Christian Grothoff
christian@grothoff.org

http://grothoff.org/christian/

These slides are based in part on materials provided by IBM’s Academic Initiative.
SMP/E

- z/OS tool for managing the installation of software products and to track modifications
- Code and its modifications are located in the SMP/E databased called the *consolidated software inventory* (CSI)
- CSI consists of one or more VSAM data sets
SMP/E Features

SMP/E controls changes at the component level:

- Selects proper “levels” of code to be installed from a large number of potential changes
- Calls system utilities to install the changes
- Keeps records of installed changes
- Can display status of software
- Allows reversal of changes
Elements

Elements are the basic building blocks of the z/OS system:

- Object modules
- Source code
- Macros
- Help-panels
- CLISTs and REXXs

Elements are associated with and dependend upon other products or services installed on the same z/OS system.
System Modifications (SYSMODs)

- SYSMOD contains the information SMP/E needs to install and track system modifications
- SYSMODs are a combination of elements and control information
Modification Text

- Contains the elements supplied by the SYSMOD
- Often only specifies the differences
Modification Control Statements (MCS)

• Indicate what elements are updated or replaced

• Specify how the SYSMOD relates to product software and other SYSMODs

• Designated by “++” as the first two characters
Types of SYSMODs

**FUNCTION**  Adds a new product or function

**PTF**  IBM correction to all z/OS installations

**APAR**  Temporary fix to a specific z/OS installation

**USERMOD**  Installation-provided modification
Function SYSMODs

- Function SYSMOD adds new elements to a system
- Function SYSMODs can provide new products, new version or release of a product or updated functions for an existing product
- All other types of SYSMODs are dependent upon the function SYSMOD
Example: Function SYSMODs

Introducing an Element (Function)

Function SYSMOD

MOD1
MOD2
MOD3
MOD4

System Libraries

LMOD2
MOD1
MOD2
MOD3
MOD4

Link Edit
Program Temporary Fixes (PTFs)

- Provided by IBM when a problem with a software element is discovered
- PTF SYSMOD contains “tested” fix for a problem; they should be installed even if you did not (yet) experience the particular problem they fix
- PTF SYSMOD is used to install the PTF
- PTF SYSMODs require presence of certain function SYSMODs
Authorized Program Analysis Report (APAR)

- Used to correct serious problems before PTF is ready for distribution
- APARs may just contain workarounds and could cause other problems; they should only be installed if you are experiencing a serious problem
- APAR can require the installation of other PTF or APAR SYSMODs
USERMOD SYSMODs

- Allow you to tailor IBM code to meet your specific needs
- Can replace, update or introduce elements
- Can have function, PTF, APAR or USERMOD SYSMODs as prerequisites
Example: USERMOD SYSMODs

Customizing an Element (USERMOD)

- LMOD2
- MOD1
- MOD2
- MOD3
- MOD4

{USERMOD}
Prerequisites and corequisites

- **Base function SYSMODs** add or replace an entire system function (and should have no dependencies)

- **Dependent function SYSMODs** provide an addition to an existing system function and have a base function SYSMOD as a prerequisite

- **PTF SYSMODs** can have function and PTF SYSMODs as prerequisites and other PTF SYSMODs as corequisites

- **APAR SYSMODs** can have APAR, PTF and function SYSMODs as prerequisites and other APAR SYSMODs as corequisites
Example: PTF prerequisites

PTF Prerequisite

PTF1

{ MOD1
  MOD2
  MOD3
  MOD4
}

LMOD2

PTF2
Tracking Elements of the System

• Need to manage prerequisites and corequisites
• The same module might be part of many different load modules

⇒ Replacing a module may require updates to many load modules!
Modification Identifiers

Each element is associated with three modification identifiers:

- Function Modification Identifier (FMID)
- Replacement Modification Identifier (RMID)
- Update Modification Identifier (UMID)

SMP/E uses modification identifiers to identify elements and their various modifications and updates.
Common MCSes: SYSMOD type

The first MCS is used to specify the type of the SYSMOD:

++USERMOD
Common MCSes: VER

+++VER is a required statement which describes the environment necessary for installing the SYSMOD:

+++VER(system-and-release-ID)
  PRE(prerequisite-PTFs)
  REQ(related user mods)
  SUP(user-mods incorporated into this one)
Common MCSes: JCLIN

++JCLIN

/* JCL used to link-edit the affected load modules */

."
Common MCSes: ZAP and MOD

+++MOD(MODULEB)  /* Name of module */
    DISTLIB(AOS12)  /* ddname of DLIB */

... Object module for MODULEB
...

