Automating BB&T's Virtual Server Environment

Presented by Tracy Dean – tld1@us.ibm.com

With Permission from the Author, Richard Baek – rbaek@bbandt.com

April 2014

Previous Experience

- Information Systems graduate from New York University
- After college, started as a z/OS Systems Programmer with IBM in Sterling Forest, NY
- Worked in communications product development for IBM in RTP, NC
- After a few years in management, returned to z/OS Systems Programming
- Short stint working on CICS systems for the Social Security Administration
- Last four years as a z/OS and z/VM systems programmer at BB&T

A Brief History

- Founded in 1872 by Alpeus Branch and Thomas Jefferson Hadley in Wilson, NC
- Current headquarters in Winston-Salem, NC
- Major expansion outside the Carolinas started in the 1990's through strategic M&A
- Expanded into the Texas market with the acquisition of Colonial Bank in 2009
- Operates over 1,800 financial centers in 12 states and Washington, DC
- Covers the southeastern US from Maryland to Florida and westward to Texas

BB&T's Growth

- In the late 1990's and early 2000's, BB&T made key acquisitions that drove growth:
  - 1995: Winston-Salem based Southern National Corporation
    - At the time, the state's fifth largest bank-holding company
  - 1998: Maryland Federal Bancorp of Hyattsville, MD
  - 1998: Franklin Bancorporation of Washington, DC
  - 1999: MainStreet Financial Corp of Martinsville, VA
  - 2000: Premier Bancshares of Atlanta, GA
  - 2001: FCNB Corp of Frederick, MD
  - 2006: Main Street Banks Inc of Atlanta, GA
  - 2009: Coastal Financial Corp of Myrtle Beach, S.C
  - 2009: Colonial Bank of Montgomery, Ala

Environment and Application Overview

- z/OS – Production and development running on 2 zEC12's
  - 11 z/OS LPARs
  - Production CICS and DB2 applications
  - Branch banking and ATM's
  - 180 Million CICS transactions per day
- z/VM – Production and development running on 2 z196's
  - Started zLinux in production in July 2011
  - Financial, mortgage and fraud applications
  - Over 140 zLinux servers across 7 LPARs
  - 22 Active IFLs
  - Over 800 GB virtual storage/memory
- Distributed systems
  - Unix, AIX and Windows servers
  - Web and online banking applications

z/VM Automation Implementation

- Needed to automate and manage the z/VM environment in the same mature manner as the z/OS environment:

- Today:
  - z/VM log / console consolidation
  - Operator console message processing
  - Automatic job scheduling
  - Monitoring page space and spool utilization

- In Progress:
  - Controlling task startup and shutdown
  - Monitoring running tasks

What Operations Manager for z/VM Provides

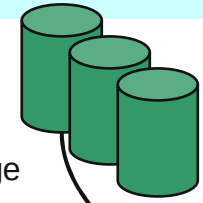
Increase productivity

- Authorized users to view and interact with monitored virtual machines without logging onto them
- Multiple users view/interact with a virtual machine simultaneously

Improve system availability

- Monitor virtual machines and processes
- Take automated actions based on console messages
- Reduce problems due to operator error

Schedule tasks



Monitor page and spool usage

Respond to system events (user state changes)

Console monitoring

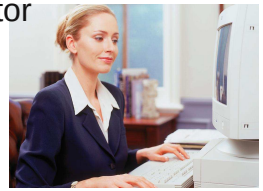
Service Virtual Machine being monitored

Idle monitor

Operations Manager for z/VM

Take action

Service Virtual Machine being monitored



- View & interact with consoles
- View spool files

Console monitoring

Automation

- Routine activities done more effectively with minimal operations staff
- Schedule tasks to occur on a regular basis

Integration

- Fulfill take action requests from performance monitoring products (e.g. OMEGAMON XE on z/VM and Linux)
- Send alerts to email, central event management systems (e.g. Netcool/OMNIbus), etc.

z/VM Log / Console Consolidation

z/VM service machines and Linux servers all have separate consoles.

