A Basic Cloning Methodology for z/VM Systems – An Update

Jim Moling
US Treasury, Bureau of the Public Debt

Thursday, June 28, 2012
VM Workshop

Disclaimers

- **The opinions & ideas expressed herein are those of the author alone and do not necessarily reflect those of Bureau of the Public Debt, furthermore, Bureau of the Public Debt is hereby absolved of any and all responsibility or liability for the information contained herein.**
- **Copyrights & Trademarks:**
  - Any and all copyrights & trademarks are hereby acknowledged to be owned by their respective parties
  - All other brands, logos and products are trademarks or registered trademarks of their respective companies
  - All rights reserved
- **Disclaimer of Endorsement:**

  Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government, and shall not be used for advertising or product endorsement purposes.
- **Disclaimer of Liability:**

  With respect to this presentation, neither the United States Government nor any of their employees, makes any warranty, express or implied, including the warranties of merchantability and fitness for a particular purpose, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

Overview

- ## Why, What & How of Cloning
  - Why cloning?
  - What are we cloning?
  - How are we cloning?

- ## Demo: Cloning Active & Static z/VM Systems

- ## Summary: What's Next…

The Why, What & How of Cloning

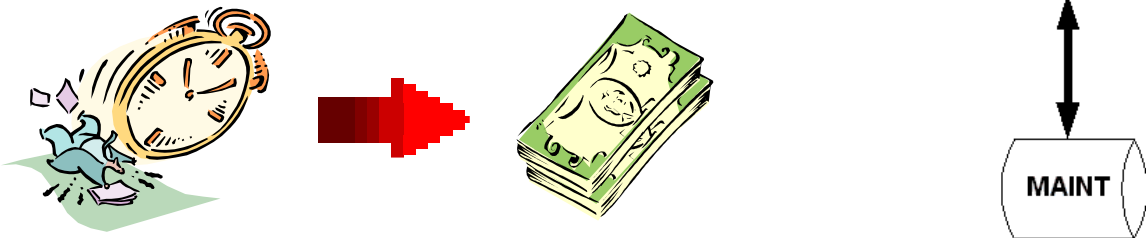- **Why Cloning?**

  - ❑ Standardized configurations

  - ❑ Facilitates maintenance testing & rollout

  - ❑ Time savings

  - ❑ Cost savings

The Why, What & How of Cloning

- ## **What Are We Cloning?**
  - Operating Systems That Run On System z
    - z/OS: Large enterprise class systems
    - z/Linux: Linux based server systems
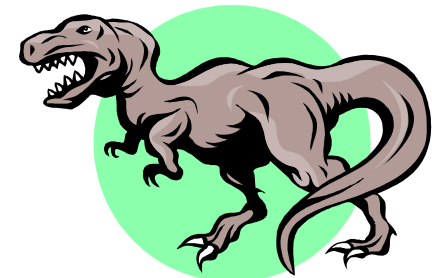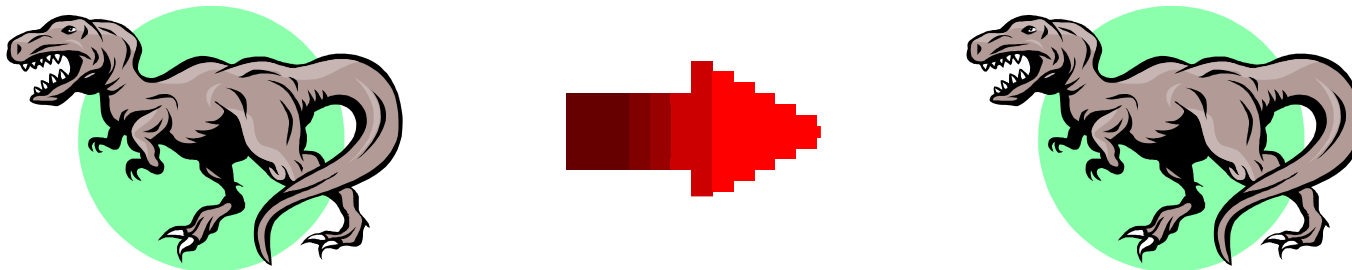   - z/VM: A Hypervisor for operating systems

z/Linux

z/VM

z/OS

The Why, What & How of Cloning

- **Cloning z/OS**
  - z/OS was the first operating system that we cloned
  - We now have multiple z/OS systems across 2 data centers
  - Takes the longest to clone: ~ 2.0-2.5 Hours
 - Cloned by running ~50 batch jobs in a z/OS environment

