clients.

Auf T1 und T2 ist die TapeServer-Komponente, auf C1 die Client-Komponente installiert.

In der Konfiguration werden dann T1, T2 und C1 als Klienten, 2 Lader und 7 Laufwerke aufgenommen.

Durch die Konfiguration der Sicherungsaufträge können folgende Datenflüsse erzeugt werden:

Klient	Zieldrive	Datenfluss
C1	D1	von C1 auf Drive1 am Server
C1	D5	von C1 auf Drive5 an T1 (ohne Umweg über Server)
T1	D4	von T1 auf Drive4 an T1 (ohne Umweg über Server)
T1	D3	von T1 auf Drive3 am Server

Die Begleitdaten (Listings, Segmentfiles etc.) werden vom TapeServer auf den **SEPsasam**-Server übertragen.

3.6 Globale Variablen (GLBV)

Zur Kommunikation zwischen Programmen werden vom **SEPsasam** Server spezielle globale Werte, sog. GLBV (global variables) erzeugt. Diese Werte liegen auf WNT-Systemen in der WNT-Registry und auf UNIX-Systemen in speziellen Dateien im Verzeichnis `{IV}/var/glbv`.

Achtung

Das Setzen bzw. Löschen von GLBVs kann zu schwerwiegenden Systemstörungen führen und darf deshalb nur von System-Spezialisten genutzt werden.

Mit dem Programm `{IV}/bin/sesam/sm_glbv` können die GLBVs gesetzt, gelesen und gelöscht werden.

Die Syntax erläutert der Aufruf `sm_glbv -h`.

3.7 SesamLoaderUtility SLU

Mit `sm_slu` können SCSI-II Lader unter Windows NT, Windows 2000, DECUNIX, HPUNIX und LINUX angesteuert werden.

Wird `sm_slu` ohne Argumente aufgerufen, so wird die folgende Kommandoübersicht ausgegeben. Die Ausgabe kann versionsabhängige Unterschiede aufweisen.

Beispiel

Usage:

```
Scan(0-7) for loaders and tapes:slu scan [bus]
Inquiry:                          slu <ID|dev> [-]i
Mode sense:                        slu <ID|dev> [-]s
Read element status:              slu <ID|dev> [-]e [s|d|p:num-num|full|empty]
Read element status with label:   slu <ID|dev> [-]l [s|d|p:num-num|full|empty]
Move medium:                      slu <ID|dev> [-]m <s|d|p:num> <s|d|p:num>
Unload for tapes:                 slu <ID|dev> [-]u
Load for tapes:                   slu <ID|dev> [-]g
Rewind for tapes:                 slu <ID|dev> [-]r
Allow removable for tapes:       slu <ID|dev> [-]a
Prevent removable for tapes:     slu <ID|dev> [-]p
Reset Semaphore File:            slu reset [ID|dev]
```

Die einzelnen Optionen werden im folgenden beschrieben.

3.7.1 sm_slu scan

Für den **SEPsam** Anwender ist vor allem das Kommando 'sm_slu scan' zur Bestimmung der Geräte-ID (SCSI-ID, Bus, LUN, Target) hilfreich. Das Kommando listet alle Geräte-IDs der angeschlossenen Lader und Laufwerke auf. Die Geräte-ID wird zur Ansteuerung des Laders in die Konfiguration von **SEPsam** aufgenommen.

Die Ansteuerung der Laufwerke erfordert, dass dem Lader die entsprechende Nummer des Laufwerks übergeben werden kann. Diese Zuordnung (relative Laufwerknnummer) muss bei der Konfiguration des Laufwerks angegeben werden.

Handelt es sich bei dem Lader um kein Kompakt-Gerät (z.B. TZ887) sondern um einen Lader mit selbständigen Einzellaufwerken, so muss das Einzellaufwerk explizit zum Entladen geöffnet werden. Auch dieses Entladen erfolgt mit dem slu-Kommando. Hierzu ist die SCSI Geräte-ID bei dem

entsprechenden Laufwerk zu konfigurieren.

Beispiel UNIX

STK-Lader mit 2 Drives (3 SCSI-Geräte)

```
bash-2.02# sm_slu scan
```

```
ID=310 Tape:    QUANTUM DLT4000          2255 (/dev/nrmt0h)
ID=340 Tape:    Quantum DLT4000          CF5F (/dev/nrmt1h)
ID=360 Loader:  STK      9730            1205
```

In der Klammer erscheint der Gerätename, der in der **SEPs**sesam Laufwerkskonfiguration eingetragen werden muss.

sm_slu 360 -e scsi zeigt dann die Laufwerksnummerierung innerhalb des Laders:

```
ID=340 Tape Drive 0 in Loader 360
ID=310 Tape Drive 1 in Loader 360
```

Beispiel Windows

TLZ9-Lader mit 1 Drive (1 SCSI-Gerät)

```
bash-2.02# sm_slu scan
```

```
ID=0000 other: WDC AC22000L          32.4 (HardDisk)
ID=0010 other: FUJITSU MPC3084AT      6021 (HardDisk)
ID=1000 other: LITEON DVD-ROM LTD163D GHR3 (CdRom)
ID=2050 Tape:  DEC      TLZ9      (C) DECA035 (Tape0)
ID=2051 Loader: DEC      TLZ9      (C) DECA035 (Loader)
```

Der Gerätename zu ID=2050 ist Tape0 (ist in der Registry wiederzufinden), die Zusammengehörigkeit von Tape0 und Loader ergibt sich aus der Identität von SCSI-ID, Bus und LUN (=205), d.h. Laufwerk und Robotermodul werden über den gleichen SCSI-Kanal bedient.

3.7.2 sm_slu inquiry

Mit der Option „-i“ können Hersteller und Typ des Gerätes abgefragt werden. Dieses Kommando kann auch zur Überprüfung eines korrekt angeschlossenen und funktionsfähigen Gerätes verwendet werden.

Beispiel

```
sm_slu 360 -i
ID=360 Loader: STK 9730
```

3.7.3 sm_slu mode sense

Verwendung zur Abfrage von Laderzustand bzw. Rückgabe des letzten Fehlerzustands.

Beispiel

```
sm_slu 360 -s
1 x Transport First=1000
30 x Slot First=0
1 x Port First=1010
2 x Drive First=1030
```

3.7.4 sm_slu read element status

Mit dieser Option wird eine Liste mit allen Elementen (Slot, Drive, Transport, Port) und deren Befüllung ausgegeben

Beispiel 1

```
sm_slu 360 -e full
Slot 0(adr= 0,src=1030,tape=1) 1=full
Slot 2(adr= 2,src= 4,tape=1)
Slot 8(adr= 8,src=1030,tape=1)
Slot 9(adr= 9,src= 0,tape=1)
Slot 10(adr= 10,src=1031,tape=1)
Slot 0(adr=1030,src= 1,tape=1)
```

Beispiel 2

```
sm_slu 360 -e empty
Slot 1(adr= 1,src= 0,tape=0) 0=empty
Slot 3(adr= 3,src= 0,tape=0)
Slot 4(adr= 4,src= 0,tape=0)
Slot 5(adr= 5,src= 0,tape=0)
Slot 6(adr= 6,src= 0,tape=0)
Slot 7(adr= 7,src= 0,tape=0)
Slot 11(adr= 11,src= 0,tape=0)
Transport 0(adr=1000,src= 0,tape=0)
Port 0(adr=1010,src= 0,tape=0)
Drive 1(adr=1031,src= 0,tape=0)
```

Beispiel 3

```
sm_slu 360 -e s:0 s=slot
Slot 0(adr= 0,src=1030,tape=1)
```

Beispiel 4

```
sm_slu 360 -e d:0 d=drive
Drive 0(adr=1030,src= 1,tape=1)
```

Beispiel 5

```
sm_slu 360 -e p:0 p=port
Port 0(adr=1010,src= 0,tape=0)
```

3.7.5 sm_slu read element status with label

Diese Option gibt eine Liste mit allen Elementen (Slot, Drive, Transport, Port) und deren Befüllung sowie die Angabe des Labels aus.

Beispiel

```
sm_slu 360 -l full l=label; weitere Parameterangaben wie vorher
```

```
Slot 0(adr= 0,src=1030,tape=1) S00001
Slot 2(adr= 2,src= 4,tape=1) S00100
Slot 8(adr= 8,src=1030,tape=1) S00004
Slot 9(adr= 9,src= 0,tape=1) S00005
Slot 10(adr= 10,src=1031,tape=1) S00101
Drive 0(adr=1030,src= 1,tape=1) S00003
```

3.7.6 sm_slu move medium

Mit dieser Option können Medien bewegt werden. Als Ziel und Quelle kann ein Element des Laders (Slot, Drive, Port) angesprochen werden. Diese Funktion kann die vom **SEPs**sesam erfasste Befüllung ändern (!) und sollte deshalb nur unter Berücksichtigung des Medien-Archivs benutzt werden (vgl. 3.3.1.5).

Das Entladen von Laufwerken erfordert ggf. ein vorausgehendes 'Unload'-Kommando für das jeweilige Laufwerk.

3.7.7 sm_slu unload for tapes

Mit dieser Option können Laufwerke entladen werden. Die Medien werden zurückgespult und ausgeworfen. Evtl. vorhandene Laufwerkklappen werden geöffnet (das Laufwerk wird dabei in den „offline“-Zustand versetzt).

Diese Option ist nicht mit dem 'Unload' des sm_loader zu verwechseln (siehe 3.3.1.5).

3.7.8 sm_slu load

Mit dieser Option können „offline“ Medien im Laufwerk wieder „online“ gebracht werden.

3.7.9 sm_slu rewind

Mit dieser Option können Medien im Laufwerk zurückgespult werden.

3.7.10 sm_slu allow removal

Mit dieser Option können mit „prevent removal“ gesperrte Laufwerk wieder freigegeben werden.

3.7.11 sm_slu prevent removal

Mit dieser Option können Laufwerke gesperrt werden. Das Entladen der Medien im Laufwerk über Bedienelemente des Laufwerks („Entladen“ Knopf am Laufwerk) wird verhindert. Mit der Option „allow removal“ wird die Sperrung wieder aufgehoben.

3.7.12 sm_slu reset

Um das parallele Absetzen von SCSI-Kommandos zu vermeiden, benutzt sls intern einen Semaphore-Mechanismus. Dieser erzeugt für die Dauer des SCSI-Aufrufs eine Sperrdatei je SCSI-ID. Wird das Kommando in dieser Phase abgebrochen, so bleibt die Sperrdatei bestehen.

Mit dem Kommandoaufruf sls SCSI-ID reset wird die Sperrdatei entfernt.

Kapitel 4

Protokolle und Listings

SEPs erzeugt auf dem Sesam-Server eine Reihe von Protokollen, die Auskunft über den Betrieb und den Status von Sicherungen und Rücksicherungen geben.

4.1 Statusprotokoll

Für jeden SEPs Sicherungstag, das ist der Zeitraum der durch zwei aufeinanderfolgende SEPs Tageswechsel festgelegt wird, wird ein Statusprotokoll geschrieben. Hier erfolgt für jeden Sicherungsauftrag ein Eintrag über den Endestatus der Sicherungen in der Datei mit dem Datum des Beginns des SEPs Sicherungstages **{IV}**/var/prot/{yyyymmdd}.status.

