

## S14 – z/VSE und CICS : Hints & Tipps

*Dagmar Kruse*                      ([dkruse@de.ibm.com](mailto:dkruse@de.ibm.com))                      *IBM Deutschland GmbH*

*Heinz Peter Maassen* ([hp.maassen@lattwein.de](mailto:hp.maassen@lattwein.de))                      *Lattwein GmbH*

- FSU von z/VSE 3.1 auf z/VSE V4.1 oder V4.2
- TCPIP 150F -APARS & „Hints and Tips z/VSE4.1“
- VSE/VSAM im z/VSE V3.1 und V4.1
  
- VSE Power JECL Erweiterungen
- VSAM Backup und Restore mit NOREWIND
- VSAM Redirector Capture Feature
  
- Verfügbare GSE-Requirements
- Password-Regeln verwalten über BSM
- Migration auf CICS TS
- EZASMI

## unbedingt beachten :

- Program Directory z/VSE V 4.2.0 (GI11-2698-04) und
- ‚IBM z/VSE System Upgrade and Service for VSE‘, **Kapitel 5**  
(SC33-8303)

## Vor dem FSU anpassen:

- **Label Area** liegt nur noch auf virtueller Platte,  
→ in \$0jCL.PROC:  

```
// EXEC PROC=STDLABEL                                (<--- remove this statement)
// VDISK UNIT=FDL,BLKS=2880,VOLID=VDIDLA,USAGE=DLA
```
- Nur noch ein Supervisor (**\$\$\$SUPI**) → IPL-Proc anpassen!
- Evtl. Prozeduren **STDLABEL**, **STDLABUP** überprüfen:
  - // DLBL BSTCNTL,'VSE.BSTCNTL.FILE',,VSAM,CAT=VSESPUC  
muss in STDLABUP enthalten sein,
  - Nicht-Standard POWER-File-Größen bzw. DLBLs  
→ Prozedur „STDLABEL“ muss angepasst werden



## Außerdem beim FSU zu beachten:

- POWER Migration auf **neues Format** im Stage 2, danach kein Restart des z/VSE3.1 Systems von DOSRES mehr möglich!  
→ **unbedingt POFFLOAD am Ende von Stage 1 !!!**
- Partition-ALLOC für VSE/POWER evtl. erhöhen,  
(s. SKALLOCA, SKALLOCB in ICCF LIB 59)
- Den Aufruf des BSMs in Proc \$BJCL auf **OS390-Mode** abändern:  
EXEC BSTPSTS,DSPACE=3M,OS390
- **Änderung bei der Passwortverwaltung (ab z/VSE3.1.1)**
  - Passwort-Regeln werden über BSM verwaltet  
via EXEC BSTADMIN PERFORM PASSWORD
  - **IESIRCVT aus Proc USERBG löschen**  
(„überschreibt“ sonst die Definitionen vom BSM)
    - Aktiviert Definitionen aus Skeleton IESLOGO, z.B: MAXNUMSO



## FSU ist ab z/VSE V.3.1 vom VSAM-VTAPE möglich!

- ➔ Details im Manual: „System Upgrade and Service“, **Appendix D**
- Base-Tape über Internet beziehen oder CD-ROM (AWS-Format)
- per FTP in vordefinierte ESDS-Datei übertragen
  - s. Appendix D.1.3 (recfm v, lrecl 32758)
- In FSU-Dialogen: ‚VIRTUAL TAPE 1‘ angeben

## Neuinstallation ab z/VSE V.3.1.1:

- **Neue Security-Konzept ist aktiv** (keine DTSECTXN)
- Umschalten auf altes Security-Konzept möglich, s. Administration Guide

- 
- Mit z/VSE4.2 ausgeliefert TCP/IP for VSE Service Pack 1.5.0 F  
Service-Stand: GA-Code + APARs (PK65428, PK66917 + PK67333)

## Verfügbare APARS für TCPIP/VSE 150F:

- PK65428 : GA-Code (Corektur von PK33472)
- PK66917 (OSA-PORT Adapter der z10, z/VSE4.1, 06/2008)
- PK67333 (Sammel-APAR, 06/2008)

→ in z/VSE4.2 integriert

- [PK70370](#) (Sammel-APAR, 09/2008, z/VSE3.1, Z/VSE4.1)
- PK71366 ( ZAPs von PK70370 für z/VSE4.2)
- [PK74055](#) (Sammel-APAR, [11/2008](#), z/VSE3.1 – z/VSE4.2)

PTFs enthalten nur Änderungen, vorherigen PTFs sind Prereq's

---

„Hints and Tips for z/VSE V.4.1“

seit Aug. 2008 auf z/VSE-Homepage:

<http://www-3.ibm.com/servers/eserver/zseries/zvse/documentation/index.html>

Im z/VSE V3.1 und V4.1 erzeugt VSE/VSAM  
häufiger SDUMPs als früher

Zusätzliche Überprüfungen wurden eingebaut,  
um Datenverluste zu vermeiden

- ungültige Relative Byte Address (RBA) oder ungültige CI-number  
→ SDUMP wird gezogen

→ Ihrer Anwendungen können daher  
unerwartet abbrechen

Einige SDUMPS wurden wieder zurückgenommen:

| z/VSE V3.1<br>-APARs | z/VSE V4.1<br>- APARs |   |
|----------------------|-----------------------|---|
| DY46855<br>(08/2008) | DY46878<br>(08/2008)  | RESTRICT SDUMPS IN SOME INVALID RBA<br>CASES                            |
| DY46900<br>(05/2008) | DY46899<br>(05/2008)  | SKIP SDUMP FOR INVALID RBA CASE IF<br>INVALID RBA CAME FROM APPLICATION |
| DY46961<br>(03/2008) | DY46956<br>(03/2008)  | MSG0S24I MSG0S29I SDUMP ISSUED BY<br>VSE/VSAM UNNECESSARILY             |

„Dumped“ Ihre Anwendung auch weiterhin?

→ unbedingt nachgehen !



# VSE/VSAM im z/VSE V3.1 und V4.1

Neue Funktion (per APAR 2.HJ 2008 verfügbar, s. nächste Seite):

**Wahlweises Annullieren Puffer nach  
„No-Record-Found“ und „ENDREQ“ - Situation**

über Programm IKQVEDA (SNAP-Dumps)

enable/disable SNAP=0012 (buffer invalidation for cluster/partition)  
(Default:disable)

Ausgangsproblem:SHR(2),AIX, Opened for Input im CICS

| z/VSE V3.1<br>-APARs | z/VSE V4.1 -<br>APARs |  |
|----------------------|-----------------------|--|
| DY46718              | DY46791<br>(10/2007)  | ENDREQ DOES NOT HELP ALLEVIATE<br>RECORD NOT FOUND SITUATION |

→ **Puffer** werden durch das APAR **stets annulliert**

→ Aber,

wenn Anwendungen bewusst ein „No-Record-Found“ erzeugen und verarbeiten, kann es zu **Performance-Einbußen** kommen

Daher jetzt wahlweises Annullieren der Puffer möglich mit:

| z/VSE V3.1 - APARs | z/VSE V4.1 - APARs   |   |
|--------------------|----------------------|---|
| DY46906            | DY46902<br>(10/2008) | ENDREQ DOES NOT HELP ALLEVIATE RECORD NOT FOUND (RNF) SITUATION WITH LSR ( <b>SNAP012 for LSR</b> ) |
| DY46909            | DY46942<br>(07/2008) | PERFORMANCE DEGREDATION WITH CERTAIN APPLICATIONS ( <b>SNAP012 for NSR</b> )                        |

- FSU von z/VSE 3.1 auf z/VSE V4.1 oder V4.2
- TCPIP 150F -APARS & „Hints and Tips z/VSE4.1“
- VSE/VSAM im z/VSE V3.1 und V4.1
  
- VSE Power JECL Erweiterungen
- VSAM Backup und Restore mit NOREWIND
- VSAM Redirector Capture Feature
  
