

Adjunct Virtual Machines

Version 2

Lauren Maietti

z/VM Development Lab Software Engineer

lmaiett@us.ibm.com

Agenda

- Introduction
- Key Concepts
 - VMDBKs
 - Virtual Configuration
- Adjunct Setup
 - Adjunct Externals
- ADJUNCT Commands and the Virtual Configuration
- Demonstration
- References
- Summary

Introduction

What is an Adjunct Virtual Machine?

- A Virtual Machine that is defined in a USER (called the principal) to help debug a live system
- Has its own virtual storage (memory), virtual CPU (vCPU), virtual devices
- Can IPL a different instance of an operating system
 - However, it's intended to run CMS
- When the adjunct is active, the principal is suspended and vice versa

Key Concepts

The Virtual Definition Block

- AKA VMDBK
- Define and control most activities related to a single virtual machine
- A VMDBK is created for each z/VM guest/virtual CPU (vCPU)
- Used by the dispatcher to run said guest/vCPU
- Virtual machine settings defined in the VMDBK are changed through
 - Directory statements
 - Configuration statements
 - Commands
- All the VMDBKs pertaining to a guest define the virtual configuration

The Virtual Configuration – The Principal

- User VMTEST, principal of the configuration
- Gets its own user VMDBK defining the user and vCPU it runs on
- Is the originator of the configuration

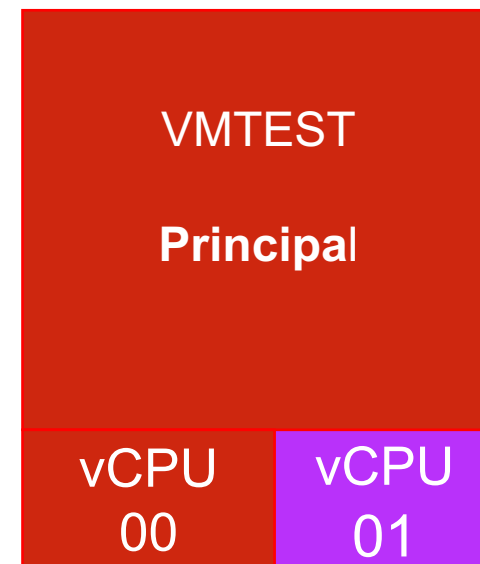


**Color signifies VMDBK grouping

The Virtual Configuration – Adding a vCPU

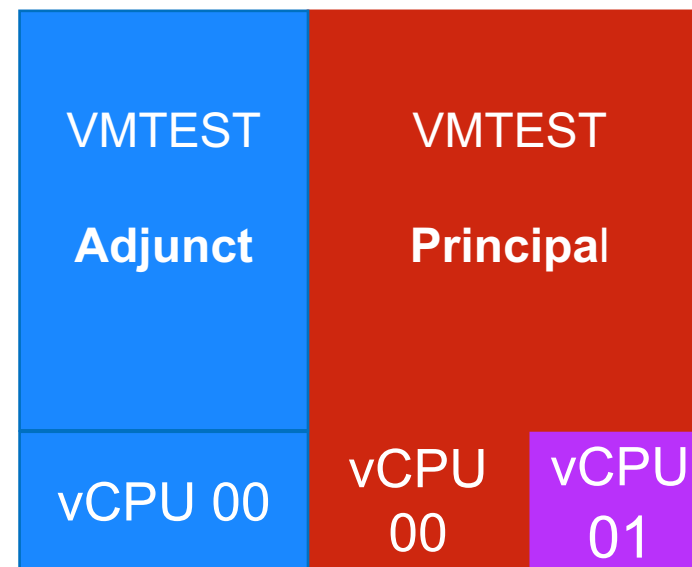
- When a new vCPU is defined it gets its own VMDBK
- vCPU VMDBK is slightly differently than a user's VMDBK

```
def cpu 1
08:03:38 25: CPU 01 defined
Ready; T=0.01/0.01 08:03:38
```



The Virtual Configuration – Adding an Adjunct

- An Adjunct Virtual Machine gets its own VMDBK once defined and started
- Much like the vCPU VMDBK, it is different from a User or vCPU VMDBK