The Why, What & How of Cloning

- **Cloning Virtual Linux Servers Was Next**
  - We have over 100 Virtual Servers (and growing)
  - Virtual Servers are cloned in seconds
    - 15-20 minutes if Flash Copy DASD feature not available
  - Initially, cloning was via a "Controller" Linux Server
    - Methodology based on IBM Virtualization Cookbook
  - Now cloning using our own modified methodology
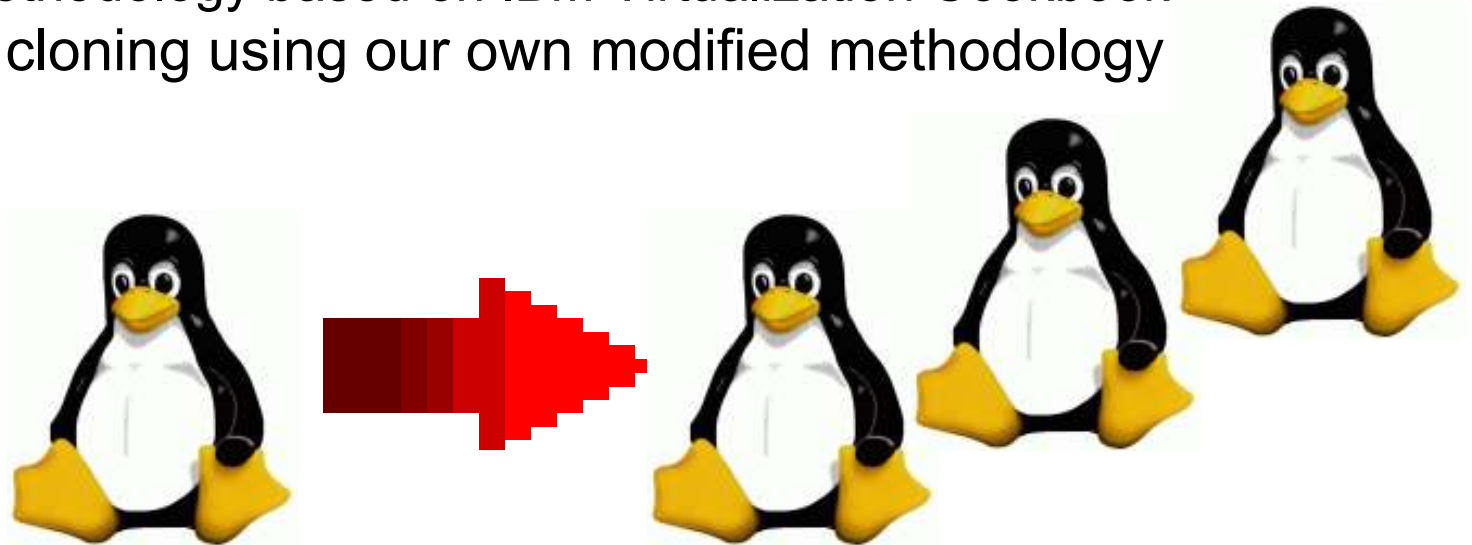
The Why, What & How of Cloning