- Verfügbare GSE-Requirements
- Password-Regeln verwalten über BSM
- Migration auf CICS TS
- EZASMI

## VSE Power JECL Erweiterungen

Im z/VSE 4.1 gibt es im VSE/Power neue JECL Erweiterungen:

- \* \$\$ **LSTDUP**, \* \$\$ **PUNDUP**
  - Kopien von Queue Einträgen mit unterschiedlichen Spooling Optionen erstellen (**Ausdruck / EMAIL / PDF / FTP** )
- Automatisches Löschen nach Ablaufdatum / Zeit
  - Neue \* \$\$ LST-, \* \$\$ PUN- Operanden: **EXPDAYS** und **EXPHRS**
- Anhängen der POFFLOAD an die vorherige Ausgabe:
  - Neuer POFFLOAD Operand: **APPEND**
- Queue Einträge nach Alter verwalten, siehe II Dialog 3.2

# VSE Power JECL Erweiterungen

## Beispiel für \* \$\$ LSTDUP:

```

* $$ JOB JNM=HUGO,CLASS=A,DISP=D
* $$ LST CLASS=A,DISP=H,LST=FEE,
* $$ LSTDUP JNM=DUPHUGO1,CLASS=M,DISP=K,DEST=DAGM,
* $$ LSTDUP JNM=DUPHUGO2,CLASS=N,DISP=H,UINF=MAAS
// JOB TEST
// LOG
* TEST OF LSTDUP
/*
/ &
* $$ EOJ

```

C  
C

## Ergibt in den POWER-Queues:

```

F1 0001 1R46I LIST QUEUE      P D C S PAGES CC
F1 0001 1R46I HUGO            03644 3 H A 1 1
F1 0001 1R46I DUPHUGO2      03646 3 H N 1 1
F1 0001 1R46I XMIT QUEUE      P D C I LINES
F1 0001 1R46I DUPHUGO1      03645 3 K M L 3

```

Achtung: nur die folgenden Operanden sind bei LSTDUP, PUNDUP erlaubt:

JNM=, CLASS=, DISP=, PRI=, COPY=, DEST=, TDISP=, REMOTE=, DIST=, SYSID=, UINF=,  
 EXPDAYS=, EXPHRS=, EXPMOM=NULL (Reset EXPDAYS+EXPHRS)

## PCOPY - Kopieren von Power Queue Einträgen:

- PCOPY **LST**,LCMA,11234,CQNUM=00001,CLASS=V,DEST=(\*,MAINT)
- PCOPY **PUN**,CICSLW,10200,CQNUM=8777,CLASS=V,DEST=(\*,MAINT)
- PCOPY **XMT**,DITTOTT,402,CQNUM=8770,CLASS=A
- PCOPY **DEL**,LCMA,8477,CQNUM=12607,CLASS=B,DISP=D

Man kann auch Jobs, die gerade Laufen von der ursprünglichen Disposition K oder L auf D ändern, wenn z.B. für einen Langläufer wie CICSICCF ein neuer Job in die RDR Queue gestellt wird.

Angezeigt wird die Disposition \*, die original Disposition wird mit dem Operanden FULL=YES angezeigt.

Siehe nächste Seite →

# VSE Power JECL Erweiterungen

## PCOPY - Kopieren von Power Queue Einträgen:

### **d lst,hugo,full=yes**

```
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I LIST QUEUE P D C S PAGES CC FORM BU
F1 0001 1R46I HUGO 11082 3 H A 1 1 +TO=(DAGM) FROM
F1 0001 D=20/10/2008 DBGP=000001 L=00000004
F1 0001 QNUM=00844 T=21:15:45
```

### **pcopy lst,hugo,11082,cqnum=844,class=v**

```
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1QK5I JOB HUGO 11096 IN LST QUEUE SUCCESSFULLY CREATED BY PCOPY
```

### **d lst,hugo**

```
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I LIST QUEUE P D C S PAGES CC FORM BU
F1 0001 1R46I HUGO 11082 3 H A 1 1 +TO=(DAGM) FROM=(DAGM)
F1 0001 1R46I HUGO 11096 3 H V 1 1 -TO=(DAGM) FROM=(DAGM)
```

## PALTER von aktiven Jobs:

**d rdr,cicslw,full=yes**

AR 0015 1C39I COMMAND PASSED TO VSE/POWER

F1 0001 1R46I READER QUEUE P D C S CARDS B

**F1 0001 1R46I CICSLW 44632 8 \* 5 53 PART=F5 FROM=(SYSA)**

F1 0001 D=28/01/2008 DBGP=000001 **ORGDP=K**

F1 0001 QNUM=01459 T=15:36:50

F1 0001 1R46I CICSLW 05559 8 L 5 53 FROM=(SYSA)

F1 0001 D=06/10/2008 DBGP=000001

F1 0001 QNUM=01865 T=15:56:59

**a rdr,cicslw,44632,disp=d**

AR 0015 1C39I COMMAND PASSED TO VSE/POWER

F1 0001 1R88I OK : 1 ENTRY PROCESSED BY A RDR,CICSLW,44632,DISP=D

**d rdr,cicslw,full=yes**

AR 0015 1C39I COMMAND PASSED TO VSE/POWER

F1 0001 1R46I READER QUEUE P D C S CARDS B

F1 0001 1R46I CICSLW 44632 8 \* 5 53 PART=F5 FROM=(SYSA)

F1 0001 D=28/01/2008 DBGP=000001 **ORGDP=D**

F1 0001 QNUM=01459 T=15:36:50

F1 0001 1R46I CICSLW 05559 8 L 5 53 FROM=(SYSA)

F1 0001 D=06/10/2008 DBGP=000001

F1 0001 QNUM=01865 T=15:56:59



## Beispiel für EXPDAYS UND EXPHRS:

```
* $$ JOB JNM=EMIL,CLASS=A,DISP=D
* $$ LST CLASS=A,DISP=H,FNO=0112,EXPDAYS=7,EXHRS=12
// JOB TEST
// LOG
* TEST OF EXPDAYS
/*
/&
* $$ EOJ
```

Expiration **DAYS** und **HOURS**. Hiermit wird die Zeit definiert nach der der Queue Eintrag automatisch von den Power Queues gelöscht wird. Falls beide Operanden DAYS und HOURS angegeben werden, so werden die Zeiten addiert. Die Liste EMIL wird in 7 Tagen + 12 Stunden gelöscht. (1 Woche bleibt die Liste erhalten.)

# VSE Power JECL Erweiterungen

IESBQUI

MANAGE BATCH QUEUES

SELECT one of the following:

QUEUE..... 1

1 = List Queue

2 = Reader Queue

3 = Punch Queue

4 = Transmit Queue

5 = Wait for Run Subqueue

6 = In-Creation Queue

(Only for selections 1 to 5)

To list only jobs whose names begin with certain characters

(Only valid for selections 1 to 4 above)

To list only jobs in a certain class

To list only this user's jobs

specify: (Only for selections 1 to 4)

1 = Oldest entry on top

2 = Newest entry on top

3 = Earliest entry to expire on top

Number of sorted entries to be shown

SPECIFY optionally:

PREFIX..... \_\_\_\_\_

CLASS..... \_

USER..... \_\_\_\_\_

**SORTED..... \_**

**LIMIT..... \_\_\_\_**

PF1=HELP

3=END

4=RETURN

**SORT=OLD** auch schon mit VSE Power 7.1 möglich! (\*)

**D lst,cicslw**

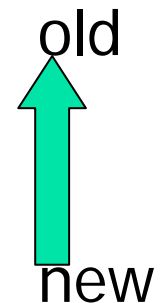
| F1 | 0001 | 1R46I | LIST QUEUE | P     | D | C | S | PAGES | CC | FORM            | B           |
|----|------|-------|------------|-------|---|---|---|-------|----|-----------------|-------------|
| F1 | 0001 | 1R46I | CICSLW     | 18527 | 8 | D | A | 100   | 1  | TO=(SYSA)       | FROM=(SYSA) |
| F1 | 0001 | 1R46I | CICSLW     | 41183 | 8 | D | A | 197   | 1  | TO=(SYSA)       | FROM=(SYSA) |
| F1 | 0001 | 1R46I | CICSLW     | 37135 | 8 | D | A | 103   | 1  | TO=(SYSA)       | FROM=(SYSA) |
| F1 | 0001 | 1R46I | CICSLW     | 52200 | 8 | D | A | 207   | 1  | TO=(SYSA)       | FROM=(SYSA) |
| F1 | 0001 | 1R46I | CICSLW     | 44632 | 8 | D | A | 13    | 1  | TO=(SYSA)       | FROM=(SYSA) |
| F1 | 0001 | 1R46I | CICSLW     | 52199 | 5 | D | A | 19    | 1  | FROM=(SYSCICSB) |             |
| F1 | 0001 | 1R46I | CICSLW     | 44631 | 5 | D | A | 21    | 1  | FROM=(SYSCICSB) |             |