Consolidation allows for centralized management and matches z/OS-style operational management.

- For all Linux on System z servers
  - OBSERVER set to Operations Manager (OPMGRM1)
- OPERATOR runs disconnected
  - SECUSER set to OPMGRM1
- z/VM service machines
  - OBSERVER or SECUSER set to OPMGRM1
- Created an Operations Manager “VIEW” to combine the console messages of all z/VM service machines
  - VIEW name is CONSOLE
- Operations staff logon to user OP1 which automatically views the console of user CONSOLE
  - Uses Operations Manager “VIEWCON USER(CONSOLE)” command
- Use BMC Mainview Console Management for zEnterprise to support multiple consoles

Monitor Service Machine Consoles

Operations Manager

syslog data

```
slog Message 1
slog Message 2
slog Message 3
…
```

OPERATOR

```
OPER Message 1
OPER Message 2
OPER Message 3
…
```

LINUX

```
LNX Message 1
LNX Message 2
LNX Message 3
…
```

TCPIP

```
TCP Message 1
TCP Message 2
TCP Message 3
…
```

Test Data

```
TEST Message 1
TEST Message 2
TEST Message 3
…
```

Data space 1

```
TEST Message 1
TEST Message 2
…
```

Data space 2

```
OPER Message 1
OPER Message 2
…
```

Data space 3

```
LNX Message 1
LNX Message 3
…
```

Data space 4

```
TCP Message 1
TCP Message 2
…
```

Data space 5

```
slog Message 1
slog Message 2
slog Message 3
…
```

Filtered

Data space 6

```
OPER Message 1
LNX Message 1
LNX Message 2
TCP Message 1
slog Message 1
OPM Message 1
TEST Message 1
OPER Message 2
…
```

Daily log

```
OPER Message 1
LNX Message 1
LNX Message 2
TCP Message 1
slog Message 1
OPM Message 1
```

Unfiltered

Example of Console VIEW (Group) Definitions

```
*
* Define a VIEW for VIEWCON command
*
DEFVIEW NAME(CONSOLE),USER(OPERATOR)
DEFVIEW NAME(CONSOLE),USER(DIRMAINT)
DEFVIEW NAME(CONSOLE),USER(TCPIP)
DEFVIEW NAME(CONSOLE),USER(SMTP)
DEFVIEW NAME(CONSOLE),USER(SNMPD)
DEFVIEW NAME(CONSOLE),USER(RACF)
*
* Define a VIEW for all zLinux servers
*
DEFVIEW NAME(ZLINUX),USER(LX*)
```

Example Of VIEWLOG Display

```
01/14/2014 11:54:18 GOMSMO0403I  SPOOL ALERT: MONITOR SPLCHECK USAGE    CONDITION
REACHED
01/14/2014 11:54:18 GOMSMO0401I  SPOOL USE: MONITOR SPLCHECK SPACE 17 PERCENT,
FILES 0 PERCENT
01/14/2014 11:54:18 GOMSMO0402I  SPOOL CHG: MONITOR SPLCHECK SPACE 0 PERCENT, FILES
0 PERCENT
01/14/2014 11:54:18 GOMACT0260I  SPOOL SPLCHECK ACTION SPLCHECK TRIGGERED BY
_GOMSMON
01/14/2014 11:54:18 GOMACT0262I  ACTION SPLCHECK BEGIN FOR _GOMSMON SERVER OPMGRS4
01/14/2014 11:54:18 GOMACT0269L  COMMAND "SPLCHECK STATUS 17"
01/14/2014 11:54:18 GOMPMO0453I  PAGE ALERT: MONITOR PAGCHECK USAGE    CONDITION
REACHED
01/14/2014 11:54:18 GOMPMO0451I  PAGE USE: MONITOR PAGCHECK SPACE 32 PERCENT
01/14/2014 11:54:18 GOMPMO0452I  PAGE CHG: MONITOR PAGCHECK SPACE 0 PERCENT
```