- Finally, We're Cloning z/VM Systems
  - Looked for a VM cloning process, didn't find it, so created it
  - We now have multiple z/VM systems across 2 data centers
  - Takes about 2-3 minutes to clone z/VM
    - But takes 20-30 minutes if Flash Copy is not available
  - Cloning is performed via a "Cloning" Virtual Machine
  - If all goes well, you'll see a demo of this process …
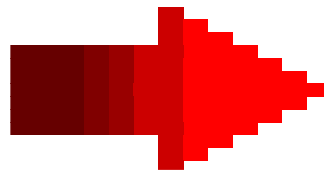
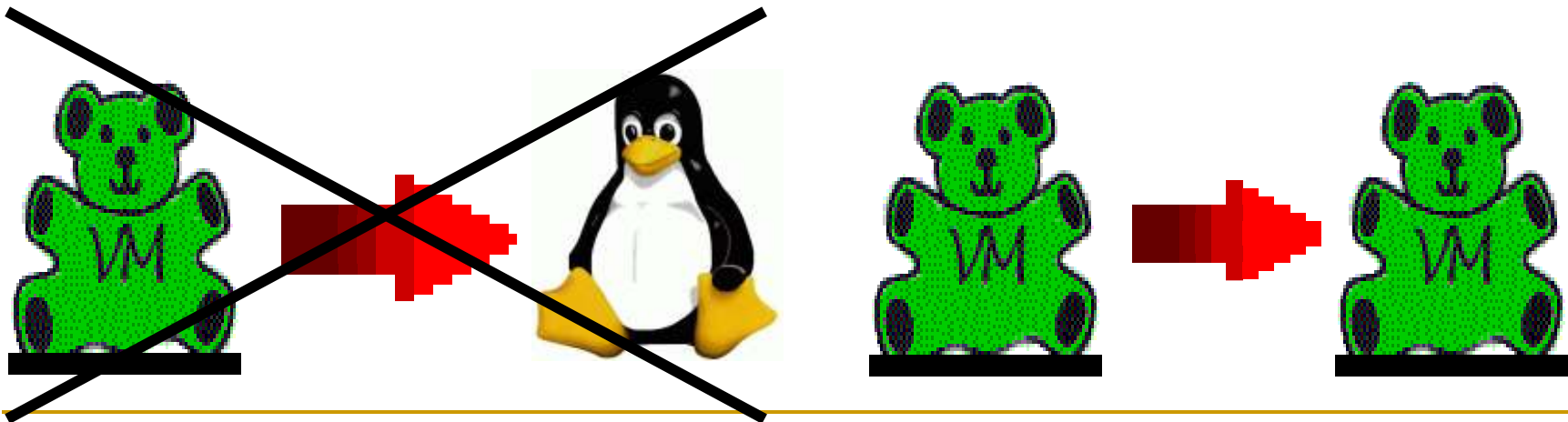# The Why, What & How of Cloning

- **How Is Cloning Accomplished?**
  - Establish standards that support duplication
  - Establish a "Base" system as the prototype
  - Develop an automated cloning procedure
  - Cloning process based on available utilities, i.e. Rexx, Flashcopy, DDR, etc.
  - Clone a system from a like system (my rule)

How We Are Cloning z/VM

- **Our z/VM Cloning Criteria**
  - Cloning process initially developed using z/VM 5.2.0
  - Running in a shared DASD environment with z/OS
    - So each z/VM system must have unique volsers
  - Based on the Shipped Configuration from IBM
    - Using a single, 3390-9 volume for SYSRES
    - Not Using Dirmaint, Shared File System, External Security, …
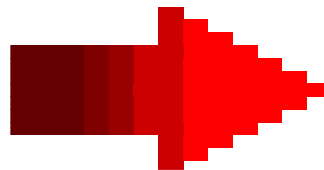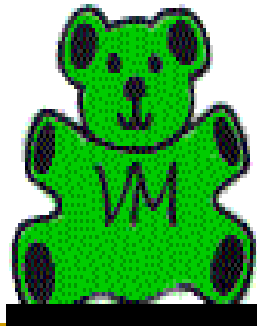
How We Are Cloning z/VM