Adjunct Setup

Adjunct Externals

- FEATURES ENABLE/DISABLE ADJUNCT
 - Configuration file statement to set the availability of the adjunct feature
- ADJUNCT directory statement
 - Include an adjunct template in a user definition
 - Allows that user to create and control the defined adjunct
- SET ADJUNCTS ON/OFF
 - Dynamical enable or disable the availability of the adjunct feature
- ADJUNCT command
 - ADJUNCT START – logon/ipl an adjunct and suspend the principal
 - ADJUNCT STOP/SUSPEND - put the adjunct in a soft stop and reactive the principal
 - ADJUNCT CP *command* – direct commands to the adjunct from the principal
 - ADJUNCT BEGIN/RESUME – reactivate the adjunct and suspend the principal
 - ADJUNCT END – logoff the adjunct and reactivate the principal
- QUERY ADJUNCT command

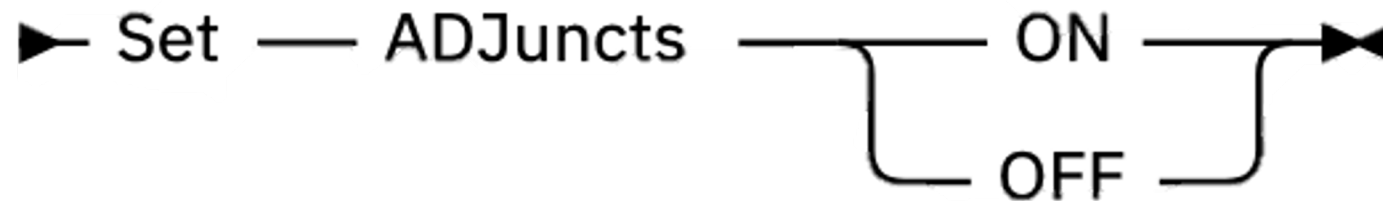
Setting up an Adjunct: An Overview

- An optional FEATURES statement in the system configuration file
 - Adjuncts are enabled by default and there is a dynamic SET ADJUNCTS command to enable/disable
 - To make a permanent change, such as keeping adjunct disabled, use a FEATURES statement
- Directory statements
 - User statement to define the adjunct template
 - ADJUNCT statement in the user definition of the user that will utilize the adjunct
- ADJUNCT Commands to control the adjunct
 - ADJUNCT START, ADJUNCT END, ADJUNCT BEGIN/RESUME, ADJUNCT STOP/SUSPEND, ADJUNCT CP *command*
- QUERY ADJUNCT
 - The query gives return codes which can be used in certain automation

Adjunct Features Statement

- The Adjunct FEATURE is enabled by default
- To disable adjunct and retain that setting
 - update the SYSTEM CONFIG file for the user
 - To avoid FEATURE statement syntax errors, add a *new* FEATURES statement
 - Don't forget to run CPSYNTAX!

SET ADJUNCTS Command



Privilege Class: A

- System Administrator can dynamically change the availability of the adjunct feature
- This setting will go away with a re-IPL
- To retain the availability setting, update the SYSTEM CONFIG with an adjunct FEATURES statement

Directory Updates: Adjunct Template

- First, define the adjunct *template* with a USER statement
- Given a common PROFILE for USER statements

PROFILE COMMON3

IPL CMS PARM AUTOOCR

CONSOLE 0009 3215

SPOOL 000C 2540 READER *

SPOOL 000D 2540 PUNCH A

SPOOL 000E 1403 A

LINK MAINT 190 190 RR

LINK MAINT 19D 19D RR

LINK MAINT 19E 19E RR

LINK MAINT 19F 19F RR

LINK MAINT 1A1 1A1 RR

LINK MAINT 120 120 RR LINK MAINT 19A 19A RR

Directory Updates: Adjunct Template

- The adjunct template can be defined as follows

```
USER MRADJUNC NOLOG 32M 1G G
```

```
    INCLUDE COMMON3
```

```
    LINK * 191 191 MR
```

- A few things to note:
 - The password is NOLOG, this is **required** so the adjunct cannot be logged into
 - The asterisk in the LINK statement does not refer to the MRADJUNC USER
 - MRADJUNC is more of an identifier than a userID