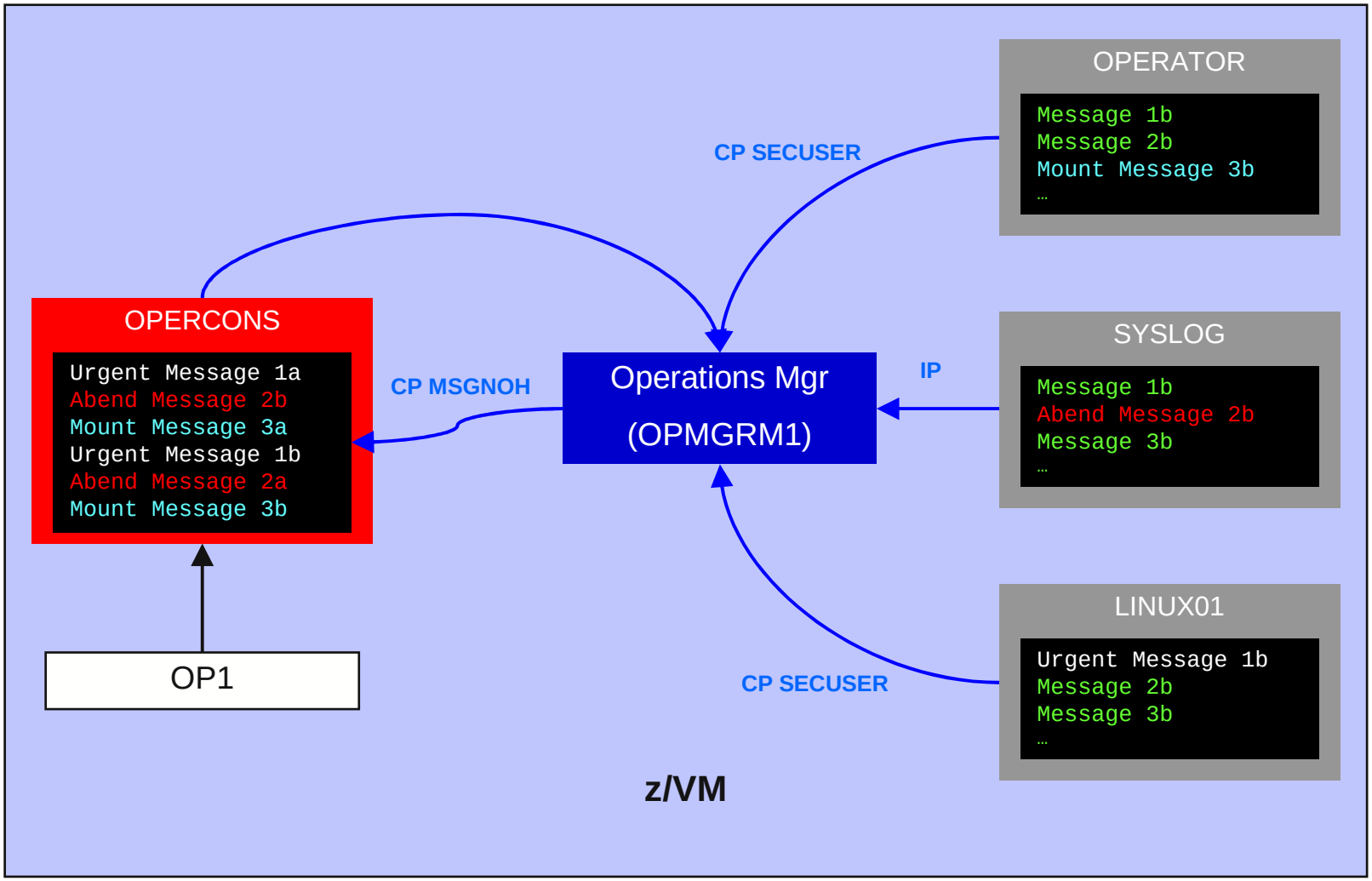
Example of VIEWCON Display

```
11:53:04 TCPIP    11:53:04 DTCSTM163I Telnet server: Conn 1: Connection opened
01/14/14 at 11:53:04

11:53:15 OPERATOR GRAF L0004 LOGON  AS  MAINT620 USERS = 75    BY S743RB   FROM
10.129.11.78

11:59:45 ZSERVE   ESAESP924I Switching from period PRIME to period PRIME on
01/14/14 at 11:59:45.
```