In der Benutzeroberfläche (GUI) zu finden unter: Protokolle-Status
Am Zeilenanfang wird ein Flag mit folgender Bedeutung gesetzt:

- 0 erfolgreich
- 1 erfolgreich, aber mit Warnung
- 2 Listing unvollständig
- 3 abgebrochen
- c abgebrochen durch Benutzer (GUI)
- X Fataler Fehler

Beispiel

Start des SESAM Sicherungstages um 2002-11-12 08:00:52 Tue (softnt6)

```
-----  
0 DISASTER\_0      D:4 2002-11-12 20:05:37 erfolgreich ohne Warnungen  
0 DISASTER\_1      D:4 2002-11-12 20:09:49 erfolgreich ohne Warnungen  
0 NT6\_E1          D:4 2002-11-12 23:41:38 erfolgreich ohne Warnungen  
0 NT6\_E3          D:4 2002-11-12 23:41:38 erfolgreich ohne Warnungen  
0 NT6\_E2          D:4 2002-11-12 23:41:39 erfolgreich ohne Warnungen
```

4.2 Tagesprotokoll

Chronologischer Bericht über die Aktivitäten eines SEPsesam Sicherungstages in der Datei `{IV}/var/prot/{yyyymmdd}.prt`.

In der Benutzeroberfläche (GUI) zu finden unter: Protokolle-Tagesprotokoll

Folgende Informationen sind im Protokoll abgelegt:

PID	ProcessIdentifier
Kennung	eindeutige Bezeichnung der Meldung mit Severity+Nummer-Modulname
Severity I	Information
Severity W	Warnung
Severity E	Error = Fehler

Beispiel

Zeitstempel	Kennung	PID	Text
2002-11-12 08:00:52	I001-STARTAL	[1300]	: Datumskennung: 20021112
2002-11-12 08:00:54	I002-START	[4131]	: die terminierten BACKUP Aufträge...
2002-11-12 08:00:55	I002-START	[4153]	: die terminierten BACKUP Aufträge...
2002-11-12 08:00:55	I002-STARTAL	[1300]	: START_ALL beendet
2002-11-12 08:00:58	I001-START	[4153]	: START auf Laufwerk 5 beendet
2002-11-12 08:00:59	I001-START	[4131]	: START auf Laufwerk 4 beendet
2002-11-12 08:05:06	I007-SEPULER	[4467]	: starte Ereignis sepuler_event.c...
2002-11-12 08:05:08	I003-MEDIA	[2350]	: INIT_4 Initialisierung eines Me...

4.3 Sicherungsprotokoll

Detailinformationen über den Ablauf einer Sicherung, z.B. SBC-Meldungen, nicht gesicherte Dateien, FTP-connect-Meldungen, Endestatus usw.

`{IV}/var/lis/{Auftragsname-yyyymmdd_lfdnr}_{S}{T}.not`

Auftragsname	wie in GUI konfiguriert
yyyymmdd	Datum des Sicherungstages
lfdnr	laufende Nummer bzgl. Sicherungstag
S	Sicherungstyp S (SMS), X (Befehlstermin) oder externer Bezeichner für INFORMIX, ORACLE, SAP, etc.
T	Sicherungslevel, für SMS: F, D, I, oder C

In der Benutzeroberfläche (GUI) zu finden unter:

Sicherungen-nach Status-selektierter Auftrag-Eigenschaften-Protokoll oder durch Anklicken der entsprechenden Sicherung im Kalenderblatt.

4.4 Liste der gesicherten Dateien

Auflistung aller gesicherten Dateien eines Sicherungsauftrages (Sicherungsjournal).
{IV}/var/lis/{Auftragsname-yyyymmdd_lfdnr}_{S}{T}.lis

Beispiel

Inhalt einer LIS Datei:

```
...  
"home/bfa/lib/" d 22.06.2000 03:22:29 22.06.2000 03:22:29 3 528 652  
"home/bfa/lib/einsaetze2" f 22.06.2000 03:22:23 22.06.2000 03:22:23 3 652 788  
...
```

Wird in der GUI im Rücksicherungsassistenten zum Aufbau der Auswahlbäume für selektive Rücksicherungen benutzt.

4.5 Segment Dateien

Partnerdatei des gleichlautenden .lis-Files. Hier befinden sich die Medien und die Segmente, auf die das Saveset geschrieben wurde.

{IV}/var/lis/{Auftragsname-yyyymmdd_lfdnr}_S{T}.sgm

Beispiel

```
SAVESET: SD20000622110016 TAPEID: 14 SEGMENT: 3  
14:0:3:S00014:3555  
14:116230016:4:S00014:1501
```

4.6 Analyseprotokolle - Logging

Zu Analyse- und Protokollzwecken ist in alle **SEPs**sesam-Programme ein Logging-Mechanismus integriert, d.h. die Programme protokollieren ihre Arbeit in Dateien unter {IV}/var/log mit. Der Umfang dieser Mitschriften kann vom Benutzer festgelegt werden.

4.6.1 Logging der Kernprogramme

Der Umfang der Logging-Dateien kann in der Datei {IV}/var/ini/debug.ini vom Benutzer festgelegt werden.

In der Datei wird hierzu der Programmname (groß geschrieben) und der Logginglevel gesetzt, z.B. SM_RESTORE=1 .

Die Bedeutung der Logginglevel ist wie folgt:

0 = Keine Logging-Datei erstellen

1 = Standard - Argumente, Aufrufe der Unterfunktionen, wichtige Variablen

2 = Detail - zusätzliche Ausgaben zur Problemanalyse

Mit dem Eintrag unter

```
[VERBOSE]
STDERR=0
```

wird das Logging, für „STDERR=1“, zusätzlich auch auf stderr ausgegeben. Dies kann bei Aufruf auf Kommandline hilfreich sein.

Die Dateien werden unter `{IV}/var/log/lgc/*.*.lgc` abgelegt und im Tageswechsel auf `*_'Datum'.lgx` umbenannt.

Sie haben folgendes Format:

```
8:05:24# Logging enabled for module SM_SEPUL_EVENT (PID=1171), parent proc

8:05:24# +++ ARGUMENTS $Id: sm_sepul_event.c,v 1.114 2002/10/24 10:03:09 P
8:05:24# --- ARGUMENTS
8:05:24# +++ MAIN:
8:05:24#     +++ CHECK_PERMIT at 2002-11-10 08:05:24
8:05:24#         Check if sepuler is passing a term with deleted event des
8:05:24#     +++ CHECK_HIGHER_PRI0  1  M  Disko_ 5 0 1
8:05:24#         +++ FILL_SCT_MEDIA_TERM descript:Disko_ 5 0 1  #
8:05:24#             Mediapool: Disko_  optional Drive-number: 5
8:05:24#         +++ i_GET_DRIVE_GROUP for media pool Disko_
8:05:24#             select grp_name from drive_groups, media_pools
8:05:24#         +++ DB_ACCESS id=get_drive_group  SQL=select g
8:05:24#             --- DB_ACCESS returns 1  msg=1
8:05:24#             --- i_GET_DRIVE_GROUP: LocalDisk (1 rows)
8:05:24#             --- FILL_SCT_MEDIA_TERM  drvgrp:LocalDisk grpflg:0 dn:
8:05:24#             get starting-time of surrounding NEWDAys
8:05:24#         +++ DB_ACCESS id=CHECK_HIGHER_PRI0  SQL=select start f
8:05:24#             --- DB_ACCESS returns 1  msg=37
```

Mit '+++' und Einrückung wird der Eintritt in eine Unterfunktion, mit '---' der Austritt markiert.

4.6.2 sm_db_main Logging

Die Logging-Einstellungen können bei Windows in der `%systemroot%\openrda.ini`, bei UNIX in `$OPENRDA_INI` verändert werden.

Das Logging wird hier auch als Tracing bezeichnet.

```
[CLIENT]
Tracefile = Pfad und Name der Tracefile (Default ..\work\oaclient.log)
```

```
[TRACECLIENT]
```

ALL = (Default FATAL | SNO | ERRORS) siehe Liste Tracing Optionen

[SERVER]

Tracefile = Pfad und Name der Tracefiles (Default ..\work\sm_db_main.log)

TraceOptions = (Default all:f) siehe Liste Tracing Parameter und Optionen

[OLEDB / ODBC32]

Tracefile = Pfad und Name der Tracefile

TraceOptions = 0-2

0=FATAL|ERROR|SNO

1=Funktionslevel Tracing

2=ausführlichste Tracing Stufe

Server Module (für oaisql und sm_db_main)

```

ALL      = Trace settings for all modules\\
SESS     = Session layer tracing\\
RTP      = Encode / decode tracing\\
XM       = Memory\\
AL       = Association\\
RDA      = RDA protocol\\
SUR      = Sequence rule\\
AE       = Application entity\\
PRESS    = Presentation layer\\
CLI      = CLI layer\\
SAO      = Reserved for ATI\\
SACF     = Reserved for ATI\\
RDAASE   = Reserved for ATI\\
SQLS     = Database server layer (only for local configuration)\\
SQLD     = Enable tracing in the Ips (only for local configuration)\\

```

Setting Server Trace Flags

Die Trace Ausgabe für den Server erfolgt gemäß ausgewähltem Server Modul plus jeweiliger Trace Stufe, die durch ein Bitfeld vorgegeben wird.

Beispiel

Tracing für „all“ mit Bitfeld „f“, d.h. FATAL—SNO—PARM—ERRORS

und Tracing für Server Modul Session Layer mit „1ff“, d.h. alle Meldungen.

TraceOptions = all:f sess:1ff

Server Trace Flag Settings

Flag	Description	Bit Mask
MAX	All tracing options set	0x1ff
FATAL	Fatal errors	0x001
SNO	Should not happen conditions	0x002
PARM	Invalid parameters	0x004
ERRORS	General errors	0x008
MAJOR_EV	Major network event	0x010
MINOR_EV	Minor network event	0x020
INFO	Informational trace	0x040
F_TRACE	Function trace	0x080
TRIVIA	Trivial messages	0x100
TIMESTAMP	Time stamp function calls	0x200

4.6.3 SMS/STPD Logging

Der Umfang der Logging-Dateien kann in den Dateien `{IV}/var/ini/sms.ini` bzw. `./stpd.ini` vom Benutzer festgelegt werden.

Mögliche Werte sind:

0=ERROR, 1=EMERGENCY, 2=WARNING, 3=NOTICE, 4=INFO, 5=DEBUG, 6=TRACE

Diese werden in den INI-Dateien unter der Sektion `[SMS.Logging]` bzw. `[STPD.Logging]` in folgender Form eingetragen:

```
[SMS.Logging]
```

```
Log_Levels=0 1 2 => bis Level 2 (Beispiel)
```

```
[STPD.Logging]
```

```
Log_Levels=0 1 2 3 4 5 6 => bis max.Level 6 (Beispiel)
```

Die Logging-Dateien werden unter `{IV}/var/log/sms` abgelegt.

Kapitel 5

Administration

5.1 Server starten und beenden

SEPs bietet eine einfache Schnittstelle zum Starten und Beenden des Servers. Mit **sm_startup** wird das System gestartet.