- Cloning process is based on shipped configuration + User volume
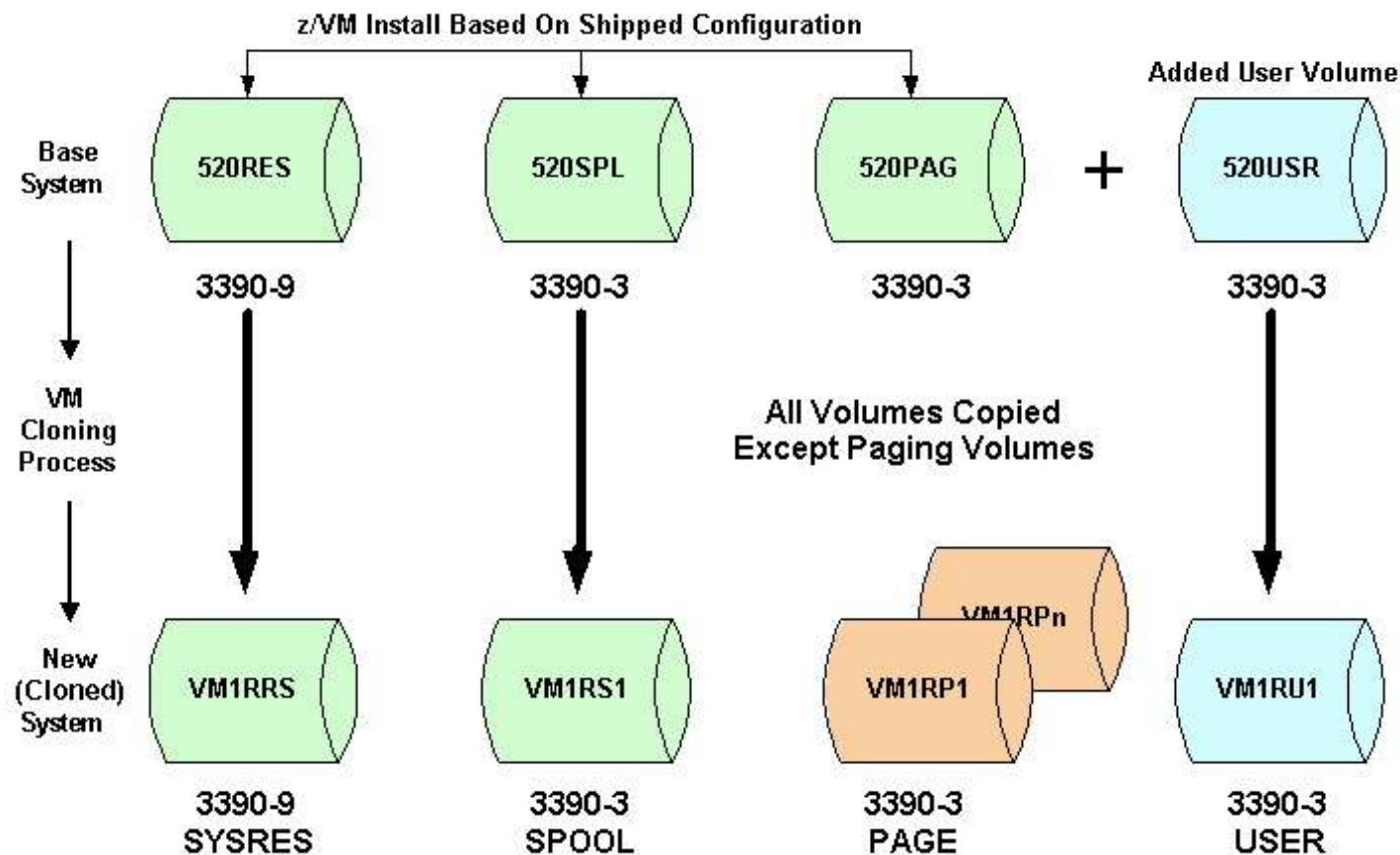
How We Are Cloning z/VM

- Two Files Must Be Modified In A Cloned System
  - SYSTEM CONFIG (Located on Parm Disk, i.e. CF1)
  - USER DIRECT (Shipped on MAINT's 2CC minidisk)
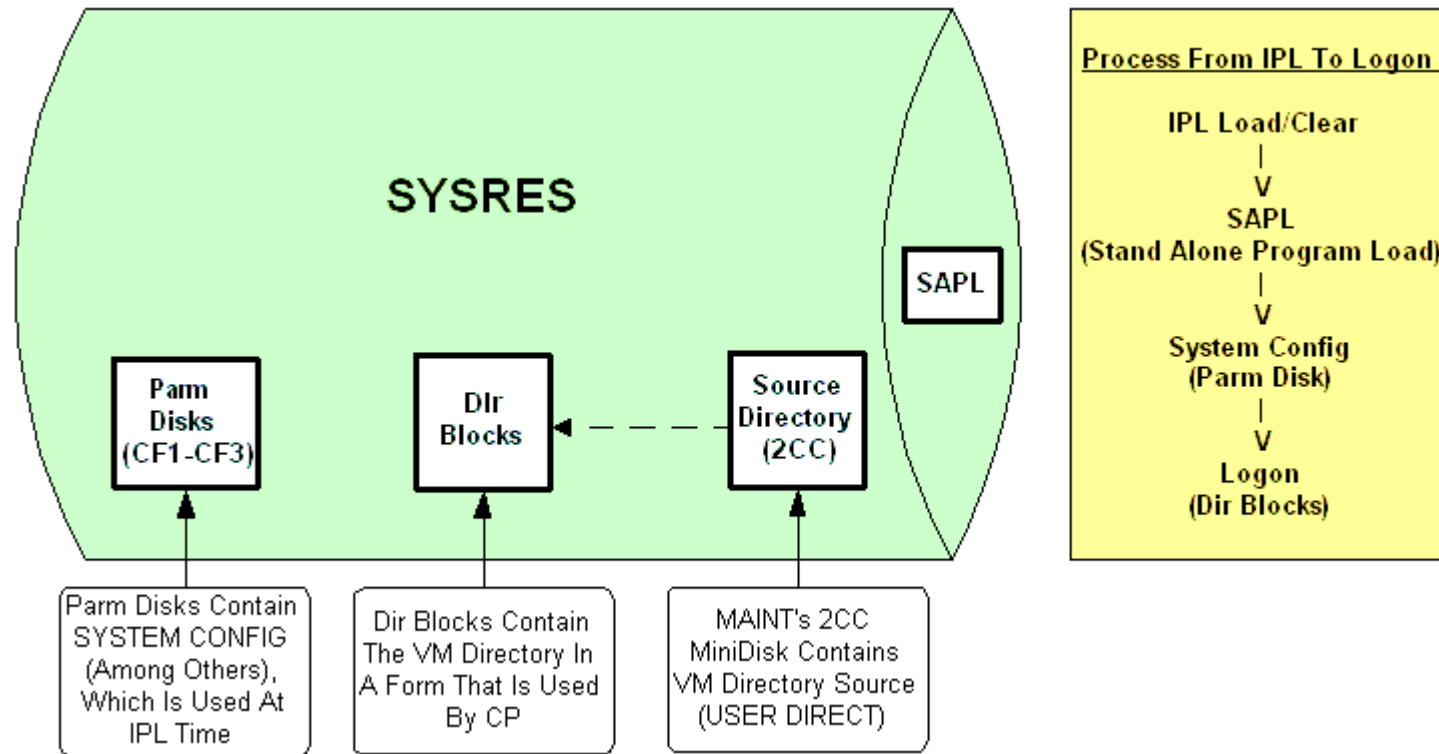
How We Are Cloning z/VM

