



# IBM Training

## System Automation for z/OS 4.1 Operations

### Student Notebook

Course code SM927 ERC 1.1

June 2019



All files and material for this course are IBM copyright property covered by the following copyright notice.

© Copyright IBM Corp. 2019. All Rights Reserved.

US Government Users Restricted Rights: Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml).

Trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this publication to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth, savings or other results.



# Contents

<b>About this course .....</b>	<b>11</b>
About the student .....	.12
Learning objectives .....	.13
Course agenda .....	.13
Typographical conventions .....	.15
References .....	.19
Naming conventions .....	.19
<b>1 Introduction to Operations using System Automation for z/OS and NetView .....</b>	<b>21</b>
Lesson 1 An introduction to System Automation for z/OS .....	.23
System Automation for z/OS: Components .....	.25
System operations component .....	.27
What can you automate? .....	.29
Summary of key automation technologies .....	.31
Architecture .....	.33
Key automation features .....	.35
Goal driven automation .....	.38
Lesson 2 Key operations features and user interfaces .....	.40
Key operations features .....	.41
User interfaces .....	.43
Single Point of Control and Single System Image .....	.44
Lesson 3 Policy structure and resource types .....	.45
Lesson 4 The role of the operator .....	.50
The role of the operator: Control, monitor, manage, analyse... .....	.52
Lesson 5 Operations using NetView for z/OS .....	.59
<b>2 Initialization.....</b>	<b>80</b>
Lesson 1 Automation environment and availability .....	.82
The automation manager environment .....	.84
The automation agent environment .....	.86
Sysplex environment example .....	.87
Automation manager availability .....	.88
Lesson 2 Initialization and initialization options .....	.89
Initialization options .....	.90
Manager initialization .....	.91
Manager start types .....	.92
Agent initialization sequence .....	.93
Agent initialization options .....	.95
Agent start .....	.97

Lesson 3 Managing the automation environment .....	98
Refreshing automation configuration files .....	100
Refreshing configuration files (continued) .....	101
Stopping the automation manager .....	102
Switching the primary automation manager .....	103
Lesson 4 Operational situations .....	104
Operational situations .....	104
Initialization scenario: System A .....	105
Initialization scenario: System B .....	107
Primary automation manager failure situation .....	108
Token mismatch situation .....	109
Agent recycle .....	110
<b>3 Commands and operations .....</b>	<b>114</b>
Lesson 1 Command basics .....	116
Command basics .....	117
Command format .....	118
Specifying resources .....	119
Routing commands to other systems .....	120
Controlling command output .....	121
Writing command output to a data set .....	122
Sorting the command output .....	123
Searching for strings .....	124
Lesson 2 Commands to display resource status .....	125
Commands to display resource status .....	126
INGLIST: Automation manager resource status display .....	127
INGLIST: An example of the primary panel .....	129
Row commands on the INGLIST panel .....	131
Setting view filters for INGLIST .....	133
INGFILT example .....	134
DISPSTAT: Display automation agent resource status .....	135
DISPSTAT primary panel example .....	136
Setting view filters for DISPSTAT .....	137
Row command E: Explanation of agent resource status .....	138
Status Display Facility .....	139
Status Display Facility panels .....	141
SDF system status display example .....	142
Status-details display example .....	143
Lesson 3 Other frequently used commands .....	144
Other frequently used commands .....	145
INGINFO: Display automation details about a resource .....	146
INGINFO: Example panel .....	147
INGINFO: Example panel .....	148
DISPINFO: Agent display of automation details for a resource .....	149
DISPINFO example .....	150
INGHIST: Display automation history for a resource .....	151
INGHIST example .....	152
INGRELS: Display resource relationships .....	153

INGRELS example .....	154
DISPTREE: Display start or stop dependencies for a resource .....	155
DISPTREE examples .....	156
INGREQ: Start or stop a resource .....	157
INGREQ verification panel .....	159
INGREQ verification panel example .....	160
INGVOTE: Managing resource votes .....	161
INGVOTE example .....	162
Lesson 4 Starting and stopping resources .....	164
Starting and stopping resources .....	165
Conditions for a successful start .....	166
Starting applications with the INGREQ command .....	167
Starting applications .....	169
Conditions for a successful stop .....	170
Considerations when stopping applications .....	171
Shutdown phases .....	172
Shutdown phase processing .....	174
Shutdown pass examples .....	175
Scope of a stop request .....	176
INGLKUP: Display or stop non-automated jobs .....	178
INGLKUP: Display or stop non-automated jobs .....	179
INGLKUP: Display jobs example .....	180
<b>4 More commands and operations .....</b>	<b>183</b>
Lesson 1 Commands to manage automation flags .....	185
Commands to manage automation flags .....	186
Changing resource settings at the automation manager .....	187
INGLIST: Resource settings management panel .....	188
Setting start-type or stop-type for a resource .....	189
DISPFLGS: Displaying automation agent flags .....	190
Determining agent flag settings for a resource .....	191
DISPFLGS examples .....	192
INGAUTO: Changing agent automation flags for a resource .....	193
INGAUTO panel .....	194
Lesson 2 Suspend and resume resources .....	195
INGSUSPD panel: Suspend resource .....	196
INGSUSPD verification panel .....	198
Suspend request completed .....	199
INGLIST showing suspended .....	200
INGVOTE showing propagated suspend request .....	202
Use INGSUSPD to resume .....	203
INGSUSPD verification panel for resume .....	204
Lesson 3 Other start/stop tools .....	205
Other start/stop tools .....	206
Pacing gates .....	207
Pacing gates dialog INGPAC .....	208
Pacing gates dialog INGPAC F11 .....	209
Pacing gates dialog INGPAC details .....	210

Pacing gates on INGLIST .....	211
Pacing gates on INGWHY .....	212
SETSTATE: Modify agent status for a resource .....	213
SETSTATE example .....	214
INGMDFY: Temporary modification of start/stop phases .....	215
INGMDFY STOP example .....	216
Lesson 4 Managing the automation environment .....	217
Managing the automation environment .....	218
INGAMS command functions .....	219
INGAMS primary panel .....	220
Suspending automation manager processing .....	222
Setting or changing runmodes .....	223
Resources qualified for a runmode .....	224
If no defined runmode is active on a system, by default, the active runmode becomes *ALL which means that no qualification restrictions are in effect. No stop requests are generated. All resources which have a desired status of MakeAvailable, and good shutdown status are started. Specifying *ALL in the runmode field on the ingrun panel replaces an active, defined runmode. ....	224
INGRUN command panel .....	225
INGRUN votes example .....	226
INGRPT: Statistics and base information .....	227
INGRPT: Summary report .....	228
DISPSYS: Display system information .....	229
INGNTFY: Managing notify operators .....	230
Lesson 5 Managing possible inhibitors .....	231
Managing possible inhibitors .....	232
An overview of resource thresholds .....	233
INGTHRES: Display and manage thresholds .....	234
INGSCHED: Managing schedules (service periods) .....	235
INGSCHED: Display schedule list .....	236
INGSCHED: Display base schedule details .....	237
INGSCHED: Overriding a base schedule .....	238
Triggers and events .....	239
INGTRIG: Display triggers .....	240
DISPEVTS: List the events in a SAplex .....	242
Lesson 6 Sysplex automation commands .....	243
INGPLEX: Command options .....	245
INGPLEX: List XCF couple data sets .....	247
INGPLEX: IPL information .....	249
Lesson 7 UNIX System Services automation .....	250
UNIX System Services automation .....	251
UNIX System Services add-on policy .....	252
<b>5 Application groups.....</b>	<b>256</b>
Lesson 1 Application moves .....	258
Initiating application moves .....	259
Evaluating move actions .....	260
Using INGGROUP for goal driven moves .....	261
INGMOVE: A command for move groups .....	263

Prepare Move policy .....	264
Move only if system is down or stopping and Move only after system failure .....	265
Lesson 2 Group management commands .....	266
Group management commands .....	267
INGGROUP: Primary panel .....	268
INGGROUP members panel example .....	270
INGGROUP: Rolling recycle .....	273
INGGROUP: Rolling recycle notification .....	274
Display group OVERRIDES example .....	275
INGMOVE example .....	276
A summary of INGMOVE commands .....	277
Lesson 3 Optional topics .....	278
Optional topics .....	279
Disruptive versus non disruptive moves .....	280
Example of a disruptive move .....	281
Example of a non disruptive move .....	282
Configuration for INGGROUP examples .....	283
Example 1 scenario: Changing the availability target (1 of 2) .....	284
Example 2 scenario: Adjusting preference values .....	287
<b>Unit 6: Application monitoring.....</b>	<b>293</b>
Lesson 1 Use DISPMTR to manage monitor resources .....	295
Display health status .....	296
DISPMTR command output example .....	297
Lesson 2 Using OMEGAMON XE for event-based monitoring .....	300
Using OMEGAMON XE for event-based monitoring .....	301
Event-based monitoring .....	302
Attributes, situations, and situation events .....	303
Event-based automation example .....	304
Lesson 3 Using Tivoli Enterprise Portal .....	305
Using Tivoli Enterprise Portal .....	306
Monitoring agent overview .....	307
SA z/OS Tivoli Enterprise Monitoring Agent with z/OS monitoring server .....	308
Automation Physical navigator example .....	309
The Resource Topology workspace .....	311
Resource Details workspace example .....	312
Resource Requests workspace .....	313
Automation Environment workspace .....	314
Automation Statistics workspace .....	315
Situations and situation events .....	317
Viewing operator requests .....	318
Take Action commands .....	319
<b>7 Centralized operations.....</b>	<b>323</b>
Lesson 1 Gateways .....	325
An overview of gateways .....	327
Gateway communication .....	328
Gateway sessions .....	329
Displaying gateway connections .....	330

Lesson 2 Status Display Facility focal point .....	331
Status Display Facility focal point .....	332
An overview of SDF focal points .....	333
Enterprise monitoring with SDF .....	334
Determining the focal point .....	335
SDF enterprise view panel example .....	336
System status summary example .....	337
Subsystem status panel example .....	338
Lesson 3 Managing resources in a multisystem environment .....	339
Managing resources in a multisystem environment .....	340
Gateways, focal point, and SPOC .....	341
A summary of SPOC .....	342
Centralizing operations .....	343
The TARGET parameter .....	344
Using the TARGET parameter: Example .....	345
The SETSTATE command panel .....	347
The SETSTATE system selection panel .....	348
Routing messages in a multisystem environment .....	349
Notify operator example view .....	351
Routing commands .....	352
RMTCMD example .....	353
INGEXEC example .....	354
<b>8 Troubleshooting .....</b>	<b>357</b>
Lesson 1 Solving problems with resources .....	359
Implement policy that is correct .....	361
Detecting problems .....	364
Analysis approach .....	365
Finding the reason for start problems .....	366
Compound status = INHIBITED, Degraded, InAuto .....	367
Observed Status = HARDDOWN .....	368
Finding the reason for stop problems .....	369
INGWHY - What does it do? .....	370
INGWHY - What is the cause? .....	371
INGWHY – Does not... .....	373
Using INGWHY: The situation .....	374
Using INGWHY: Scroll to next reason .....	376
Lesson 2 Collecting information .....	377
NVINFO collects NetView diagnostic information .....	378
NETLOG, syslog and joblogs .....	379
INGLKUP resource REQ=COLLECT .....	380
Lesson 3 Using trace and debug .....	381
Automation manager State Trace Table .....	382
Trace services for the automation manager and agent .....	383
AOCTRACE dialog .....	384
AOCTRACE for automation debugging .....	385
Lesson 4 Solving message automation problems .....	386
NetView message types, AT, MRT, CRT .....	387

CNM493I shows match in an automation table . . . . .	391
CNM493I lists commands executed . . . . .	392
Tracing and simulating messages . . . . .	393
Finding SA message automation . . . . .	394
List Automation Table commands . . . . .	396
INGMSGSS to display exceptional messages . . . . .	397
Finding status messages . . . . .	398
DISPACF for JES2 messages . . . . .	399





# About this course

---



## IBM System Automation for z/OS 4.1 Operations



© Copyright IBM Corporation 2018

Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This course introduces and explains the System Automation for z/OS® (SA z/OS) commands that are used for system operations. In this course, the System Automation for z/OS automation manager and automation agent run in a z/OS 2.2 environment. The automation platform, Tivoli® NetView for z/OS is at version 6 release 2. The course uses several automation scenarios in single and multisystem configurations to demonstrate the concepts that are taught in the lessons. This training class is delivered in an environment with multiple opportunities for hands-on lab exercises.

For information about other related courses, see the IBM education training paths website:

[ibm.com/training/](http://ibm.com/training/)

Details	
<b>Delivery method</b>	Classroom
<b>Course level</b>	ERC 1.0  This course is a new course.
<b>Product and version</b>	IBM System Automation for z/OS 4.1  Including updates June 2018: APAR OA55386 and Service Management Unite 1.1.4
<b>Duration</b>	3.5 days
<b>Skill level</b>	Basic



**Note:** This course applies to IBM System Automation for z/OS (System Automation), but can be also used for IBM Automation Control for z/OS. Functions not provided by IBM Automation Control for z/OS are in *underlined italics*. The terms automation and the automation product are used when there are no differences. Alternatively the product name is used to describe its functions. At the time of writing IBM Automation Control for z/OS was still at a IBM System Automation for z/OS V3.5 level

## About the student

This course is designed for users of IBM System Automation for z/OS, especially operators, administrators, and system programmers.

Before taking this course, make sure that you have attended the following course or have equivalent skills:

- *SM917 - IBM System Automation for z/OS 4.1 Architecture*

Before taking this course, make sure that you have the following skills:

- Basic z/OS operations skills
- Basic NetView skills

Recommended courses before this course are:

- *TZ203 - IBM Tivoli NetView for z/OS 6.1 Fundamentals*
- *TZ213 - IBM Tivoli NetView for z/OS 6.1 Automation*

# Learning objectives

The learning objectives are to get the required knowledge to use System Automation for z/OS or IBM Automation Control for z/OS to perform the following operations tasks:

- Start and stop automated resources
    - Troubleshooting
  - Monitor and control automated resources
  - IPL and shut down systems and a SAplex
- 

## Objectives

---

When you complete this course, you can use IBM System Automation for z/OS 4.1 or IBM Automation Control for z/OS to perform the following tasks:

- Start and stop automated resources
- Monitor and control automated resources
- IPL and shut down systems and an SAplex

---

© Copyright IBM Corporation 2018

# Course agenda

The course contains the following units:

1. [Introduction to Operations using System Automation for z/OS and NetView](#)

This unit provides an introduction to operations using System Automation for z/OS and NetView. This unit repeats some of the contents from the architecture course like introduction to System Automation and its components, component System Operations its architecture and its key

features. This unit details the role of the operator in controlling automation and resources and teaches how to use NetView for operations.

## 2. [Initialization](#)

This unit describes the initialization and synchronization of the automation manager and automation agents. It explains how each component is initialized, how to refresh the automation configuration file, and the management of automation managers.

## 3. [Commands and operations](#)

This unit introduces operator commands that you use to manage automated resources. Some of the commands that are explained, and demonstrated in this unit are SDF, INGLIST, DISPSTAT, INGREQ, INGINFO, and DISPINFO.

## 4. [More commands and operations](#)

This unit introduces more operator commands that are used to manage the automation environment. The lessons include topics on how to use commands such as INGAMS, INGRUN, INGRPT, DISPSYS, and INGPLEX.

## 5. [Application groups](#)

This unit introduces application groups, group types, group natures, and their attributes. The unit highlights and explains the benefits of groups when managing automated business applications in an enterprise. The lessons include topics on group management commands such as INGGROUP, INGMOVE, INGVOTE, and INGLIST.

## 6. [Application monitoring](#)

The primary objective of this unit is performance monitoring of automation managed resources. The lessons introduce and explain automation monitor resources, exception-based, and event-based monitoring, OMEGAMON XE, and the Tivoli Enterprise Portal.

## 7. [Centralized operations](#)

This unit reinforces topics that are covered in previous lessons in the course. You learn to use Automation to manage resources in a multisystem environment. The lessons in this unit include topics on managing gateway sessions, and managing enterprise automation from a focal point. You learn to use the Status Display Facility (SDF), Single Point of Control (SPOC) techniques, and the TARGET parameter to route commands.

## 8. [Troubleshooting](#)

This unit is building on what you have learned to determine why resources won't start or stop. Collecting information, using trace and debug and solving problems with messages.

# Typographical conventions

In this course, the following typographical conventions are used.

Convention	Usage
<b>Bold</b>	Important text is emphasized in <b>bold</b> .
<i>Italics</i>	New terms appear in <i>italics</i> when they are defined in the text.
<b><i>Bold Italics</i></b>	Important new terms appear in <b><i>bold italics</i></b> when they are defined in the text.
Monospace	Code examples, output, and system messages appear in a monospace font.
UPPERCASE	Mainframe commands and parameters can be shown in UPPERCASE letters.
<u><i>Italics underlined</i></u>	Functions not provided by IBM Automation Control for z/OS are in <u><i>underlined italics</i></u>

## Commonly used acronyms

Acronyms written in bold are frequently used. Acronyms written in italics are from SA z/OS or NetView.