**d lst,cicslw,sort=old**

AR 0015 1C39I COMMAND PASSED TO VSE/POWER

F1 0001 1R46I FOR 'D LST,.,LIMIT=016,SORT=OLD' COLLECTED 007 OF 00007 ENTRIES

| F1 | 0001 | 1R46I | LIST QUEUE | P     | D | C | S | PAGES | CC | FORM         | B          |
|----|------|-------|------------|-------|---|---|---|-------|----|--------------|------------|
| F1 | 0001 | 1R46I | CICSLW     | 18527 | 8 | D | A | 100   | 1  | D=24/07/2008 | T=12:01:47 |
| F1 | 0001 | 1R46I | CICSLW     | 41183 | 8 | D | A | 197   | 1  | D=01/08/2008 | T=09:06:33 |
| F1 | 0001 | 1R46I | CICSLW     | 37135 | 8 | D | A | 103   | 1  | D=22/08/2008 | T=10:57:36 |
| F1 | 0001 | 1R46I | CICSLW     | 52199 | 5 | D | A | 19    | 1  | D=29/08/2008 | T=08:12:38 |
| F1 | 0001 | 1R46I | CICSLW     | 52200 | 8 | D | A | 207   | 1  | D=29/08/2008 | T=08:13:21 |
| F1 | 0001 | 1R46I | CICSLW     | 44631 | 5 | D | A | 21    | 1  | D=24/09/2008 | T=17:38:08 |
| F1 | 0001 | 1R46I | CICSLW     | 44632 | 8 | D | A | 13    | 1  | D=24/09/2008 | T=17:39:22 |



**Oldest on Top**

(\*) z/VSE 3.1

## SORT=OLD/NEW auch schon mit z/VSE 3.1

```
d lst,cicslw,sort=new
```

```
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
```

```
F1 0001 1R46I FOR 'D LST,.,LIMIT=016,SORT=NEW' COLLECTED 007 OF 00007 ENTRIES
```

```
F1 0001 1R46I LIST QUEUE P D C S PAGES CC FORM B
```

```
F1 0001 1R46I CICSLW 44632 8 D A 13 1 D=24/09/2008 T=17:39:22
```

```
F1 0001 1R46I CICSLW 44631 5 D A 21 1 D=24/09/2008 T=17:38:08
```

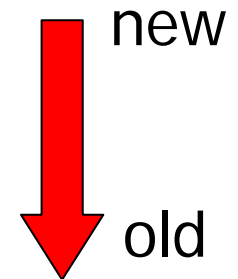
```
F1 0001 1R46I CICSLW 52200 8 D A 207 1 D=29/08/2008 T=08:13:21
```

```
F1 0001 1R46I CICSLW 52199 5 D A 19 1 D=29/08/2008 T=08:12:38
```

```
F1 0001 1R46I CICSLW 37135 8 D A 103 1 D=22/08/2008 T=10:57:36
```

```
F1 0001 1R46I CICSLW 41183 8 D A 197 1 D=01/08/2008 T=09:06:33
```

```
F1 0001 1R46I CICSLW 18527 8 D A 100 1 D=24/07/2008 T=12:01:47
```



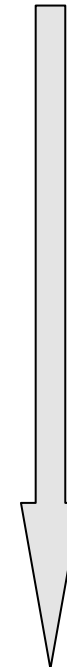
Newest on Top

## Display größte Queue Einträge

d biggest

AR 0015 1C39I COMMAND PASSED TO VSE/POWER

| F1 | 0001 | 1R4BI | 016 | BIGGEST  | SORTED | C | I | CARD/LINE | DBGP    | QNUM  | SUF | PAGES | QUE |
|----|------|-------|-----|----------|--------|---|---|-----------|---------|-------|-----|-------|-----|
| F1 | 0001 | 1R4BI | 001 | LCBZ     | 00285  | W | P | 78344     | 0000104 | 01539 |     |       | XMT |
| F1 | 0001 | 1R4BI | 002 | LCVS     | 56876  | T | L | 19645     | 0000040 | 01170 |     | 324   | LST |
| F1 | 0001 | 1R4BI | 003 | CICSLW   | 52200  | A | L | 15754     | 0000026 | 01840 |     | 207   | LST |
| F1 | 0001 | 1R4BI | 004 | LHMHPREP | 42176  | R | L | 12569     | 0000025 | 01755 |     | 226   | LST |
| F1 | 0001 | 1R4BI | 005 | CICSLW   | 41183  | A | L | 15022     | 0000025 | 01830 |     | 197   | LST |
| F1 | 0001 | 1R4BI | 006 | LC0CPGA  | 19490  | T | L | 11679     | 0000024 | 01801 |     | 213   | LST |
| F1 | 0001 | 1R4BI | 007 | JDSVT    | 04637  | P | L | 10811     | 0000023 | 01688 |     | 177   | LST |
| F1 | 0001 | 1R4BI | 008 | LC00     | 19414  | X | L | 10890     | 0000022 | 01541 |     | 182   | LST |
| F1 | 0001 | 1R4BI | 009 | DTRPTF03 | 27983  | B | L | 13927     | 0000019 | 01690 |     | 281   | LST |
| F1 | 0001 | 1R4BI | 010 | LS0      | 48443  | L | L | 7323      | 0000016 | 01815 |     | 110   | LST |
| F1 | 0001 | 1R4BI | 011 | CICSLW   | 37135  | A | L | 7847      | 0000014 | 01338 |     | 103   | LST |
| F1 | 0001 | 1R4BI | 012 | CICSLW   | 18527  | A | L | 7611      | 0000013 | 01763 |     | 100   | LST |
| F1 | 0001 | 1R4BI | 013 | STARTVCS | 56149  | A | L | 11025     | 0000012 | 01238 |     | 145   | LST |
| F1 | 0001 | 1R4BI | 014 | JDSVT    | 04634  | P | L | 5767      | 0000011 | 01829 |     | 92    | LST |
| F1 | 0001 | 1R4BI | 015 | DTRPTF03 | 19395  | B | L | 5501      | 0000008 | 01476 |     | 110   | LST |
| F1 | 0001 | 1R4BI | 016 | GODB2    | 56602  | L | L | 4050      | 0000008 | 01642 |     | 78    | LST |



## Neu mit z/VSE 4.1

pdisplay/pdelete/palter/phold/prelease/poffload  
select/ **poffload backup|pickup|save**  
mit Operand: CRDAYS=**|>|<|¬=|>=|<=**nnn

Listeinträge anzeigen, die älter als 100 Tage sind:

**d lst,crdays>100**

```
AR 0015 1C39I  COMMAND PASSED TO VSE/POWER
F1 0001 1R46I    LIST QUEUE    P D C S  PAGES  CC FORM
F1 0001 1R46I  PRIMSL2    08590 9 D A      3    1
F1 0001 1R46I  COMKRUS    09507 9 D A     22    1
F1 0001 1R46I  PRIMSL2    09610 9 D A      3    1
```

## Neu mit z/VSE 4.2

pdisplay/pdelete/palter/phold/poffload select  
mit Operand: CRAGE=|>|<|¬=|>=|<= hhmm