- ## Conventions/Standards For Cloning
  - ❑ Established a naming standard for System ID (SYSID)
  - ❑ Established a naming standard for system volumes
  - ❑ Added a User volume to the configuration
  - ❑ Created a minimal System Config "model file" for cloning
  - ❑ Created a minimal VM Directory "model file" for cloning

New (Cloned) System

VM1RRS — 3390-9 SYSRES

VM1RS1 — 3390-3 SPOOL

VM1RP1 — 3390-3 PAGE

VM1RU1 — 3390-3 USER
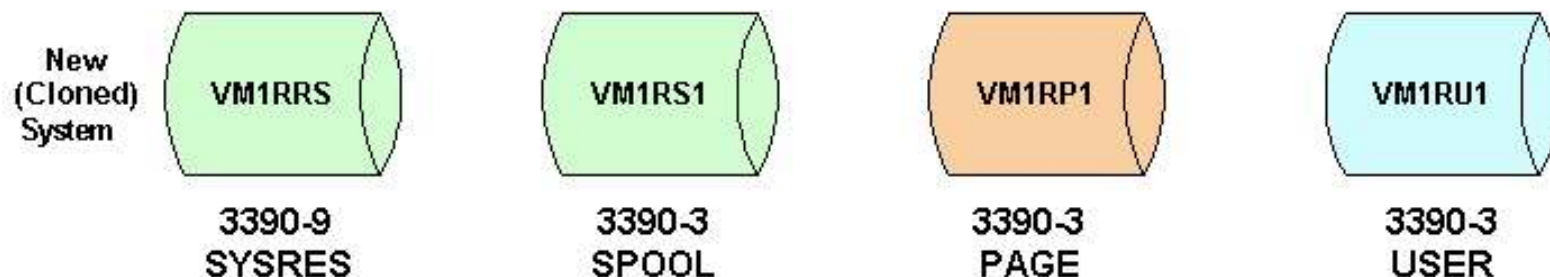
How We Are Cloning z/VM

- ## Naming Standard – SYSID
  - Using a 4-char SYSID to drive the cloning process
    - All system files can co-exist since names are unique (based on SYSID), i.e. System Configs, VM Directories, TCP/IP files, etc.
    - Volsers are also based on SYSID to make them unique
  - SYSID format: **VMnx**, where:
    - **VM** = Common prefix so VM systems & volumes are unique (from z/OS & z/Linux in our case)
    - **n** = a number that represents the data center
    - **x** = a letter to designate the VM system