Console Message Processing

- Allows message coloring, highlighting and suppression
- Also allows for command/REXX actions when message is issued
- Primarily used for message suppression from the VIEWCON command
- Search for messages by message ID or any message text
- Messages still appear in VIEWLOG and remain in the Operations Manager log in memory and on disk
- Suppress regular informational messages not required by Operations
- Makes VIEWCON display easier to read and slower to autoscroll

Create a Centralized Operations Console for z/VM and Linux



Example Of Message Suppression

```
*-------------------------------------------------------
* Suppress messages from VIEWCON - still go to VIEWLOG
*-------------------------------------------------------
DEFRULE NAME(DVH2140I),MATCH(*DVHWAI2140I*),ACTION(SUPPRESS)
*
DEFRULE NAME(DVH2142I),MATCH(*DVHWAI2142I*),ACTION(SUPPRESS)
*
DEFRULE NAME(DVH2143I),MATCH(*DVHWAI2143I*),ACTION(SUPPRESS)
*
DEFRULE NAME(DVH2288I),MATCH(*DVHREQ2288I*),ACTION(SUPPRESS)
*
DEFACTN NAME(SUPPRESS),INPUT(SUP)
```

Schedule Events and Actions in Operations Manager

- Define schedules
  - Hourly, daily, weekly, monthly, or yearly, nth weekday of the month
  - Once on specified month, day, year, and time
  - At regular intervals
    - Every x hours and y minutes
  - Within a specified window of time
    - Specify start time
    - Specify conflicting schedules
    - Specify maximum time to defer this schedule
  - Within limits
    - Restrict to specific days of the week: Monday through Sunday plus holidays
    - Restrict to certain hours of the day
- Specify the action associated with the schedule
  - Actions specified are the same as those for console and spool monitoring

Automatic Job Scheduling

- Allows for the scheduled execution of commands, programs, or REXX EXECs
  - At set intervals (every 5 minutes, hourly, daily, weekly)
  - At set times and set dates (1:00 AM every Tuesday, 15th day of each month)
- Multiple “batch” servers for execution
- Usage at BB&T
  - Back up the Operations Manager log for audit purposes
  - Back up Velocity performance data for capacity planning
  - Monthly RACF SMF record archive / monthly system reporting
  - Regularly check system resources / minidisk utilization
  - Daily directory backups

Example Job/Command Scheduling

```
*-------------------------------------------------------
* Run minidisk check twice each day
*-------------------------------------------------------
DEFSCHD NAME(DISKCHK1),WHEN(08:00),ACTION(DISKCHK)
DEFSCHD NAME(DISKCHK2),WHEN(20:00),ACTION(DISKCHK)
DEFACTN NAME(DISKCHK),COMMAND(DISKCHK),OUTPUT(LOG),ENV(SVM)
*-------------------------------------------------------
* Schedule to send logs to z/OS daily at 00:01
*-------------------------------------------------------
DEFSCHD NAME(SENDLOG),WHEN(00:01),ACTION(SENDLOG)
DEFACTN NAME(SENDLOG),COMMAND(LOGGER),OUTPUT(LOG),ENV(SVM)
*-------------------------------------------------------
* Schedule to send RACF SMF to z/OS 2nd Monday of each month
*-------------------------------------------------------
DEFSCHD NAME(SENDSMF),WHEN(2NDMON-00:45),ACTION(SENDSMF)
DEFACTN NAME(SENDSMF),COMMAND(SMFPROC),OUTPUT(LOG),ENV(SVM)
```