Listeinträge anzeigen, die älter als 48 Stunden sind:

**d lst,crage>4800**

```
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1R46I LIST QUEUE P D C S PAGES CC FORM
F1 0001 1R46I PRIMSL2 08590 9 D A 3 1
F1 0001 1R46I COMKRUS 09507 9 D A 22 1
```

# VSE Power JECL Erweiterungen

## Consolmeldungen ausblenden

```
F4 0004 // JOB TEST
      DATE 06/10/2008, CLOCK 15/29/48
F4 0004 * TEST OF EXPDAYS
F4 0004 EOJ TEST
      DATE 06/10/2008, CLOCK 15/29/48, DURATION    00/00/00
F1 0001 1Q34I  LST WAITING FOR WORK ON 00E
F4 0001 1Q34I  F4 WAITING FOR WORK
v msg,1q34i,nocons
AR 0015 1C39I COMMAND PASSED TO VSE/POWER
F1 0001 1Q8HI MESSAGE 1Q34I BEEN ENABLED, NOW DISABLED FOR CONSOLE
F4 0004 // JOB TEST
      DATE 06/10/2008, CLOCK 15/30/39
F4 0004 * TEST OF EXPDAYS
F4 0004 EOJ TEST
      DATE 06/10/2008, CLOCK 15/30/39, DURATION    00/00/00
V msg,alldisab,cons
```

Alle ausgeblendeten Meldungen werden weiterhin in die  
HardCopy Datei weggeschrieben !



Moderne Bandlaufwerke haben heute eine Kapazität von 80 oder mehr GB. 3390-Model 3 aber nur 2,7 GB. Daher lohnt es sich über VSAM Struktur und Datensicherung neue Gedanken zu machen.

Daten liegen heute in einem RAID Verbund- wo niemand mehr die physikalische Location einer VSAM Datei nachvollziehen kann.

1. Ist es noch sinnvoll, je Volume einen Userkatalog zu definieren ?
2. Oder kann man mit einem User Katalog 10 und noch mehr Volumes verwalten?
3. Was geschieht mit den bereits bestehenden Sicherungsbändern? Kann man diese sinnvoll auf 1 Kassette speichern und wenn ja wie?

User Kataloge können Dataspaces auf einem oder mehreren Volumes verwalten. Durch die Reduzierung der Anzahl von Userkatalogen wird die Datei Verwaltung einfacher. Man sollte aber folgendes beachten:

Kataloge sind auf exakt 1 Volume limitiert und können maximal 16 Extents gross sein. Deshalb sollte man den User Katalog mit genügender Größe anlegen. Der Default aus II reicht nicht für mehr als 90 Cluster auf einer 3390 aus.

Daumenregel: Je 3390 10 Cyl. Katalog Grösse definieren. Werden mehrere Volumes einem Usercatalog zugeordnet so sollten  $n \cdot 10$  CYL. definiert werden.

## Alte Backup Bänder zu 1 Backup Tape konsolidieren!

```
// JOB COPY BACKUP: INPUT TAPES auf 1 BACKUP TAPE für RESTORE mit NOREWIND !
// ASSGN SYS010,700          INPUT BACKUP UCAT120 (3390-3)
// ASSGN SYS011,701          INPUT BACKUP UCAT122 (3390-3)
// ASSGN SYS012,702          INPUT BACKUP UCAT124 (3390-3)
// ASSGN SYS013,790          OUTPUT TAPE = UCAT120 + UCAT122
// UPSI 1
// EXEC DITTO
$$DITTO TT      INPUT=SYS010,OUTPUT=SYS013,NFILES=EOV
$$DITTO FSR     INPUT=SYS011,NFILES=1                SKIP VOL1
$$DITTO BSR     INPUT=SYS013,NFILES=1                SKIP EOVS
$$DITTO TT      INPUT=SYS011,OUTPUT=SYS013,NFILES=EOV
$$DITTO FSR     INPUT=SYS011,NFILES=1                SKIP VOL1
$$DITTO BSR     INPUT=SYS013,NFILES=2                SKIP EOVS
$$DITTO TT      INPUT=SYS011,OUTPUT=SYS013,NFILES=EOV
/*
```

Warum ist das so ???



# VSAM Backup und Restore mit NOREWIND

## Backup Multiple Kataloge auf 1 TAPE !

```
// JOB BACKUP7
// ASSGN SYS005,790                OUTPUT = TAPE
// TLBL TAPE1,'BACKUP.FILE1'
// DLBL IJSYSUC,'UCAT220',,VSAM
// EXEC IDCAMS,SIZE=AUTO
  BACKUP (*) COMPACT -
  STDLABEL(TAPE1)
  NOREWIND
/*
// ASSGN SYS005,790
// TLBL TAPE2,'BACKUP.FILE2'
// DLBL IJSYSUC,'UCAT222',,VSAM
// EXEC IDCAMS,SIZE=AUTO
  BACKUP (*) COMPACT -
  STDLABEL(TAPE2) -
  NOREWIND
/*
/&
```

Mit COMPACT 170 MB, ohne COMPACT 409 MB !

Achtung:  
Der Operand:  
**NOREWIND**  
muss in jedem  
Step aufgeführt  
sein !!!

# VSAM Backup und Restore mit NOREWIND

## Restore: Multiple Kataloge von 1 TAPE !

```
// JOB RESTOR13
// ASSGN SYS004,790
// TLBL TAPE1,'BACKUP.FILE1,
// DLBL IJSYSUC,'UCAT220',,VSAM
// EXEC IDCAMS,SIZE=AUTO
  RESTORE OBJECTS((CPG.* -
  VOLUMES(PRD220)) -
  STDLABEL(TAPE1) -
  NOREWIND
/*
// TLBL TAPE2,'BACKUP.FILE2'
// EXEC IDCAMS,SIZE=AUTO
  RESTORE OBJECTS(* -
  VOLUMES(PRD222)) -
  CATALOG(UCAT222) -
  STDLABEL(TAPE2) -
  NOREWIND
/*
/&
```

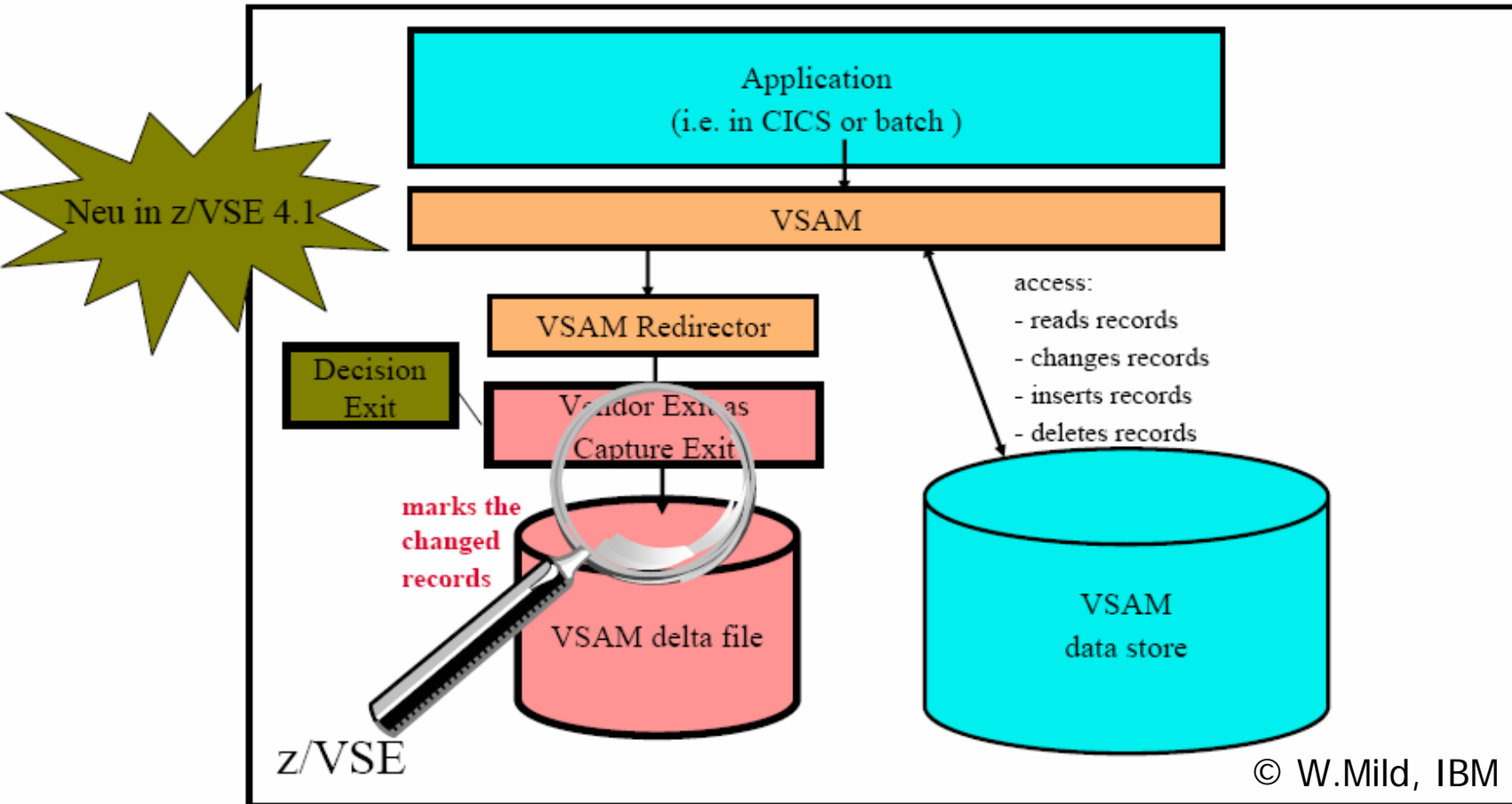
Achtung:  
Der Operand:  
**NOREWIND**  
muss in jedem Step  
aufgeführt sein !!!

