



# Das z/OS Betriebssystem und das TSO-Subsystem

Präsentation zum Assignment  
“REXX-Skript zum Auslesen einer Datenbank”

Vladimir Zhelezarov

.....

# Quellen

- IBM Corporation (2010)  
IBM Knowledge Center  
<https://www.ibm.com/support/knowledgecenter/zosbasics/com.ibm.zos.zmainframe/toc.htm>
- Ebbers, M. et al (2011)  
Introduction to the New Mainframe. z/OS Basics. IBM Red Book
- Bogdan, M.; Spruth, W. (2013)  
Enterprise Computing. Einführung in das Betriebssystem z/OS

# Mainframe Betriebssysteme

Die wichtigsten Betriebssysteme für den IBM Mainframe sind:

- z/OS:
  - Allzweck-Betriebssystem;
- z/VM:
  - Hypervisor für Virtuelle Maschinen;
- z/VSE:
  - Allzweck, aber mit begrenzter Funktionalität. Umstieg auf z/OS wird empfohlen;
- Linux für System Z:
  - Vollfähiges Linux auf dem Mainframe;
- z/TPF:
  - Spezialisiert als Transaktionenmonitor.

# z/OS Grundkonzepte

- Durch Trennung von Architektur und Implementierung wird eine volle Vor- und Rückwärtskompatibilität erreicht, sowie auch Sicherheit für die Investition in Mainframe-Hardware und Software;
- Der Fokus bei dem Mainframe und bei z/OS ist auf einer sehr schnellen Ein-/Ausgabe Verarbeitung bei der gleichzeitigen Arbeit von sehr vielen Benutzer;
- Das “z” in z/OS steht für “z”ero days downtime – Ausfallsicherheit ist Top-Priorität;
- z/OS unterscheidet den Problem-Modus, für die Benutzer und ihrer Anwendungen; und Supervisor-Modus – für Kernel-Aufrufe (Supervisor Calls). Somit ist eine bessere Zugriffsrechtenkontrolle gezielt;
- Mehrfache Redundanz in vielen Aspekte – in Hardware, sowie auch im Betriebssystem;
- 24-, 31-, or 64-bit Addressierung:
  - legacy-Anwendungen werden unterstützt;
  - neue Anwendungen nutzen besser die erhöhte Leistung der modernen Hardware