Directory Update: Include Adjunct Template in Principal User

▶▶—ADJUNCT—templateuser————▶◀

- This statement is included in the user that is intended to use the adjunct
- Using the MRADJUNC USER previously defined

```
USER OPERATOR SLIMEY 256M 1G
```

```
    INCLUDE COMMON3
```

```
    ADJUNCT MRADJUNC
```

```
    MDISK 0191 3390 2463 0005 X1USR7 MRD RREAD WWRIT MMULT
```

```
    MDISK 0192 3390 2650 005 X2USR8 MR RREAD WWRIT MMULT
```

```
    MDISK 0340 3390 2655 150 X2USR8 MR RREAD WWRIT MMULT
```

```
    MDISK 0399 3390 2805 030 X2USR8 MR RREAD WWRIT MMULT
```

- A few things to note:
 - The ADJUNCT MRADJUNC statement makes the previously defined adjunct available to this user ID
 - There can only be one ADJUNCT template included per principal user definition
 - The MDISK 191 statement includes modes M R and D. The D stands for deferred link.
 - Once all updates are in, don't forget to run DIRECTXA on the updated directory file!

Controlling the Adjunct – ADJUNCT Command

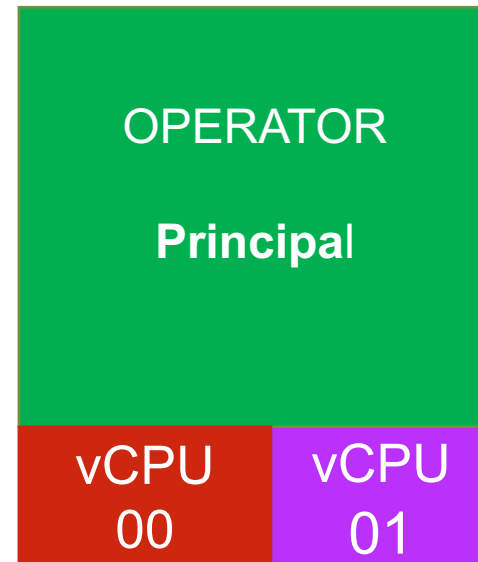


Privilege Class: G

- This command is used to control the adjunct defined for the principal

Adjunct Commands and the Virtual Configuration

Virtual Configuration: Before an Adjunct



*Green signifies the virtual machine that is active

*Grey signifies the virtual machine that is inactive

ADJUNCT START

- Creates and IPLs an adjunct, analogous to LOGON/IPL for a user
- However, it also suspends execution of the principal configuration (“soft stop”)
- This command is only successful when the ADJUNCT FEATURE is ENABLED and an adjunct template is configured for the issuing user

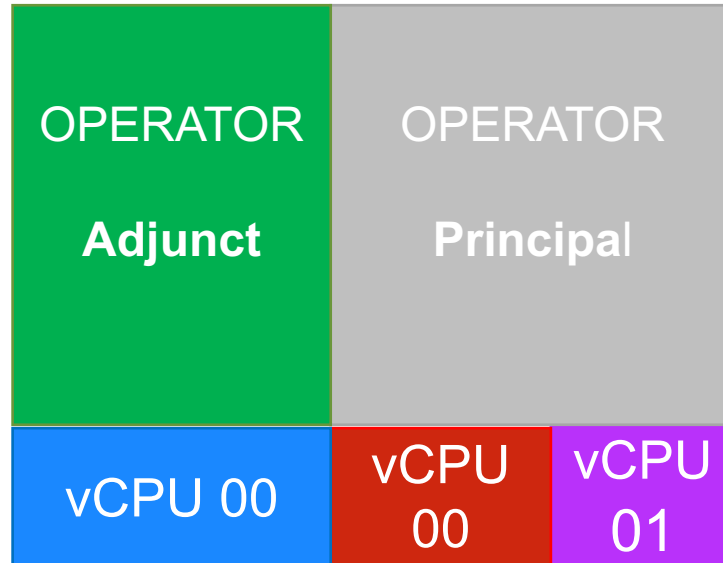
```
query adjunct
08:17:52 25: HCPYJT5768I OPERATOR Adjunct does not exist
Ready(05768); T=0.01/0.01 08:17:52
```

```
adjunct start
08:18:37 HCPLNM102E DASD 0191 forced R/O; R/W by OPERATOR
08:18:37 * MSG FROM OPERATOR: Hello World - Userid = MRADJUNC
```

```
query adjunct
08:19:25 HCPYJT5777I OPERATOR Adjunct is currently active
Ready(05777); T=0.01/0.01 08:19:25
```

```
adjunct start
08:54:28 HCPYJT5769E OPERATOR Adjunct already exists
Ready(05769); T=0.01/0.01 08:54:28
```