Es wird nach dem Aufstarten aller Dämonen bzw. Services zunächst die allgemeine **sesam**-Bearbeitungsqueue `qu_all` initialisiert und dann in dieser das Programm `sm_startup` gestartet. `sm_startup` prüft die **SEPs** Umgebung und startet `sm_config_drives` zur Initialisierung der Laufwerke.

Wird der **SEPs**-Server mit „`sm_startup`“ gestartet, so wird mit „`sm_main status`“ geprüft, ob die Dämonen bzw. Services aktiv sind. Sind die Prozesse nicht aktiv, so wird „`sm_main start`“ aufgerufen und `sm_startup` beendet sich vorzeitig.

Für das Beenden des **SEPs**-Servers dient **sm_shutdown** der **sm_os_startup stop** aufruft, welcher u.a. `sm_main stop` zum Beenden der Services/Dämonen verwendet.

5.2 Reboot Konfiguration (Autostart nach Reboot)

Der **SEPs** ist so aufgebaut, dass er sich nach dem Booten des Rechners automatisch startet. Um diese Funktion zu unterbinden, geht man wie folgt vor:

Unter Unix-Systemen.

Wechseln in das `/bin/sesam`-Verzeichnis und Eingabe von **sm_setup unset_reboot**. Um den Dienst wieder zu starten, wird **sm_setup set_reboot** eingeben.

Unter Windows-Systemen:

WNT: Start-Einstellungen-Systemsteuerung-Dienste.

Auswahl von **SEPs** Server bzw. **SEPs** Tape Server. Startart auswählen, dann „Deakti-

viert“ selektieren. Auch der Dienst sesam Multiplexing Server 0. muss deaktiviert werden.

Windows 2000 / Windows XP: Start-Einstellungen-Systemsteuerung.

Dann Ordner „Verwaltung“ anwählen und „Dienste“ selektieren. Doppelklick auf **SEPs**sesam Server bzw. **SEPs**sesam Tape Server; bei Starttyp „Deaktiviert“ verwenden.

5.3 Funktionsprüfung

Zur Überprüfung der **SEPs**sesam Komponenten ist die Anzeige **GUI:laufende Meldungen** und das Starten eines Sicherungsauftrags der GUI ausreichend (Sofortstart Sicherung). Das Backup sollte anlaufen und Durchsatzmengen anzeigen. Der Funktionscheck war erfolgreich, falls die Sicherung erfolgreich beendet wird.

Ablauf Funktionscheck

GUI: Anzeige „laufende Meldungen“

Möglicher RMI Fehler:

RMI Server am SEP sesam Server hat keine Socketverbindung zu <Service-Adresse> <Port>

GUI: Sofortstart Sicherung

Möglicher Datenbank Fehler:

ODBC/sm_db_main Verbindung zur Datenbank fehlgeschlagen

Möglicher SEPuler Fehler:

sepuler Sicherung wird nicht gestartet

Überprüfung über Zeitplan => Termine

Termin soll spätestens eine Minuten nach Ablauf verschwinden

Möglicher Queue Manager Fehler:

Queue Manager Sicherung wird nicht gestartet

Überprüfung über „Monitoring Laufwerke“

Prüfe Prozess sm_qm_main und falls notwendig sm_startup wiederholen

SM-Kommandos für Backup

Möglicher CTRLC/SSH Fehler:

sm_ctrlc Verbindung zum Klienten nicht möglich

Überprüfen der Namensauflösung (Name und Adresse)

Überprüfen ob Klient erreichbar

Überprüfen ob Klient über CTRLC/SSH erreichbar

sm_ctrlc am Klienten prüfen, ob dieser am richtigen Port aktiv ist

sm_ctrlc/ssh Verbindung auf Kommandozeile prüfen

Möglicher SBC Fehler:

sbc-Komponente am Klienten prüfen

Möglicher STPD Fehler:

stpd am Server prüfen

FTP-Fehler im Sicherungsprotokoll überprüfen

1. Prüfe Daemon und Prozessliste
2. Prüfe FTP-Verbindung mit ftp Host 11001, z.B. ftp A0032 11001

Möglicher SMS Fehler:

SMS Dataserver prüfen

sms-Fehler im Sicherungsprotokoll überprüfen

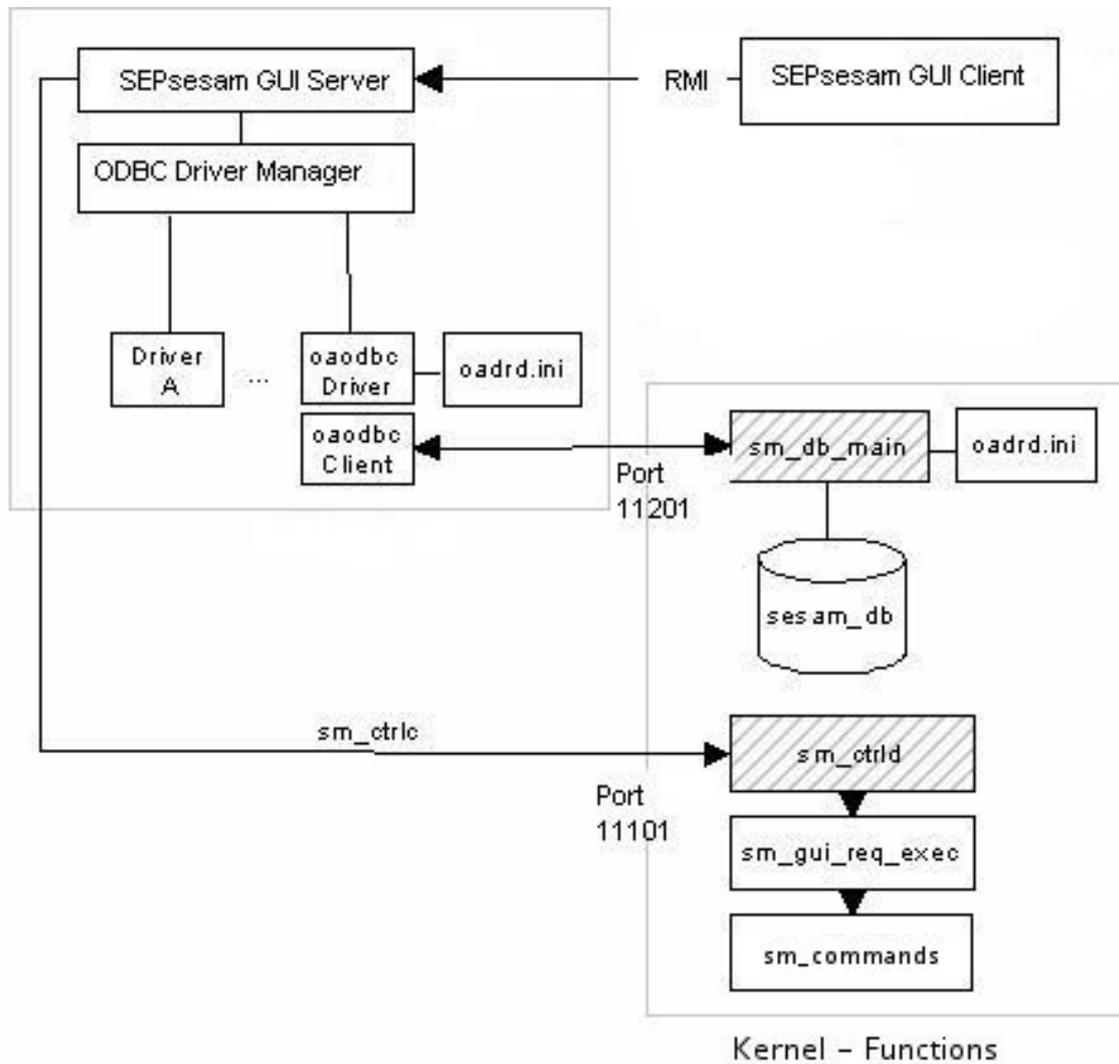
Prüfe SMS Dataserver in Prozessliste

ggfs. Log-Level erhöhen und Logging prüfen

5.4 GUI-Server Administration

Für den ODBC-Zugriff ist am GUI-Server-Rechner ein ODBC Driver Manager wie z.B. odbc32 (Standard) auf WNT, Intersolv Driver Manager auf Compaq, unixODBC auf LINUX notwendig. Der Driver Manager verwaltet alle ODBC-Datenquellen. Für die **SEPs**sesam Datenbank muss eine Datenquelle „sesam_db“ konfiguriert sein.

Die Datenquelle wird über den oaodbc Driver und den oaodbc Klienten angesprochen. Dieser Klient kommuniziert mit dem **SEPs**sesam Datenbank Server (sm_db_main) und der entsprechenden Datenquelle am Server.



ODBC-Verbindung

Die Voraussetzungen werden bei der Serverinstallation automatisch realisiert.

5.4.1 GUI-Server Kommandos

Ein Aufruf einer Interaktion wird vom GUI über den RMI und vom RMI Server über CTRL an das Modul `sm_gui_request` weitergereicht. Dieses stellt die Parameter für die Funktionsaufrufe bereit und ruft das benötigte Programm auf.

Im folgenden werden je GUI-Komponente (Kopfleiste, unteres Fenster, etc.) die SEPsasam Programme und die jeweiligen Interaktionen aufgelistet.

	in der Menue- und der Symbolleiste
sm_restore	Aktivitäten-Rücksicherung-Start ausführen
sm_search_lis	Aktivitäten-Rücksicherung-Suche nach Dateien
sm_lis_db	Aktivitäten-Rücksicherung-Aufbau des Baumes für selektiven Restore
sm_sho stream file pre	Konfiguration-Schnittstellen-Pre
sm_sho stream file post	Konfiguration-Schnittstellen-Post
sm_sho stream file alarm	Konfiguration-Schnittstellen-Alarm
sm_sho stream file disaster	Konfiguration-Schnittstellen-Desaster
	im unteren Anzeigefenster
sm_sho stream cur	laufende Anzeige zyklisch
sm_break	Sicherung bzw. Rücksicherung abbrechen
	im Objektbaum ()linkes Fenster
sm_remote_tcpip	Komponenten-Topologie-neuer Klient-UNIX
sm_remote_wnt	Komponenten-Topologie-neuer Klient-WNT
sm_inventory	Komponenten-Lader-Inhalt-Archivabgleich
sm_loader	Komponenten-Lader-Laderaktion
sm_config_loaders	Komponenten-Lader-neuer Lader/Laufwerk löschen
sm_config_drives	Komponenten-Laufwerke-neues Laufwerk/Laufwerk löschen
sm_arch new	Komponenten-Medien-Neu
sm_arch rel	Komponenten-Medien-Löschen
sm_cal.sheet	Zeitplanung-Kalenderblatt
sm_sho stream pre	Sicherungen-nach Status-Auswahl Saveset-Pre
sm_sho stream post	Sicherungen-nach Status-Auswahl Saveset-Post
sm_sho stream not	Sicherungen-nach Status-Auswahl Saveset-Protokoll
sm_daily_schedule	Zeitplanung-Zeitpläne-Eigenschaften anlegen/ändern
sm_process_sts	Monitoring-Prozesse
sm_drive_sts	Monitoring-Laufwerke
sm_sho stream statprl	Protokolle-Status
sm_sho stream sesamprl	Protokolle-Tagesprotokoll
sm_sho stream lic	Lizenzinfo

5.5 Zugriffseinschränkung der Befehlsstermine

5.5.1 Problembeschreibung

Der Befehlsstermin ermöglicht die Ausführung von beliebigen Programmen auf Sicherungsklienten. Aus Sicherheitsgründen muss die Ausführung einschränkbar sein. Die Zugriffssicherheit eines Rechners ist Aufgabe des jeweiligen Systemadministrators: Nur dieser darf eine Berechtigung erteilen bzw. unterbinden.