SYSID=VM1R

New (Cloned) System

VM1RRS	VM1RS1	VM1RP1	VM1RU1
3390-9 SYSRES	3390-3 SPOOL	3390-3 PAGE	3390-3 USER
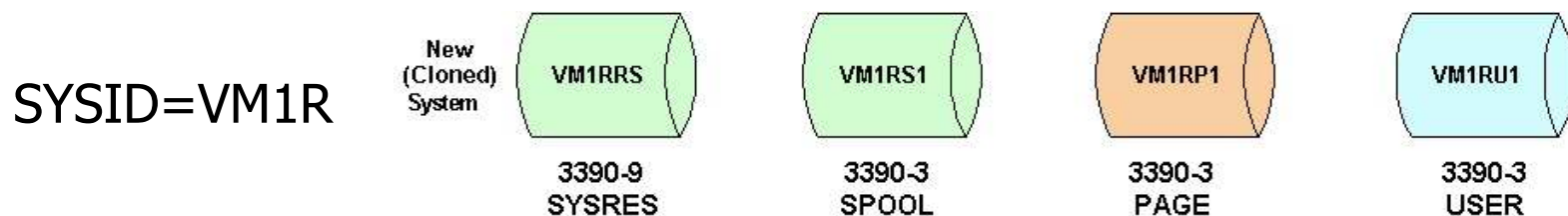
How We Are Cloning z/VM

- **Naming Standard – System Volsers**
  - Volser format: VMnx + V# = **VMnxV#**
    - First 4 characters are the SYSID, i.e. **VMnx** (such as VM1R)
    - 5th character indicates volume type: **V** = R, P, S, U
    - 6th character is a number to keep same volume types unique: **#** = 1-9 (except SYSRES is always **S**, i.e. **RS**)
  - *Note that current cloning process does not include TDISK volumes – we just haven't needed them*
    - TDISK volume type would be **T**, i.e. VM1R**T**1
    - Cloning can be modified to include TDISK's (if you modify execs)

	V#=RS	V#=S1	V#=P1	V#=U1
SYSID=VM1R	New (Cloned) System			
	VM1RRS	VM1RS1	VM1RP1	VM1RU1
	3390-9 SYSRES	3390-3 SPOOL	3390-3 PAGE	3390-3 USER

How We Are Cloning z/VM

Added User Volume

520USR

3390-3
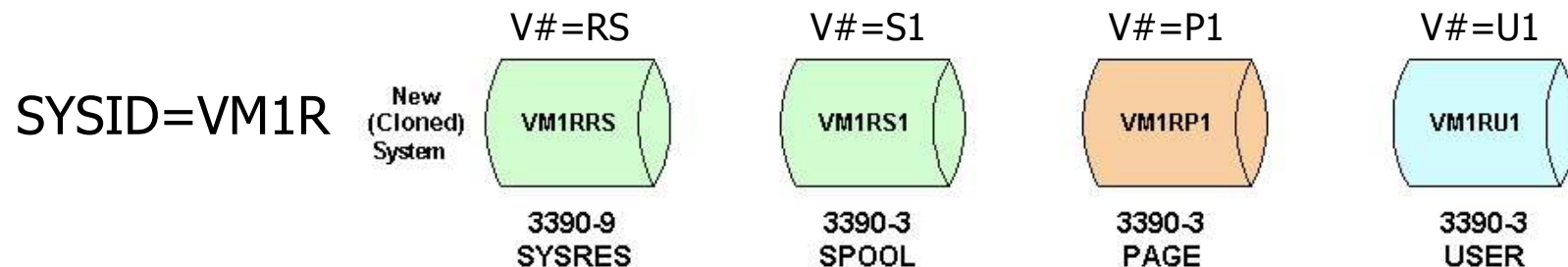
- **Why Add A User Volume?**
  - Simplifies moving to a new version of z/VM
  - Simplifies maintenance via full SYSRES replacement
  - Easy to clone – just 1 more DASD volume to copy
  - Conventions:
    - Always allocate new minidisks on the User volume
      - This means no more allocations on the SYSRES volume!
    - Using low-end cylinders for VM users (we only have a few)
    - Using high-end cylinders for VM tools, utilities & ISV software

Base System | 520RES | 520SPL | 520PAG | + | 520USR

3390-9   3390-3   3390-3   3390-3
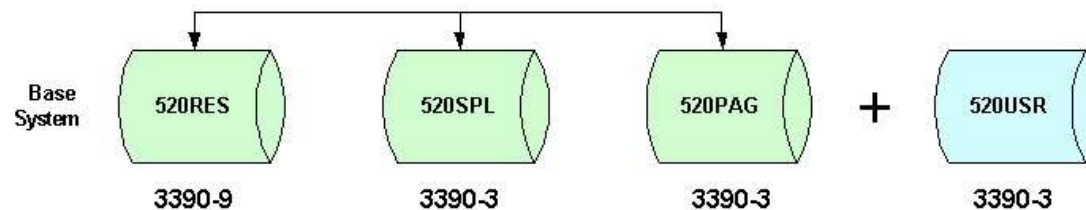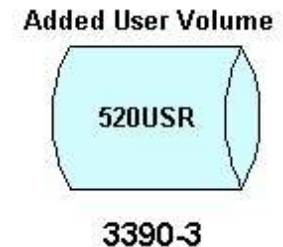
How We Are Cloning z/VM