System Page Space Monitoring

- Operations Manager allows you to set up regular monitoring of your page space utilization
- Different actions can be implemented depending upon the % of use
- Additionally, actions can be taken due to the rapid growth in paging usage
- Usage at BB&T
  - Regardless of utilization, issue utilization message to log every 30 minutes
  - Every 5 minutes, if utilization is between 40 and 59 percent issue warning (email)
    - Limit it to 2 emails per hour
  - Every 5 minutes, if utilization is between 60 and 100 percent, issue error (text message)
    - Limit to 4 texts per hour
  - Every 5 minutes, check for page usage growth of 10% or more
    - Limit to 2 per hour

Example of Page Utilization Statements

```
DEFPMON NAME(PAGCHECK),INTERVAL(30),USAGE(000-100),ACTION(PAGCHECK)
DEFACTN NAME(PAGCHECK),+
  COMMAND(MSGNOH OPMGRM1 CURRENT PAGING UTILIZATION IS &4 %),+
  ENV(SVM)
*
DEFPMON NAME(PAGWARN),INTERVAL(5),USAGE(040-059),+
  ACTION(PAGWARN),LIMIT(2,3600)
DEFACTN NAME(PAGWARN),COMMAND(PAGWARN &4),ENV(SVM)
*
DEFPMON NAME(PAGHIGH),INTERVAL(5),USAGE(060-100),+
  ACTION(PAGHIGH),LIMIT(4,3600)
DEFACTN NAME(PAGHIGH),COMMAND(PAGHIGH &4),ENV(SVM)
*
DEFPMON NAME(PAGGROW),INTERVAL(5),INCREASE(010-100),+
  ACTION(PAGGROW),LIMIT(2,3600)
DEFACTN NAME(PAGGROW),COMMAND(PAGGROW &5),ENV(SVM)
```

System Spool Space Monitoring

- Operations Manager allows you to set up regular monitoring of your spool utilization
- Different actions can be implemented depending upon the % of use
- Additionally, actions can be taken due to the rapid growth in spool usage
- Usage at BB&T
  - Regardless of utilization, issue utilization message to log every 30 minutes
  - Every 5 minutes, if utilization is between 70 and 84 percent issue warning (email)
    - Limit to 2 emails per hour
  - Every 5 minutes, if utilization is between 85 and 100 percent, issue error (text message)
    - Limit to 4 texts per hour
  - Every 5 minutes, check for spool usage growth of 10% or more
    - Limit to 2 per hour

Example of Spool Utilization Statements

```
DEFSMON NAME(SPLCHECK),INTERVAL(30),USAGE(000-100),ACTION(SPLCHECK)
DEFACTN NAME(SPLCHECK),COMMAND(SPLCHECK STATUS &4),ENV(SVM)
*
DEFSMON NAME(SPLWARN),INTERVAL(5),USAGE(070-084),+
  ACTION(SPLWARN),LIMIT(2,3600)
DEFACTN NAME(SPLWARN),COMMAND(SPLCHECK WARNING &4),ENV(SVM)
*
DEFSMON NAME(SPLERROR),INTERVAL(5),USAGE(085-100),+
  ACTION(SPLHIGH),LIMIT(4,3600)
DEFACTN NAME(SPLERROR),COMMAND(SPLCHECK ERROR &4),ENV(SVM)
*
DEFSMON NAME(SPLGROW),INTERVAL(5),INCREASE(010-100),+
  ACTION(SPLGROW),LIMIT(2,3600)
DEFACTN NAME(SPLGROW),COMMAND(SPLCHECK GROWTH &5),ENV(SVM)
```