Zur Zugriffsbeschränkung auf UNIX-Systemen werden i. a. Dateien am Rechner erstellt, welche einen Zugang explizit ermöglichen bzw. unterbinden, wie z.B. die Datei /etc/ftpusers.

Folgende Vorgehensweise hat sich aufgrund ihrer Einfachheit und Robustheit bewährt:

- einfache Änderung der Zugangsberechtigung durch Änderung der Datei
- einfache Übersicht der aktuellen Berechtigungen durch Ausgabe der Datei
- sichere Verwaltung durch ausschließlichen root-Zugang, Dateien in „/etc“ sind nur durch 'root' editierbar.

Zur Zugriffsbeschränkung auf WNT-Systemen können Einträge in der Registry verwendet werden. Die Registry berücksichtigt alle WNT-Benutzerberechtigungen und lässt Zugriffe auf Einträge nur entsprechend der jeweiligen Rechte zu. Eine Überwachung der Einträge wird unterstützt.

5.5.2 Befehlseinschränkung auf UNIX-Systemen

Für die Beschränkung des **SEPsesam** Befehlstermins wird die Datei `sesam_cmdusers.allow` im Verzeichnis `/etc` verwendet.

Ein Befehlstermin kann nur ausgeführt werden, wenn sowohl der Benutzer als auch der Dateiname des auszuführenden Programms in `/etc/sesam_cmdusers.allow` aufgeführt sind.

Format der Datei `/etc/sesam_cmdusers.allow`:

```
{user} {command} | *
```

Beispiele

a) Zugang für Benutzer oracle Programm `sm_oracle_client`:

```
oracle sm_oracle_client
```

b) Zugang für Benutzer informix mit beliebigem Programm:

```
informix *
```

Die Kommandoausführung wird immer nur mit dem ausgewählten Benutzer ausgeführt

```
su - {user}
```

Hinweis

Die Login Scripten des Benutzers, z.B. `informix/.profile`, dürfen keine Eingaben, z.B. „Auswahl der Datenbank mit 1 oder 2“ erwarten. Dadurch würde die Kommandoausführung blockiert werden.

5.5.3 Befehlseinschränkung auf WNT-Systemen

Für die Beschränkung des **SEPsesam** Befehlstermins werden die Registry-Einträge unter `HKEY_LOCAL_MACHINE/SOFTWARE/SEP Elektronik GmbH/sesam/CommandEvents` ver-

wendet. Jeder Benutzer wird als Ordner, das auszuführende Programm als Key mit beliebigem Wert abgelegt. Als Wert lassen sich zusätzliche Informationen zur Befehlsfreigabe ablegen.

Hinweis

Der Ordner CommandEvents darf nur mit den zwingend notwendigen Zugriffsrechten ausgestattet werden, damit ein ungewollter Zugriff vermieden wird!

Ein Befehlstermin kann nur ausgeführt werden, wenn sowohl der Benutzer als auch der Name des auszuführenden Programms aufgeführt sind.

Wird ein '*' als Programm Key angegeben, so können beliebige Kommandos ausgeführt werden.

Sonderfall Kommando „cmd“:

Wird ein CMD-Kommando (MS-DOS Kommando Interpreter), z.B. cmd /c dir c:\angegeben, so wird zunächst der Registry Key cmd gesucht. Ist dieser nicht definiert, so wird das jeweilige Subkommando, im Beispiel dir, gesucht.

Beispiele

a) Zugang für Benutzer oracle Programm sm_oracle_client:

Key

HKEY_LOCAL_MACHINE/SOFTWARE/SEP Elektronik GmbH
/sesam/CommandEvents/oracle/sm_oracle_client

notwendig. Wert beliebig.

b) Zugang für Benutzer informix mit beliebigem Programm:

Key

HKEY_LOCAL_MACHINE/SOFTWARE/SEP Elektronik GmbH/sesam/CommandEvents/informix/*

notwendig. Wert beliebig.

c) Zugang für Benutzer Achim mit beliebigem CMD-Kommando:

Key

HKEY_LOCAL_MACHINE/SOFTWARE/SEP Elektronik GmbH
/sesam/CommandEvents/Achim/cmd

notwendig. Wert beliebig.

5.6 Konfiguration einer Firewall

SEPs sesam2000 ermöglicht für die Verwendung von Firewalls die Einschränkung der benutzen Ports.

Feld clients.access_options

Der Wert dieses Feldes (z.B. -s, -o port=first-last) wird direkt für den Remoteaufruf verwendet. Da die sinnvollen Optionen bei den verschiedenen Zugriffsarten (CTRLD, SSH) sehr unterschiedlich sind, ist dieses Feld frei editierbar. (GUI)

Erläuterung bzgl. CTRLD: Mit Port Range (-o ...) ergibt sich folgender Ablauf:

5.6. KONFIGURATION EINER FIREWALL

SESAM Server	FireWall	Klient
		CTRLD_MAIN listening auf Port {ctrld_port}
SM_CTRLC		
Source Port=random erster freier Port		
Source Port für CTRLD Klienten =erster freier Port am Klienten aus {first_port}-{last_port} Default unbeschränkt		
Target Port={ctrld_port}		
SM_CTRLC wählt lokalen Port aus und verbindet sich zu CTRLD_MAIN:{ctrld_port}		
SM_CTRLC schlägt für die weitere Kommunikation den Range vor		
		innerhalb des vorgegebenen Ranges:
		CTRLD listening auf erstem freien Port (stderr)
		CTRLD listening auf zweitem freien Port (stdin)
		CTRLD listening auf drittem freien Port (stdout)
		wird von CTRLD_MAIN an SM_CTRLC übermittelt
SM_CTRLC wählt weitere 3 Ports und verbindet sich zu den Ports des CTRLD am Klienten		

Die weitere Kommunikation erfolgt über diese 3 Ports.

Erläuterung bzgl. CTRLD: Mit Single Port Range (-s) ergibt sich folgender Ablauf:

SESAM Server	FireWall	Klient
		CTRLD_MAIN listening auf Port {ctrld_port}
SM_CTRLC		
Source Port=random erster freier Port		
Source Port für CTRLD Klienten =über {ctrld_port} (single port)		
Target Port={ctrld_port}		

```

| |
SM_CTRLC wählt lokalen Port aus | |
und verbindet sich zu          | |
CTRLD_MAIN:{ctrlld_port}      | |
| |
SM_CTRLC schlägt für die weitere| |
Kommunikation single port vor | |
| |
| | CTRLD listening auf {ctrlld_port} stderr, stdin
| | und stdout
| | wird von CTRLD_MAIN an SM_CTRLC übermittelt
| |

```

Weitere Kommunikation über diesen Port.
Übergabe Source Port:

```
-s # single port communication
-o port={first_port}-{last_port}
```

Übergabe Target Port:

```
-p {ctrlld_port}
```

Felder `clients.stpd_options` und `clients.stpd_port`

Darin wird zum einen der Destination Port des STPD Servers und zum anderen die Sourceports des sbc in folgendem Format gespeichert:

Für einen Klienten mit Firewall Einschränkung: `clients.stpd_options: [first_port-last_port] # Default: NULL`

Für einen Tape Server mit speziellem STPD Port: `clients.stpd_port: [stpd_port] # Default: NULL (== 11001)`

Die Angaben sind optional. Der Wert in diesen Feldern wird für den sbc Aufruf verwendet.

Erläuterung bzgl STPD:

Klient	FireWall	Tape Server
		STPD_MAIN listening on Port {stpd_port}
SBC		
Source Port=erster freier Port		
aus {first_port}-{last_port}		
Default unbeschränkt		
Target Port={stpd_port}		
sbc/XBSA wählt lokalen Port aus		

sm_db "insert into defaults (key,value,user_name) values ('mestersesam',1,'sesam')"

sm_db "delete from defaults where key='mestersesam'" macht den **SEPs** wieder zu einem normalen Server.

Achtung: Diese Funktion ist Lizenzpflichtig!

5.8 Lizenzierung des SEPs-Server

Der **SEPs** Server ist nach Zeit, Rechnernamen und TCPIP-Adresse lizenziert.

In `{IV}/var/ini/sm.lic.ini` sind die lizenzierten Werte und deren Verschlüsselung abgelegt.

Lizenzkategorien: TIMEOUT = Laufzeit in Tagen ab Installation

SERVER = Name des sesam Servers

TCPIP = TCPIP-Adresse des sesam Servers

Ein negatives Ergebnis der Lizenzprüfung blockiert die Ausführung von Backups.

Rücksicherungen von bereits erfolgten Sicherungen sind jedoch jederzeit möglich.

Falls `TIMEOUT <= 30` Tage, dann wird unabhängig von SERVER & TCPIP geprüft (Demolizenz).

Falls `TIMEOUT > 30` Tage, dann werden auch SERVER & TCPIP kontrolliert.

Bei Unstimmigkeiten durch evtl. Umbenennung des Rechners oder geänderte TCPIP-Adresse wird der TIMEOUT auf 7 Tage reduziert, um dem Anwender die Kontaktaufnahme mit Ihren zuständigen Distributor bzw. mit der **SEP AG (support@sep.de)** in einem akzeptablen Zeitrahmen zu erlauben.

Manipulationen in der Lizenzdatei führen dazu, dass der TIMEOUT auf 0 gesetzt wird, d.h. das System wird sofort gesperrt.

Die Lizenzprüfung beginnt *10 Tage vor Ende der Lizenzierung* mit dem Senden von Warnungen über die SESAM-Alarmschnittstelle, in das Tagesprotokoll und an die GUI.

Bei Lizenzierungsproblemen wenden Sie sich bitte umgehend an **support@sep.de**, weitere Informationen finden Sie unter **www.sep.de**.

Kapitel 6

Desaster Recovery

6.1 Desaster Recovery Konzept

Als Desaster wird hier die Zerstörung des **SEPs** Servers oder eines Sicherungsklientens und seiner Daten angesehen.

Für eine vollständige Wiederherstellung eines Systems sind außer den Sicherungsdaten auch die Meta Informationen des Systems notwendig.

Für die Sicherung der Meta Informationen sind für Linux 'disk_info' Sicherungen und für Windows 'system_state' Sicherungen notwendig. Damit werden die Konfigurationen der Systeme erfasst und stehen im Desaster Fall zur Verfügung. Im Falle von UNIX Systemen kann diese Konfiguration auch durch eine Neuinstallation erstellt werden. Im Desaster Fall sind dann lediglich Rücksicherung der Sicherungsdaten auszuführen.