- **Application (APL, SA z/OS resource)**
- **Application Group (APG, SA z/OS resource)**
- *Automation Manager Configuration (AMC)*
- Automated operator (AOP, SA z/OS resource)
- Automatic restart manager (ARM)
- *Automation control file (ACF)*
- *Automation Status File (ASF)*
- **Automation Table (AT, previously also MAT)**
- *Availability Target (AVT)*
- *BCP Internal Interface (BCPi)*
- *Command Authorization Table (CAT)*
- Couple data set (CDS)
- Coupling facility (CF)
- Coupling Facility Resource Management (CFRM)
- Central processor complex (CPC)
- **Cross-system coupling facility (XCF)**
- Customer Information Control System (CICS)
- Data definition (DD)
- Data set name (DSN)
- **Extended Multiple Console Support (EMCS)**
- Extensible Markup Language (XML)
- Family of IBM licensed programs for relational database management (DB2®)
- Generation data group (GDG)
- **Geographically Dispersed Parallel Sysplex (GDPS)**
- **Hardware Management Console (HMC)**
- IBM Tivoli Monitoring (ITM)
- *IBM Tivoli NetView for z/OS (NetView)*
- *IBM System Automation for z/OS (SA z/OS)*
- IBM WebSphere MQ (*formerly MQ Series*) (MQ)

- IBM Workload Scheduler (IWS or previously also TWS, OPC)
- Information Management System (IMS)
- **Initial program load (IPL)**
- **Interactive System Productivity Facility (ISPF)**
  - Job control language (JCL)
  - Job Entry Subsystem (JES)
  - Library lookaside (LLA)
  - Lifecycle reporting (LCR)
  - Logically partitioned (LPAR) mode
- **Message processing facility (MPF)**
  - **Monitor resource (MTR, SA z/OS resource)**
  - Multi-access spool (MAS)
  - Multiple Console Support (MCS)
  - Multiple Virtual Storage (MVS)
  - *NetView-to-NetView Task (NNT)*
  - *Network resource (NTW, SA z/OS resource)*
  - OpenEdition for MVS (OMVS)
  - Partitioned data set (PDS)
  - Partitioned data set extended (PDSE)
- **Policy Data Base (PDB)**
- **Primary Automation Manager (PAM)**
  - Primary POI task (PPT)
  - ProcOps Service Machine (PSM)
  - Program operator interface (POI)
  - Program temporary fix (PTF)
- **Program to program interface (PPI)**
  - Resource Access Control Facility (RACF)
  - Resource Measurement Facility (RMF)
  - Runtime environment (RTE)
- **SA z/OS Subplex (SAplex)**
- **Secondary Automation Manager (SAM)**
- Security Access Facility (SAF)

- Short Message Services (SMS)
- Simple Object Access Protocol (SOAP)
- ***Status display facility (SDF)***
- **Subsystem interface (SSI)**
- Supervisor Call (SVC)
- System Automation for Integrated Operations Management (SA IOM)
- System Display and Search Facility (SDSF)
- System Management Facility (SMF)
- *Terminal Access Facility (TAF)*
- Time Sharing Option (TSO)
- Tivoli Enterprise Monitoring Agent (TEMA)
- Tivoli Enterprise Monitoring Server (TEMS)
- Tivoli Enterprise Portal (TEP)
- Tivoli Enterprise Portal Server (TEPS)
- UNIX® System Services (USS)
- Virtual Operator Station Task (VOST)
- Virtual Storage Access Method (VSAM)
- Virtual Telecommunications Access Method (VTAM)
- **Write to operator (WTO)**
- **Write to operator with reply (WTOR)**

# References

## IBM System Automation for z/OS manuals

- Customizing and Programming
- Planning and Installation
- Defining Automation Policy
- User's Guide
- Messages and Codes
- Operator's Commands
- Programmer's Reference
- Product Automation Programmer's Reference
- TWS Automation Programmer's Reference and Operator's Guide
- Service Management Unite Automation Installation and Configuration Guide
- End-to-end Automation
- Monitoring Agent Configuration and User's Guide

## IBM Automation Control for z/OS manual:

- Getting Started

# Naming conventions

- This course is about the System Automation for z/OS functions as they apply to a sysplex. System Automation for z/OS supports dividing a sysplex into multiple subplexes, called **SAplex**. Whenever you see reference to a sysplex, it also applies to a SAplex.
- Whenever you see a command, its is an automation product command. Other commands are specified explicitly, like MVS command.





# 1 Introduction to Operations using System Automation for z/OS and NetView

IBM System Automation for z/OS 4.1



## Unit 1: Introduction to Operations using System Automation for z/OS and NetView



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit provides an introduction to operations using System Automation for z/OS and NetView. This unit repeats some of the contents from the architecture course like introduction to System Automation and its components, component System Operations its architecture and its key features. This unit details the role of the operator in controlling automation and resources and teaches how to use NetView for operations.

# Objectives

When you complete this unit, you can perform these tasks:

- Describe the basic components of System Automation for z/OS
  - Describe the automation architecture
  - Describe typical policy resource types and the configuration file structure
  - Describe key automation and operations features
  - Describe the role of the operator and typical tasks
  - Operations using NetView for z/OS
- Notes:
    - This unit repeats some of the contents from the architecture course
    - This course applies to IBM System Automation for z/OS (System Automation), but can be also used for IBM Automation Control for z/OS. Functions not provided by IBM Automation Control for z/OS are in *underlined italic*. The terms automation and “automation product” are used when there are no differences

## Objectives

When you complete this unit, you can perform the following tasks:

- Describe the basic components of System Automation for z/OS
- Explain what a resource is
- Describe the roles of the automation manager and the automation agents in a SAplex
- Describe the purpose of the policy database
- Describe what operator interfaces exist



**Note:** This course applies to IBM System Automation for z/OS (System Automation), but can be also used for IBM Automation Control for z/OS. Functions not provided by IBM Automation Control for z/OS are in *underlined italic*. The terms automation and “automation product” are used when there are no differences. At the time of writing IBM Automation Control for z/OS was still at a IBM System Automation for z/OS V3.5 level

# Lesson 1 An introduction to System Automation for z/OS



## Lesson 1. Overview



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson provides the base for the course by repeating some of the contents from the architecture course to provide introductions to System Automation and to component System Operations, its architecture, its key features, and to goal driven automation.

## Lesson agenda

This is a repeat from the architecture course and covers

IBM System Automation for z/OS:

- Overview
- System Operations component
- Architecture
- Goal driven automation

*Lesson agenda*

This is a repeat from the architecture course and covers IBM System Automation for z/OS overview and the architecture and goal driven automation of its System Operations component.

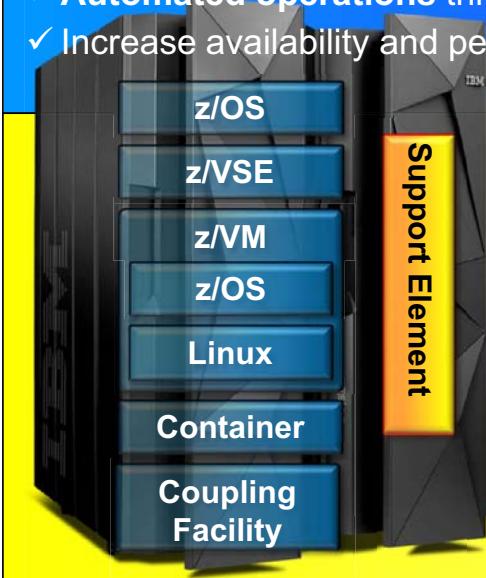


**Important:** Please review whether you need this required level set.

# System Automation for z/OS components

## System Operations

- ✓ Automate repetitive and complex tasks in **z/OS systems and end-to-end**
  - ✓ Helps operators to perform their tasks in a more robust way
  - ✓ Reduces z/OS specific skill requirements
- ✓ **Automated operations** through monitoring of applications and messages
- ✓ Increase availability and performance through pro-active automation



## Processor Operations

- ✓ z Systems external automation and single point of control
- ✓ **Faster LPAR startup**, recovery, shutdown
- ✓ **Save software costs** through LPAR capacity management
- ✓ **Higher availability**: Don't miss any hardware alerts
- ✓ **Save energy** with power management

© Copyright IBM Corporation 2019

1-5

*System Automation for z/OS: Components*

System Automation for z/OS has two key components that are designed to automate processor and system operation. These components are called processor operations and system operations.

## ***The system operations component***

The **system operations** component and NetView as the prerequisite automation platform, can help operators by automating many system console operations and selected operator tasks such as startup, monitoring, recovery, and shutdown of z/OS subsystems, components, applications, UNIX (USS), and sysplex resources. The system operations component can also automate operator console messages, messages in the job log, initiate timer-based actions, and prevent shortages of critical z/OS resources.

The automation product integrates with **OMEGAMON XE** to help you to increase availability and performance through proactive automation.

## ***The processor operations component***

**Processor operations** helps operators manage more systems with greater efficiency. This means that one operator, even from a remote location, can configure, initialize, monitor, shut down, and

recover a central processor complex (CPC), Logical Partitions (LPAR), and multiple systems in parallel and respond to various detected conditions. An operator, using one standard interface, can do all that across multiple types of systems such as z/OS, z/VSE, z/VM, and Linux.

Clients can save software costs through LPAR capacity management, for instance by adjusting capacity across LPARs and WLM capacity groups automatically.

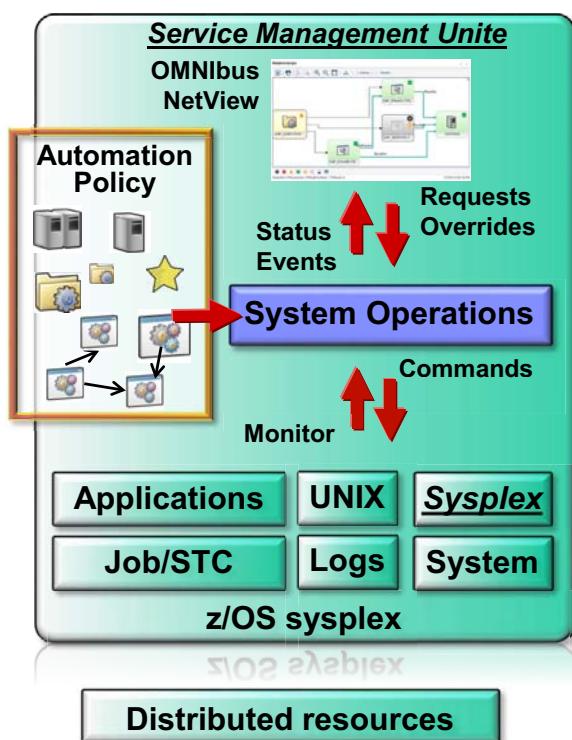
Hardware alerts and priority messages that must be responded to in a matter of seconds can be automated to increase availability.

Save energy with power management based on time of day or system load.



**Note:** The I/O Operations component is not covered as V3.5 was the last release supporting it.

# Component System Operations overview



## Policy-based and goal driven automation

- Start, recover and shutdown z/OS and distributed applications and systems using:

- Timers, events, triggers, service periods
- Dependencies and groups
- Thresholds, active and external monitors

- JES, CICS, IMS, DB2, WebSphere, OMEGAMON, UNIX, SAP, GDPS...

## Message monitoring and response

- WTO, WTOR, joblog, NetView, hardware

- Escalation using OMNIbus, email, SMS...

## Prevent outages of critical resources:

- WTO and AMRF buffers, spool, sysplex
- SYSLOG, LOGREC, SMF, dump data sets

## Easier operations at the application level

- 3270 or Service Management Unit

- Single point of control, single system image

© Copyright IBM Corporation 2019

1-6

### System operations component

The system operations component, which exploits NetView, automates many system console operations and selected operator tasks such as startup, monitoring, recovery, and shutdown of z/OS subsystems, components, applications, UNIX (USS), and sysplex resources and entire systems and sysplexes and even distributed resources (SA z/OS only).

The system operations component provides comprehensive, Plug and Play automation for mission-critical software like JES, CICS, IMS, DB2, IBM Workload Scheduler, WebSphere, OMEGAMON, UNIX, SAP, and GDPS (SA z/OS only), and many more.

Applications can be started or stopped with **timers, events, triggers, or service periods**.

**Dependencies** can be automatically established, like starting TSO after VTAM.

Resources can be members of **groups**, which can greatly reduce the complexity of automation definition and operations.

**Server groups** control which and how many group members are started. Group members can represent, for instance, application servers.

**Thresholds** can be defined to alert operators or stop recovery.

Various **active monitors** including a fast control block scanner and a monitor for UNIX resources are available.

Clients can specify their own monitor and interval. Monitor resources can determine the **health status** using external monitors or OMEGAMON.

**Messages** like write-to-operator (WTO) or WTOR, or even messages from job logs or internal CICS and IMS messages can be automated without writing scripts. Automation can be different depending on codes in the message or depending on how often the message appeared.

**Escalation** to various notification targets can be set up easily using pre-defined or user-defined alert points on a resource level to complete the following tasks:

- Create incidents in IBM Tivoli Service Request Manager
- Create events in IBM Tivoli Netcool OMNIbus
- Alarm staff using email, pager, or SMS using IBM System Automation for Integrated Operations Management (SA IOM)

The system operations component provides automation procedures that enable recovery of the following **z/OS components** and data sets:

- SYSLOG data sets
- LOGREC data sets
- System Management Facility (SMF) data sets
- Write-to-operator (WTO) buffers
- JES spool
- z/OS dump data sets
- Action Message Retention Facility (AMRF) buffers
- Sysplex resources, for example, coupling facility (SA z/OS only)

All that you have to do is specify to system operations in an ISPF dialog what resources you want to automate and monitor, and what your policies are for automation and monitoring. The automation engine, which is identical on all systems, starts, monitors, recovers, and shuts down resources and complex applications according to your configuration and goals.

At any time, the operator can monitor and control from an enterprise-wide single point of control at the application level. You have the choice of using a 3270-based NetView console or the graphical user interfaces Service Management Unite (part of SA z/OS) or the Tivoli Enterprise Portal, which is monitoring only. When exceptions occur or goals must be changed, an operator just issues requests or overrides. SA z/OS offers a SAplex single system image, system boundaries are removed.

**Goal driven automation** greatly simplifies operations. Operators just request what they want, and automation takes care of any dependencies and resolution of affected or even conflicting goals.

# What can you automate?

- Automate messages from z/OS, applications, logs, hardware
- Prevent outages of critical resources
- Start, recover, and stop z/OS and cross platform resources
  - Started tasks (STCs) and jobs independent of scheduler
  - UNIX System Services (USS) resources
  - Cross sysplex and cross platform
- IPL, startup, and shutdown pacing
- Change system configuration like day or night shift
- Automatic or manual application move or switch in SAplex
- DB2, CICS, and IMS automation
- Escalation of problems
- Pro-active automation using integration with monitors and OMEGAMON
- Integration with scheduling
- Problem determination

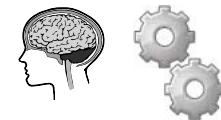
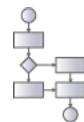
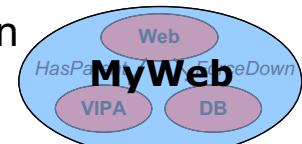
What can you automate?

- Basic automation scenarios like automating messages from many sources, starting, recovering, and stopping of z/OS, UNIX System Services (USS) resources and even any resources across sysplexes and platforms
- Prevent outages of critical resources like MVS consoles, JES spool, logrec, SMF, page or dump data sets, couple data sets, or syslog
- Advanced automation scenarios like:
  - IPL, startup, and shutdown pacing
  - Different system configurations
  - Pro-active automation using integration with monitors
  - Automatic or manual application move or switch inside SAplex
- Escalation of problems to a status console or to an alert console
- DB2 automation
- CICS and IMS automation, even console access and internal messages and transactions
- Integration with scheduling: IBM Workload Scheduler has an automation workstation that allows

to send any command to automation, synchronously and asynchronously

# Summary of key automation technologies

- **Policy-based automation** can replace scripts
- Plug and play automation using best practices
- Powerful **sysplex-wide automation** and operation
- **Goal driven automation** to keep applications in line with business goals
- **Grouping** of resources for reduced complexity and management at business application level
- **Relationships** between resources for accelerated startup and shutdown, and correct recovery
- Manage by resource state, not by message
  - Status change triggers automation
- Monitor resources with health status and actions



Animated

© Copyright IBM Corporation 2019

1-8

## Summary of key automation technologies

**Policy-based automation** is a sophisticated methodology that allows to easily incorporate business goals into an automation framework. As the term implies, policy-based automation uses simple policy definitions specified in fill in the blanks ISPF-based panels, with no program scripting or special education required.

Policy-based automation includes resource information, groups of resources, and relationships in the decision-making process before taking action. Resource information defines resource class and name; how to start, stop, and monitor the resource; and what the preferred systems are (SA z/OS only). Resources can be members of cluster-wide (SA z/OS only) groups and can have relationships.

IBM ships **Plug and Play automation** that includes best practices automation for many z/OS applications. This can help to complete the following tasks:

- Reduce time and effort in creating a policy or updating one
- Improve automation quality

**Goal driven automation** greatly simplifies operations. Goal driven automation tries to keep applications in line with business goals specified in the policy or by the operator. Automation can take care of any dependencies by issuing start or stop commands to put other resources into the required status. It can resolve conflicting goals and even can remember goals.

**Grouping of resources** and definition of aggregate or business applications can greatly reduce the complexity of automation definition and operations.

Exploiting application groups is beneficial in many ways:

- Automation definition and operations can be greatly simplified through the grouping of resources and even business-application definitions.
- Application groups let you monitor important business applications and help verify that everything they require is available.
- Groups can make operations easier by showing the aggregated status of resources and by group actions such as startup or shutdown.
- Through groups, operators are freed from knowing the various pieces that make up an application, their dependencies, how to start or stop them, and so on.

**Relationships** between resources to achieve accelerated startup and shutdown, and correct recovery. Resources can have complex dependencies of different kinds to other resources inside and outside of the application (group) they belong to. This gives you the power to define these dependencies, so that resources get what they need, are started in the right order as quickly as possible and are shut down fast without interference.

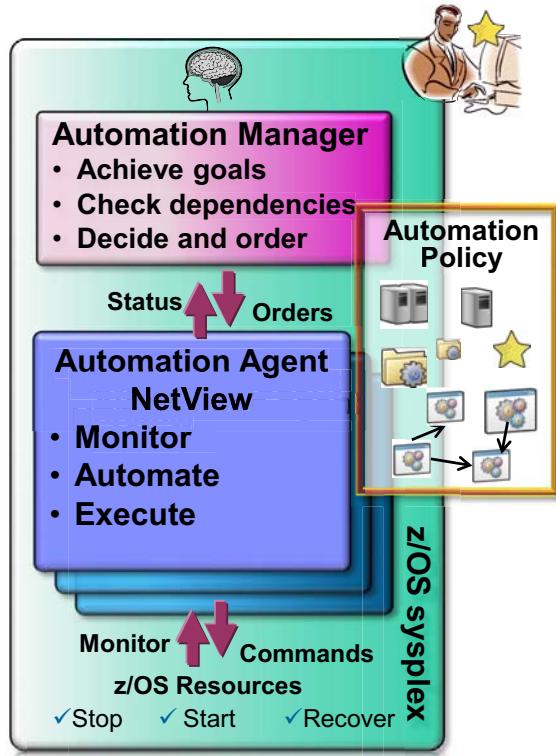
### Manage by state, not by message

Automation uses monitoring to keep track of the resource status. Although automation can use messages to update the status and also can automate messages, it is the status change that triggers automation. For example, child resources are started after the parent is up or a dependent resource is forced down when the resource it depends on terminates.

Also, messages are forgotten, but the status is preserved even across restarts and IPLs.

**Monitor resources** enable integration with any monitor and include easy to exploit OMEGAMON access to update the health status of monitor resources, which can trigger **pro-active automation**. The **health status** of monitor resources is propagated along monitor relationships to applications and is aggregated into the overall resource status.

# Architecture



- The automation policy defines the resource automation and goals ★
- Automation manager (AM)
  - Keep resource status in line with goals and dependencies by sending start and stop orders to agent
- NetView-based automation agent role:
  - Monitor and automate resources
  - Inform AM and execute AM orders
  - Operator console
- Operators
  - Control automation
  - Detect and manage problem statuses
  - Override goals

© Copyright IBM Corporation 2019

1-9

Architecture

## The automation manager

The automation manager orders the agent to load the same policy.

In the automation manager's sphere of control, it maintains status information about each resource, it controls resources, their dependencies to other resources and their desired status (goal). The automation manager uses all of this information to determine which applications to start and stop, and when to do so. The manager does not issue start and stop commands; it sends orders to automation agents which issue the start and stop commands.

## The automation agent

The automation agent runs in the NetView address space which provides the operator console. Access to the agent is through the NetView command facility. Agent commands display and allow some modification of the resource status, automation goals and status, and policy.

The automation agent uses the NetView automation table message-processing function for automation purposes. Messages are generated as the result of some event or situation. Many automation policy definitions use messages to trigger actions. A NetView automation table is the

detection mechanism for messages. The automation product creates entries in the NetView automation table for specific messages. The messages are assigned to a NetView **Autotask** to distribute workload and to ensure serialization of the messages as they are automated. Based on the type of message an action is routed to an automation operator. The action can include update of the agent status for an application or z/OS commands.