- **Cloning Models Give Consistent Results**
  - Cloning will only change volsers for a new system
  - Cloning models should have basic definitions – you decide what they should be
    - System Config Model: fn=<Source Sysid> ft=$CONFIG$
    - User Directory Model: fn=<Source Sysid> ft=$DIRECT$
  - You can then modify the new files after cloning
    - New System Config: fn=<New Sysid> ft=CONFIG
    - New User Directory: fn=<New Sysid> ft=DIRECT


```
Parm          Dir        Source
Disks         Blocks     Directory
(CF1-CF3)                (2CC)
```

Note: fn means 'file name' & ft means 'file type'

How We Are Cloning z/VM

- **Preparation For Cloning**
  - Setup a cloning virtual machine called: VMCLONER
    - Make sure it has the same authorities as MAINT, i.e.
      - Dir Option: DEVMAINT (Allows SET MDISK to work)
      - Dir Option: LNKNOPAS (Allows LINK w/o password)
      - Other Dir Options: MAINTCCW LNKE LNKS
  - Wrote Rexx execs to perform the cloning & IPL processes
    - Rexx execs only use utilities shipped with z/VM
      - DDR, CPFMTXA, Flashcopy, etc.
    - 3 Rexx execs accomplish Cloning & IPL
      - CLONEVM   – Performs the cloning process
      - IPLVM   – Performs an IPL of the cloned system
      - TESTIPL   – IPLVM front-end with instructions for 1st IPL

How We Are Cloning z/VM

- Two Cloning Modes To Choose From
  - Active Mode – The original cloning method
    - Cloning source (volumes) based on VM system you logged onto
    - The original cloning process was based on active mode only
    - Use this method after installing the z/VM starter system
    - Warning: *This mode does not support cloning an <u>active</u> system that you are <u>not</u> logged onto!*
  - Static Mode – Cloning enhancement
    - Cloning source is from static, non-active VM system volumes
    - This method allows you to establish a "golden image" for cloning:
      - *Install the starter system and clone it using Active Mode*
      - *IPL & customize the cloned system, i.e. add SFS, External Security, DirMaint, etc.*
      - *After customizing, shut it down, then use it as a golden image via static cloning from another VM system, i.e. the starter system*

How We Are Cloning z/VM

- **Choice Of Naming Standards**
  - ❑ Using our Naming Standards is optional
  - ❑ Depends on specified parameters whether or not our SYSID and Volser naming standards are enforced
  - ❑ If you do use your own standards, make sure they're correct!
  - ❑ See the comments in the default parameter file for details
    - Default parameter file: CLONEVM $PARMS$ A1

How We Are Cloning z/VM

- **Overview Of The Cloning Process**
  - Execute the CLONEVM exec: CLONEVM *sysid*
    - The cloning process consists of 11 steps
    - There are many prompts during it's execution, so that it can be easily aborted if something doesn't look right
    - Execution results are written to a log file: sysid LOG A1
  - Execute the TESTIPL exec: TESTIPL *sysid*
    - Front-end to the IPLVM exec
    - Calls the IPLVM exec after displaying 1st IPL instructions
    - IPLVM execution results are written to: sysid IPL A1
  - Hopefully you'll see how this works during the demo …

How We Are Cloning z/VM

- **Recent enhancements to the VMCLONE exec:**
- Changed: 2012-01-24 - Enhanced To Ask If Generic Or Standard Naming
- Conventions Should Be Used When Creating A New
- Parameter File.
-
- Changed: 2012-01-27 - Static Mode Is Enhanced By Using The DIRENT
- Utility (From IBM VM Download Website) In
- Order To Access The VM Directory And System
- Config Files Directly From The SYSRES Source
- Volume (Rather Than Making Copies Of Them On
- The Running System).
-
- Changed: 2012-02-22 - Enhanced so that the VM Directory and System
- Config files are copied to the minidisks on
- the new SYSRES volume after cloning even when
- cloning mode is Active, rather than copying to
- the local minidisks before cloning, so that
- only the cloned system is updated - this way
- the local system is not affected.
-
- Changed: 2012-02-28 - Enhanced to check the Syntax of the new System
- Config file with the CPSYNTAX utility.