## Seit z/VSE 4.1 gibt es Erweiterungen zum VSAM Redirector.

- Das **VSAM Capture Exit** ist ein Teil von VSAM Redirector. Es werden alle Änderungen im VSAM abgegriffen, ein Delta mit allen Änderungen wird gebildet und in einem anderen VSAM Cluster (das ist die "Delta File") gespeichert.
- Jeder Delta Datensatz enthält die VSAM Daten und zusätzliche Informationen über den Zeitpunkt (timestamp) und wer (partition, phase name, origin value, etc.) diesen Datensatz verändert hat. Ein Filter kann Zusatzentscheidungen treffen.
- **Diese Datensätze können asynchron von einem anderen Programm über Systemgrenzen hinweg abgegriffen werden.**
  - zum Beispiel können die VSE Konnektoren benutzt werden um den Prozess zu automatisieren und die Daten abzugreifen und in eine Datenbank einfügen.

# VSAM Redirector Capture Exit

## VSAM Redirector Capture.





Um den VSAM Redirector Client oder das VSAM Capture Exit zu aktivieren muss man das Skeleton SKRDCFG in der ICCF Library 59 benutzen. (\*)

Der VSAM Redirector Client und das VSAM Capture Exit werden automatisch ab z/VSE 4.1 installiert. (\*)

(\*) siehe Anhang.

# VSAM Redirector Capture Exit

Das VSAM Capture Exit kann in 3 Ausprägungen benutzt werden: \_

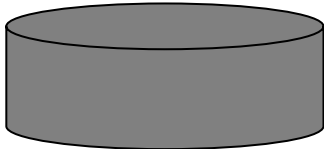
- MQSeries
- **VSAM Delta Cluster**
- Lokale Verarbeitung

Einstellungen bei VSAM Delta Cluster: Man unterscheidet 2 Methoden nämlich:

- Der **kumulativer Mode** benutzt eine VSAM KSDS DELTA Datei, deren Schlüssel identisch mit dem Original Cluster ist oder die RRN/RBA. Daher kann ein Satz sich nur einmal in der DELTA Datei befinden. Nur die letzte Änderung ist im der Delta File vorhanden. Das genügt um die Datei mit anderen DB's zu synchronisieren.
- Der **Journaling Mode** verwendet eine KSDS oder ESDS als VSAM Delta Cluster. Alle Änderungen werden in den Delta Cluster hinzugefügt mit Verwendung des TOD Clock als Key (bei KSDS). Somit werden alle Änderungen gespeichert.
- Beide Methoden können zur Synchronisation einer Datenbank verwendet werden. Unter Umständen kann die Reihenfolge entscheidend für die DB Logik sein. Dann muss der Journaling Mode verwendet werden.

# VSAM Redirector Capture Exit

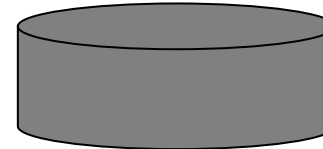
## ■ Kumulativer Mode



|       |        |
|-------|--------|
| 00001 | Insert |
| 00002 | Update |
| 00003 | Insert |
| 00004 | Delete |

Nur die letzte  
Veränderung wird in der  
Delta Datei gespeichert.

## ■ Journaling Mode



|       |        |
|-------|--------|
| 00001 | Insert |
| 00002 | Insert |
| 00003 | Insert |
| 00002 | Update |
| 00001 | Delete |
| 00004 | Insert |
| 00001 | Insert |
| 00002 | Update |
| 00004 | Update |
| 00004 | Delete |

Für alle drei VSAM Capture Exit Redirection Modi (MQSeries, VSAM Delta Cluster, und lokale Verarbeitung), kann man zusätzlich ein **DECISION EXIT** verwenden:

- Der Aufruf des Decision Exits erfolgt immer, wenn ein Delta Record/ Message geschrieben werden soll.
- Das Decision Exit entscheidet, ob ein Delta Record/ Message geschrieben wird oder ob er ignoriert wird. Diese Entscheidung kann über den Inhalt des Satzes getroffen werden.
- Das Decision Exit erhält den alten und den neuen Inhalt eines Satzes (Nur bei UPDATE).

# VSAM Redirector Capture Exit

Das VSAM Capture Exit erstellt Delta Records, die mit einem Delta Header in der Länge von 38 oder 42 Bytes beginnen.

|          |         |   |
|----------|---------|---|
| TODClock | 8 Bytes | TOD Clock Zeitpunkt des Updates (TOD = Time of day) |
| JobName  | 8 Bytes | Job Name des Programms                              |
| Phase    | 8 Bytes | Phasename des Programms                             |
| Origin   | 8 Bytes | Origin Wert, z.B. Label Name                        |
| PartID   | 2 Bytes | Partition ID  |
| OpCode   | 1 Byte  | 'I'=insert,'U'=update,'D'=delete                    |
| Flags    | 1 Byte  | X'01'= RRD/RBA follows                              |
| RecordLe | 2 Byte  | Länge des Satzes (exclusive Header und RBA/RRN)     |
| RBA/RRN  | 4 Bytes | RBA oder RelRecNo des Satzes                        |

Nur bei ESDS, RRDS, oder VRDS Dateien, folgt die 4-Byte RRN/RBA dem Capture Header.

# VSAM Redirector Capture Exit

Parameter zur Genrierung des VSAM Capture Exit:

**MODE= JOURNALING** oder  
**MODE= CUMULATIVE**

**ORIGIN=**

Gibt einen frei wählbaren 8-stelligen Namen an. Dieser Name wird in den Delta Header geschrieben

Dieser erscheint in jedem Delta Satz oder einer MQ Message und kann später ausgewertet werden.

Falls ORIGIN nicht definiert wird ist der aktuelle DLBL Name der Default Origin.

**DELTATYPE=KSDS| ESDS (Optional).**

ESDS ist nur erlaubt, falls MODE=JOURNALING angegeben wird.

**DELTADD=**

Gibt den DLBL Namen desDelta Clusters an. Dieser Parameter darf nur für Journaling oder Kumulativen Mode verwendet werden.