Bei Ausfall des **SEPs** Servers stehen die Rücksicherungsmöglichkeiten der GUI nicht zur Verfügung. Für die Rücksicherung sind die Daten der entsprechenden Sicherungssätze notwendig. Diese Daten werden durch die **Desaster-Schnittstelle** - oder falls auch diese Daten nicht vorhanden sind - durch den 6.3 **Desaster Recovery ohne Metadaten**, d.h. dem Auffinden der relevanten Sicherungssätze direkt vom Sicherungsmedium, bereitgestellt.

6.2 Desaster-Schnittstelle

Zur Vorbereitung auf den Desaster Fall des **SEPs** Servers gibt es im **SEPs** die sogenannte Desaster-Schnittstelle.

Diese Schnittstelle dient zur raschen Wiederherstellung des **SEPs** Servers!

Das Bedienungskonzept ist wie folgt:

- der Backupserver sichert seine eigenen Daten mittels ganz normalen Sicherungen
- der Backupserver sichert seine Konfigurationsdaten mittels 'disk_info' (Linux) oder 'system_state' (Windows) Sicherungen

- die Metadaten zu diesen Sicherungen werden auf andere Systeme kopiert, sodass diese im Desasterfall für eine Rücksicherung via **SEPs sesam** verfügbar sind.

Nach der Installation des **SEPs sesam** steht in dessen var-Verzeichnis eine Datei `./var/skel/templates/sm_disaster` (`sm_disaster.cmd` in Windows) zur Verfügung, die der Benutzer nach `./bin/sesam/sm_disaster` (`sm_disaster.cmd`) umkopieren und nach Belieben ändern kann.

Diese Prozedur wird von allen Sicherungen, deren Namen 'DISASTER' (= englisch für Disaster) oder 'SESAM' enthält im Erfolgsfall ausgeführt. Dabei werden 2 Argumente übergeben:

- das Label, die Quelle des Auftrags und der Barcode des gerade benutzten Mediums
- das vollständige Restorekommando, mit dem die Daten des Savesets restauriert werden können.

Die **Desasterstrategie** schaut demnach wie folgt aus

- der Benutzer richtet mind. 1 Sicherung mit geeignetem Namen (z.B. SESAM_BACKUP) ein, die das var-Verzeichnis des **SEPs sesam** sichert (damit werden Listings, Loggings, Datenbank, INI-Dateien, etc. gesichert)
- Diese Sicherung sollte täglich mit dem Modus COPY oder FULL gesichert werden.
- Der Benutzer programmiert `sm_disaster(.cmd)` in dem Sinne, dass der Inhalt der beiden Eingabeargumente vom **SEPs sesam** Rechner auf andere Rechner an definierte Stellen kopiert wird (Emails, Dateikopien, Disketten etc.). Damit stehen die Informationen, wann und auf welches Medium die letzte Eigensicherung des **SEPs sesam** stattgefunden hat, zur Verfügung.

Im Ernstfall sind dann nach der Reparatur des Rechners folgende Schritte abzuarbeiten:

- vom Distributionskit **SEPs sesam** neu installieren
- an definierter Stelle bestimmen, welches Medium die letzte Eigensicherung des **SEPs sesam** beinhaltet und, falls gewünscht, in eine Prozedurdatei kopieren (z. B. `sm_disas.sh (.cmd)`)
- Medium in das Laufwerk einlegen und das Restorekommando ausführen oder `sm_disas.sh (.cmd)` starten

Beispiel

Die Metadaten der Desasterschnittstelle stellen sich wie folgt dar:

```
SESAM_00003 d:/Programme/SEPs sesam SC20020710200512  
sbc -r -l full -o over -S softnt6:11001 -d SMS:Tape0 -t SESAM_00003:3 -s SC20020710200512 -n 5 -R /
```

SESAM_00003 ist das Medium

d:/Programme/SEPs sesam ist das Quellverzeichnis der Datensicherung

SC20020710200512 ist der Sicherungssatz auf dem Medium

sbc -r ... ist das Restorekommando

6.3 Disaster Recovery ohne Metadaten

Wenn keine Metadaten aus der Disaster Schnittstelle verfügbar sind dann müssen die nötigen Daten auf den Medien gesucht werden.

SEPs bietet das Programm „sm_recover“ (siehe 3.3.1.6) zum Auffinden der Metadaten bzw. zum Rücksichern eines Sicherungssatzes direkt vom Medium.

Der folgende Auszug zeigt ein Beispiel einer sm_recover Sitzung:

```
florix:/opt/sesam/var/ini # sm_recover
num device      name          typ
  1 disk1        DISK_HARD
  2 /dev/nst0    TZ88
Eingabe Laufwerksnummer, (r) Liste nochmal zeigen, (q) abbrechen: 2
```

Mediumanalyse: Restauration von Daten oder Dateilisten von Sicherungssätzen

Die Restauration von Daten während SEPs aktiv ist könnte zu Konflikten mit gleichzeitig laufenden Laufwerksaktivitäten führen.

Wenn Sie sicher sind, dass keine Termine aktiv werden, dann braucht SEPs nicht blockiert zu werden. Eine Blockierung schützt

vor Konflikten mit anderen Aktivitäten, diese werden aber eben auch nicht ausgeführt (zB. auf anderen Laufwerken).

Wollen Sie SEPs blockieren? [j|n] > j

warte bis SEPuler - die Zeitsteuerung des SEPs - seine aktuelle Arbeit beendet

...

SEPuler ist jetzt blockiert ...

Wollen Sie Label, Savesetnamen und Segmentnummer direkt eingeben (keine Suche auf dem Medium) (j/n) > n

spule Band zurück und suche alle Sicherungssätze ...

Header:

STATUS=SUCCESS

MSG="Tapelabel = WEEK00002,TapeID = 2.000000,Tapeinfo = SESAM-volume, BlockSize = 65536, Creation Date = 2004-02-19 12:27:57"

Segments:

1, SavesetName = SC20040219122608, SavesetInfo = job=SESAM_BACKUP,UNIX

2, SavesetName = SC20040219122622, SavesetInfo = job=ntpc23_all,UNIX

3, SavesetName = SC20040219122611, SavesetInfo = job=softnt15_all,WNT

4, MSG="1043: Invalid tape format. Saveset Header should follow TapeMark"

5, MSG="1043: Invalid tape format. Saveset Header should follow TapeMark"

6, SavesetName = SC20040219122611, SavesetInfo = job=softnt15_all,WNT

7, MSG="1027: Skipping tapemarks on tape failed."

```
Trailer:
STATUS=SUCCESS
Tape Label: WEEK00002
Next Tape Label:
Tape Id: 2.000000
Tape BlockSize: 65536
Tape Info:
Creation Date: 2004-02-20 10:00:19
Tape EOM Flag: FALSE
MSG="OK"
Medium ist jetzt abgesucht
```

es befinden sich 4 Sicherungssätze auf dem Medium

```
I001-RECOVER Kopfdaten des analysierten Mediums
I002-RECOVER Label           : WEEK00002
I003-RECOVER Erzeugt am      : 2004-02-19 12:27:57
I004-RECOVER Kopfinformation : SESAM-volume,
I005-RECOVER BlockGrösse     : 65536
I006-RECOVER Info           :
I007-RECOVER Zuletzt geschrieben : 2004-02-20 10:00:19
I008-RECOVER Geteilte Sicherungen: FALSE
I009-RECOVER Nächstes Label   :
```

Liste der Sicherungssätze

```
1 SC20040219122608 Task: job=SESAM_BACKUP; Source:
2 SC20040219122622 Task: job=ntpc23_all; Source:
3 SC20040219122611 Task: job=softnt15_all; Source:
4 SC20040219122611 Task: job=softnt15_all; Source:
```

Bitte Nummer eines Sicherungssatzes oder q(uit) für Ausstieg eingeben > 1

Sollen vom Saveset die Dateien kopiert oder eine Dateiliste erstellt werden ?
(k/l) > k

Bitte Zielverzeichnis eingeben (/ bedeutet Originalpfad) > /

Dürfen gleichnamige Dateien am Zielort überschrieben werden (j/n) > j

Starte Kontrollprozess SMS_WATCH in Queue qu_tape_trans2
Kontrollprozess SMS_WATCH gestartet.

```
I011-RECOVER Bearbeite Sicherungssatz SC20040219122608 auf Medium WEEK00002 (/)
I016-RECOVER Dateien von Saveset SC20040219122608 auf Medium WEEK00002 nach /
kopiert.
```

Protokoll in Datei /opt/sesam/var/prot/recover_20040311144838.prt

6.3. DESASTER RECOVERY OHNE METADATEN

Liste der Sicherungsätze

- 1 SC20040219122608 Task: job=SESAM_BACKUP; Source:
- 2 SC20040219122622 Task: job=ntpc23_all; Source:
- 3 SC20040219122611 Task: job=softnt15_all; Source:
- 4 SC20040219122611 Task: job=softnt15_all; Source:

Bitte Nummer eines Sicherungsatzes oder q(uit) für Ausstieg eingeben > q

Kapitel 7

System Recovery

7.1 System Recovery für UNIX, LINUX

Um für Systemplattenabstürze an UNIX, LINUX Klienten gewappnet zu sein, müssen die entsprechenden Systemverzeichnisse gesichert werden.

Im Falle eines Systemplattenabsturzes werden nach dem Einbau einer neuen Platte und der Installation des Betriebssystems diese Daten mit einem Restore wieder zurückkopiert.

Wurden **disk_info** Sicherungen ausgeführt so kann mit Hilfe dieser Sicherungen, sofern eine identische Hardware zur Verfügung steht, das System automatisch wiederhergestellt werden. Voraussetzung ist ein per Installations CD gebootetes System mit SEPs sesam Klient. Wir empfehlen die Verwendung der **SEPs sesam** Bare System Recovery CD.

Zu beachten ist, dass eine volle Restaurierung nur sinnvoll ist, wenn an der Hardware nichts geändert wurde. Ansonsten müssten die entsprechenden Geräte- und/oder Treiberdateien in einem selektiven Restore ausgefiltert oder nach dem vollen Restore nachbearbeitet werden.

7.2 System Recovery für Windows

Bei Windows muss zusätzlich zur Sicherung der Daten auch die Sicherung des System States verfügbar sein.

Im **System State** werden alle systemrelevanten Daten abgelegt (Active Directories, Boot Files, Registry, etc.).

Um den System State zu sichern, muss als Sicherungsquelle „system_state“ in einem Disaster Backup Auftrag, z.B. **SESAM_SYSTEM_STATE** eingetragen werden.

Diese Aufträge sollen regelmäßig, gemeinsam mit der Sicherung der oben erwähnten Verzeichnisse ausgeführt werden.

Hinweis

Auf folgenden Windows Systemen wurde der Disaster Recovery getestet.

NT4.0 (SP3) Workstation
Windows 2000 (SP3)
Windows 2000 Server (SP3)
XP Professional
Windows 2003 Server (SP3)

7.2.1 Vorgehen im Disaster Fall

Installieren des Betriebssystems:

Das Installationsverzeichnis muss mit dem ursprünglichen übereinstimmen. Zu beachten ist dabei die korrekte Gruppenzuordnung und Netzwerkkonfiguration!
Ebenso muss der ursprüngliche Service Pack installiert werden.

Installation des SEPsesam Klienten

Verfahren gemäss LIESMICH/README_....txt (siehe 2.4).