Virtual Configuration: After ADJUNCT START



ADJUNCT SUSPEND/STOP

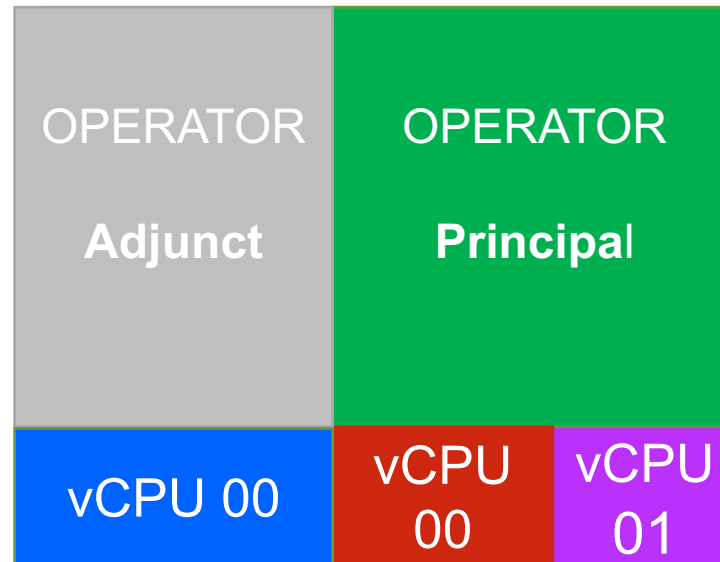
- Puts the Adjunct in a soft stop (with everything left as is)
- Reactivates the principal user (picks up where it was left off)

```
query adjunct
08:19:25 HCPYJT5777I OPERATOR Adjunct is currently active
Ready(05777); T=0.01/0.01 08:19:25
```

```
adjunct stop
Ready; T=0.01/0.01 08:43:38
```

```
query adjunct
08:42:34 25: HCPYJT5776I OPERATOR Adjunct is suspended
Ready(05776); T=0.01/0.01 08:42:34
```

Virtual Configuration After ADJUNCT STOP/SUSPEND



ADJUNCT CP *command*

- Allows principal user to direct CP commands at the ADJUNCT when suspended
- An adjunct must exist and be suspended for this command to succeed

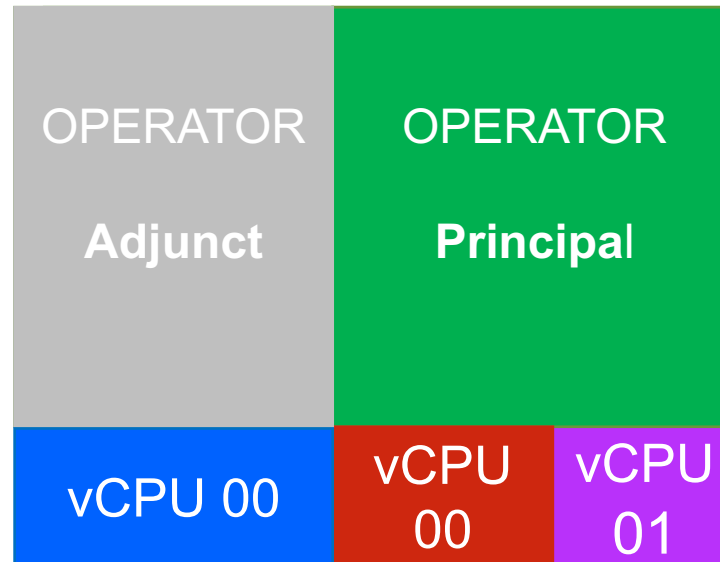
```
query adjunct
08:42:34 25: HCPYJT5776I OPERATOR Adjunct is suspended
Ready(05776); T=0.01/0.01 08:42:34
```

```
adjunct cp link * 191 191 mw
08:46:44 DASD 0191 LINKED R/W; R/W BY OPERATOR
Ready; T=0.01/0.01 08:46:44
```

```
query adjunct
08:42:34 25: HCPYJT5776I OPERATOR Adjunct is suspended
Ready(05776); T=0.01/0.01 08:42:34
```

```
q link 191
08:50:19 25: OPERATOR 0191 R/W, OPERATOR 0191 R/W ADJUNCT
Ready; T=0.01/0.01 08:50:19
```