**SHARE=NONE**

| **ENDREQ** | **TCLOSE** (Optional). Gibt an, ob ENDREQ oder TCLOSE nach jedem Insert eines

Delta Records erfolgen soll.

# VSAM Redirector Capture Exit

Weitere Parameter zur Genrierung des VSAM Capture Exit:

## **DECEXIT=**

Name der Phase die das Decision Exit implementiert. Beispiel hierzu findet man in der ICCF Library 59 unter Skeleton SKDECEXT.

## **DELTACAT=**

Optionaler Parameter der die 44 Byte Fileld des User Katalogs beinhaltet, in dem der Delta Cluser definiert wurde. Der Parameter darf nur verwendet werden, wenn der Parameter DELTADD nicht verwendet wird. Er muss angewendet werden, wenn DELTACLU gesetzt ist.

## **DELTACLU=**

Optionaler Parameter der die 44 Byte Fileld des Delta Cluster enthält in dem die Delta Sätze gespeichert werden. Dieser Parameter darf nur gesetzt sein, falls DELTADD nicht angegeben wurde.

Er muss verwendet werden wenn DELTACAT definiert wird.

## **DSNSTR=**

Ein optionaler Parameter (Default ist 255). Gibt die Anzahl der Strings (0-255) an, für Dataset Name

Sharing verwendet werden. 0 bedeutet, dass kein Dataset Name Sharing angewendet wird.

## **IGNOREERROR=**

NO | **YES** (Optional). Falls auf YES gesetzt werden alle Fehler ignoriert und die Verarbeitung wird fortgesetzt. Dies kann zu inkonsistenten Daten führen, da nicht alle Änderungen geloggt werden.

# VSAM Redirector Capture Exit

Beispiel für das VSAM Capture Exit:

\*

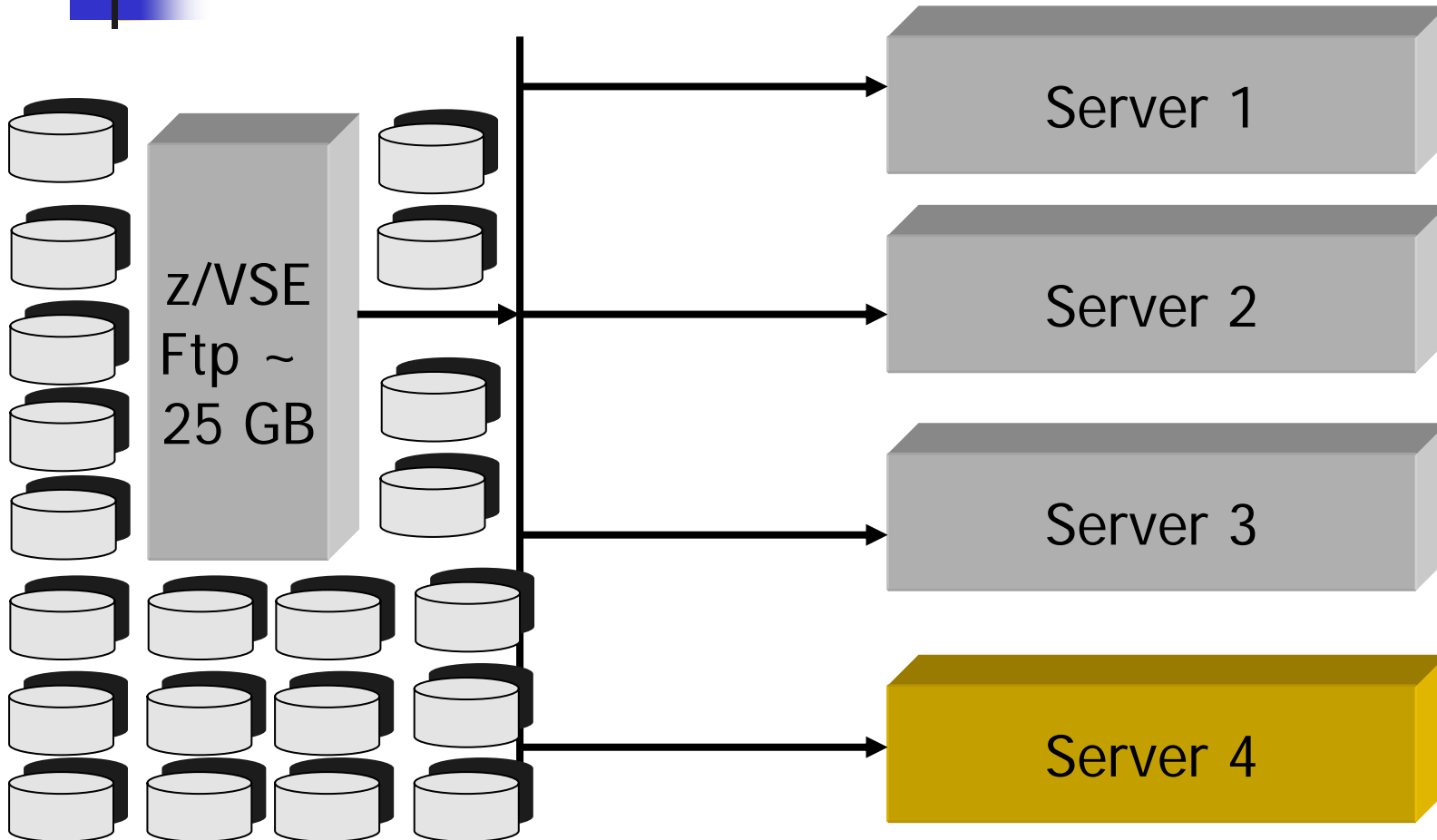
```
IESRDENT CATALOG='VSESP.USER.CATALOG',           X
          CLUSTER='CUSTOMER.KSDS.FILE',           X
          'IESVSCAP',                               X
          MODE=JOURNALING,                          X
          DELTADD='DELTAFI',                        X
          DELTATYPE=KSDS,                           X
          SHARE=ENDREQ,                              X
          ORIGIN='TEST2'
```