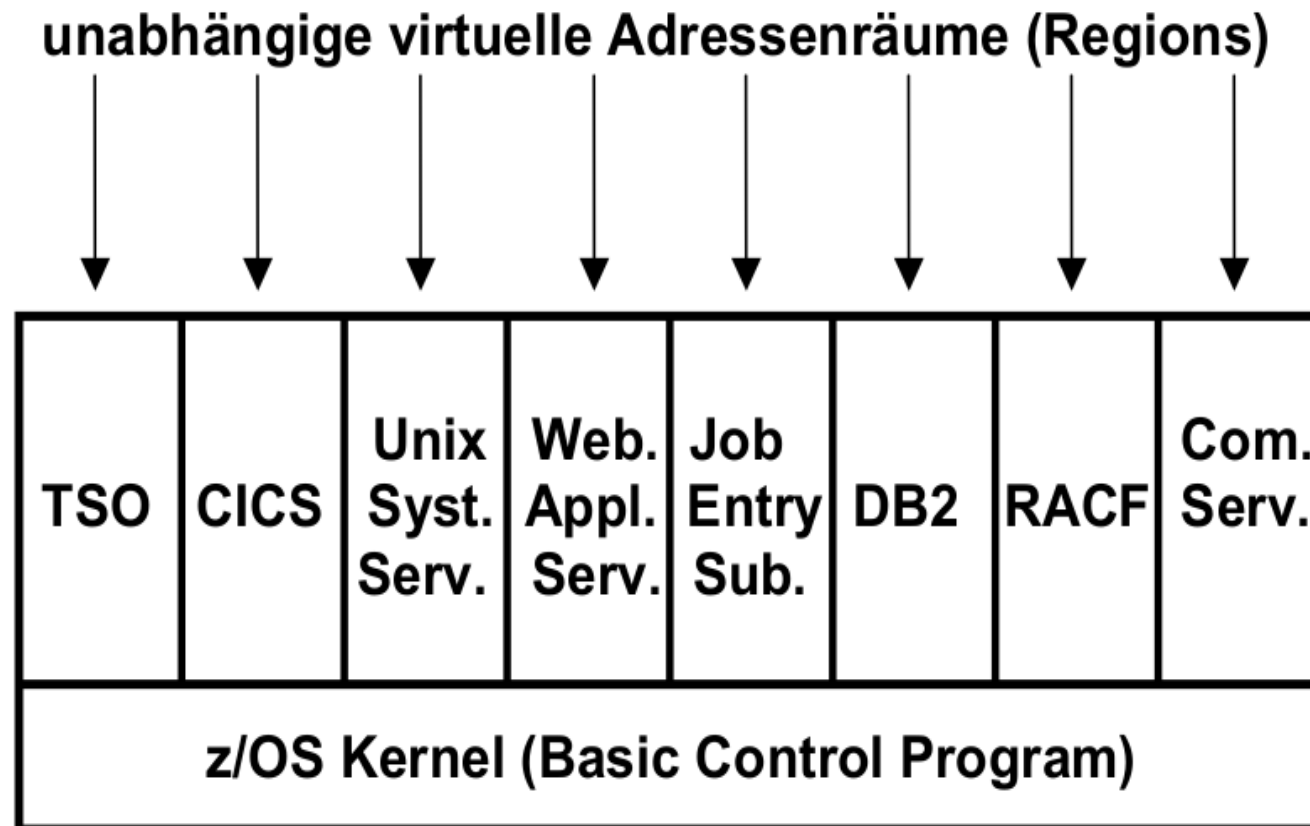
# z/OS: Speicherverwaltung

- Virtual Storage: jedem Programm erscheint den ganzen Speicher zur Verfügung zu haben. In Realität - eine Kombination aus Hauptspeicher und Plattenspeicher;
- Address Space: Der Bereich von virtuelle Adressen, die z/OS einem Prozess zuweist. Jeder Benutzer hat eigenen Adress Space, wo er Programme mit TCBs starten kann;
- TCB (Task Control Block): Ähnlich einem Process Control Block bei Unix (=Threads);
- Vorteile:
  - Erlaubt die Addressierung von mehr Speicher als im System verfügbar; Eventuelle Fehler bleiben beschränkt im Adressraum; Anwendungen sind voneinander isoliert;
- Adressraum-Isolation:
  - Unterscheidung zwischen Programme und Daten, die zu jedem Adressraum gehören;
  - Aufteilung auf private und geteilte Speicherbereiche, für die Interprozesskommunikation;
  - Schützt das Betriebssystem von den Benutzer und die Benutzer voneinander;
- Kommunikation zwischen Adressräume:
  - SRB (service request block): asynchrone Kommunikation;
  - Cross memory services: synchrone Kommunikation.

# z/OS: Datenspeicherung

- Die Daten in z/OS werden als Datasets gespeichert;
- Dataset: eine Datei, die aus Records besteht;
- Record: bestimmte Anzahl von Bytes, die zusammengehörige Daten beinhalten. Ein Record wird als Einheit betrachtet im Gegenteil zu Windows/Unix, wo die Dateien meistens sequenzielle Datenstreams sind;
- Sequentieller Dataset: Die Records sind nacheinander gespeichert;
- Partitioned DataSet: Besteht aus Directory und Members; Das Verzeichnis hat die Adressen von alle Members; Die Members haben sequenzelle Records;
- Virtual Storage Access Method Dataset: bezeichnet gleichzeitig den Dataset-Typ und die Zugangsmethode:
  - Dataset in Schlüsselfolge (KSDS): Wahlfreier Zugang über Schlüssel;
  - Dataset in Zugangsfolge (ESDS): Die Records werden sequentiell gespeichert und aufgerufen;
  - Relativ Record Dataset (RRDS): Auf die Record wird über Nummer zugegriffen;
  - Linear Dataset (LDS): Byte-Stream Daten, ähnlich wie Windows und Unix.

# Module in z/OS: Überblick





# Module in z/OS: Erläuterung

- TSO: time sharing option. Interaktive Verarbeitung von Benutzeranfragen;
- CICS: customer information control system. Der am weitesten verbreitete, IBM proprietäre Transaktionsmonitor;
- Unix System Services. Unix Schnittstelle sowie ein Unix kompatible Shell;
- WebSphere Web Application Server;
- Job Entry Subsystem. Subsystem für die Stapelverarbeitung;
- DB2. Eine Relationale Datenbank;
- RACF: Resource Access Control Facility. Sicherheitssystem;
- Communication Server. Sichere Kommunikation über TCP/IP, SNA und UNIX Schnittstellen.



# Batch-Verarbeitung

- Batch- (Stapel-) verarbeitung: für wiederholbare, geplante Aufgaben, wie z.B. Kundensaldoberechnung bei einer Bank, Lohnabrechnungen in einer Firma usw.
- Die Jobs werden in eigenen Prozesse ausgeführt;
- JCL (Job Control Language): die IBMs Script-Sprache zur Steuerung von Jobs (Aufträge für Stapelverarbeitung)

**Statement Bezeichnung beginnt in Spalte 12**



```
//SPRUTHC JOB (123456) , ' SPRUTH' , CLASS=A , MSGCLASS=H , MSGLEVEL=(1 , 1) ,  
//          NOTIFY=&SYSUID , TIME=1440  
//PROCLIB JCLLIB ORDER=CBC . SCBCPRC  
//CCL      EXEC PROC=EDCCB ,  
//          INFILE=' SPRUTH . TEST . C (HELLO1) '  
//          OUTFILE=' SPRUTH . TEST . LOAD (HELLO1) , DISP=SHR '
```