The automation agent receives orders from the automation manager and issues commands that are based on defined automation policy.

The agent sends the status changes for each application to the primary automation manager. The primary automation manager sends orders to the automation agent, requesting the start or stop of applications.

## ***The operator***

The operator role is to control automation, to detect and react on problem statuses, messages, and alerts as well as problem analysis, execution of automation change requests by managing automation, policy and resources.

Operators can use requests to override any goals.

# Key automation features

- Comprehensive automation for z/OS and UNIX applications, for example:
  - User definable **start types** and three **stop types**
  - Desired available status, startup and restart options
  - Warning and alert **thresholds** to stop recovery
  - Exception and **captured messages**
  - Built-in ASCB scan monitor reduces CPU overhead for monitoring
  - Monitoring of **UNIX** processes, files, and ports
- **Server group** with *availability* and *satisfactory targets* and member *preferences*
- Operator notification and **alerting** to SDF, SA IOM, OMNIbus, problem management, and so on
- **Runmodes** for selective startup like base, online, databases, and so on
- **Pacing gates** can prevent startup or stop of too many applications
- Non-disruptive cluster-wide policy activation
  - Synchronized with NetView automation table load
- **Automation reports** and availability and recovery time reporting
- Comprehensive **automation infrastructure** makes extension easy

## Key automation features

On this slide are the key automation features.

At the core is comprehensive automation for z/OS and UNIX applications.

Highlights include:

- User definable **start types** with multiple passes and stages that also affect message automation as well as prestart and poststart policy
- Three **stop types** with multiple passes and an INIT and FINAL phase
- **Desired available status** to initially start or stop an application or to accept its current observed status
- **Startup and restart options** like whether the application is started after any failure or only for specific ABEND codes.
- **Thresholds** can be set that let the operator know if certain errors are occurring infrequently, frequently, or have reached a critical level where the recovery process must end to avoid endless loops. This is done by specifying how many times an error must happen in a certain time period for each error situation.
- **Exception and captured messages**

Special support exists for **UNIX resources** running in z/OS UNIX System Services: Monitoring of UNIX processes including UNIX user ID, command path, and filter as well as files, and ports.

Automatic or manual move of an entire application inside a system or Parallel Sysplex (SA z/OS only) is provided by **move groups** that have member preference values.

**Server groups** can represent, for instance, application servers, and control which and how many group members are started through preferences and availability and satisfactory targets. System Automation supports sysplex-wide group members.

Move groups and server groups allow to define preferred locations of a resource and what should happen when a system is unavailable or available again.

Various active monitors including a fast control block scanner and a monitor for UNIX resources are available.

**Escalation** to various notification targets can be set up easily using pre-defined or user-defined alert points on a resource level to:

- Create incidents in IBM Tivoli Service Request Manager
- Create events in IBM Tivoli Netcool OMNIbus
- Alarm staff using email, pager, or SMS using IBM System Automation for Integrated Operations Management (SA IOM)

**Runmodes** can be defined to partially start the system or to start backup applications that are normally started on another system.

The policy management can be incremental and is non-disruptive during **policy activation**. Policy activation is synchronized with the NetView automation table and message revision table load.

If the resource consumption of too many applications during the start and stop phases becomes a problem, **pacing gates** can be defined to limit the number of applications of a kind that can be started or stopped at the same time.

**Automation reports** display statistical information about the automation agent and some basic information about the automation manager.

For availability and recovery time reporting automation collects and records job-related information, and writes **System Management Facility** (SMF) records at specific events in the lifetime of a resource.

Comprehensive **automation infrastructure** makes extension easy. Automation supplies commands that provide your automation procedures with a simple, standard way of interfacing with the automation control file, the automation status file, and the NetView log file.

Using these commands in automation procedures provides you with the following advantages:

- Reduced development time: Less code must be written.
- Portable code: Automation policy information that is unique to an enterprise can be kept in the automation control file rather than distributed among many automation procedures. The

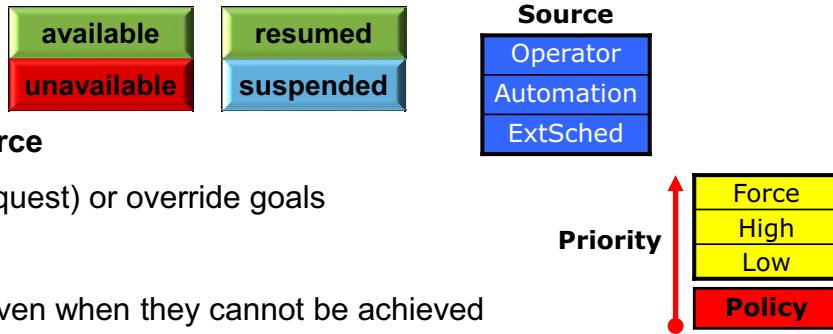
automation procedures implement a number of different rules for handling a situation and the automation control file is used to select which rules are applicable to the current situation.

- A consistent, documented interface. Messages like Write-to-operator (WTO) or WTOR or even from job logs or UNIX can be automated without writing scripts. Automation can be different depending on codes in the message or depending on how often the message appeared.

As it was covered previously, it is not mentioned here that the system operations component provides automation procedures that enable recovery of many z/OS components and data sets.

# Goal driven automation

- Availability goal
- Suspended goal
- Goals come from a **source**
- Operator can create (request) or override goals
- Goals have a **priority**
- Goals are **persistent**, even when they cannot be achieved
  - Unlike commands that are entered and forgotten
  - Cancel requests when you no longer need them
- Automation manager tries to achieve goals while taking care of dependencies
- Easier and safer operations with a single action at application group level
  - **Goals can be propagated** to group members and along active dependencies as votes putting all required resources in the desired status
  - **Conflicting votes are resolved** by priorities and request sources
- Resource status adjusted to goals, dependencies, configuration, and status



## Goal driven automation

Goal driven automation is very different from the command driven automation of other products:

- Goal driven automation can keep applications in line with business goals, dependencies, configuration, and status
- Goals are defined in the policy, can be overridden or created by the operator
- Goals are persistent
- Goals can be propagated

The administrator defines the goals for the application according to business requirements in the policy. Goals are either available (up) or unavailable (down). Application group goals control, which and how many members are started.

The operator can change or override goals and create start or stop goals by entering requests. Operator requests can have a higher priority than the Desired Available state from the policy.

Goals are persistent. They are not like commands that are run and forgotten. You must cancel requests when you no longer need them.

Operations at the application level means starting an application having several components and dependencies with one request.

- The request is propagated as votes to group members and along the dependency tree
- When multiple dependencies exist, conflicting goals can be the result. Conflicting goals are resolved. Requests have a **priority** and have a **source**. Automation uses priorities and sources when determining which request to accept. Sources might be, for example, automation scripts (**AUTOOPS**), human operators (**OPERATOR**), or **EXTERNAL**, like IBM Workload Scheduler

This process can prevent operations errors, like shutting down a resource that is needed by another application. Operators can use goal driven automation to simplify operations by entering or removing a request at the application level. They can use the automation product to manage dependencies and goals that are affected and to resolve conflicting goals.

The automation product automatically adjusts resource status to business goals, dependencies, configuration, and status.

# Lesson 2 Key operations features and user interfaces



## Lesson 2. Key operations features and user interfaces



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson introduces the key operations features and the user interfaces available in NetView with 3270-based operator dialogs and the dynamic, highly customizable Status Display Facility as well as an overview of the graphical user interfaces are available:

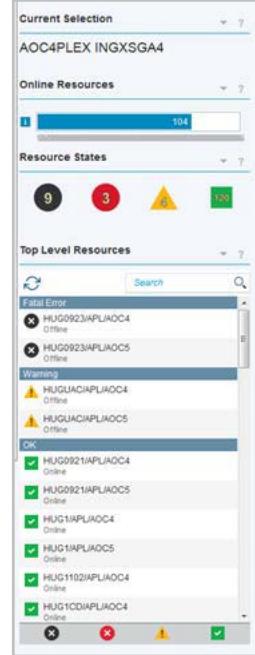
- Service Management Unite (SMU): A new way to operate your mainframe.
- The Tivoli Enterprise Portal to monitor the status of automation on z/OS systems from a browser-based interface.



**Note:** The announcement of IBM Service Management Suite for z/OS, V1.5.0 contains a statement of general direction that IBM intends to discontinue System Automation for z/OS support for collecting automation data using the Monitoring Agent and feeding that data into the Tivoli Enterprise Portal (TEP).

# Key operations features

- **Operations at the application level** lowers complexity
  - Automation takes care of dependencies and components
  - Status aggregation that includes health status
- **Goal driven automation** can reduce errors
  - Goals are not changed by IPL
- Replace consoles with a **single NetView console**
  - Dynamic **Status Display Facility** and powerful operator commands and dialogs
  - Event and status history and logs
  - **Single point of control**
  - Parallel sysplex-wide **single system image**
  - Convenient panel tailoring, sorting, and filtering capabilities
  - Automation flags to switch automation on or off
    - Scope can be system-wide down to single messages or transactions
- *Suspend and resume automation of applications*
- *System IPL complete notification*
- Web-based user interfaces
  - **Service Management Unite** (SA z/OS only)
  - Tivoli Enterprise Portal



© Copyright IBM Corporation 2019

1-13

## Key operations features

On this slide is a selection of key operations features in the automation product.

**Operations at the application level** means starting an application consisting of several components with one request.

**Goal driven automation** simplifies operations. System operators enter or remove a request at the application level. The automation product manages any dependencies and goals that are affected. It can resolve conflicting goals and handle goals even over restarts and IPLs.

## Replace your consoles with a single NetView console:

At any time, the operator can monitor and control from an enterprise-wide (SA z/OS only) single point of control at the application level. You have the choice of using a 3270-based NetView console or the graphical user interface of the Tivoli Enterprise Portal. Both are easy to use, powerful, and provide a common interface for all types of resources. When exceptions occur or goals must be changed, an operator just issues requests and overrides. SA z/OS offers a SAplex single system image, system boundaries are removed.

The single point of control provided by Automation Control for z/OS is limited to three systems in the sysplex.

The 3270-based NetView console includes the dynamic Status Display Facility (SDF) and powerful operator commands and dialogs as well as the netlog that shows all messages and WTOs that are sent to NetView.

Operators can tailor the layout of most display panels that are horizontally scrollable to pin or hide to determine order of columns and to sort by key in different directions and columns.

**Automation flags** allow to switch automation on or off, from a system-wide scope down to single messages or transactions.

**System IPL complete** notification.

**Suspend and resume resources** is a new and easy way to temporarily suspend automation for specific resources and their dependents without impacting the operations team by generating false alarms.

Operators can take resource in and out of automation easily.

# User interfaces

- Service Management Unite**
- 3270 based operator dialogs
- Status display facility (SDF)
- Tivoli Enterprise Portal

MVSA - SYSTEM STATUS SUMMARY

Resources

- >APPLS
- >GROUPS
- >MONITORS

Messages

- >WTOR
- >MESSAGES

Special Items

- >GATEWAY

CMD Name	Type	System	Sus	Compound	Desired	Observed
INGKYST0	SA z/OS - Command Dialogs					
Domain Id : ING02		INGLIST				
Operator Id : FREI						
A Update	B Start	C Stop	D INGRESL	E INGVOTE	F INGINFO	G INGINFO
H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTR	N INGINFO
R Resume	S Suspend	T INGTWS	U User	X INGWHY	Z scroll	
CMD Name	Type	System	Sus	Compound	Desired	Observed
A01B	APL	S1		PROBLEM	AVAILABLE	HARDDOWN
A02G	APL	S1		INHIBITED	AVAILABLE	SOFTDOWN
A03B	APL	S1		DIR	SATISFACTORY	AVAILABLE
A04G	APG	S1		INHIBITED	AVAILABLE	SOFTDOWN
A05B	APL	S1		DENIED	AVAILABLE	SOFTDOWN
A06B	APL	S1		SATISFACTORY	UNAVAILABLE	SOFTDOWN
B01B	APL	S1		INHIBITED	AVAILABLE	SOFTDOWN
B02B	APG	S1		SATISFACTORY	UNAVAILABLE	SOFTDOWN

© Copyright IBM Corporation 2019

## User interfaces

With system operations commands, you can control and maintain all of the resources SAplex-wide from a single point of control.

**IBM Service Management Unite** at the top is a new customizable dashboard user interface that can increase productivity of day to day operations even for expert users.

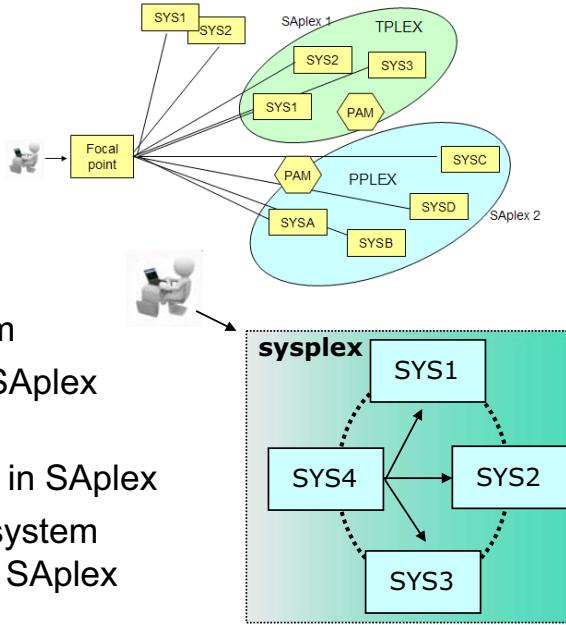
**SDF** to the left is the abbreviation of **Status Display Facility**. It consists of a set of hierarchical panels in NetView which are delivered as samples. The system administrator can also define user-specific panel layouts. If the status of an object changes, it is dynamically reflected on the panels and is propagated up in the hierarchy.

The **INGLIST command** at the bottom is available in NetView and provides details about all resources from the point of view of the automation manager. It displays information about a resource, such as statuses, flags, and schedules.

The **Tivoli Enterprise Portal** support allows you to monitor the status of automation on z/OS systems and z/OS sysplexes (SA z/OS only) using a TEP client. On the TEP workspaces, you get an overview on the resources with all their different states. The graphical summaries are combined with the detailed tabular views.

# Single Point of Control and Single System Image

- Automation commands can be sent to any connected system in the SAplex or enterprise
  - Responses automatically routed back to user



## Single System Image

- SAplex acts like a large single system
- Allows actions on subsystem within SAplex without specifying system name
- Host system is known on all systems in SAplex
- Selection panel displayed when subsystem runs on more than one system in the SAplex

© Copyright IBM Corporation 2019

1-15

### *Single Point of Control and Single System Image*

You can logon to any NetView and send automation commands to any connected system in the SAplex or enterprise by specifying the TARGET parameter. The responses are automatically routed back to the user. The INGLIST and DISPSTAT commands can even display resources from different systems in one panel.

**Single System Image** means that a SAplex acts like a large single system. It allows actions on subsystems within the SAplex without specifying the system name. Its host system is known on all systems in SAplex.

A selection panel is displayed when subsystem runs on more than one system in the SAplex.

# Lesson 3 Policy structure and resource types



## Lesson 3. Policy structure and resource types

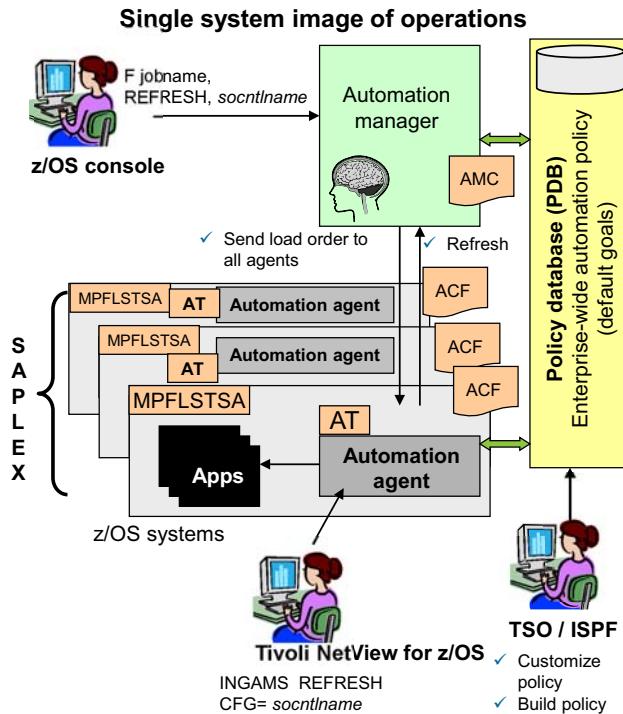


Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson introduces:

- The process from policy definition to policy build and policy activation
- The automation configuration file structure
- Operated resources

# Automation policy structure and activation



- Use the ISPF-based customization dialog to:
  - Define and customize an automation policy for systems, sysplexes, applications, groups...
- Run the build process to create these items:
  - Automation configuration file
    - Automation manager configuration (AMC) for managers
    - Automation control file (ACF) for agents
  - Automation table (AT)
  - MPF list member
  - Message Revision Table (MRT)
- Configuration file is loaded or refreshed with these commands:
  - The **INGAMS** command
  - MVS modify command

© Copyright IBM Corporation 2019

1-17

## Automation policy structure and activation

System Automation for z/OS operation is policy-based. Automation policy is defined through ISPF customization dialogs and saved in a policy database (PDB). A single PDB can hold automation policies for all SAplices in an enterprise.

The policy database provides the input to a build process, which creates the automation configuration file containing the applicable automation policies. System Automation for z/OS uses the automation configuration file to drive automation within the SAplice. There is one automation configuration file for each SAplice. All Automation control files (ACF) for the agents of an SAplice must be in the same data set. A stand-alone system is treated as a monoplex.

The message processing facility (MPF) list member is generated whenever a new configuration file is built. The list member can be added to the MPF list in the z/OS parmlib data set. Use of the MRT instead of MPF is also a good option.

The automation table (AT) and message revision table (MRT) can be built for the enterprise, a SAplice, or a single system.

The configuration file is loaded or refreshed with these commands:

- The INGAMS command
- MVS modify command against the automation manager

## Resource Types (partial list)

Code	Resource Type	Resource Description
APL	Application	A started task or other automatable entity, typically a started task
APG	Application Group	A group of resources that are defined and managed within the automation manager
MTR	Monitor	A resource that represents a monitor, sets and propagates the health status
SYS	System	A resource that represents a z/OS system
GRP	Group of systems	A standard or sysplex group of systems or <i>subgroups</i>
SYG	System group	A generated group containing all resources that are running on a system (not an APG!)
REF	<u>Resource reference</u>	Resource reference for End-to-end (E2E) automation
DMN	<u>Remote domain</u>	SAplex or non-z/OS domain hosting remote resources

Naming convention is: `resource_name/resource_type/system`  
Sysplex application groups do not specify the system

### Application (APL) Resources

The Application (APL) resources represent the individual address spaces (and other entities) that are to be automated. For example, you might define Application (APL) resources for JES, VTAM, and TSO.

### Application Group (APG)

Application Group (APG) is a group of Applications (APL), Resource references (REF), or Application Groups. It is used to manage the applications and to link them to systems. For an Application to be linked to a system, it must be a member of an Application Group that is linked to the system.