Homegrown Automation

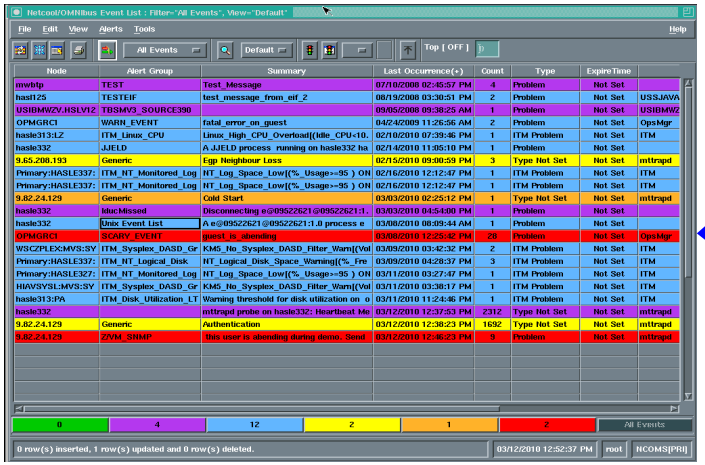
- Examples of our homegrown automation used in conjunction with Operations Manager
  - Process and archive Velocity performance reports
  - Archive Velocity MXG data to z/OS for processing
  - Migrate RACF SMF data to z/OS for processing
  - Back up Operations Manager logs to meet audit requirements
  - Utility to search Operations Manager logs for server specific messages
  - Daily DIRECTORY backups

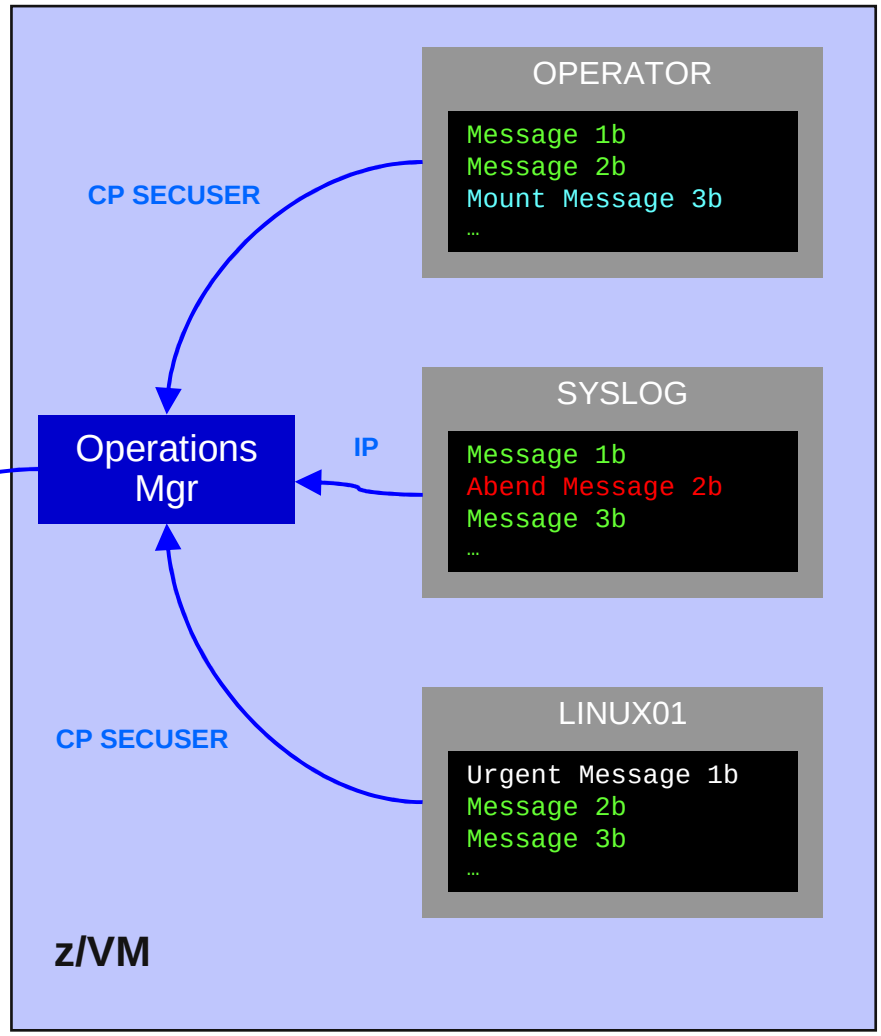
Future plans for Operations Manager for z/VM