\*



# VSAM Redirector Capture Exit

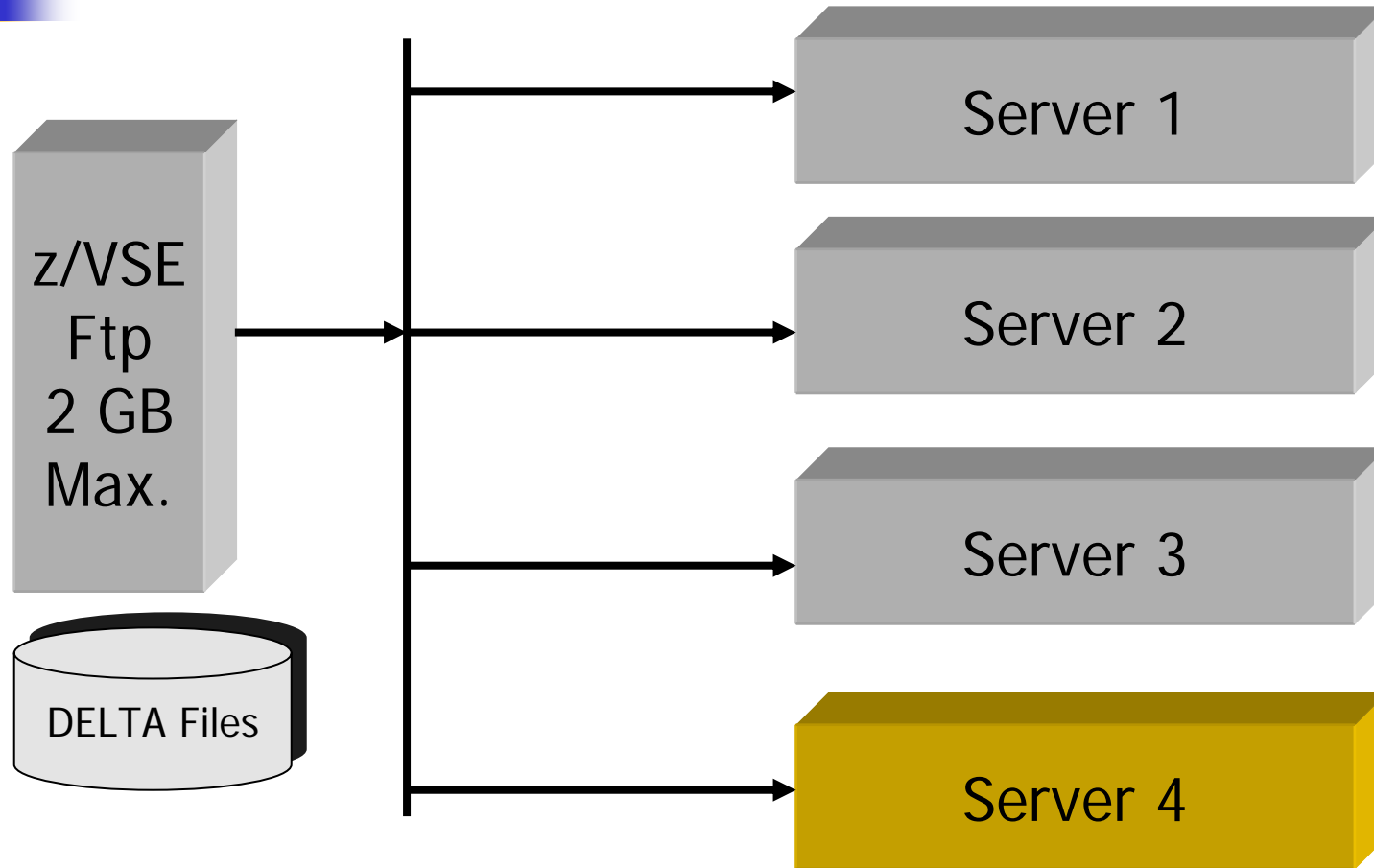
Kundenszenario ohne VSAM Capture Exit:



Um 19:30 starten FTP's die ca. xx GB Daten 1:1 in die DB der Serverumgebung senden.  
Server 4 startet um 3:30 Uhr eine zeitkritische Verarbeitung, die aber nur die Daten vom Host benutzt, die bis zu dieser Zeit auf die Server 1-3 übertragen wurden.

# VSAM Redirector Capture Exit

Kundenszenario mit VSAM Capture Exit:



Die Delta File Records könnten jede Stunde übertragen werden, daher wären die Daten auf der Server Seite praktisch ab 20:00 Uhr abends aktuell.

- FSU von z/VSE 3.1 auf z/VSE V4.1 oder V4.2
- TCPIP 150F -APARS & „Hints and Tips z/VSE4.1“
- VSE/VSAM im z/VSE V3.1 und V4.1
  
- VSE Power JECL Erweiterungen
- VSAM Backup und Restore mit NOREWIND
- VSAM Redirector Capture Feature
  
- **Verfügbare GSE-Requirements**
- **Password-Regeln verwalten über BSM**
- **Migration auf CICS TS**
- **EZASMI**

# Verfügbare GSE-Requirements

„Make FCB Name displayable and alterable“  
(Requirement an VSE/POWER von 2007)

Mit **z/VSE4.2** implementert:

```
// PDISPLAY LST|XMT,FULL=YES  
// PALTER LST,jobname,FCB=$$BFCB22  
// PALTER LST,jobname,CFCB=$$BFCB22
```

You can display the name of the FCB-image phase of a list entry by using the operand FULL of the PDISPLAY command, and alter it by using the PALTER command. However, you should note that only the name is altered: *the list data itself is not reformatted according to the new FCB-image phase.*

You can use the new **CFCB** operand to manipulate spool entries according to their name of the FCB-image phase.

➔ Details in „**VSE/POWER 8.2 Administration and Operation**“

## Verfügbare GSE-Requirements

„Jobname TAPESRVR should be changeable“

(Requirement von letzter Tagung)

Mit **z/VSE4.2** implementert:

```
// VTAPE START .....,JNM=powerjobname,...
```

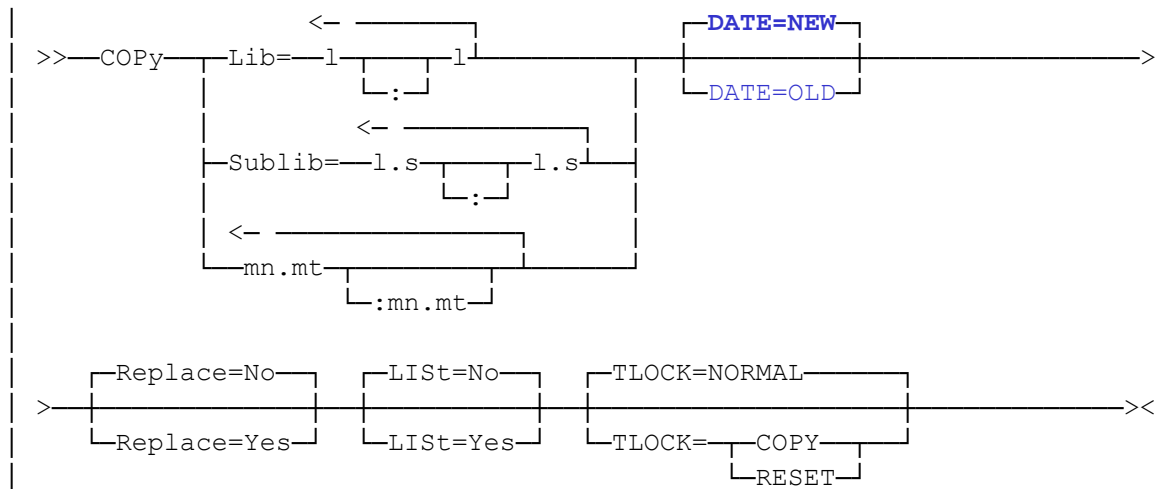
JNM='jobname':

Identifies the job starting the tape server. The default is 'TAPESRVR'.  
The maximum character length of jobname is 8.

# Verfügbare GSE-Requirements

“Parameter Date=New|Old for LIBR-Functions COPY, Move and REName”  
 (Requirement von letzter Tagung)

**Seit z/VSE3.1 implementiert für COPY und MOVE:**



Requirement für RENAME akzeptiert!

## EXEC BSTADMIN

```
PERFORM|PF PASSWORD [HISTORY|NOHISTORY]  
                    [LENGTH(minimum-pw-length)]  
                    [REVOKE(number-invalid-pws)|NOREVOKE]  
                    [WARNING(days-before-pw-expires)|NOWARNING]]
```

- **HISTORY**: neue PWD wird mit letzten 12 PWD\_Generationen verglichen,
    - **Achtung: Anzahl der History PWDs kann nicht geändert werden!**
  - **LENGTH**: Länge des PWD (3-8-stellig) , Default 3-stellig
  - **REVOKE**: Nach n fehlerhaften Versuchen, wird Userid gesperrt, Default: 5 Versuche
  - **WARNING**: Warnung bevor PWD expired, Default=7 Tage
- ➔ Details im z/VSE Administration Guide, Kap.3

# Passwort-Verwaltung über Basic Security Manager

0 exec bstadmin

BG-0000 BST901A ENTER COMMAND OR END

0 status

BG 0000 ...

BG 0000 PASSWORD PROCESSING OPTIONS:

BG 0000 12 GENERATIONS OF PREVIOUS PASSWORDS BEING MAINTAINED.

BG 0000 AFTER 5 CONSECUTIVE UNSUCCESSFUL PASSWORD ATTEMPTS,

BG 0000 A USERID WILL BE REVOKED.

BG 0000 PASSWORD EXPIRATION WARNING LEVEL IS 7 DAYS.

BG 0000 A PASSWORD CAN HAVE 3 TO 8 CHARACTERS.

.....

0 perform password nohistory

BG 0000 BST904I RETURN CODE OF PERFORM IS 00

BG-0000 BST901A ENTER COMMAND OR END

0 status

.....