MOD is followed by the object deck whereas ZAP is followed by a set of superzap control statements.
Where SMP/E Keeps Data

- Distribution libraries – all the elements used as input for running your system (important for backup)
- Target libraries – executable code needed to run the system
- Consolidated Software Inventory (CSI) – information about the structure of the z/OS system
SMP/E Libraries
The Consolidated Software Inventory

The CSI contains entries for the elements in the distribution and target libraries, grouped into zones:

- Global zone
- Target zone(s)
- Distribution zone(s)
SMP/E Commands for Zones

- **ZONECOPY** – create a copy of a zone (for example, to create a zone for testing)

- **ZONEEXPORT** – serialize a zone to a sequential data set (for example, for backups)

- **ZONEDELETE** – delete a zone

- **ZONEMERGE** – merge one zone into another

- **ZONERENAME** – rename a zone
SMP/E zones and libraries

CSI Global Zone

CSI Target Zone

CSI Distribution Zone

TLIBs

DLIBs
SMP/E Commands

**RECEIVE**  Place a SYSMOD in to the SMP/E library and create CSI entries for it.

**APPLY**  *Select* received SYSMODs for installation in the target libraries (TLIBs); SMP/E ensures that prerequisites are also installed in the proper sequence using the correct functional levels.

**ACCEPT**  Take *selected* SYSMODs and install them into the associated distribution libraries.
SMP/E Commands
Examples: RECEIVE

SET BDY(GLOBAL).  // work in global zone
RECEIVE.          // receive everything
RECEIVE HOLDDATA. // ... only hold data
RECEIVE SYSMODS.  // ... only SYSMODS
RECEIVE FORFMID(HOP0001). // ... specific product
Example: RECEIVE in JCL (1/2)

//jobname JOB ...
//RECEIVE EXEC PGM=GIMSMP
//SMPCSI DD DSN=SMPE.GLOBAL.CSI,DISP=SHR
//SMPOUT DD SYSOUT=* 
//SMPRPT DD SYSOUT=* 
//SMPPRINT DD SYSOUT=* 
//SMPCNTL DD *

SET BOUNDRY(GLOBAL).
RECEIVE SYSMODS HOLDDATA
  ORDER(ORDERSERVER(ORDRSRVR))
    CONTENT(PTFS(UQ12345,UQ98765))
  FORTGZONES(ZOS14)).
Example: RECEIVE in JCL (2/2)

//ORDRSRVR DD *
<ORDERSERVER
  keyring="MRWKRNG"
  certificate="SMPE Client Certificate">
</ORDERSERVER>
/*

In addition to the specified PTFs “UQ12345” and “UQ98765” all requisites that are not already present will also be received.
Examples: APPLY

SET BDY(ZOSTGT1).  // specify target zone
APPLY PTFS.        // apply all PTF SYSMODs
APPLY SELECT(UQ12345).  // ... specified PTF
APPLY APARS USERMODS. // ... all APARs & USERMODs
APPLY PTFS FORFMTID(HOP0001) // ... PTFS for product HOP0001
APPLY ... CHECK.    // simulate only
Examples: RESTORE

SET BDY(ZOSTGT1).  // specify target zone
RESTORE SELECT(UZ001).  // remove PTF UZ001
RESTORE SELECT(UZ001) GROUP.  // ... PTF UZ001 and deps
Examples: ACCEPT

SET BDY(ZOSTGT1). // specify target zone
ACCEPT PTFS.    // accept all PTFs
ACCEPT SELECT(UZ001). // ... specified PTF
Questions