## Monitor (MTR)

The Monitor Resource (MTR) entry type allows you to obtain the health state of an object in two different ways:

- Actively, by polling, that is, executing a monitoring command periodically
- Passively, by processing events

Monitor resources are connected to application resources (APLs) or application group resources (APGs). The health status of the monitored object is propagated to the APLs or APGs and results in a health status there.

## System (SYS)

System (SYS) defines a single system that is to be automated. It includes definitions for automatically responding to events that happen on the system. Connect the Application Groups to it to instruct automation that it must run the resources in the APG on the System.

## Group (GRP) or SAplex

*Group (GRP)* is a collection of one or more systems (SYS). While Groups are required for most functions to automate systems, they are not required for the single-system automation model used by Automation Control.

For SA z/OS:

- a Group of type *SYSPLEX* contains only systems or *subgroups* that belong to the same XCF group, representing a logical sysplex, also called **SAplex**.
- a Group of type *STANDARD* is used for all other Groups, for example for documentation purposes or to represent a physical sysplex (a group of systems within different XCF group IDs, such as a GDPS site).
- A SubGroup Entry Type can be used for intermediary grouping of the systems within a group. Typical uses are for ProcOps or for documentation purposes.

## SYG

Indicates the group of all resources of a system. SYG is created automatically. This is the application group that contains all resources of a particular system.

## Resource reference (REF)

This entry type is used for end-to-end automation and defines a reference to a resource running on a remote domain (for example another z/OS sysplex or SAplex).

### Remote domain (DMN)

This entry type defines a remote domain (for example another z/OS sysplex or SAplex) where resources are running which are in the scope of end-to-end automation.

#### Naming convention

Naming convention is:

resource\_name/resource\_type/system

Sysplex application groups do not specify the system

## Lesson 4 The role of the operator



### Lesson 4. The role of the operator



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson introduces the role of the operator which is to use automation to execute changes and to detect problems in the automated environment.

## The role of the operator



- Control automation
- Detect problem statuses, messages, and alerts  
- Manage resources 
- Create or override goals 
- Problem analysis 
- Additional tasks

### Lesson agenda

- Control automation
- Detect problem statuses, messages, and alerts
- Manage resources
- Create or override goals to start, move or stop resources
- Analyze problems
- Additional tasks like managing timers, checking system, sysplex management and more



# The role of the operator

- Control automation
  - Load a automation configuration file
  - Manage automation managers
  - Set a runmode
- Detect problem statuses, messages, and alerts
- Manage resources
  - Set status, for example to return resource to automation
  - Suspend and resume resources
  - Set automation flags
- Start, move or stop resources through goals
  - Create, delete, or modify requests, override service periods
  - Set next start type
  - Set availability targets
  - Set triggers, events
- Analysis of
  - Automation decisions
  - Resource policy, statuses, history...
  - Automation log and syslog
- Additional tasks: Manage timers, sysplex, commands



HardDown



*The role of the operator: Control, monitor, manage, analyze...*

The operator role is to control automation like loading or refreshing a automation configuration file, switching roles of automation managers, setting runmodes...

Operators should use their user interface of choice to detect and react on problem statuses, messages, and alerts. This can involve analysis of automation decisions, resource policy, statuses, history, and messages as well as of the automation log and syslog.

For example, when a resource problem is fixed after a harddown status, the operator can set the status to a softdown status to return the resource to automation.

Execution of change requests also can require to suspend or resume resources, to create or override goals using requests or service periods, or group availability targets.



# Control automation

- Update policy including timers, AT, and MRT for a SAplex
  - Load a automation configuration file dynamically using the INGAMS or modify command
  - Start or restart automation manager with a new configuration file using any start type
  - Temporary policy change on a single system can be done with ACF command
- Manage automation managers
  - Promote to primary automation manager
  - Start secondary automation managers
  - Fix takeover file in case of I/O problems
- Set a runmode for a system
  - Dynamically using the INGRUN command
  - During IPL or agent restart by responding to AOF603D WTOR

## Control automation

### Update policy including timers, AT, and MRT for a SAplex:

- Load a automation configuration file dynamically using the INGAMS command or z/OS modify command
- Start or restart automation manager with a new automation configuration file using any start type
- Temporary policy change on a single system can be done with the ACF command

### Manage automation managers

- Start secondary automation managers. This is usually automatic using a sysplex server group
- If you need to move the primary automation manager to another system, promote a secondary automation manager there to primary automation manager
- In case of I/O problems, use INGAMS DISABLE to stop the automation manager to write to the takeover file. It also deallocates the takeover file from the automation manager

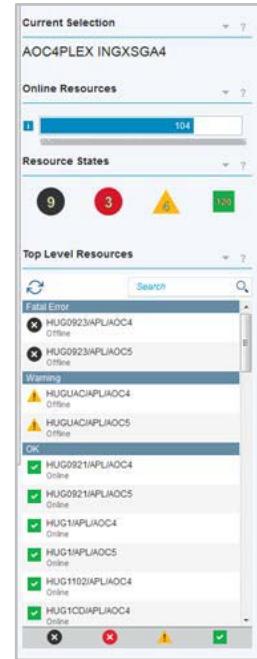
### Set a runmode for a system

- Dynamically using the INGRUN command
- During IPL or agent restart by responding to AOF603D WTOR

# Detect problem statuses, messages, and alerts



- Available user interfaces dependent on your implementation:
  - Status Display Facility (Dynamic): Colored states and details...
  - **Service Management Unite** (SA z/OS only)
  - SA INGLIST dialog in NetView (Refresh required): Watch compound states
  - NetView Notify operators
    - Can be sent to the MCS console when no operator is logged on
- Escalation
  - Send email using **NetView INFORM policy**
  - Send event to Netcool OMNIbus or other Event Integration Facility (EIF) receiver
  - Send SMS text messages with confirmation using System Automation for Integrated Operations Management (SA IOM)
  - Send event to the Tivoli Enterprise Portal
  - User defined escalation



© Copyright IBM Corporation 2019

1-23

*Detect problem statuses, messages, and alerts*

Operators should use their user interface of choice to detect and react on problem statuses, messages, and alerts. Available user interfaces dependent on your implementation:

- Status Display Facility (Dynamic): Colored states and details...
- **Service Management Unite** (SA z/OS only)
- SA INGLIST dialog in NetView (Refresh required): Watch compound states
- NetView Notify operators
  - Can be sent to the MCS console when no operator is logged on

If you don't want to watch a console, the following escalation options can be implemented:

- Send email using NetView INFORM policy
- Send event to Netcool OMNIbus or other Event Integration Facility (EIF) receiver
- Send SMS text messages with confirmation using System Automation for Integrated Operations Management (SA IOM)
- Send event to the Tivoli Enterprise Portal
- User defined escalation

# Manage resources

- Automation is broken
  - Analyze why automation is broken: ABEND code, threshold...
  - Problem must be fixed
  - Return resource to automation: Set agent status to AUTODOWN
- Suspend and resume resources as requested by owner or change
  - Use command INGSUSPD to suspend and resume automation for specific resources (APGs, APLs, REFs, and MTRs)
  - RESUME automation restores the normal behavior (**except Job Log Monitoring**)
- Set or unset automation flags temporarily for major and minor resources
  - Example: Stop automation for a message or transaction or MVS component
  - Not recommended for APLs as it blocks goal driven automation
  - Use INGAUTO command



HardDown



Manage resources

Typical resource management tasks are:

Automation is broken

1. Analyze why automation is broken: ABEND code, threshold...
2. Problem must be fixed
3. Return resource to automation: Set agent status to AUTODOWN

Suspend and resume resources as requested by owner or change: Use command INGSUSPD to suspend and resume automation for specific resources (APGs, APLs, REFs, and MTRs).

RESUME automation restores the normal behavior (except Job Log Monitoring)

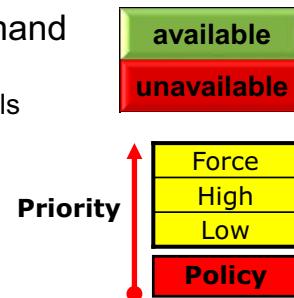
Set or unset automation flags temporarily for major and minor resources using the INGAUTO command. Using automation flags is NOT recommended to stop automation of APLs as it blocks goal driven automation.

Example: Stop automation for a message or transaction or MVS component.

## Start, move or stop resources



- Create, delete, or modify requests using INGREQ command
  - For start, just remove any make unavailable goals
  - Use low priority first to avoid affecting other, possible important goals
  - If your vote is not winning, investigate reasons using INGWHY.  
If you MUST achieve your goal, use force priority
  - Use a comment to tell your colleague why
  - Specify expiration, if known
- Override service periods either for single resource or all resources linked to a service period
  - Cold start will forget any overrides
- Move resources using INGMOVE command
- Set next start type using INGSET command
- Modify startup or shutdown using INGMDFY command
- Set group availability targets using INGGROUP command
- Warm start will forget any goal changes and all status about E2E environment (DMN and REF) since last policy refresh
- Cold start will also forget any service period overrides



*Start, move or stop resources*

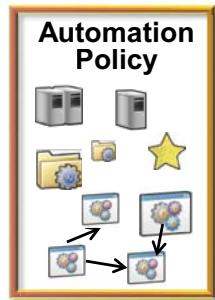
- **Create, delete, or modify requests** using INGREQ command.  
For start, just remove any make unavailable goals:
  - Use low priority first to avoid affecting other, possible important goals
  - If your vote is not winning, investigate reasons using INGWHY.  
If you MUST achieve your goal, use force priority
  - Use a comment to tell your colleague why
  - Specify expiration, if known
- **Override service periods** either for single resource or all resources linked to a service period
- **Set group availability targets** using INGGROUP command
- **Move resources** using INGMOVE command
- **Set next start type** using INGSET command
- Modify startup or shutdown using INGMDFY command
- **Warm start** will forget any goal changes and all status about E2E environment (DMN and REF) since last policy refresh since last policy refresh

- **Cold start** will also forget any service period overrides



## Problem analysis

- Analyze automation decisions and problems
  - If order is not sent, check for inhibitors using INGWHY command and resolve:
  - Check relationships using INGRELS, DISPTREE commands or SMU
  - Check for problem statuses, flags, dependencies and captured messages using INGINFO (manager) and DISPINFO (agent) commands
  - Set events for triggers using INGEVENT command
  - Check history using INGHIST command and automation log and syslog
  - Display resource errors recorded in the status file using DISPERRS command
  - Use ASF to display records from the automation status file
- Health status: For monitors use DISPMTR command
- Review resource policy
  - Use commands INGINFO and DISPINFO, DISPAPG for groups
  - Use command ACF REQ=DISP ENTRY=abc
  - Use policy reports



© Copyright IBM Corporation 2019

1-26

### Problem analysis

Analyze automation decisions and problems. If order is not sent, check for inhibitors using INGWHY command and resolve:

- Check relationships using INGRELS, DISPTREE commands or SMU
- Check for problem statuses, flags, dependencies and captured messages using INGINFO (manager) and DISPINFO (agent) commands
- Set events for triggers using INGEVENT command
- Check history using INGHIST command and automation log and syslog
- Display resource errors recorded in the status file using DISPERRS command
- Use ASF to display records from the automation status file

Health status: For monitors use DISPMTR command

To review resource policy

- Use commands INGINFO and DISPINFO, DISPAPG for groups
- Use command ACF REQ=DISP ENTRY=abc
- Use policy reports



# Additional tasks

- Manage timers temporarily using SETTIMER command
- Check SYSTEM status using DISPSYS command
  - IPL history and PARMLIB comparison using the INGPLEX IPL command
- Sysplex management
  - Couple data sets using the INGPLEX CDS command
  - Coupling facilities and structures using the INGCF and INGSTR commands
- Send commands to selected systems using INGSEND command
- Display all jobs that are not controlled by SA z/OS using INGLKUP
- Drain JES2 using DRAINJES command
- Manage CICS, DB2 or IMS using INGCICS, INGDB2 or INGIMS commands
- IBM Workload Scheduler integration supported by INGTWS command
- Display automation operators, operators that receive automated messages, gateways, consoles, dumps, SLIP traps, flags, events, triggers, thresholds, view filters...

## Additional tasks

- Manage timers temporarily using SETTIMER command
- Check SYSTEM status using DISPSYS command. The IPL history and PARMLIB comparison can be done using the INGPLEX IPL command
- Sysplex management
  - Couple data sets using the INGPLEX CDS command
  - Coupling facilities and structures using the INGCF and INGSTR commands
- Send commands to selected systems using INGSEND command
- Display all jobs that are not controlled by SA z/OS using INGLKUP
- Drain JES2 using DRAINJES command
- Manage CICS, DB2 or IMS using INGCICS, INGDB2 or INGIMS commands
- IBM Workload Scheduler integration supported by INGTWS command
- Display automation operators, operators that receive automated messages, gateways, consoles, dumps, SLIP traps, flags, events, triggers, thresholds, view filters...

# Lesson 5 Operations using NetView for z/OS



## Lesson 5. Operations using NetView for z/OS



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to use NetView for z/OS operations.

It provides a NetView overview, a description of the NetView automation platform, the NetView console and the following logs

- CANZLOG: Consolidated Audit, NetView & z/OS Log
- Netlog: Automation log

## Operations using NetView for z/OS

---

- NetView overview
- NetView for z/OS, the automation platform
- NetView, the better console
  - Using NetView from a 3270 Session
  - CANZLOG: Consolidated Audit, NetView & z/OS Log
  - Using NetView log

## NetView overview

---

NetView® for z/OS provides functions to help maintain the highest degree of availability for IBM System z networks.

It provides:

- An extensive set of tools for managing complex, multi-vendor, multi-platform networks and systems from a single point of control
- Support for both IP and SNA networks
- Advanced automation facilities for events from many sources
- Enhanced operations and message management support
- A set of user interfaces to meet the needs of any user and management functions that work with other products to provide a complete picture of your networks and systems

---

30

*NetView overview*

The NetView for z/OS program provides functions to help maintain the highest degree of availability for IBM System z® networks. It offers an extensive set of tools for managing and maintaining complex, multi-vendor, multi-platform networks and systems from a single point of control. It provides advanced correlation facilities to automate any network or system event and provides support for both TCP/IP and SNA networks. It also provides a set of user interfaces to meet the needs of any user and management functions that work with other products to provide a complete picture of your networks and systems.

NetView has enhanced operations and message management support to improve and simplify operator interactions and to provide more control in automating and managing day-to-day operations

## NetView for z/OS, the automation platform

NetView for z/OS has extensive automation facilities:

- Automation table (IF ... THEN ....)
- Message revision table to revise messages
- Command revision table to revise commands
- Automation tasks for isolation and load distribution
- Timers
- Unlimited sources and targets of automation
- Rexx scripts with automation function packages
- Powerful pipes instead of scripts

You can automate from:

- Messages (VTAM, z/OS, JES, NetView, user, etc.) - solicited and unsolicited
- Alerts (SNA and non SNA)
- Full screen applications, SNMP traps, TEC events

NetView for z/OS has extensive automation facilities:

- **Automation table** (IF ... THEN ....)
- **Message revision** table to revise messages. This function enables user-defined modification of attributes such as color, route code, descriptor code, display and syslog settings, and text of original z/OS messages (rather than copies).

The message revision table can override actions taken by the z/OS message processing facility (MPF) and can generally replace the MPF. It also provides statistics and usage information, includes a test mode, and is active even when the NetView program is not. Finally, the message revision table is under the control of the NetView system programmer rather than the z/OS

system programmer, thus simplifying the administration of message processing.

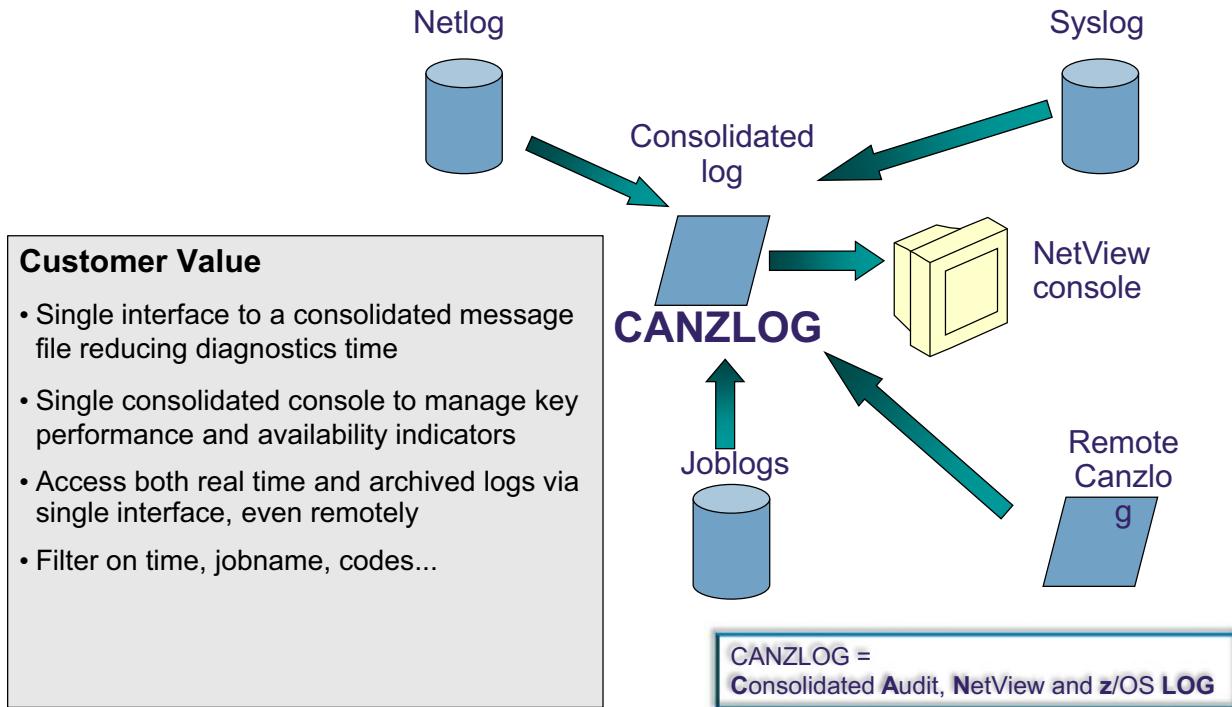
For example, you can take the following actions:

- Revise messages before they are presented to the system log, console or automation.
- Treat a message differently depending on its source.
- Suppress messages entirely.
- Automate only.
- The message revision
- **Command revision** table to revise commands. This function enables user-defined modification of z/OS commands without needing to transfer the commands to the NetView application address space. Commands can be deleted; parameters and keywords can be added, removed, or modified; nicknames can be expanded (such as creating new command or parameter synonyms); and explanatory WTO messages can be issued. Command revision supersedes the MVS command management function in the NetView program.
- **Automation tasks** for isolation and load distribution. These are operator station tasks (OSTs) that do not require a terminal or operator. Like other OSTs, autotasks can receive messages and issue commands. Autotasks are limited only in that they cannot run full-screen applications. You can define one or more autotasks for automation and have them started during NetView initialization. Then, the automation table, command lists, command processors, and timer commands can all issue commands under the autotasks. The autotasks can receive messages and present them to the automation table or to installation-exit routines. Thus, many of the other automation facilities can use autotasks.
- **Timers:** These initiate automated actions. Both operators and automation procedures can issue timer commands to schedule other commands, command lists, and command processors at a specified time, after a specified delay, repeatedly after specified intervals or in complex, timed combinations.
- Unlimited sources and targets of automation
- **Rexx** scripts with automation function packages. These are user-written programs that can be used as if they are NetView commands. A **command list** or **command processor** can be used by an operator to accomplish a complex operation with a single command or can perform an entire, complex procedure without operator intervention.
- Powerful pipes instead of scripts