BG 0000 PASSWORD PROCESSING OPTIONS:

BG 0000 NO PASSWORD HISTORY BEING MAINTAINED



## Warum überhaupt auf CICS TS migrieren?

- bietet sehr viele Vorteile
  - bessere Speichernutzung
  - neue Funktionen und Weiterentwicklungen
- Anwendungen (Command-Level) aus CICS/VSE weitestgehend kompatibel
- Nach z/VSE 4.2 läuft nur noch CICS TS

Aber die Umstellung braucht Zeit

➔ Planen Sie die Migration **JETZT** als Projekt !

## Oft behindert durch:

- **MACRO-Level wird nicht mehr im CICS TS unterstützt**  
(seit 1984 nicht mehr weiterentwickelt)  
→ Migration auf COMMAND-Level !!! (seit 1977 verfügbar)

## Häufige Fehler bei der Migration:

- **Standardeinstellungen** vom CICS/VSE werden **einfach übernommen!**
  - Neue /geänderte Parameter nicht angepasst
- alten Tabellen / Prozeduren ungeprüft verwendet!
- **Speicherbelegung** in neuem (E)DSA-Konzept nicht überprüft und angepasst
- Nicht genügend getestet

## Hilfreich sind

- Redbook „Migration to VSE/ESA 2.4 and CICS Transaction Server for VSE/ESA 1.1” (SG24-5595), 1999
  - gilt unverändert, bis auf Migration zum neuen BSM-Security-Konzept
- IBM z/VSE Administration Guide V4.1.1 (SC33-8304-01)
  - Migration auf externe Security
- Skeletons, Tabellen, Programme in ICCF Libr. 59
- GSE-Vorträge & Workshops, u.a.
  - ‚CICS Transaction Server‘ vom 9.12.- 11.12.2008

H.P. Maassen

## CICS Voraussetzungen für die EZA Interfaces (ab z/VSE 4.1) .

Um die neuen EZA API (EZASMI Macro und EZASOKET Call Interface) zu benutzen, muss in einer CICS TS Umgebung ein "Task-Related-User-Exit" (TRUE) gestartet werden. Der Name ist: **EZATRUE**. Dieses Programm stellt die Verarbeitungsumgebung für das EZA API her und lädt die entsprechenden Workbereiche.

Das Programm EZATRUE kann über das Programm **EZASTRUE** über eine der folgenden Möglichkeiten gestartet oder beendet werden:

- Transaction EZAT (**EZAT START** startet EZATRUE, **EZAT STOP** beendet den Exit).
- EZASTRUE als PLTPI Eintrag ( StartUp CICS) and als PLTSD Eintrag (Stop beim Shutdown)
- Aufruf über EXEC CICS LINK des Programms EZASTRUE mit der folgenden COMMAREA Parameter Liste:

## **COMMAREA Parameter Liste zum Aufruf von EZASTRUE.**

| Offset | Length | Description                                 |
|--------|--------|---|
| 0      | 8      | Eyecatcher "EZATRUE"                        |
| 8      | 1      | Request Type: "S" Start Request "T" Request |
| 9      | 1      | Return Code vom EZASTRUE:                   |
|        |        | 0 EZATRUE Start/Termination erfolgreich     |
|        |        | 4 EZATRUE bereits im Start- oder Term       |
|        |        | 8 Fehler bei Start/Terminate EZATRUE        |
|        |        | 16 Parameter Liste fehlerhaft               |

Status

**MVC COMAR,=CL8`EZATRUE`**

**MVC COMAR+8(2),=C`S`**

**EXEC CICS LINK PROGRAM(`EZASTRUE`) COMMAREA(COMAR) LENGTH(10)**

**CLI COMAR+9,C`0`**

**BE GESTARTET**

## Einschränkungen und Voraussetzungen für EZASMI .

Die folgenden Einschränkungen beziehen sich auf das EZAMI Macro API:

- Es muss **CICS/TS** sein nicht für CICS/VSE verfügbar!
- Es kann **nicht** in einer ICCF Pseudo Partition laufen.
- Locks: Es dürfen keine Locks gehalten sein, bei Verwendung dieser Befehle.
- INITAPI/TERMAPI Macros: Die INITAPI/TERMAPI Macros müssen **innerhalb einer Task** durchgeführt werden.
- Speicher, der von Socket Calls verwendet wird, muss den gleichen PIK haben, wie es im PSW steht.
- Falls Storage Protection im CICS TS definiert wurde, muss für alle Programme die EZASMI Macro API benutzen, EXECKEY(CICS) definiert sein. Das gilt auch für Programme, die EZASMI Programme über LINK aufrufen.
- TASKDATAKEY(CICS) für die Transaktion ist **nicht erforderlich**.

## Einschränkungen und Voraussetzungen für EZASMI .

### Addressability Mode (Amode) Überlegungen

- Das EZASMI Macro API kann nur aufgerufen werden, wenn sich das Programm im **A-Mode 31** befindet.
- Wird das EZASMI Macro API in CICS Transaktionen verwendet so muss vorher das EZA „Task-Related-User-Exit“ (TRUE) gestartet werden.

# Haben Sie noch Fragen?





Vielen Dank  
für Ihre  
Aufmerksamkeit

# Anhang

Der VSAM Redirector Client und das VSAM Capture Exit werden automatisch mit z/VSE 4.1 installiert. Dazu gehören die folgenden 5 Phasen:

|                |   |   |
|----------------|---|---|
| IKQVEX01.PHASE | > | <i>generische</i> Exit Phase                                      |
| IESREDIR.PHASE | > | VSAM Redirector Client Phase                                      |
| IESVSCAP.PHASE | > | VSAM Capture Exit Phase   |
| IESRDCFG.PHASE | > | Konfiguration   |
| IESRDANC.PHASE | > | Redirector Anchor Phase   |
| IESRDLDA.PHASE | > | Dieses Programm dient der Registrierung einer neuen Konfiguration |

# VSAM Redirector Capture Exit

Um den VSAM Redirector Client oder das VSAM Capture Exit zu aktivieren muss man das Skeleton SKRD CFG in der ICCF Library 59 benutzen.

1. Assemble/link des Members IESRDCFG, dann katalogisieren in die PRD2.CONFIG.
2. Laden der Phase IESRDCFG.PHASE in die SVA (optional).
3. Kopieren **IESVEX01.PHASE** in die PRD2.CONFIG with the name **IKQVEX01.PHASE**, to activate the exit phase.
4. Laden der Phase: IKQVEX01 in die SVA (optional). **Achtung: Das Original steht immer noch in der IJSYSRS.SYSLIB.**  
Falls man eine neue Konfiguration aktivieren möchte (ohne IPL) muss man ->
5. Die Phase IESRDANC.PHASE in die SVA laden (Das sollte nur einmal je IPL durchgeführt werden.)
6. Jetzt das Programm IESRDLDA ausführen. Die Phase IESRDCFG muss schon in die SVA geladen worden sein.(Siehe unter Step 2). Das Programm IESRDLDA wird dann eine neue Version der Konfiguration aktivieren und die Änderungen werden sofort aktiv. Betroffene VSAM Dateien die bereits geöffnet sind müssen erst geclosed und dann neu geöffnet werden! (Gilt besonders für CICS Files.)

## Space Allocation ohne KeyRange

- **Primary Space wird auf dem ersten Volume zur “Define Time“ angelegt.**
- **Falls VSE/VSAM mehr Speicherplatz während des Ladevorgangs oder bei der Verarbeitung braucht, wird sekundär Allocation auf dem 1. Volume angefordert.**
- **Nur für den Fall, dass auf dem 1. Volume kein Space mehr zur Verfügung steht und weiterer Platz benötigt wird, legt VSAM auf dem **2. Volume einen Space in der primären Grösse an.****
- **Das geschieht auch dann, wenn keine Sekundär Allocation vorgesehen wurde. Erst wenn dieser Platz belegt ist wird die sekundär Allocation auf dem 2. Volume durchgeführt.**