**Label, Spalte 3 - 10**

# Interaktive Verarbeitung mit TSO

- TSO/E (Time Sharing Option/Extensions) oder kurz TSO: Benutzerschnittstelle für die interaktive Verarbeitung von Anfragen;
- Kommunikation hat ursprünglich über einem IBM 3270 Terminal stattgefunden, aktuell wird ausschließlich dafür ein Software-Emulator benutzt;
- Nach dem Login ist der Benutzer im TSO Native Modus (ähnlich einer DOS Eingabeaufforderung) – dieser wird so direkt selten gebraucht wegen seiner begrenzten Funktionalität;
- Am häufigsten wird, anstelle von TSO Native, die ISPF Schnittstelle benutzt. ISPF (Interactive System Productivity Facility) ist Menü-basiertes Interaktives System mit erweiterter Leistung;

# TSO: Login

```
----- TSO/E LOGON -----

Enter LOGON parameters below:                                RACF LOGON parameters:

Userid   ==> ZPROF

Password ==>

New Password ==>

Procedure ==> IKJACCNT                                       Group Ident  ==>

Acct Nbr  ==> ACCNT#

Size      ==> 860000

Perform   ==>

Command   ==>

Enter an 'S' before each option desired below:
      -Nomail      -Nonotice      -Reconnect      -OIDcard

PF1/PF13 ==> Help    PF3/PF15 ==> Logoff    PA1 ==> Attention    PA2 ==> Reshow
You may request specific help information by entering a '?' in any entry field
```

# TSO: Native Mode

```
READY
  alloc dataset(zschol.test.cntl) volume(test01) unit(3390) tracks
space(2,1) recfm(f) lrecl(80) dsorg(ps)
READY
listds
  ENTER DATA SET NAME -
  zschol.test.cntl
  ZSCHOL.TEST.CNTL
  --RECFM-LRECL-BLKSIZE-DSORG
    F      80      80      PS
  --VOLUMES--
    TEST01
READY
```

# CLIST und REXX

- CLIST (command list) und REXX (Restructured Extended Executor) bieten Skript-Fähigkeiten an, um über TSO Aufgaben zu steuern oder zu automatisieren;
- CLIST ist eine Liste mit Befehle, die alle zusammen ausgeführt werden;
- REXX ist interpretive Sprache, kann aber auch kompiliert werden um die Ausführungszeit zu beschleunigen;
- Die Befehle werden in einer Datei gespeichert und dann als Skripten ähnlich aus TSO aufgerufen werden:
  - Für CLIST:
    - TSO EXEC 'CLIST AREA.COMMND'
  - Für REXX:
    - TSO EXEC 'PRAK.REXX.EXEC(MEMBER)'
- CLIST findet man nur auf dem Mainframe; REXX existiert auch auf andere Plattformen.

# REXX Beispiel

```
/* A short program to greet you */
/* First display a prompt */

say 'Please type your name and then press ENTER:'
parse pull answer /* Get the reply into answer */

/* If nothing was typed, then use a fixed greeting */
/* otherwise echo the name politely */

if answer='' then say ' Hello Stranger! '
              else say ' Hello ' answer '!'
```



# ISPF

- Bietet Zugang zu vielen der Funktionen an, die am häufigsten vom Benutzer benötigt werden;
- Nach dem Aufruf von ISPF über TSO taucht das Haupt-Menü von ISPF auf. Durch Angabe von kurzen Befehlen oder Nummern und die Benutzung von Tastaturkürzel kann man zu den Untermenüs navigieren und deren Funktionalität benutzen;
- Manche ISPF Software-Clients bieten auch GUI (Benutzeroberfläche) an, die direkt mit der Maus steuerbar sind.

# Anlegen eines Datasets in ISPF

```
Menu  RefList  Utilities  Help
-----
Allocate New Data Set
Command ==>
Data Set Name . . . : ZCHOL.TEST.CNTL
Management class . . . (Blank for default management class)
Storage class . . . . (Blank for default storage class)
Volume serial . . . . TEST01 (Blank for system default volume) **
Device type . . . . . (Generic unit or device address) **
Data class . . . . . (Blank for default data class)
Space units . . . . . TRACK (BLKS, TRKS, CYLS, KB, MB, BYTES
                             or RECORDS)
Average record unit . . (M, K, or U)
Primary quantity . . . 2 (In above units)
Secondary quantity . . . 1 (In above units)
Directory blocks . . . 0 (Zero for sequential data set) *
Record format . . . . F
Record length . . . . 80
Block size . . . . .
Data set name type : (LIBRARY, HFS, PDS, or blank) *
                    (YY/MM/DD, YYYY/MM/DD)
Expiration date . . . . YY.DDD, YYYY.DDD in Julian form
Enter "/" to select option DDDD for retention period in days
    Allocate Multiple Volumes or blank)

( * Specifying LIBRARY may override zero directory block)

( ** Only one of these fields may be specified)
F1=Help F2=Split F3=Exit F7=Backward F8=Forward F9=Swap F10=Actions F12=Cancel
```