You can automate from:

- Messages (VTAM, z/OS, JES, NetView, user, etc.) - solicited and unsolicited
- Alerts (SNA and non SNA)
- Full screen applications, SNMP traps, TEC events

# CANZLOG: Consolidated Audit, NetView & z/OS Log



© Copyright IBM Corporation 2019

1-32

## CANZLOG: Consolidated Audit, NetView & z/OS Log

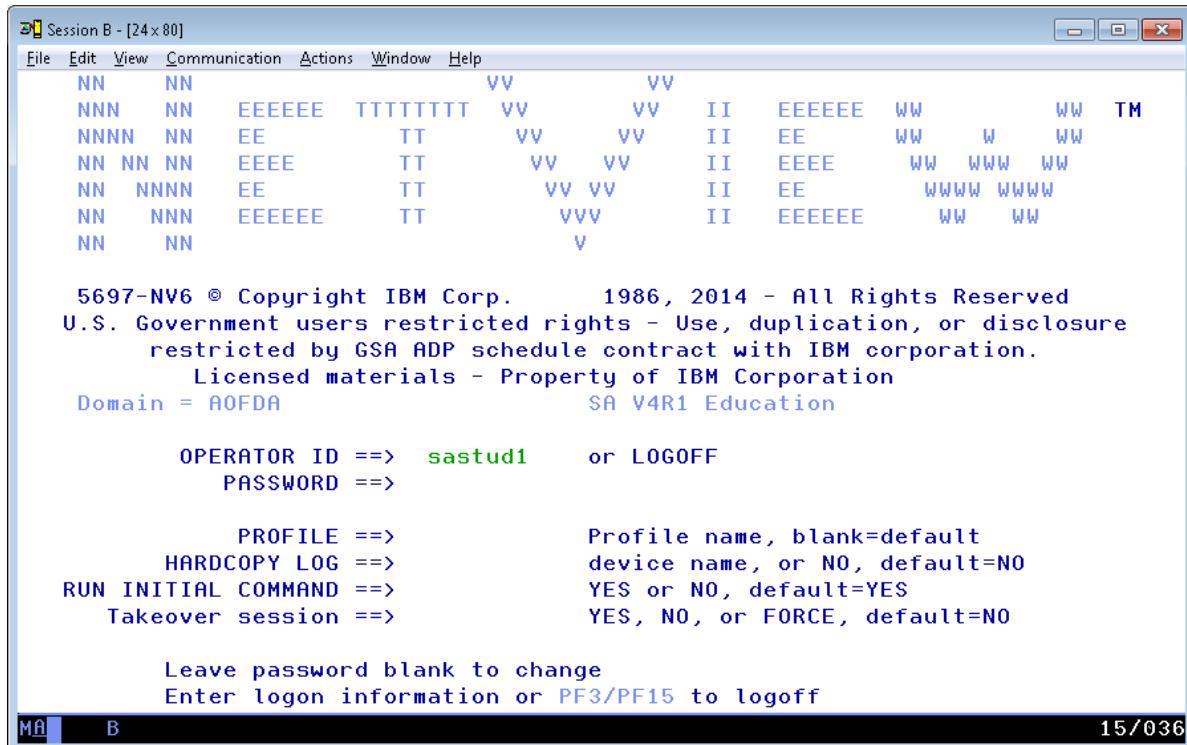
The consolidated audit, NetView, and z/OS log (Canzlog) provides a comprehensive means of displaying the following kinds of information:

- MVS messages. This is similar to the SYSLOG function, but, unlike SYSLOG, logging on to TSO is not necessary
- NetView messages. This is similar to the NETLOG function
- Broadcast messages
- DOMS
- Command echoes
- Trace and audit messages

CANZLOG provides:

- Single interface to a consolidated message file reducing diagnostics time
- Single consolidated console to manage key performance and availability indicators
- Access both real time and archived logs via single interface, even remotely
- Filter on time, jobname, codes..., see example later

# Logon to NetView



© Copyright IBM Corporation 2019

1-33

## Logon to NetView

To establish a session with the NetView program, enter the following command, where applid is the name of the NetView application to which you are logging on. LOGMODE and DATA are optional parameters, logmode specifies information about your terminal session, and data specifies information that is inserted in the OPERATOR ID and PASSWORD fields of the NetView logon panel:

```
logon applid(applid) logmode(logmode) data(data)
```

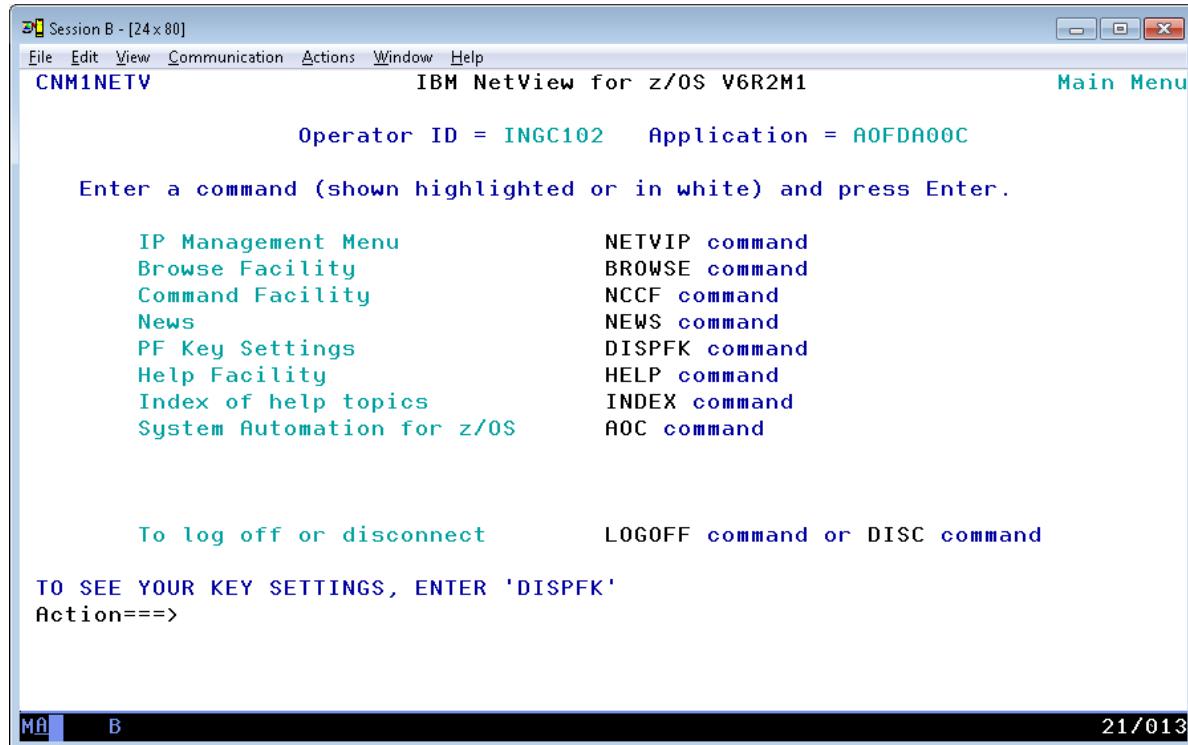
When you log on, the NetView program queries the device for screen size and color attributes if the logmode specifies to issue the query. Otherwise, the NetView program uses the screen size specified in the logmode. The command facility adapts to use the entire width and depth of the screen.

1. Type your operator identification (for example, OPER1) in the space next to the OPERATOR ID field, where the cursor is located.
2. Enter your password or password phrase next to the PASSWORD field.
3. You can override your default profile member in the DSIPRF data set using the PROFILE ==> field. If operators are defined in an SAF product, you cannot enter a profile value because no

PROFILE field is displayed. Instead, the operator attributes are specified in the NETVIEW segment of the SAF product.

4. If you do not want to use an initial command, type no in the RUN INITIAL COMMAND field. If you want to use an initial command, leave this field blank or type yes. The initial command is defined in your profile or NETVIEW segment and set up by your system programmer to eliminate some manual procedures like setting the PF-keys.
5. If the operator ID is already logged on and you want to take over the session, enter YES as the takeover value.
6. Press Enter. If you left the PASSWORD field blank and the NetView program is using an SAF product such as RACF to check passwords or password phrases, you can change your password in the following panel.
7. Press the Clear or Enter key to clear the screen and go to the NetView Main Menu. After the NetView program processes the operator profile, the following panel is displayed:

## NetView mainmenu



### NetView mainmenu

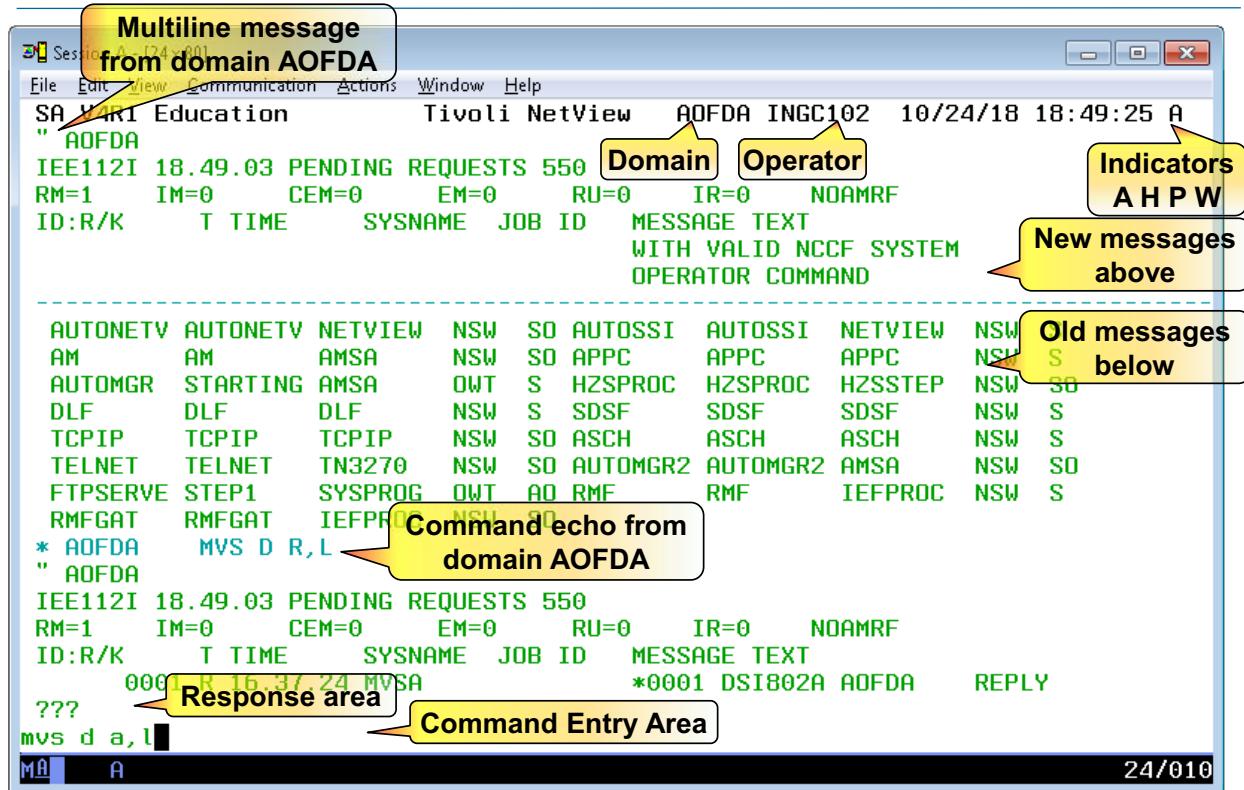
After logging on you see the NetView Main Menu displayed by the **NetView Command Facility (NCCF)**, a core NetView component. The command facility is used to send commands and receive messages. It also provides base functions and services for other components, such as intercomponent communication, presentation services, database services, and automation facilities.

The NetView Main Menu automatically recognizes whether an option on the menu is active or inactive. The NetView Main Menu displays only active options. For example, if the Automated Operations Network and Session Monitor are not active, those options are not displayed on the menu.

If a command on the NetView Main Menu is backlit, it is only partially available. That means that some functions are available using the command, but not all functions.

The first line of the panel shows the panel name to the left.

## NetView Console



### NetView Console

The first line of the NetView console shows two customizable names.

The next field lists the application identifier or *domain* (AOFDA) and your *operator* identifier (INGC102). The next two fields list the current date and time.

The last two fields contain a combination of A, H, P, W, or a blank, which indicates whether messages can be written to the panel. The *A, H, P, and W indicators* are described in the following list:

- A The *autowrap* indicator means that AUTOWRAP is active. If autowrap is on and the display is full of data, it is automatically overlaid with new data. If autowrap is not on, press the Clear or Enter key to allow new data to overlay the display screen.
- H The *held-screen* indicator means that the screen does not roll forward unless it is unlocked by the operator. You can use this indicator if you need time to read the screen before it is erased, or to freeze the screen while you mark messages for deletion or enter a command.
- P The *pause* status indicator. A command list running on the operator task is pausing for operator input and does not continue until the operator enters information.
- W The *wait* indicator. A command list running on the operator task is waiting for messages or other events, such as for a specified amount of time to elapse.

The dashed line separates the latest messages from the older ones. The messages are continually updated. You can use this line to locate the most recent messages. The most recent messages are the ones directly above the line. The oldest messages displayed on the screen are at the bottom of the screen, below the line.

To rearrange the messages on the screen, press the Enter key. This redisplays the messages in sequential order and removes the dashed line.

Generally, messages are no longer displayed as the screen scrolls. Examples of exceptions include *reply messages*, *held messages*, and *windowed responses*.

Near the bottom of the screen is a line that begins with the ??? indicator. This line is the *response area*. Look here for error messages.

Between the response area and the first line of the panel is the message area that displays commands, responses, and messages from the system.

Most NCCF messages have the following format:

```
type domid code msgno text
```

Where:

- type  
Message type. For more information about message type symbols, refer to HDRMTYPE in the DSITIB macro.
- domid  
Domain or application of the message origin
- code  
Message Code
- msgno  
Message number you can use to look up more information using the online help.
- msgtext  
Text of the message

At the bottom is the Command Entry Area. Using the *input* command you can increase the command entry to a maximum of three lines. When you press any action key (Enter, PF, PA, or Clear), the command area is erased. You can use *program function (PF)* or *program access (PA)* keys to send commands to the system.

You can have multiple components active at the same time. Use the *ROLL* function to move among active components in a continuous loop. The PF key that is supplied by the NetView product for ROLL is PF6.

If you are in a component other than command facility with a panel displayed you can be interrupted by a message from another component. This message is displayed on the command facility screen. After the message is displayed, the NetView program displays \*\*\* at the bottom of the command facility screen. You can press Enter to return to the panel you were using when the interrupt occurred.

# Frequently used NetView commands

- **No prefix** NetView command or program
- **MVS** MVS or other system command
- **USS** USS command
- **TSO** TSO command
- **BR item** Browse NETLOGA, NETLOGI, any member
- **WHO** Lists NetView users logged on
- **WRAP** Toggle AUTOWRAP
- **BLOG** Browse the log using filters
- **ROLL** Move to next active panel in a continuous loop (PF6)
- **HELP cmd** Display help for command cmd (PF1)
- **WINDOW** Command prefix for windowed responses (PF10)
- **Netvasis** Command prefix for using lower case (alternative override)
- **INPUT n** Set the input area to n lines
- **LISTA dd** List allocated DD names and data sets
- **LIST STATUS=TASKS** List all active tasks (PF22 lists all inactive tasks)
- **TASKUTIL** Display CPU and storage use for NetView tasks
- **AUTOTBL status** Display AT status

## *Frequently used NetView commands*

- No prefix: NetView command or program
- MVS: MVS or other system command
- USS: USS command
- TSO: TSO command
- BR item: Browse NETLOGA or NETLOGI, or any member
- WHO: Lists NetView users logged on
- WRAP: Toggle AUTOWRAP
- BLOG: Browse the log using filters
- ROLL: Move to next active panel in a continuous loop (PF6)
- HELP cmd: Display help for command cmd (PF1)
- WINDOW: Command prefix for windowed responses (PF10)
- Netvasis: Command prefix for using lower case (an alternative is the override command)
- INPUT n: Set the input area to n lines

- LISTA dd: List allocated DD names and data sets
- LIST STATUS=TASKS: List all active tasks (PF22 lists all inactive tasks)
- TASKUTIL: Display CPU and storage use for NetView tasks
- AUTOTBL status: Display AT status

## Long output > window

```
CNMKWIND OUTPUT FROM LIST STATUS=TASKS LINE 71 OF 106
TYPE: OST TASKID: AUTNET1 RESOURCE: AUTNET1 STATUS: ACTIVE
TYPE: OST TASKID: AUTNET2 RESOURCE: AUTNET2 STATUS: ACTIVE
TYPE: OST TASKID: AOFARCAT RESOURCE: AOFARCAT STATUS: ACTIVE
TYPE: OST TASKID: AUTREC RESOURCE: AUTREC STATUS: ACTIVE
TYPE: OST TASKID: AUTRPC RESOURCE: AUTRPC STATUS: ACTIVE
TYPE: OST TASKID: AUTSHUT RESOURCE: AUTSHUT STATUS: ACTIVE
TYPE: OST TASKID: AUTSYS RESOURCE: AUTSYS STATUS: ACTIVE
TYPE: OST TASKID: AUTXCF RESOURCE: AUTXCF STATUS: ACTIVE
TYPE: OST TASKID: AUTXCF2 RESOURCE: AUTXCF2 STATUS: ACTIVE
TYPE: OST TASKID: AUTPLEX RESOURCE: AUTPLEX STATUS: ACTIVE
TYPE: OST TASKID: AUTPLEX2 RESOURCE: AUTPLEX2 STATUS: ACTIVE
TYPE: OST TASKID: AUTPLEX3 RESOURCE: AUTPLEX3 STATUS: ACTIVE
TYPE: OST TASKID: AUTEVT1 RESOURCE: AUTEVT1 STATUS: ACTIVE
TYPE: OST TASKID: AUTEVT2 RESOURCE: AUTEVT2 STATUS: ACTIVE
TYPE: OST TASKID: GATAOFDA RESOURCE: GATAOFDA STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK01 RESOURCE: AUTWRK01 STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK02 RESOURCE: AUTWRK02 STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK03 RESOURCE: AUTWRK03 STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK04 RESOURCE: AUTWRK04 STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK05 RESOURCE: AUTWRK05 STATUS: ACTIVE
TYPE: OST TASKID: AUTWRK06 RESOURCE: AUTWRK06 STATUS: ACTIVE
BNH183I CURRENT SCROLL VALUE IS 9
CMD==> ■
MA B 24/009
```