How We Are Cloning z/VM

- Cloning Process …
  - Determine the SYSID for a new VM system, i.e. VM1R
  - Logon to the VMCLONER virtual machine
  - Execute the CLONEVM exec: **CLONEVM** vm1r
  - It looks for a parameter file, i.e. VM1R $PARMS$ A1
  - If exists, asks if you want to make changes (via XEDIT)
  - If it doesn't exist, it will create a new one and enter XEDIT
  - Edit the parameter file to specify (or change if not 1st time):
    - The SYSID of the new system: VM1R
    - The DASD addresses for the new system
    - The Source SYSID that you're cloning from
    - The Source Volser Prefix you're cloning from
    - The Source Model files for the System Config & User Directory
    - Finally, save the parameter file with these changes, i.e. FILE command

How We Are Cloning z/VM

- Enhanced to ask which type of parameters to use:

```
Which Type Of Parameters Do You Want To Use, Standard Or Generic?

(1) Standardized Volser Parameters (SYSRES=xxxRES, SPOOL=xxxSP1, PAGE=xxxPG1, etc.)

(2) Generic Volser Parameters (SYSRES=xxxxxx, SPOOL=xxxxxx, PAGE=xxxxxx, etc.)


==> Please Select The Type Of Parameter File To Create, Either 1 Or 2 (Enter=1):
```

How We Are Cloning z/VM

- Editing the parameter file:

```
 VM1R      $PARMS$  A1  F 80  Trunc=80 Size=161 Line=0 Col=1 Alt=0
====>
00000 * * * Top of File * * *
00001 * ================================================================
00002 * ============> Parameter File For Cloning A New VM System <============
00003 * ================================================================
00004
00005 *   < Specify Source & Target Information For Cloning A New System >
00006
00007 * ----------------------------------------------------------------
00008 * New (Target) System Information
00009 * ----------------------------------------------------------------
00010 * Specify The New (Target) System & DASD Volumes That Will Be Cloned To:
00011
00012   NewID   = VM1R
00013
00014   NewResA = BA65
00015   NewResV = VM1RRS
          .
          .
```

How We Are Cloning z/VM

- Cloning Process …
  - 11 steps are executed by the CLONEVM exec:
    1. GetParms   – Reads/Edits cloning parameters from a file
    2. DefVols   – Defines volsers & verifies addresses
    3. ClipVols   – Clips the volsers of all target DASD
    4. LinkVols   – Links to all source & target volumes
    5. CloneDir   – Clones the VM Directory source file
    6. CloneCfg   – Clones the System Config file
    7. CloneVol   – Clones the new system volumes
    8. WriteDir   – Writes VM Directory to new SYSRES
    9. PageVols   – Formats the new PAGE volume(s)
    10. Transfer   – Transfers files if cloning a static system
    11. FirstIPL   – First IPL of cloned system to clear spool

How We Are Cloning z/VM

- Cloning Process …
  - Once CLONEVM has completed, then run TESTIPL
    - The TESTIPL Rexx exec provides instructions on performing the first IPL of a newly cloned system, then calls the IPLVM Rexx exec
      - *Specify the new SYSID as a parameter to TESTIPL*
        - *For example: **TESTIPL vm1r***
    - The IPLVM Rexx exec allows you to easily run another VM system "2nd level"
      - *The IPLVM exec defines terminals so you can dial into VMCLONER locally to check out the new system*
      - *TCP/IP access is also possible, but you'll have to set it up to work in your environment! (see IPLVM exec comments)*

How We Are Cloning z/VM

- Instructions displayed by the TESTIPL exec

```
TESTIPL: About To Perform An IPL Of VM System: VM1R ...
----------------------------------------------------------------
> If This Is The First IPL Of VM1R, Then Follow These Guidelines:

1) When The SAPL Screen Is Displayed, Specify: PROMPT FN=VM1R

2) Press PF10 To Proceed With The IPL Using These Options

3) When Prompted For Start Options, Specify: COLD DRAIN

4) When The TOD Prompt Is Given, Specify: NO

5) There Will Be Several Prompts For Spool Files, Reply: GO
----------------------------------------------------------------
==> About To Begin The IPL, Continue (Y/N) ? (Enter=Y):
```

Where Do We Go From Here…

- Lots of room for improvement in this cloning process, but it's a start!

- Some of the improvements that are in the works:
  - Working on panel displays for entering cloning parameters
  - Adapt to work with Shared File System (SFS)
  - Adapt to work with directory maintenance software, i.e Dirmaint, etc
  - Adapt to work with external security software, i.e. RACF, etc

- If you're interested in a copy of the current execs (and this presentation) you can get them from the VM Workshop website!

Questions?

- Any Questions?

- If not, then happy cloning!