- Integration with Netcool/OMNIbus
  - http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101492
- Manage startup and shutdown of z/VM and Linux virtual machines
- Additional virtual machine monitoring

Integration with Netcool/OMNIbus

IBM Tivoli Netcool/OMNIbus





Controlling Startup/Shutdown

- Startup:
  - AUTOLOG1 starts RACF
  - AUTOLOG2 starts Operations Manager for z/VM (OPMGRM1)
  - Run an Operations Manager for z/VM startup configuration file
  - Convert to running Operations Manager for z/VM configuration file dynamically
- Shutdown:
  - Dynamically switch to Operations Manager for z/VM shutdown config file
  - Stops all active monitoring/actions and begins orderly shutdown of z/VM and Linux virtual machines
    - Continues to collect all console data
  - Shutdown z/VM – which will shutdown Operations Manager for z/VM

Monitoring Running Tasks

- DEFEMON
  - Monitors for state change of a virtual machine. Events from the CP system service *VMEVENT are captured and compared against defined event monitors. If conditions for the event monitor are met, the associated action is scheduled for execution.

- DEFMMON
  - Monitor for the desired state of a virtual machine. A check is performed to determine if the virtual machine is logged off. If it is, the specified action is scheduled for immediate execution. The action typically issues a command to bring the virtual machine to a logged on status.

- DEFIMON
  - Use the DEFIMON command to define an idle monitor. A check is performed to determine if the RULE, MACHINE, SCHEDULE, SPOOL, EVENT, PAGE, or IDLE resource is triggered a specified number of times within a specified time frame. If a resource is not triggered, the action associated with the idle monitor is scheduled to run.

Automation Demos Available

1. Send an e-mail based on a console message
2. **Send an alert to Netcool/OMNIbus based on a console message, hold and unhold messages**
   a. Using POSTZMSG interface to Netcool/OMNIbus
   b. Using SNMP interface to Netcool/OMNIbus
3. **Send a message or email if spool approaches full**
   a. Send a message if spool usage is too high on any member of an SSI Cluster
   b. Send an email if spool usage is too high on a single system
4. View and clean up spool files
5. Automated spool cleanup
6. Archiving DIRMAINT's log files when disk gets full
7. Process a file of test messages as a console
8. Process Linux syslog data as a console
9. Create a central operations console on one z/VM system
10. Create a central operations console across multiple z/VM systems
    a. When the systems are in an SSI cluster
    b. When the systems are not in an SSI cluster
11. Integration with OMEGAMON XE on z/VM and Linux - take action based on CPU usage of Linux guest
12. **Monitor service machines for logoff – and autolog them**
13. Send an email if page space approaches full
14. Monitor SSI connectivity between 2 cluster members
15. Suppress passwords on Linux consoles
16. Autolog a Linux Guest and Send Message if Doesn't Start Successfully
17. View consoles of Linux guests, Linux syslog data, and CMS user IDs or service machines


धन्यवाद
Hindi
多謝
Traditional Chinese
감사합니다
Korean
Спасибо
Russian
Gracias
Spanish
Thank You
English
Obrigado
Brazilian Portuguese
شكراً
Arabic
Danke
German
Grazie
Italian
多谢
Simplified Chinese
Merci
French
நன்றி
Tamil
ありがとうございました
Japanese
ขอบคุณ
Thai