© Copyright IBM Corporation 2019

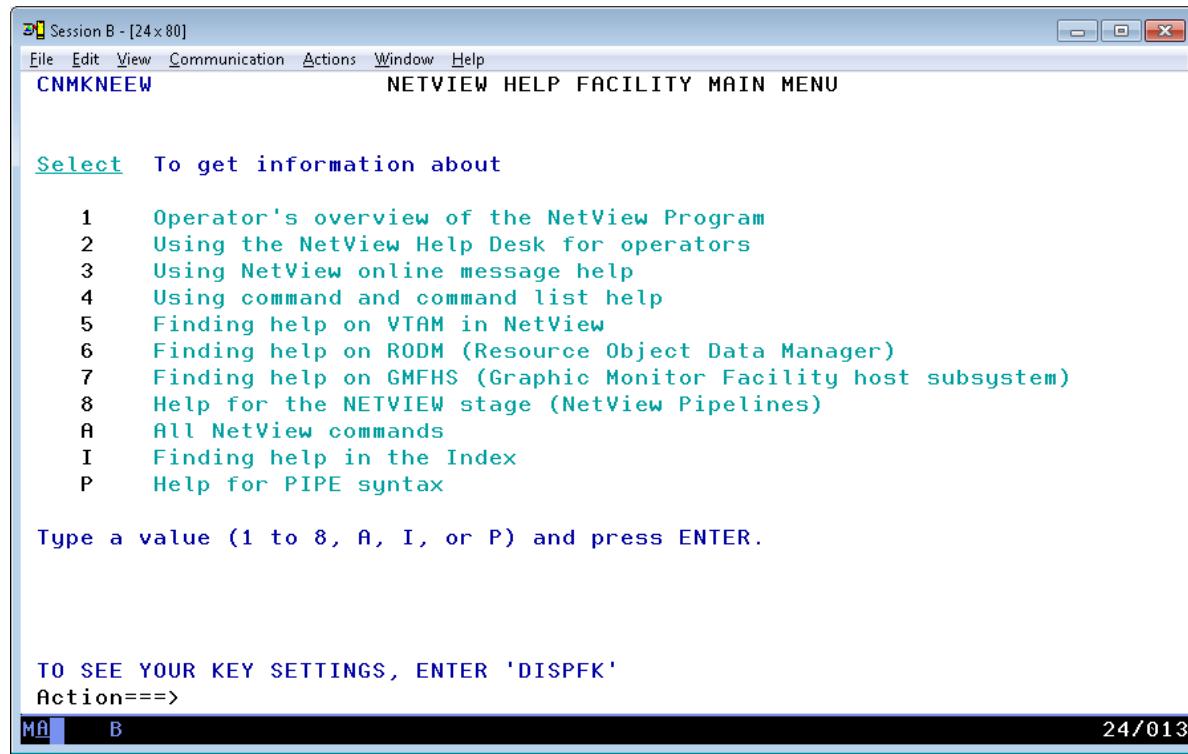
1-37

### Long output > window

Windowed responses are messages that are displayed in a scrollable window using the NetView WINDOW command. This prevents the message responses from being overwritten by subsequent messages; you can also navigate through the information using standard BROWSE commands.

You can window responses by prefixing the command with `window`, which is usually PF10. Now instead of looking at the output in line mode you can view it captured in a full screen window. If it is multiple pages you can scroll using F7 and F8. You can use FIND commands and the ALL command which I will explain more about in one second. Try this command, `uss onetstat` but hit F10 instead of ENTER. Now more about the ALL command. It is kind of a filter mixed with a find. Let's say you are looking at the output from the `onetstat` command but you are only interested in ports associated with `inetd`. On the command line issue, `all inetd`. You have now successfully filtered out only the `inetd` owned ports. Now type `all` again just by itself. Everything is back. We will explore more uses for the ALL command when we talk about logs and browsing but one last thing about WINDOW. Hit F2 and the command is reissued and a fresh set of results is displayed.

## NetView help



The screenshot shows a terminal window titled "Session B - [24x80]". The title bar includes standard menu options: File, Edit, View, Communication, Actions, Window, and Help. The main title "NETVIEW HELP FACILITY MAIN MENU" is displayed above the menu bar. The menu bar also shows the session identifier "CNMKNEEW". The window content displays a list of help topics:

```
Select To get information about
1 Operator's overview of the NetView Program
2 Using the NetView Help Desk for operators
3 Using NetView online message help
4 Using command and command list help
5 Finding help on VTAM in NetView
6 Finding help on RODM (Resource Object Data Manager)
7 Finding help on GMFHS (Graphic Monitor Facility host subsystem)
8 Help for the NETVIEW stage (NetView Pipelines)
A All NetView commands
I Finding help in the Index
P Help for PIPE syntax

Type a value (1 to 8, A, I, or P) and press ENTER.
```

Below the list, there is a note: "TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'". The bottom of the window shows the prompt "Action==>" followed by the letters "MA" and "B". In the bottom right corner, the date "24/013" is visible.

© Copyright IBM Corporation 2019

1-38

### *NetView help*

This is the NetView help facility main menu. You can display it by entering: `help netview`

The following NetView for z/OS mainframe online help is available, depending on your installation and configuration:

- General help and component information
- Command help
- Message help
- Sense code information
- Recommended actions
- Help desk

## Netlog filtered for ERROR

```
STATMON.BROWSE ACTS NETWORK LOG FOR 09/27/18 (18270) COLS 017 094 22:25 A
DOMAIN: AOFDA SCROLL ==> CSR
---2---+---3---+---4---+---5---+---6---+---7---+---8---+---9---
AOFDA 17:58:42 " more potential errors in the security controls on this sys
AOFDA 21:58:50 " more potential errors in the security controls on this sys
AOFDA % 00:31:00 E IEFC452I FRUSAG - JOB NOT RUN - JCL ERROR
AOFDA % 00:31:00 E IEE122I START COMMAND JCL ERROR
AOFDA 01:37:28 " configuration error.
AOFDA 01:37:28 " configuration error.
AOFDA 01:58:58 " more potential errors in the security controls on this sys
AOFDA 05:59:05 " more potential errors in the security controls on this sys
AOFDA 09:59:13 " more potential errors in the security controls on this sys
AOFDA P 22:22:47 - DSI064A OPENACB FAILED, ACBOFLG = X'02', ACBERROR = X'5A',
AOFDA 22:22:51 - DSI556I DSILOG : VSAM DATASET 'OPEN' COMPLETED, DDNAME =
AOFDA 22:22:52 - DSI556I DSILOG : VSAM DATASET 'OPEN' COMPLETED, DDNAME =
AOFDA 22:22:55 - DSI556I DSISVRT : VSAM DATASET 'OPEN' COMPLETED, DDNAME =
AOFDA 22:23:31 U AOF869I 1 ERROR(S) ENCOUNTERED PROCESSING MEMBER ACFZ999
AOFDA 22:23:31 U AOF791I AUTOMATION CONTROL FILE PROCESSING COMPLETED WITH
AOFDA % 00:31:00 E IEFC452I FRUSAG - JOB NOT RUN - JCL ERROR
AOFDA % 00:31:00 E IEE122I START COMMAND JCL ERROR
AOFDA % 00:31:00 E IEFC452I FRUSAG - JOB NOT RUN - JCL ERROR

CMD==> █
TO SEE YOUR KEY SETTINGS, ENTER 'DISPFK'
MA B 23/009
```

© Copyright IBM Corporation 2019

1-39

### Netlog filtered for ERROR

The network log is the automation log. It contains commands, responses, automation table matches, and messages from z/OS or from inside NetView.

The log is really a pair of VSAM files that flip from one to the other as they fill up. Data in the inactive log remains available for browsing until it becomes active again at which point all data is erased. There is a batch utility provided for copying the inactive log to a sequential file for archiving.

To browse the log the most common command is BR NETLOGA. BR is an abbreviation for browse and NETLOGA is the active log. BR NETLOGI will allow you to browse the inactive log which will contain the old data from the previously active log. Common commands like FIND, FIND P or FIND PREV, and ALL xxx are also supported. You can also use M for max in conjunction with F7 and F8. You can quickly identify a problem by entering ALL IEF450. This will filter out only the job abend messages.

Another way to browse the log is to use the BLOG command:

The BLOG command activates the network log browse facility based on filters. You can select which records to display using any combination of filters like message ID or text and more. Here I have filtered for messages containing ERROR.

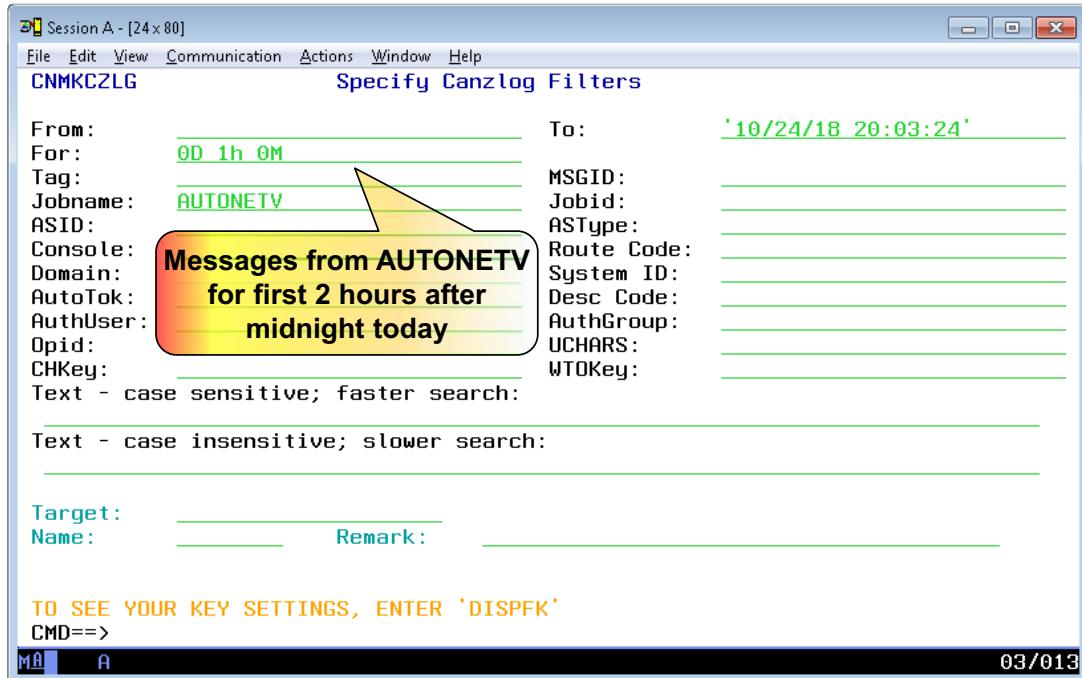
Most network log messages have the following format:

```
number taskid domid code time type text
```

Where:

- number: The sequential serial number of NETLOG. This screenshot starts at column domid.
- taskid: Generally the ID of the logging operator or task, but, if applicable, can be another name such as an SAF user ID.
- domid: Generally the originating domain, but, if applicable, can be another name such as an SAF user ID, a PPI name, a PDS member, or a TAF session ID. These names can be up to 8 characters and can therefore overwrite the code field, which otherwise is the next field.
- code: If the previous field is a domain ID, which is 5 or fewer characters, this field is the code (see “Message Codes” on page 284). The domid field can cause this field to be overwritten or missing.
- time: The time that the task logged the message.
- type: Message type. For more information about message type symbols, refer to HDRMTYPE in the DSITIB macro.
- msgno: Message number you can use to look up more information using the online help.
- msgtext: Text of the message.

## CANZLOG relative time specification



© Copyright IBM Corporation 2019

1-40

### CANZLOG relative time specification

The consolidated audit, NetView, and z/OS log (Canzlog) provides powerful filters. Here I am using a relative time specification to display messages from job AUTONETV for the first 2 hours after midnight today.

## Student exercise

---



---

© Copyright IBM Corporation 2019

1-41

*Student exercises*

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you completed this unit, you can perform the following tasks:

- Describe the basic components of System Automation for z/OS
- Describe the automation architecture
- Describe typical policy resource types and the configuration file structure
- Describe key automation and operations features
- Describe the role of the operator and typical tasks
- Operations using NetView for z/OS

### *Summary*

Now that you completed this unit, you can perform the following tasks:

- Describe the basic components of System Automation for z/OS
- Describe the automation architecture
- Describe typical resource types the policy database
- Describe key automation and operations features
- Describe the role of the operator and typical tasks
- Operations using NetView for z/OS



## 2 Initialization

---

IBM System Automation for z/OS 4.1



## Unit 2: Initialization



© Copyright IBM Corporation 2019

Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit describes the initialization and synchronization of the automation manager and automation agents. It explains how each component is initialized, how to refresh the automation configuration file, and the management of automation managers.

## Objectives

---

When you complete this unit, you can perform the following tasks:

- Describe the automation environment
- Describe automation manager initialization
- Describe automation agent initialization
- Use the appropriate commands to refresh the automation control files
- List and explain the manager and agent start options
- Build various situations that illustrate initialization events

### *Objectives*

When you complete this unit, you can perform the following tasks:

- Describe the automation environment
- Describe automation manager initialization
- Describe automation agent initialization
- Use the appropriate commands to refresh the automation control files
- List and explain the manager and agent start options
- Build various situations that illustrate initialization events

# Lesson 1 Automation environment and availability



## Lesson 1. Automation environment and availability



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson reviews the automation environment and explains the availability of the automation manager.

## Automation environment and availability

- The automation manager environment
- The automation agent environment
- Sysplex environment
- Automation manager availability

© Copyright IBM Corporation 2019

2-4

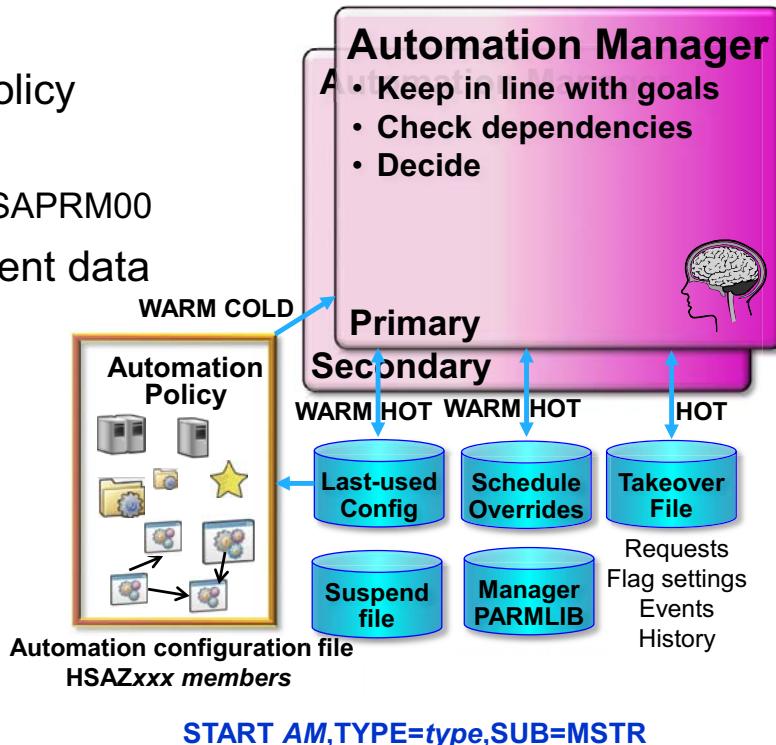
### *Lesson agenda*

- The automation manager environment
- The automation agent environment
- Sysplex environment
- Automation manager availability

## The automation manager environment

Shared data sets:

- Configuration file = Policy
- Manager PARMLIB
  - Initialization member HSAPRM00
- Takeover File: persistent data
- Schedule Overrides
- Name of last-used configuration file
- Suspend file



© Copyright IBM Corporation 2019

2-5

*The automation manager environment*

The automation manager is a z/OS started task. It can be started either from the command line, with the z/OS start command, or automatically at IPL, with an entry in the COMMNDxx member. The default procedure name is AUTOMGR.

This slide shows several of the required data sets for the automation manager:

- HSAPLIB: The DD name of the parameter library data set for the automation manager. The parameter library data set contains at least one initialization member that is named HSAPRM with a two-character suffix. The default name of the initialization member is HSAPRM00. In the initialization member, there are statements that define how the automation manager is initialized and operates. You can override the default initialization member in HSAPLIB in a command-line start command, or within the start procedure JCL. Here is an example that shows the selection of member HSAPRM01 in a start command.

START AUTOMGR, M=01

- HSAOVR: The DD name of the schedule override file. HSAOVR points to a VSAM data set that contains schedule and resource overrides to defined service periods. Information in the HSAOVR data set is used for WARM or HOT starts of the automation manager.
- Takeover file: The name of this data set is specified in initialization member HSAPRMxx. The takeover file is a VSAM data set that contains persistent data for the manager to use during HOT starts of the automation environment. Persistent data includes information such

as votes, events, automation manager flag changes, history, and application group changes.

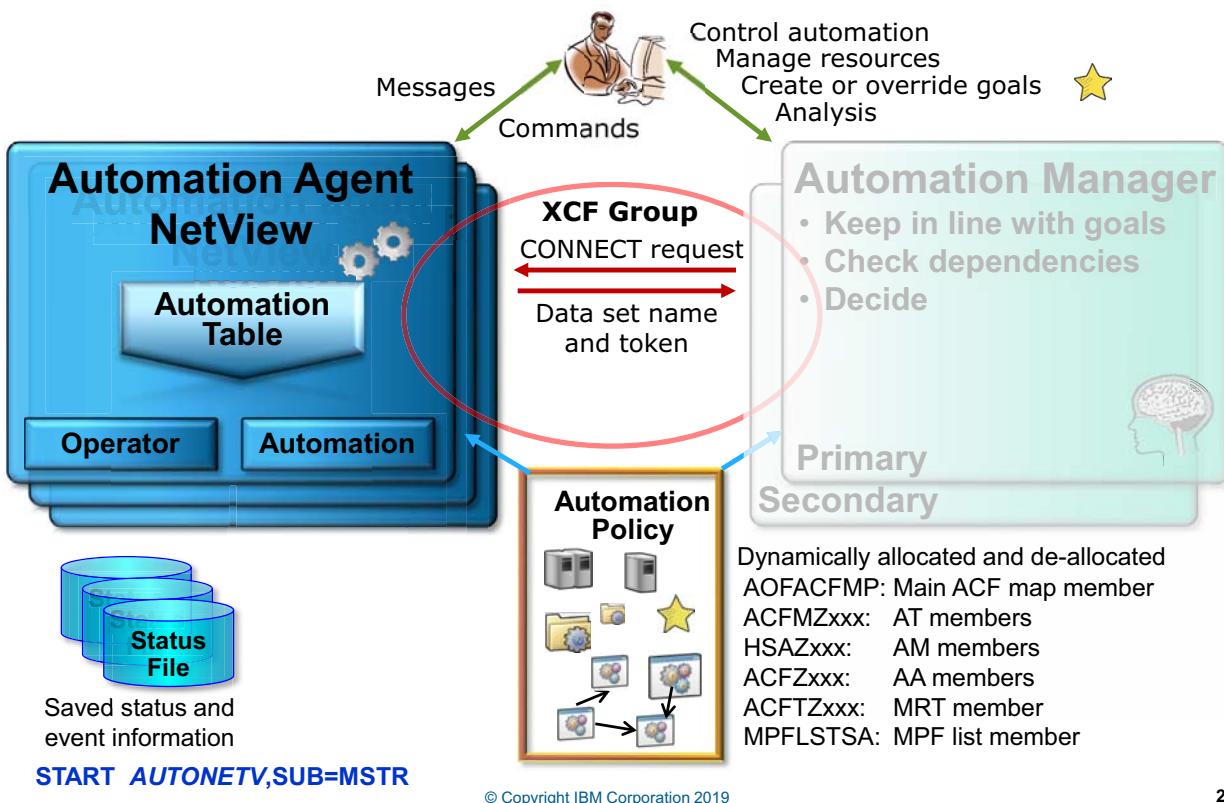
- HSACFGIN: This DD name points to a data set that contains the name of the last-used automation configuration file. During WARM and HOT starts, the automation manager uses information in the **hsacfgin** data set to reconnect to the previously used automation configuration file. The last-used name is updated whenever the automation manager is started COLD and when the automation configuration file is refreshed.