Virtual Configuration after ADJUNCT CP *command*



ADJUNCT BEGIN/RESUME

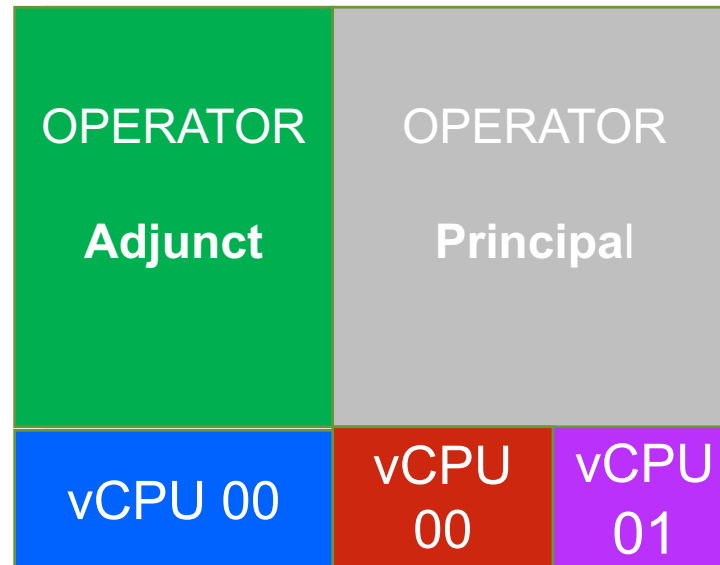
- Puts principal in a soft stop (keeps everything as it is)
- Reactivates the adjunct user (picks up where it was left off)

```
query adjunct
08:42:34 25: HCPYJT5776I OPERATOR Adjunct is suspended
Ready(05776); T=0.01/0.01 08:42:34
```

```
adjunct begin
Ready; T=0.01/0.01 08:52:14
```

```
query adjunct
08:52:57 HCPYJT5777I OPERATOR Adjunct is currently active
Ready(05777); T=0.01/0.01 08:52:57
```

Virtual Configuration after ADJUNCT BEGIN/RESUME



ADJUNCT END

- Logs off the Adjunct & reactivates the principal

```
query adjunct
08:52:57 HCPYJT5777I OPERATOR Adjunct is currently active
Ready(05777); T=0.01/0.01 08:52:57
```

```
adjunct end
Ready;
```

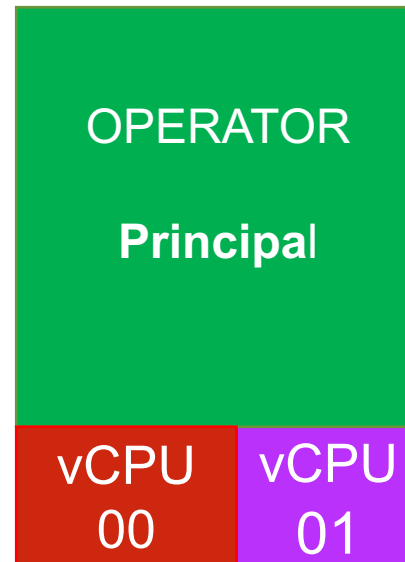
```
query adjunct
HCPYJT5768I MAIETTI Adjunct does not exist
Ready(05768);
```

```
adjunct begin
HCPYJT5768E MAIETTI Adjunct does not exist
Ready(05768);
```

```
adjunct stop
HCPYJT5768E MAIETTI Adjunct does not exist
Ready(05768);
```

```
adjunct end
08:57:17 25: HCPYJT5768E OPERATOR Adjunct does not exist
Ready(05768); T=0.01/0.01 08:57:17
```

Virtual Configuration after ADJUNCT END



Demonstration

References

- Primary Sources in z/VM Library - <https://www.vm.ibm.com/library/>
 - CP Planning and Administration
 - CP Command and Utility Reference

Summary

- Adjuncts run side by side with a principal
 - Only one virtual machine may be active at a time
 - Can be used to debug a live system
 - Can run anything but is meant to run CMS
-
- Have fun use cases for Adjunct? Email me:
lmaiett@us.ibm.com