Dabei ist der Server anzugeben, der die Platten- und System.State-Sicherung verwaltet hat.

Im Falle eines Systemplattenabsturzes werden nach dem Einbau einer neuen Platte und der Installation des Betriebssystems die Daten mit einem Restore wieder zurückkopiert.

Wiederherstellung des Systems:

1. Der defekte Rechner muss mit identischer Hardware wieder in den ursprünglichen technischen Zustand versetzt werden.
2. Installation des identischen Betriebssystems vor dem Disaster Fall, mit den identischen Haupt-servicepacks (z.B. SP4 bei W2k oder SP2 bei WXP).
3. Einstellung der Netzwerkeigenschaften mit derselben IP und demselben Namen wie vor dem Defekt.
4. Sollten es sich bei dem Disasterrechner um einen Exchangeserver oder einen Domainserver handeln, müssen diesen zuvor standardmäßig installieren. (dcpromo auf der cmd eingeben).
5. Installation des SEPsesam Klienten mit Namensverweis zum SEPsesam Server.
6. Den Rechner neu starten und mit F8 im Modus **Verzeichnisdienstwiederherstellung** starten. Der **SEPsesam** Server muss mit *ping* erreicht und von *nslookup* aufgelöst werden können.
7. Auf dem SEPsesam Server die Daten Rücksicherung starten (mit überschreiben), warten bis beendet.
8. Auf dem sesam Server die **system_state** Rücksicherung starten (mit überschreiben), warten bis beendet.
9. Den Rechner neu starten.
10. Auf vollständige Funktion prüfen und bei Problemen das Restore Logfile auf dem SEPsesam Server überprüfen.

Im Falle eines **SEPs**esam Server Desasters können die Rücksicherungen nicht per GUI sondern müssen per Kommandoaufruf gestartet werden, siehe 6.2.

Hinweis

Falls %SYSTEMROOT% nicht gleich SESAMROOT ist (z.B. c:\WINNT und d:\Programme\SEPs(esam) gibt es Probleme beim Restore. Ändern Sie deshalb die Datei <SESAMROOT>\var\ini\sm.ini unter [PATHES] den Wert der Variable gv_rw_tmp=c:\temp\.

Um dies zu umgehen, installieren Sie den **SEPs**esam Klienten am besten dort wo Ihr System liegt.

Kapitel 8

Sondermodule

Im folgenden werden einige optionale Module beschrieben. Weitere Sicherungsmodule, z.B. ORACLE, INFORMIX, SAP/R3 werden in der jeweiligen Dokumentation beschrieben.

8.1 Online-Backup/Restore Domino-Server

8.1.1 Systemvoraussetzungen

Voraussetzungen am Klienten

- Lotus Notes Domino Server 5.0.8
- Sesam NT Client (zB 1.5.1.14)

Zusatzmodule (bin/sesam):

- sbcln.dll

sm.ini Modifikations Beispiele:

```
[SBC_DATABASE]
DB_CLIENTS=SBCLN:sbcln.dll
sbc bindet dann mit Keyword SBCLN:databasename die DLL ein.
```

```
[SBC_NOTES]
tmp_dir=c:\lotus\domino\data\tmp
eventuell Temp Verzeichnis erzeugen oder nach var/tmp von Sesam setzen.
```

```
[SERVER]
```

```
sesam_server=SesamServerName
```

setzen.

PATH Variable anpassen nach \lotus\domino nnotes.dll (6.7 MB) soll erreicht werden und/oder \lotus\notes nnotes.dll (6.7 MB)

8.1.2 Full Backup

Prinzip:

```
sbc -b -S softnt17 -s notes.sset SBCLN:c:\lotus\domino\data\*.nfs
```

GUI

Sicherungstyp: Lotus Notes

Beispiel für Quelle

c:\lotus\domino\data\bookmark.nsf (oder *.nsf)

Achtung

Keyword SBCLN: wird beim Sicherungstyp Lotus Notes aus der SEPs sesam DB Tabelle task_types entnommen und automatisch vor der Quellenangabe eingefügt.

8.1.3 Full Restore

Domino Server herunterfahren

Prinzip:

```
sbc -r -S softnt17 -s notes.sset SBCLN:c:\lotus\domino\data\*.nfs
```

Im GUI über entsprechende Auswahl im Rücksicherungsassistenten.

8.2 Online-Backup/Restore MS SQL Server

8.2.1 Systemvoraussetzungen

Voraussetzungen am Klienten

- MS SQL Server 7.0

8.2.2 Beschreibung MS SQL

MS SQL 7.0 besteht aus mehreren Windows NT Diensten, der Hauptdienst ist jedoch MSSQL-Server. Zwei weitere Dienste sind MSDTC und SQLServerAgent. Die Datenbanken liegen bei der Standardinstallation unter mssql7\data jeweils als Daten (.mdf) und Log-Datei (.ldf).

Die eigentlichen Namen für die Backup Quelle (zB SBCMSQL:pubs) werden mittels Enterprise Manager gewählt.

Mit sqlmangr{.exe} (Enterprise Manager) für die entsprechenden Tables system User erlauben (unter location/sqlserver/database/users doppelklicken) damit die Sicherung und Rücksicherung im Hintergrund gestartet werden kann.

8.2.3 Full Backup

Prinzip:

```
sbc -b -S decunix:11001 -s sql.sset -d /tmp SBCMSQL:pubs
```

GUI:

Sicherungstyp: MS SQL Server

Beispiel für Quelle

pubs

Achtung

Keyword SBCMSQL: wird beim Sicherungstyp MS SQL Server aus der SEPsesam DB Tabelle task_types entnommen und automatisch vor der Quellenangabe eingefügt.

8.2.4 Full Restore

Prinzip:

```
sbc -r -a r -S decunix:11001 -s sql.sset -d /tmp SBCMSQL:pubs
```

Im GUI über entsprechende Auswahl im Rücksicherungsassistenten.

Anschließend mit sqlmangr{.exe} (Enterprise Manager) auf die entsprechend Location/Datenbank gehen und bei Backup Restore Database entsprechende ComboBox 'first backup to restore' wählen.

8.3 Online-Backup Exchange-Server 5.5

Hinweis

Die Einbindung von Exchange Server 2000 wird in einem eigenen Dokument beschrieben.

8.3.1 Systemvoraussetzungen

Voraussetzungen am Klienten

- Exchange Server 5.5

- Sesam NT Client (zB 1.5.1.14)

Zusatzmodule (bin/sesam):

- edbbcli.dll
- sbc_msex{.exe}
- start_service.bat (Achtung bei Englisch/Deutsch Versionen sind die Dienstnamen evt verschieden)
- stop_service.bat

sm.ini Modifikations Beispiele:

```
[PATHES]
gv_rw_smbx_log=C:\sesam_client\nt_cli.1.5.1.14\var\log\smbx\ (Verzeichnis muss dann angelegt
werden)
gv_rw_work=C:\sesam_client\nt_cli.1.5.1.14\var\work\ gv_rw_tmp=C:\sesam_client\nt_cli.1.5.1.14\var\tmp\
```

sm_node Anpassungen für Sub Modul sm_ctrl_main

unter Dienste

sm_node

Startart

Anmelden als Dienstkonto zB NTTEST\Administrator Passwort (damit der System User Exchange administrieren darf)

8.3.2 Beschreibung EXCHANGE

MS Exchange 5.5 lässt sich nur auf NT Domain Servern installieren.

MS Exchange besteht aus mehreren Windows NT Diensten, die JetDB Datenbanken verwalten. (Warnung: bei Englisch/Deutsch-Versionen sind die Dienstnamen evtl. verschieden)

Dienstname (net start/stop) Beschreibung Datenbank

MSExchangeMTA message transfer agent mtadata\ ...

MSExchangeES

MSExchangeIS information store mdbdata\priv.edb

mdbdata\pub.edb

MSExchangeDS directory service dsadata\ds.edb

8.3.3 Full Backup

Als Backup-Quelle wird der Exchange Server Name gewählt. Im GUI sollte dann Exchange Sicherung ausgewählt werden, damit als Call sbc_msex aufgerufen wird.

Prinzip:

```
sbc_msex -b -S decunix:11001 -s exchange.sset -j exchange -d /tmp QSWNT400
```

Beispiel mit sm_ctrlc

```
sm_ctrlc -l system -p 11301 qswnt400 sbc_msex -b -C softnt10 -S softnt10 -l full -s SC20011022151425  
-d SMS:disk1 -t DISK00001:1 -j ex_IS -i 'ex_IS,qswnt400,sbc_msex,' qswnt400
```

8.3.4 Information Store Backup

Als Backup Quelle wird der Exchange Server Name und Submodul mit Slash getrennt gewählt.

Beispiel

```
sbc_msex -b -S decunix:11001 -s exch_IS.sset -j exch_IS -d /tmp QSWNT400\IS
```

8.3.5 Full Restore

Beispiel

```
sbc_msex -r -S decunix:11001 -s exchange.sset -j exchange -d /tmp QSWNT400
```

Im GUI über entsprechende Auswahl im Rücksicherungsassistenten.

Kapitel 9

Problemanalyse und -behebung

Die folgende Beschreibung soll dazu dienen, dass Fehler möglichst exakt und schnell erkannt werden können.

Hierzu bietet SEPsesam folgende Möglichkeiten

- diverse Protokolle (inkl.Email-Versand)
- Tagesprotokolldateien per Programm von Fremdsprachen in deutsch übersetzbar um die Analyse zu erleichtern.
- Logging-Mechanismus mit einstellbaren Levels

Besonders im Szenario „**SEPsesam** läuft Wochen lang problemlos mit hohem Durchsatz“ und “plötzlich“ sinkt der Durchsatz oder bestimmte Rechner werden nicht mehr gesichert wirkte **SEPsesam** in den meisten Fällen als Detektor für hardware- und/oder softwaretechnische Defekte oder Änderungen im Netz des Betreibers (Adressen geändert, Backbone falsch angebohrt, Sternkoppler defekt. etc).

Bei Störungen im Funktionsablauf des **SEPsesam** empfiehlt sich folgender Ablauf der Analyse:

1. mit **sm_main status** prüfen, ob alle Prozesse laufen (siehe 3.5)
ggfs. mit **sm_main reload ...** den fehlenden Prozess nachstarten.
2. Tagesprotokoll (siehe 4.2)
3. Spezifische Protokolle zu Sicherungen, Rücksicherungen (siehe 4.3)
4. Logging-Dateien (siehe 4.6)

9.1 Probleme mit der Oberfläche (GUI)

Probleme mit den Java-Sicherheitseinstellungen sind z.T. schwer zu erkennen, da z.B. der GUI-Klient nicht startet, der Konfigurationsfehler aber auf dem GUI-Server zu suchen ist.