The box that is shown in the lower part of the slide represents the automation configuration file that contains the following information:

- Automation Manager Configuration (AMC): HSAZxxx members
- Automation control file (ACF): (used by agent)
- NetView automation tables (AT)
- NetView message revision table (MRT)
- z/OS Message Processing Facility list (MPF)

In the automation customization dialogs, the build option generates automation configuration file members from information in a policy database. Many customers use generation data groups (GDGs) for the automation configuration file. Using GDGs provides for a backup copy of the files. During customization, the value of the SCOPE parameter in the customization dialogs specifies whether to generate the NetView automation table for a single systems, a SAplex, or an enterprise.

## The automation agent environment



2-6

*The automation agent environment*

The automation agent runs in NetView. NetView is started with the SUB=MSTR parameter. The name of the supplied NetView start procedure is INGENVSA. If the procedure name is the same as the SUBSYS value, then the SUB= parameter defaults to MSTR.

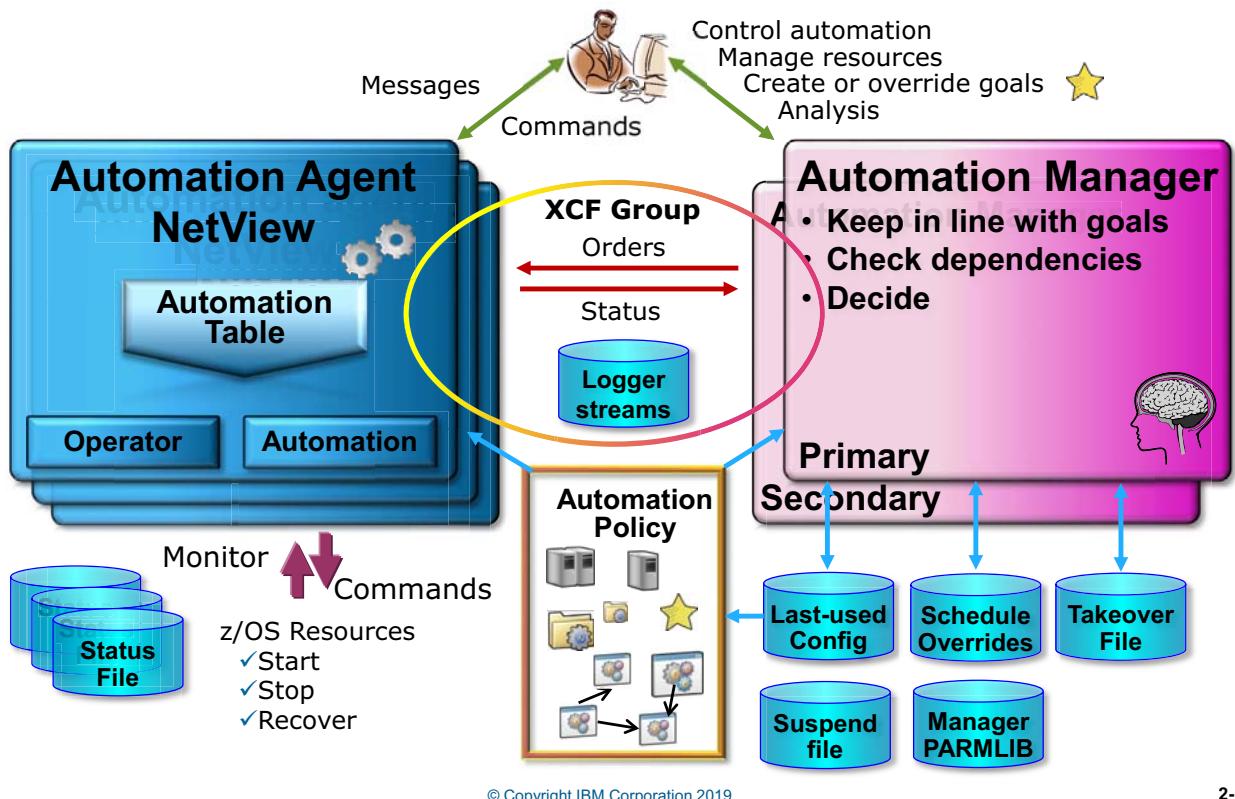
When an agent initializes, it connects with its primary automation manager and receives the name of the automation configuration file and a token. The agent uses this information to locate its automation control file in the automation configuration file, and to ensure that it is consistent with the manager files. The automation agent and manager communicate through z/OS cross-system coupling facility (XCF).

Several data sets used by the agent are shown:

- The automation configuration file contains automation control files that the agent reads at initialization. The agent dynamically allocates the configuration file, and then deallocates it when it is finished reading its automation data.
- The automation status file is a VSAM data set that contains saved status, threshold information, and error events.

The NetView automation table and optionally the NetView message revision table are loaded after the automation configuration file is processed.

# Sysplex environment



2-7

Sysplex environment example

This slide shows an example of a sysplex configuration. Two automation managers are running in different systems of the sysplex in the same SAplex.

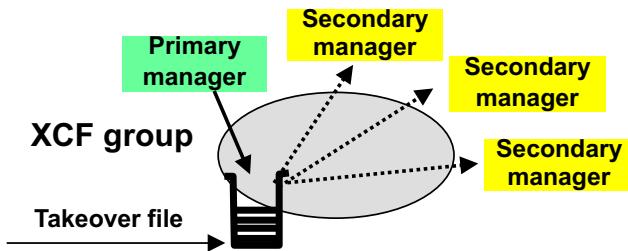
For Automation Control for z/OS the difference is that secondary automation managers run in the same system.

Each agent reads its own automation control files, automation table, and message revision table from the same configuration file, however policy which is linked to multiple systems is shared. Each agent has its own automation status file.

The two managers share all the manager-related files, or each automation manager can have its own manager PARMLIB file. When a second automation manager starts in the SAplex, and it discovers that a primary automation manager is already active; it initializes as a secondary automation manager. For secondary automation manager initialization, start type can be anything, cold, warm, or hot. In the role of a secondary automation manager, the manager does not communicate with the agents or process any automation information. It allocates the same files as the primary automation manager; however, it does not read any data sets except the parameter member in manager PARMLIB data set. The secondary automation manager waits to take over when requested or needed.

## Automation manager availability

- Primary automation manager makes automation decisions
- Secondary automation managers wait to become primary
- A takeover means that automation decisions are delayed but not lost
  - Elapsed time depends on the system load
- Event notification is a z/OS service
  - XCF SYSGONE or XCF MEMBER LEAVE
- Typically, no interaction occurs between the primary automation manager and the secondary automation manager
  - Secondary automation managers do not affect the workings of primary automation managers
  - Data for restart is saved on a shared disk



© Copyright IBM Corporation 2019

2-8

### Automation manager availability

If the primary automation manager fails, a secondary automation manager takes over. The secondary automation manager initializes with a HOT start by reading the current persistent data from the HSATKOVF file and the schedule overrides from the HSAOVR file. No information is lost; the takeover is seamless, except for delays in agent-manager communication.

Automation managers are aware of other managers because each joins the automation product XCF group at initialization. The default name of the XCF group is INGXSG. The XCF group name is defined in INGXINIT or HSAPRMxx. All automation managers in the same SAplex join the same XCF group.

XCF Group processing is a z/OS service that signals members of a group when specific events occur. These two events are important:

- MEMBER LEAVE: This event occurs when a member is no longer available. For example, an automation manager address space terminates normally or abnormally.
- SYSGONE: This event is generated when a system leaves the sysplex, either normally or abnormally.

Automation manager processing uses these signals to determine whether an automatic takeover is necessary.

# Lesson 2 Initialization and initialization options



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson describes how the managers and agents start and initialize. Start options and the differences between them are explained.

# Initialization options

- Automation manager initialization options
  - HOT
  - WARM
  - COLD
- Automation agent initialization options
  - STOP
  - PAUSE
  - NOSTART
  - [RUNMODE=X] - Optional
  - CONTINUE

## Initialization options

The automation manager initialization options for its start type are

- HOT
- WARM
- COLD

The automation agent initialization options are:

- STOP
- PAUSE
- NOSTART
- [RUNMODE=X] - Optional
- CONTINUE

## Manager initialization

- Checks for the primary automation manager
- If starting as primary, the start type is determined from one of the following methods:
  - The START command: S AM,TYPE=HOT,SUB=MSTR
  - The JCL procedure: EXEC PGM=HSAPINIT,PARM='START=HOT'
  - The HSAPRMxx parameter: START=HOT
- Defaults to a HOT start if no specific start-type information is available
  - WARM start if no valid takeover file present
- Takeovers are always HOT starts
  - No data is lost during a HOT start

### Manager initialization

When an automation manager starts, it first tries to determine its position. It checks for a primary automation manager. If a primary manager is already running, the address space initializes as a secondary automation manager. In this case, it does not do anything other than join the XCF group.

The primary automation manager can initialize in one of three ways: HOT, WARM, or COLD. You can specify the start type in these ways:

- Parameter on the command line in a **START** command
- Parameter within the start-procedure JCL
- The **TYPE=** parameter in the HSAPRMxx member

HOT starts use existing data, such as data stored in the takeover file. There is no loss of data with a HOT start. A takeover is effectively a HOT start by the new primary automation manager. Sometimes the desired type cannot be used. For example, if a HOT start is requested and there is no valid information in the takeover file, then a WARM start occurs. The automation manager can be enabled for automatic restart manager (ARM).

## Manager start types

	HOT	WARM	COLD
<b>Configuration data set name</b>	Last used DSN	Last used DSN	DSN comes from parmlib or from an operator PROMPT
<b>Schedule overrides</b>	Taken from last run	Taken from last run	<i>Contents deleted</i>
<b>Persistent data which includes votes and runmodes</b>	Taken from last run	<i>Contents lost</i>	<i>Contents lost</i>

2-12

### Manager start types

This slide shows what is kept and what is lost with each type of start. During a cold start, the name of the automation configuration file is written to the data set specified in the HSACFGIN DD statement. The manager reads the name of the last-used automation configuration file during WARM or HOT starts.

When schedule overrides are created, they are stored in the VSAM file that is identified in the HSAOVR DD statement. On WARM or HOT starts, this file is read and the schedule overrides from the previous automation manager are retained. On COLD starts, the VSAM file is reformatted and the old data is lost.

Persistent data (for example: votes, automation flags, group settings) is stored in the VSAM takeover file and reinstated when HOT starts are done. Hot starts also allow the carry-over of active runmodes. (introduced in another unit). On WARM or COLD starts, the persistent data file is reformatted and the old data is lost; runmodes are not carried over. When starting the automation manager with **PROMPT=YES**, a write to operator with reply (WTOR) is issued, requesting the name of a valid automation configuration data set name.

## Agent initialization sequence

- Contacts the primary automation manager
  - Receives the data set name and token
  - Dynamically allocates the automation control file unless it is preallocated
- Reads the automation configuration file main member and compares tokens from the primary automation manager
- Saves policy data and timestamps in CGlobals
- Issues AOF767I and AOF603D messages to an operator and processes the response
- Loads ACF information from the data set
- Loads the NetView automation table  
Syntax problem will stop initialization
- Loads the NetView message revision table (optional)

### Agent initialization sequence

At initialization, the agent receives the name of the automation control file and its token from the primary automation manager. The name of the automation control that the agent uses is in the automation configuration file map. Automation configuration file member AOFACFMP contains the name of the main automation control file member for the agent of a system. The main control file contains the member names of the automation control files for the resources in the agent's NetView domain.

The automation product then issues a AOF767I write to operator (WTO) message with initialization options followed by the AOF603D WTOR. The operator replies with one of the options or allows the automation product to continue initialization after a 2-minute delay (2 minutes is the default for the delay). There are exits available to change options and to process the operator reply, if required.

After the WTOR is processed, the agent loads the automation control files, automation table, and optionally, a message revision table from the automation configuration file. Any automation table syntax problem will stop initialization. After the automation configuration file is loaded, the agent registers itself with other agents in the same SAplex. Registration allows the agent to send system automation commands to other agents in the SAplex. The agent also registers with the automation manager and exchanges information with the automation manager. The agent is now initialized, and the automation product is fully operational.

## 2 Initialization

### *Lesson 2 Initialization and initialization options*

The RESYNC command causes the automation product to rerun selected parts of its initial status determination routines or to reinitialize parts of its environment setup. The command ACF ATLOAD, reloads the NetView automation table.

# Agent initialization options

```
AOF767I AUTOMATION OPTIONS: 000
  . STOP      - CANCEL AUTOMATION
  . PAUSE     - SUSPEND AUTOMATION
  . NOSTART   - DO NOT AUTOMATE SUBSYSTEM STARTUP
  . RUNMODE=x - SET RUNMODE (CURRENT *ALL)
  . ENTER     - CONTINUE
*0003 AOF603D ENTER AUTOMATION OPTIONS OR 'R' (RE-DISPLAY) - DOMAIN AOFDA
```

- Messages at automation agent initialization
  - AOF767I: Lists valid responses to message AOF603D
  - AOF603D: WTOR - reply with agent initialization option
    - The typical response is a null reply - ENTER

## Agent initialization options

This slide shows the form of the AOF767I message with the valid responses to the AOF603D WTOR. The AOF603D message contains the name of the domain that issued the message. This information can be important. The message is broadcast to all consoles and can drive automation in all NetView domains. Using the domain ID simplifies filtering in automation tables (act on messages from specific domains). In most cases, a null reply is used. The message is redisplayed when you reply with an invalid value.

If you do not enter a reply to the message, the system proceeds with the default response (null) at the end of the administrator-defined delay. A NetView common global variable controls how long initialization waits for the operator to reply to the AOF603D WTOR. The default value is 2 minutes.

If you are not sure how to respond to the AOF603D message, reply with PAUSE. The agent reissues the AOF767I message with an AOF606D WTOR and waits. The AOF606D message is similar to the AOF603D message. Agent initialization is complete when you see message AOF540I.

To ensure a successful initialization, check that message AOF869I indicates no errors. You can also issue the INGAMS command to check that the agents are in status READY.

If there are errors with the automation table, you should see message AOF715A and a request to enter the name of a valid automation table. If you know that you do not want to run the automation

## 2 Initialization

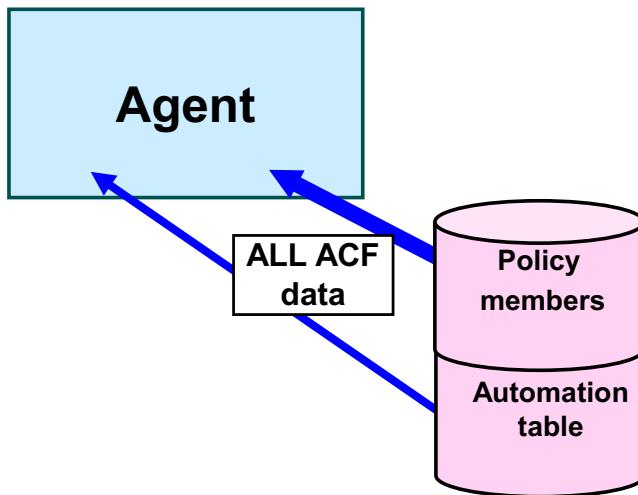
### Lesson 2 Initialization and initialization options

agent at all, enter STOP. Agent initialization does not take place. Recycle NetView if you want to run the agent.

If you do not want automation product to start any subsystem, even though the policy says it should, enter NOSTART as part of the response. Nostart is similar to setting the automation manager Hold flag to YES. This option might be useful when testing. The value (\*ALL) in the example indicates that no defined runmode is active on the system. Runmodes are explained in another unit.

## Agent start

Loads all information from the automation configuration file



### Agent start

At initialization, the agent dynamically allocates the automation configuration file, and reads its own control information. The agent writes each automation fragment that it reads to CGlobals. It deallocates the data set when it is finished reading.

# Lesson 3 Managing the automation environment



## Lesson 3. Managing the automation environment



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson teaches how to use the INGAMS command to manage and refresh the automation environment. Scenarios include:

- Refreshing the automation configuration file
- Stopping or switching an automation manager
- Operational situations

## Managing the automation environment

---

- Refreshing the automation configuration file
- Stopping or switching an automation manager
- Operational situations

## Refreshing automation configuration file

- INGAMS REFRESH

- INGAMS REFRESH CFG=configuration\_dataset\_name | \*
- Primary automation manager:
  - Allocates a new data set and refreshes data from the automation configuration file
  - Sends refresh requests to each agent
  - Updates configuration information with the new data set name
- Each agent
  - Reads its automation control file from the new data set and checks the token
  - Refreshes the agent policy information
  - Refreshes current agent status information
  - Passes status information to the manager
  - Reloads the NetView automation table (default)

© Copyright IBM Corporation 2019

2-18

### Refreshing automation configuration files

The INGAMS command, when used with the REFRESH keyword, refreshes automation control information for the SAplex. Typically, refreshing automation control information is done after modifications to automation policy. In the INGAMS command, the “CFG=” value is the name of a partitioned data set, a generation data group (GDG) name, or an asterisk (\*). An asterisk indicates a reload of the current configuration file.

Each agent receives a refresh request. In some cases, there might be no actual update of the automation control file for a system. The refresh action updates only changed fragments of the automation configuration file. Agent flags can be reset as a result of a refresh.

This simplified description does not include all the checking that occurs. Tokens are always checked. Also, a check is made that there is automation configuration file information in the data set for all systems in the SAplex. The complete refresh request is denied if any of these checks fail. The INGAMS command is described in greater detail in [Unit 4, Lesson 4](#) on page 219. For more information about the complete syntax, see *System Automation for z/OS: Operator's Commands*.

## Refreshing automation configuration files (continued)

- Refresh from the z/OS console
  - **F AM,REFRESH,configuration\_dataset\_name**
- Runtime policy changes are lost when the automation configuration file is refreshed

© Copyright IBM Corporation 2019

2-19

### Refreshing configuration files (continued)

You can also use the z/OS subsystem Modify (F) command to refresh automation configuration information:

**F automgr,REFRESH,configuration\_dataset\_name**

where:

- *automgr* is the job name for the automation manager address space
- *configuration\_dataset\_name* is the name of the new configuration file

The data set name can be either a fully qualified data set name, or a GDG name. Asterisks are not allowed in this command format.