Fehlerbeschreibung	Ursache, Behebung
GUI Server nicht erreichbar	* Netzverbindung zum SEPs unterbrochen => läuft der Rechner ? * GUI-Serverprozess läuft nicht => sm_main reload rmi
Verbindung zur Datenbank fehlgeschlagen	DB- oder RMI-Server laufen nicht => sm_main reload db => sm_main reload rmi
GUI läßt sich nicht starten	Problem mit Java-Rechten Rechneintrag in sm_java.policy fehlt: Beispiel <pre>// NET permission java.net.SocketPermission "mycname:*", "connect,accept,resolve";</pre>
Problem beim Schreiben/Lesen auf dem Arbeitsverzeichnis:	Eintrag für das Arbeitsverzeichnis in sm_java.policy fehlt Beispiel Windows: <pre>// FILE permission java.io.FilePermission "D:\sesam\var\-", "read,write";</pre> Beispiel Linux/Unix/Tru64 <pre>// FILE permission java.io.FilePermission "/sesam/var/-", "read,write";</pre>
Es erscheint kein Online-Handbuch	* kein Adobe Acrobat Reader installiert => unter www.adobe.de kostenloser Download * Reader in der GUI nicht konfiguriert => GUI:Konfiguration-Vorbelegungen-Online
Es erscheint keine Online-Hilfe	* kein Browser installiert => bitte installieren * Browser in der GUI nicht konfiguriert => GUI:Konfiguration-Vorbelegungen-Online

Tipps

- Die Zugriffsberechtigungen für fehlende Rechner können mit dem Programm **sm_setup allow_gui {rechner} {user}** in das serverseitige java-policy-File eingetragen werden.
- Nach Änderungen in der sm_java.policy muß der GUI-Server mit **sm_main reload rmi** neu gestartet werden. Eine Änderung zur Laufzeit ist aus Sicherheitsgründen im Konzept von Java nicht vorgesehen.
- Die Syntax für die Pfadnamen in der sm_java.policy ist plattformabhängig.
- Um zu testen, ob eventuelle Probleme an der Java Security liegen können, kann man die Zeile *permission java.security.AllPermission;* entkommentieren. Dadurch ist die Einschränkung auf

die explicit aufgeführten Rechte ausgeschaltet.

Zur Vermeidung von Sicherheitslücken ist diese Zeile nach dem Test unbedingt wieder zu kommentieren!

9.2 Fehlermeldungen der Kernsoftware

Netzwerkprobleme gehören zu den häufigsten Fehlerursachen, d.h. ein Rechner läuft nicht problemlos und/oder die hardwaretechnische Verbindung zu ihm ist gestört. Hier hat der Betreiber dafür zu sorgen, dass die Kommunikation einwandfrei läuft (**Netzwerk Prüfung (NW-Check)**).

Netzwerk Prüfung

Mit Hilfe von Standard Kommunikationsprogrammen müssen die Verbindungen und die Adressauflösung geprüft werden (ping, nslookup, Adressauflösung etc.).

Zusätzlich muss die Verbindung mit dem jeweiligen SEPsesam Zugriffsprogramm (CTRL/SSH) geprüft werden (siehe 3.4).

Die Adressauflösung muss unbedingt konsistent sein, d.h. wenn für einen TCP/IP Namen die Auflösung eine IP-Adresse wiedergibt, so muss die Auflösung der IP-Adresse denselben TCP/IP Namen wiedergeben!

Beispiel

```
# nslookup decunix
Server:  seplinux2.sep.de
Address: 193.28.59.40

Name:    decunix.sep.de
Address: 193.28.59.94

# nslookup 193.28.59.94
Server:  seplinux2.sep.de
Address: 193.28.59.40

Name:    decunix.sep.de
Address: 193.28.59.94
```

In Fällen, bei denen unter 'Ursache, Behebung' nur => **support@sep.de** angegeben ist, ist eine Analyse in der Logging-Datei nötig, die zu diesem Zweck an den Support gesendet werden muss. **support@sep.de** teilt Ihnen mit, welche Dateien benötigt werden.

Hinweis

Unter *{val}* erscheinen in den Meldungen die jeweiligen aktuellen Werte.

9.2.1 BACKUP - Probleme mit Sicherungen

Fehlermeldung	Ursache, Behebung
E001-BACKUP Fehler in PRE-Bearbeitung zu Sicherung {val}: {val}	Bug im PRE-Script => benutzerprogrammiert Autorisierung nicht konfiguriert => (siehe 3.4) NW-Check
E002-BACKUP SMS_WATCH-Prozess kann nicht in Queue {val} gesetzt werden	Queuemanager läuft nicht => sm_shutdown — sm_startup
E003-BACKUP konnte gv_bck_msg- SMS_BACKUP nicht lesen	support@sep.de
E004-BACKUP {val} {val}	allgemeiner Fehler => support@sep.de
E005-BACKUP Fehler in POST-Bearbeitung zu Sicherung {val}: {val}	Bug im POST-Script => benutzerprogrammiert Autorisierung nicht konfiguriert => (siehe 3.4) => NW-Check
E006-BACKUP {val} Fehler siehe Sicherungsprotokoll	Fehler während Datentransfer => Sicherungsprotokoll an support@sep.de
E007-BACKUP Parameterfehler: ungültige Option {val} (Zeile {val})	=> sm_backup -h Usage befolgen
E008-BACKUP SBC Version auf dem Klienten unterstützt keine Verschlüsselung: {val}	alte SBC-Version auf dem Klienten => support@sep.de kontaktieren, Versionen abklären und installieren

9.2.2 BASICS - grundlegende Fehler

Fehlermeldung	Ursache, Behebung
E001-BASICS STR_CHAIN Stringüberlauf	=>support@sep.de
E002-BASICS {val} Fehler beim Öffnen der Datei {val} mit Modus:{val}	zu wenig Plattenplatz => erweitern fehlendes Verzeichnis => Verzeichnis anlegen =>support@sep.de
E003-BASICS {val} {val} existiert noch nicht - Neustart	sm_startup
E004-BASICS GET_LOCAL_TCPIP_ADRESS konnte keine benutzbare WinSock DLL finden.	=>support@sep.de
E005-BASICS HAL_REDIR Stringüberlauf: {val} {val}	=>support@sep.de
E006-BASICS CHECK_SMS autom. SMS-Neustart misslungen - SMS nicht verfügbar.	=> sm_shutdown — sm_startup; ggf. nach einigen Minuten wiederholen oder Reboot des Rechners
E007-BASICS {val} Fehler beim Löschen der Datei(en) {val}	=>support@sep.de
E008-BASICS Fehler bei Kommandoausführung {val}: {val}	NW-Problem => NW-Check Autorisierung nicht konfiguriert => (siehe 3.4)
E009-BASICS Fehler bei Dateikopie {val} {val}: {val}	Prüfen ob Filesystem voll ist; =>support@sep.de
E010-BASICS FILE Kopie {val} -> {val} kann Quelle nicht zum Lesen öffnen	=>support@sep.de
E011-BASICS FILE Kopie {val} -> {val} kann Ziel nicht zum Schreiben öffnen	=>support@sep.de
E012-BASICS FILE Kopie {val} -> {val} ungültiger Status {val}	=>support@sep.de
E014-BASICS konnte {val} nicht in Queue {val} stellen	Queuemanager läuft nicht => sm_shutdown — sm_startup
E013-BASICS SHO_FROM_TO Fehler in Eingabe von-bis : {val}-{val}	=> Usage
E015-BASICS Fehlende Parameter - mehr Information mit	=> Usage

9.2.3 BREAK - unerwartetes Abbrechen

Fehlermeldung	Ursache, Behebung
E001-BREAK Queue {val} konnte nicht blockiert werden.	Queuemanager läuft nicht => sm_shutdown — sm_startup =>support@sep.de
E002-BREAK Queue {val} konnte nicht gestartet werden.	wie E001-BREAK
E003-BREAK Abbruch durch {val} um {val} gescheitert ({val})	=>support@sep.de
E005-BREAK falscher Eingabewert {val}	=> sm_break -h Usage
E004-BREAK Keine Parameter eingegeben - siehe sm_break -h	=> sm_break Usage

9.2.4 CONFDRI - Konfiguration Laufwerke

Fehlermeldung	Ursache, Behebung
E001-CONFDRI Submit {val} in queue {val} mit Fehler.	Konfiguration Laufwerke Queuemanager läuft nicht => sm_shutdown — sm_startup
E002-CONFDRI Fehler bei Aktivierung des SesamMultiplexStream-Datenservers.	SMS-Server läuft nicht => sm_main reload sms
E003-CONFDRI Automatische Erkennung und Konfiguration der Hardware mit Fehlern beendet.	=> über GUI Lader und Laufwerke konfigurieren =>support@sep.de

9.2.5 CONFLOA - Konfiguration Lader

Fehlermeldung	Ursache, Behebung
E001-CONFLOA konnte Laderbefüllung nicht ermitteln	neu installierter Lader => Verkabelung prüfen => sm_slu scan
E002-CONFLOA konnte die Laufwerke des Laders {val} nicht ermitteln: {val}	neu installierter Lader => Verkabelung prüfen => sm_slu scan
E003-CONFLOA konnte die Startbefüllung des Laders nicht ermitteln: {val}	neu installierter Lader => Verkabelung prüfen => sm_slu scan

9.2.6 DATABAS - Probleme mit Datenbank

Fehlermeldung	Ursache, Behebung
E001-DATABAS Fehler bei Zugriff auf Datenbank: {val}	=>support@sep.de
E002-DATABAS DB_SEL_FIRST {val}: kann keine Memory allokieren.	=>support@sep.de
E003-DATABAS DB_SEL_ALLOC {val}: die Recordsliste ist leer, kann kein neues Element allokieren	=>support@sep.de
E004-DATABAS DB_SEL_ALLOC {val}: kann keine Memory allokieren.	Prüfen ob Filesystem voll ist; =>support@sep.de
E005-DATABAS {val} Feld {val} nicht im Record gefunden	=>support@sep.de
E006-DATABAS {val}: Recordliste ist leer kann angeforderte Daten nicht holen	=>support@sep.de
E007-DATABAS unerwarteter NULL-Wert	=>support@sep.de
E008-DATABAS CONVERT_DATE_TIME: falsches Argument {val}	=>support@sep.de
E009-DATABAS OA_MAKE_DB_REC_LIS {val}: {val}	=>support@sep.de
E010-DATABAS OA_MAKE_DB_REC_LIS {val}: zu viele Spalten (max {val})	=>support@sep.de
E011-DATABAS {val} {val}: kann keine Memory allokieren ({val} Bytes)	=>support@sep.de
E012-DATABAS {val} SQL-Pipe liefert nicht die erwarteten Zeichen	=>support@sep.de
E013-DATABAS {val} {val}: Fehler beim Öffnen einer Pipe	=>support@sep.de
E014-DATABAS OA_DO_SQL {val}: Fehler in SQL-Kommando: {val}	=>support@sep.de
E015-DATABAS OA_DO_SQL {val}: Fehler von DAMISQL: {val}	=>support@sep.de
E016-DATABAS {val} PSQL Kommando erzeugt keine Ausgabe.	=>support@sep.de
E017-DATABAS {val} {val}: zu viele Spalten	=>support@sep.de
E018-DATABAS {val} erzeugte unerwartetes Ergebnis: {val}.	=>support@sep.de

9.2.7 DBFPACK - Datenbank-File-Packer

Fehlermeldung	Ursache, Behebung
E001-DBFPACK kann Dateien nicht finden	falschen DBF-Namen angeben => {IV}/var/db/schema/sesam/*.dbf
E002-DBFPACK 3.Argument für Option 'P' fehlt	=> sm_dbf_pack -h Usage
E003-DBFPACK Argumente für Option 'i' fehlen	=> sm_dbf_pack -h Usage
E004-DBFPACK es fehlen Argumente für die DBF-Analyse	=> sm_dbf_pack -h Usage
E005-DBFPACK kann Datei {val}{val} nicht zum Lesen öffnen.	=> falschen DBF-Namen angeben => {IV}/var/db/schema/sesam/*.dbf
E006-DBFPACK kann Datei insert.tmp nicht zum Schreiben öffnen.	=> {IV}/var/work prüfen
E007-DBFPACK Record i{val} hat falsche Länge	=>support@sep.de
E008-DBFPACK kann Datei {val}{val} nicht öffnen, um den neuen Zähler einzusetzen	=>support@sep.de

9.2.8 DRIVES - Probleme mit Laufwerken

Fehlermeldung	Ursache, Behebung
E001-DRIVES falsches Medium {val} für MOUNT-Versuch in Laufwerk {val}	=> Medium mit Lader oder manuell wechseln => Label des Mediums prüfen (Archivabgleich) oder sm_drive info Drive Nummer
E002-DRIVES konnte kein Label vom Medium in Laufwerk {val} lesen.	SMS-Server läuft nicht => sm_shutdown — sm_startup Medium defekt => auswechseln
E003-DRIVES Laufwerk {val} ist nicht online ({val})	wie E001-DRIVES
E004-DRIVES gv_ro_sms ist noch nicht definiert (INI prüfen, Neustart)	=> sm_startup
E005-DRIVES SMS konnte Label auf dem Medium nicht finden	wie E001-DRIVES
E006-DRIVES SMS in nicht erlaubten Zustand	=> sm_shutdown — sm_startup
E007-DRIVES die Pipes des SMS laufen nicht - Neustart SMS	sm_main reload sms
E008-DRIVES Fehler bei Login zu SMS	sm_main reload sms
E009-DRIVES Fehlertext: {val}	=>support@sep.de
E010-DRIVES Fehler bei Initialisierung des Mediums in Laufwerk {val}: {val}.	wie E001-DRIVES
E001-GETVOL Anforderung Medium {val} mit Fehler: {val}	wie E001-DRIVES =>support@sep.de

9.2.9 HOSTS - Probleme mit Rechnern

Fehlermeldung	Ursache, Behebung
E001-HOSTS Rechner {val} nicht in Datenbank gefunden: {val}	=> nochmals konfigurieren
E002-HOSTS RemoteCopy {val}: ungültiger Modus:{val}	=>support@sep.de
E003-HOSTS RemoteCopy {val} {val} {val} -> {val}: Fehler bei {val}	=>support@sep.de => evtl. NW-Check
E004-HOSTS RemoteCopy {val} {val} {val} -> {val}: {val}	=>support@sep.de
E005-HOSTS RemoteCopy Fehler beim Lesen von {val} (Num.{val}): {val}	=>support@sep.de => evtl. NW-Check
E006-HOSTS RemoteCmd {val} ungültiger AccessMode - {val}	=> in GUI Klienteneigenschaft Accessmode ändern
E007-HOSTS RemoteCmd {val} Fehler: {val}	=>support@sep.de => evtl. NW-Check
E008-HOSTS Konfiguration eines {val}-Klienten mit Fehler beendet ({val}).	=>support@sep.de => evtl. NW-Check => danach wiederholen
E009-HOSTS WNT-Rechner {val} wurde NICHT einkonfiguriert.	=> NW-Check => wiederholen
E010-HOSTS lokale Datei {val} kann nicht geöffnet werden um SESAM Rechner dort einzutragen.	=> Existenz prüfen

9.2.10 LOADERS - Probleme mit Ladern

Fehlermeldung	Ursache, Behebung
E001-LOADERS Aktion {val} des Laders {val} ist misslungen	=> Probleme am Lader => Verkabelung prüfen => sm_slu scan = SCSI-Zugang prüfen => evtl. kurz ausschalten
E003-LOADERS Das spezifizierte Fach {val} des Laders {val} ist belegt.	Unload auf bereits belegten Slot => im Archiv Slotinfo prüfen und ggf.berichtigen => Archivabgleich falls manuelle Veränderungen im Lader
E004-LOADERS Achtung Lader {val} ist autounload-fähig! Autounload wird aktiviert!	keine Aktion nötig
E005-LOADERS Achtung für das Laufwerk {val} konnte die interne Laufwerknummer {val} nicht im Lader {val} gefunden werden! Bitte Laufwerkskonfiguration prüfen.	=> Laderkonfig.prüfen; ggf. berichtigen

9.2.11 MEDIA - Probleme mit Medien

Fehlermeldung	Ursache, Behebung
E001-MEDIA {val} Fehler: {val}	=>support@sep.de
E002-MEDIA Fehler bei Submit {val} in Queue {val}	Queuemanager läuft nicht => sm_shutdown — sm_startup
E003-MEDIA Fehler beim Laden des Mediums aus Fach {val}.	Fehler aus EXXX-LOADERS
E004-MEDIA Archiv-Verwaltung gibt keinen Status zurück.	=>support@sep.de
E005-MEDIA Archivaktion {val} fehlerhaft beendet: {val}	=>support@sep.de
E006-MEDIA Archivaktion {val} fehlerhaft beendet - unbekannter EXIT-Code: {val}	=>support@sep.de

9.2.12 RESTORE - Rücksicherungsprobleme

Fehlermeldung	Ursache, Behebung
E001-RESTORE LIS_DB {val}	=>support@sep.de
E002-RESTORE Rücksicherungsauftrag {val} nicht konfiguriert - bitte nachholen	=>support@sep.de
E003-RESTORE Rücksicherung {val} fehlerhaft beendet: {val}	=>support@sep.de
E004-RESTORE selektive Generations-Rücksicherung hat keine Savesets gefunden	=>support@sep.de
E005-RESTORE selektive Generations-Rücksicherung kann den benötigten Memory nicht belegen ({val}).	=>support@sep.de
E006-RESTORE SEARCH_LIS {val} ist ungültiger Auftragsname.	=>support@sep.de

9.2.13 SBC_COM - Meldungen des Sesam Backup Clients

Fehlermeldung	Ursache, Behebung
E000-SBC_COM falscher Parameter	=> Usage sm_sbc_com -h
E001-SBC_COM falsche Anzahl an Parametern	=> Usage sm_sbc_com -h
E002-SBC_COM fehlende oder falsche Pflichtangabe Auftrag	=> Auftragsname mu ss gegeben werden
E003-SBC_COM Fehler in Datenbank SELECT auf Tabelle	=>support@sep.de
E004-SBC_COM fehlende Pflichtangaben zu Label, Mediapool oder Laufwerknummer	=> Pflichtangaben einpflegen
E005-SBC_COM FIND_DRIVES_OF_POOL für Medienpool {val} ist kein Laufwerk konfiguriert	=> in GUI Medienpool vervollständigen
E006-SBC_COM eine externe Sitzung {val} wurde noch nicht eröffnet	=> zuerst externe Sitzung öffnen
E007-SBC_COM fehlende Pflichtangabe zu Savesetname	=> Pflichtangabe einpflegen
E008-SBC_COM GET_SEGM_AND_OFFSETS konnte die ausgewählte Datei {val} im Saveset {val} nicht finden	=> Dateinamen berichtigen
E009-SBC_COM GET_SEGM_AND_OFFSETS Formatfehler in Zeile [{val}]	=>support@sep.de
E011-SBC_COM falscher Parameter -s {val} : bitte savesetname@starting-time wie von Open-Restore zurückgegeben benutzen	=> savesetname@starting-time wie von Open-Restore zurückgegeben benutzen
E013-SBC_COM Restore nicht möglich, da die Sicherung {val} nicht erfolgreich war	evtl.anderen Saveset des gleichen Auftrages benutzen
E014-SBC_COM INQUIRE_INFO falscher Typ	=> Usage sm_sbc_com -h
E015-SBC_COM INQUIRE_INFO Saveset {val} existiert nicht	=> evtl. anderen Saveset des gleichen Auftrages benutzen
E016-SBC_COM INQUIRE_INFO kann LIS-Datei {val} zu Saveset {val} nicht zum Lesen öffnen	Schreibweise auf Fehler prüfen
E017-SBC_COM während Queue-Eintrag (submit) sm_sbc_com_ext	=>support@sep.de
E018-SBC_COM Fehler beim Datenbank UPDATE in Tabelle	=>support@sep.de
E019-SBC_COM CONNECT_BACKUP Fehler beim Laden des Mediums {val} in Laufwerk {val}	Fehler aus Exxx-LOADERS
E020-SBC_COM LOTUS_SAVESET keine Sicherung mit Datei {val} gefunden	evtl.anderen Saveset benutzen
E021-SBC_COM Fehler bei Dateikopie {val} {val}	=>support@sep.de
E022-SBC_COM Medium {val} ist derzeit nicht verfügbar	wird anderweitig benutzt => warten bis frei => andere Aktion abbrechen

9.2.14 SEPULER - Meldungen des SEP_sesam Schedulers

Fehlermeldung	Ursache, Behebung
E001-SEPULER Fehler bei Initialisierung der Queue {val} (Typ {val})	=> Queuemanager läuft nicht => sm_shutdown — sm_startup
E002-SEPULER Fehler bei Submit {val} in Queue {val} (Typ {val})	=> wie E001-SEPULER
E003-SEPULER Duplizierung des PrimaryKeys für Eintrag in DB:results	=>support@sep.de
E005-SEPULER ungültiges Kommando: {val}	=>support@sep.de
E006-SEPULER Rücksicherungsauftrag {val} ist noch nicht konfiguriert - bitte nachholen	=> nachholen im Rücksicherungsassistenten
E007-SEPULER READ_INI Fehler {val} gemeldet	=>support@sep.de
E008-SEPULER Der Laufwerksgruppe {val} sind keine Laufwerke zugeordnet.	=> andere Gruppe oder Laufwerk wählen => Konfiguration ändern
E009-SEPULER Textüberlauf: {val} -> {val}	=>support@sep.de
E010-SEPULER falscher Zyklustyp {val}	=> anderen Typ wählen
E011-SEPULER Fehler bei der Aktualisierung der benutzerdefinierten Liste des Zeitplans {val}.	=>support@sep.de
E012-SEPULER Die Vorausberechnung für die nächste Ausführung für den Zeitplan {val} errechnet immer dasselbe Ergebnis.	=>support@sep.de
E013-SEPULER Für den Medienpool {val} konnte keine Laufwerksgruppe bestimmt werden.	=> Konfiguration berichtigen

9.2.15 SONSTIGE

Fehlermeldung	Ursache, Behebung
E001-START Laufwerk nicht verfügbar ({val})	=> Konfiguration berichtigen
E002-START Laufwerk {val} existiert nicht ({val})	=> Konfiguration berichtigen
E001-STARTUP Verzeichnis {val} wurde nicht gefunden	=> Problem nach Installation => Verzeichnis manuell anlegen
E001-WATCH Empfangene Performance Daten mit ungültigem Format: {val}	SMS-Server läuft nicht => sm_main reload sms Diese Meldung darf vorübergehend erscheinen!
E002-WATCH Bearbeitung Mediumende fehlerhaft beendet: {val}	=>support@sep.de