## Stopping the automation manager

- F AM,STOP,type,disp
  - Type:
    - NORMAL: Complete all work in progress (default)
    - IMMED: Complete current work item
    - FORCE: Terminate all work in progress
    - DEFER: Wait for an interval in the STOPDELAY parameter.  
If a secondary automation manager is running, terminate the primary automation manager; otherwise, wait until the last agent ends
  - Disp:
    - NORESTART: Terminate normally (default)
    - RESTART: Terminate and begin ARM restart
- P automgr
  - Stops the manager with the default type and disp parameters

AM is the job name that is supplied in the \*BASE PDB

© Copyright IBM Corporation 2019

2-20

### Stopping the automation manager

Two commands can be used to stop an automation manager address space: z/OS STOP and z/OS MODIFY. More options are available with the MODIFY command:

- The **type** parameter controls how much work is allowed to finish normally before the manager stops. DEFER is useful when doing a final shutdown. The last active agent issues this request, and then terminates itself so that all started tasks are removed from the system.
- The **disp** parameter specifies whether ARM should restart the automation manager. NORESTART should be used for final shutdown before system IPL. To use the RESTART option, you must define the automation manager to z/OS ARM.

A normal z/OS STOP command uses the default values for the **type** and **disp** parameters to shut down the address space. These commands are valid for both primary and secondary automation managers. If a NORMAL shutdown is requested for the only active automation manager, it waits for all active automation agents to shut down. More specifically, the manager waits for the INGPXDST task to end. A WTOR requests confirmation of the manager shutdown before all agents are stopped. The z/OS CANCEL command also terminates the automation manager address space immediately.

# Switching the primary automation manager

- Use the INGAMS command to switch automation managers
  - Full panel interface: Enter row-command A (Manage) next to a secondary automation manager
  - Line mode command: SET automgr MODE=PAM where automgr is the job name of a secondary automation manager
- Use the z/OS console only when necessary
  - F automgr,SETMODE,PAM,disp
  - automgr – The job name of a secondary automation manager
  - disp
    - NORESTART: Prior primary automation manager terminates normally (default)
    - RESTART: ARM restarts the previous primary automation manager as a secondary automation manager

## *Switching the primary automation manager*

To switch the primary manager task, issue the INGAMS command from either the INGAMS panel or the command line. The command always targets a secondary automation manager. The z/OS MODIFY command, with the SETMODE parameter also switches the automation manager. The modify command is issued with the start-procedure name of a secondary automation manager. This method forces the current primary to terminate. If RESTART is specified, the former primary manager restarts as a secondary. The primary role is switched.

Note: The MODIFY command must be entered on the home system of the secondary automation manager.

# Lesson 4 Operational situations

## Operational situations

- SAplex initialization situation
  - System A initializes as the first system in the SAplex
  - System B initializes and joins SAplex
- Primary automation manager fails the situation
- Token mismatch situation
- Recycle the agent situation

For Automation Control for z/OS the differences are that secondary automation managers run in the same system and that each system requires its own primary automation manager

For Automation Control for z/OS the differences are that secondary automation managers run in the same system and that each system requires its own primary automation manager.

## **Initialization scenario: System A**

### System A IPLs as the first system in the SAplex

- Manager starts
  - No takeover data or configuration information; issue a WTOR for the operator response
  - Receives the configuration data set and the takeover file names from HSAPRMxx
  - Starts writing persistent data to the takeover file
  - Starts writing schedule overrides to the override file
- Agent starts:
  - Receives the automation configuration file data set name and the token from the manager
  - Reads policy from the automation control file
  - Loads the NetView automation table
  - Loads the message revision table (optional)
  - Starts writing changed status to automation status file

#### *Initialization scenario: System A*

During a SAplex-wide IPL, as the first system IPLs, an automation manager address space is started with the default start type of HOT. The manager joins the XCF group as the first manager in the group and starts as a primary automation manager.

Because there is no persistent data or schedule override data, a WTOR is issued. The operator must decide to continue with a WARM start or with RETRY. The name of the data set holding the automation manager configuration files is obtained from the HSAPRMxx member. The manager dynamically allocates this data set, and reads the AOFACFMP member from it. The AOFACFMP member contains the name of the main configuration member for the manager in the enterprise. Within this member is a configuration token value, which the manager reads and saves. The configuration data set is then deallocated. Current manager definitions are then written to the takeover file. From then on, any new persistent data is written to the takeover file, and any schedule overrides are written to the override file.

The first agent starts and joins the same XCF group, and identifies the current primary manager. It connects to the primary automation manager and receives the name of the data set that contains the automation configuration file, and a token that specifies the level of file that the manager is using.

The agent allocates the data set and reads the AOFACFMP member that contains the automation control files for its system. The agent loads automation information from the files and stores the

common global variables. The configuration data set is deallocated. After initialization, the agent saves any changes to the automation status file.

## **Initialization scenario: System B**

### System B IPLs and joins SAplex

- The manager starts as secondary
- The agent starts with the same steps as with system A
- Both agents synchronize the status updates and start communication using XCF (Gateway sessions in SA 4.1 were replaced by XCF)

### *Initialization scenario: System B*

System B IPLs and joins the SAplex (the same sysplex group). An automation manager address space is also started. The new manager joins the XCF group, and discovers an active primary automation manager. The new manager starts as a secondary automation manager. It allocates the same files as the primary automation manager, but it does no processing.

The NetView agent address space also starts. It connects to the current primary automation manager (on system A) and retrieves the configuration file data set name and token. It uses the information in the configuration file to initialize, similar to the first agent.

## Primary automation manager failure situation

### Primary automation manager on system A fails

- System B secondary automation manager becomes the primary automation manager; effective HOT start
  - Read persistent data from the takeover file
  - Read schedule overrides from the override file
  - Issue a load request to all agents and update agent status
- Automation manager on system A restarts
  - Initializes as a secondary automation manager

#### *Primary automation manager failure situation*

The primary automation manager on system **A** fails. XCF detects the failure. A signal is passed to the secondary automation manager on system **B**. The secondary automation manager initiates a takeover, reads the takeover file, and uses the information to set current definitions, status, and other persistent data.

The new primary automation manager sends a load request to each automation agent, passing the name of the current configuration data set and the token. Each agent resynchronizes the status of the applications. This action by the agent sends status information to the new primary automation manager.

The automation manager on system A is now restarted, either manually or by automation. The manager on system A initializes as a secondary automation manager because the automation manager on system B is now the primary.

## Token mismatch situation

- The entire system can be affected
- Use the previous configuration  
  INGAMS REFRESH CFG=\*
- Alert your automation product administrator

© Copyright IBM Corporation 2019

2-26

### *Token mismatch situation*

The automation configuration file is critical to the automation product. The inability to initialize renders the entire system unavailable.

If a token mismatch is encountered when refreshing the automation configuration file, the swiftest and safest course of action is to revert to a previous version by issuing the following command:

```
ROUTE ALL,MODIFY AM REFRESH *
```

If you use GDGs for the configuration data sets, then specify minus 1 (-1) to load the previous version.

## Agent recycle

- The agent receives the automation configuration file data set name and token from the manager
- The agent reads the automation control file information
- The agent reads resource status from its automation status file

### Agent recycle

During a recycle of the NetView address space, the automation agent reconnects to the primary automation manager after NetView starts. The agent receives the configuration data set name and token, and then retrieves its policy information from the automation control file. The agent then reads the last-stored status for each application from the automation status file and uses this information during the restart of automation.

## Student exercise



© Copyright IBM Corporation 2019

2-28

### *Student exercises*

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you completed this unit, you can perform the following tasks:

- Describe the automation environment
- Describe automation manager initialization
- Describe automation agent initialization
- Use the appropriate commands to refresh the automation control files
- List and explain the manager and agent start options
- Build various situations that illustrate initialization events

### *Summary*

Now that you completed this unit, you can perform the following tasks:

- Describe the automation environment
- Describe automation manager initialization
- Describe automation agent initialization
- Use the appropriate commands to refresh the automation control files
- List and explain the manager and agent start options
- Build various situations that illustrate initialization events





## 3 Commands and operations



### Unit 3: Commands and operations



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit introduces operator commands that you use to manage automated resources. Some of the commands that are explained, and demonstrated in this unit are SDF, INGLIST, DISPSTAT, INGREQ, INGINFO, and DISPIINFO.

## Objectives

---

When you complete this unit, you can perform the following tasks:

- Use the online help function
- Display resource status with SDF, INGLIST, and DISPSTAT
- Issue manager and agent commands to display resource information, relationships, and automation flags
- Use INGREQ to start and stop resources

### *Objectives*

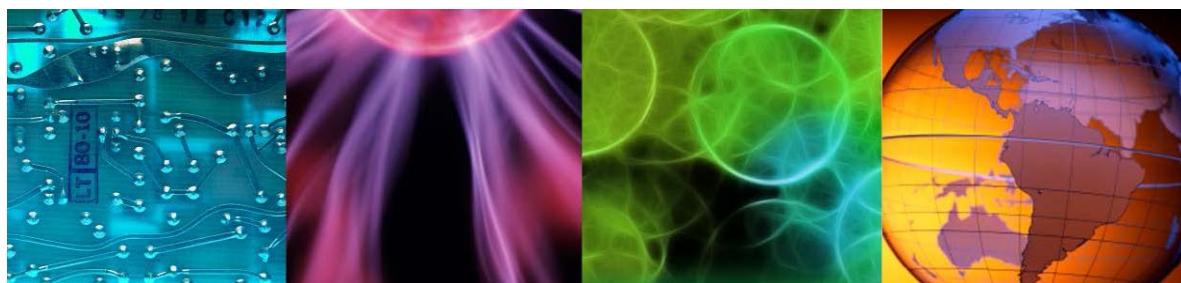
When you complete this unit, you can perform the following tasks:

- Use the online help function
- Display resource status with SDF, INGLIST, and DISPSTAT
- Issue manager and agent commands to display resource information, relationships, and automation flags
- Use INGREQ to start and stop resources

# Lesson 1 Command basics



## Lesson 1. Command basics



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson covers command basics and command format. You learn to specify resources, route commands to other systems, and to control command output.

# Command basics

- Program Function (PF) keys
  - Most panels display a list of available PF keys and their function

PF1=Help	PF2=End	PF3=Return	PF4=DISPSTAT	PF5=Filters	PF6=Roll
PF8=Forward	PF9=Refresh	PF10=Previous	PF11=Next	PF12=Retrieve	

- Online help is available
  - Commands
    - `HELP command_name`
    - PF1 while command line contains `command_name`
  - Messages
    - `HELP message_number`
    - PF1 while command line contains `message_number`
  - Within a panel, press PF1

© Copyright IBM Corporation 2019

3-4

## Command basics

On each panel is a set of program function (PF) keys that are used for actions that can be taken on the panel. The example that is shown on this slide is from the INGLIST panel. The PF keys are typically displayed at the bottom of the panel. You can also use the DISPFK command to display the settings for PF keys.

Within Automation, there are several of ways of getting online help:

- The AOCHELP command lists all Automation commands. Commands can be run from the list or help panels that are displayed for each command on the list.
- Use the HELP command (PF1) to display information about commands, messages, automation manager status values, and to get context-specific help.
- The EXPLAIN command displays the meaning and possible causes for the manager and agent status values for a resource.
- Use the AOC command to display a menu-driven dialog that leads to most commands.

The command and message help source can be modified to suit installation needs. Details are in the *Tivoli NetView for z/OS for z/OS Customization Guide*.

## Command format

- Access the full screen (panel) interface for commands
  - Enter the *command\_name* without parameters
- Every command has a general form
  - *Command\_name*
  - *keyword\_1,...,keyword\_n*,
  - *keyword\_1=value\_1,..., keyword\_n=value\_n*
- Some commands support abbreviated keyword values; for example, Win or W for winning
- Examples
  - INGLIST
  - INGVOTE STATUS=W,EXCLUDE=SVP,USER=*username*

### Command format

Automation commands can display panels or return the data if all of the required keyword parameters and values are entered. Command results can be displayed on Automation commands panels, written to a file, or sent to a program. Automation displays a panel with data and input fields when a command with no parameters is entered on the command line. No panel is displayed when a command is entered with required keywords and parameters.

You can separate keywords or parameters with a comma or a space. Values can be abbreviated if the abbreviation contains enough characters to make it unambiguous. For example, STATUS=W or STATUS=WIN can be used to specify WINNING as the value for the STATUS parameter. Commands are parsed for accuracy before being submitted to Automation.

# Specifying resources

- Automation manager resource naming:
  - name/type[/system]
  - Scope is sysplex or subplex
- **Name** specifies the name of the resource
- **Type** specifies the type of the resource
  - APL, APG, REF, EVT, MTR, SVP, SYG, SYS, DMN
- **System** specifies the system or domain name where the resource is defined. Not used for sysplex resources
- Wildcard \* and placeholder % for one character
  - CICS\*/AP\*              Name starts with CICS. Type=APL or APG
- Selection panel displayed when resource is not unique

## Specifying resources

the term resource denotes any entity that can be automated. The automation manager administrates resources. A resource is identified by its name. The resource name must be unique within the automation manager's name space. The format of the resource name is name/type[/system]. The three components of the resource name can be the following:

- **Name**: subsystem name for an application (APL) or the automation name for an application group (APG).
- **Type**: Optional type of the resource.
- **System**: Specifies the system or domain name where the resource is defined.  
The system parameter is not used for sysplex resources.

You can also use an asterisk (\*) to substitute one or more components.

You can use the percentage sign (%) as a place holder for one character.

If the resource that you specified is not unique within the domain of the automation manager, a selection panel is displayed where you can select what you want to work with.

## Routing commands to other systems

```
>-----+-----+-->
'--TARGET=--.-| Destination |---' | -OUTDSN=dsname-----|
      | -| Dest_list |---|           | -OUTMODE=-.LINE-----|
      '|-*ALL-----'|           | -AUTO---|
                           | -NETLOG-'
```

- TARGET parameter routes commands to other systems
- Default is local system or SAplex
- The destination can be of these types:
  - A particular system or NetView domain in the SAplex or enterprise
  - A subset of systems in the SAplex or enterprise
  - An SAplex (sysplex group name (GRP))
  - All systems currently active in the local SAplex (\*ALL)

### Routing commands to other systems

The scope of many system operations commands is SAplex-wide or enterprise-wide. Most commands that affect the automation agent support the TARGET parameter, which specifies where the command is processed. The TARGET parameter is required when routing the command from a focal point system to a remote system or SAplex. If the TARGET parameter is omitted, Automation processes the command on the local system. A target destination can be of these types:

- A single system that is identified by the system name
- A single system that is identified by the NetView domain ID
- A SAplex
- A list of destinations that are separated by commas or spaces

The \*ALL keyword sends the command to all active systems in the local SAplex. \*ALL cannot be used in a destination list.

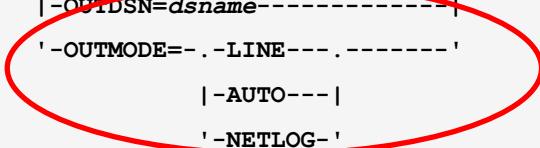
For example, to monitor the shutdown status of multiple systems enter the following command:

DISPSTAT STATUS=AUTODOWN TARGET=(MVSA,MVSB)

Note: Not all commands that support the TARGET parameter also support the \*ALL keyword.

## Controlling command output

```
>---.-----|-----|----->  
'-TARGET=-.-| Destination |.-.-'| -OUTDSN=dsname-----|  
     |-| Dest_list |---|      '-OUTMODE=-.LINE-----'  
     '-*ALL-----'| -AUTO---|  
                  '| -NETLOG-'
```



- No OUTMODE: Panel is displayed in operator sessions
- LINE
  - Line mode command output
    - No color attributes
    - Different field sequence
    - Not processed by the automation table
    - Not written to NetView log (netlog)
    - Can be used with NetView PIPEs
- AUTO
  - Command output to netlog for unattended automation tasks; otherwise, to console
- NETLOG
  - Command output to the NetView log only

© Copyright IBM Corporation 2019

3-8

### Controlling command output

Values for the optional parameter OUTMODE specify the type of command output. For example, OUTMODE=NETLOG specifies that the command output goes to the NETLOG only.

When no OUTMODE is specified a panel is displayed in operator sessions.



**Attention:** The format of line mode command output may change in a new release. Therefor it is recommended to use INGDATA for automation purposes.

## Writing command output to a data set

```
>-- .----- .----- .-->
' -TARGET=-.-| Destination | -.-' | -OUTDSN=dsname-----|
      | -| Dest_list | ---|     ' -OUTMODE=-. -LINE--- .-----'
      '-*ALL-----'           | -AUTO---|
                           ' -NETLOG -'
```

- Controls command output to a named, existing data set
- Sequential data set or a member of a partitioned data set
- Minimum record length 80 bytes, recommended are:
  - VB 1024 for INGLIST
  - 256 for DISPSTAT, DISPGW, INGIMS
  - 128 for INGMSG, INGVOTE, DISPMTR, INGHIST
- Forces OUTMODE=LINE

© Copyright IBM Corporation 2019

3-9

### Writing command output to a data set

The OUTDSN parameter is used to write command output to an existing sequential or partitioned data set.



**Attention:** If OUTDSN specifies a data set that is already allocated to NetView (for example DSILIST), the exclusive ENQ that is issued when OUTDSN is used will not be removed until NetView is stopped. Thus, it is recommended that you do not use pre-allocated data sets for OUTDSN.

## Sorting the command output

```
.-----.
|      |
v      |
>--SORT--- | sort-order |-coln-----.
```

- Most of the commands in fullscreen mode support the SORT and FIND
- sort\_order
  - A The list is displayed in ascending order. This is the default
  - D The list is displayed in descending order
- Coln: The column to be sorted on. You can specify more than one column, separated by a comma or a blank

### Sorting the command output

Most of the commands in fullscreen mode support the SORT and FIND subcommands. The SORT subcommand lets you change the order that data is displayed in.

coln specifies the column number or title (header) to be sorted on. You can specify more than one column separated by a comma or a blank.

The sort order can be ascending (A) or descending (D).

## Searching for strings

```
>--FIND-- search-string--| direction |-----
```

- The FIND subcommand searches the display data for a specified string.
- *search\_string*: The string that is to be searched for. The search starts at the first line shown in the display.
- *direction*
  - N Find the next occurrence (forward direction). This is the default
  - P Find the previous occurrence (backward direction)

### Searching for strings

The FIND subcommand searches the display data for a specified string. It can be abbreviated to F. The RFIND (repeat find) command is used to find the next occurrence of the string in the display data. It can be abbreviated to RF.

You can specify the string that is to be searched for and the direction.

# Lesson 2 Commands to display resource status



## Lesson 2. Commands to display resource status



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn the commands to display the resource status.

# Displaying resource status

- INGLIST:  
Automation Manager resource status display (static)
- DISPSTAT:  
Automation Agent resource status display (static)
- SDF:  
Automation Agent resource status display (dynamic)

## *Commands to display resource status*

This lesson explains three commands that can be used to display resource status in an automation environment. DISPSTAT and INGLIST give static displays: a snapshot at the time either command is entered. Both commands can be issued manually or by programs. DISPSTAT shows the automation agent view of the resource status, while INGLIST shows the automation manager view of the resource status. Several views are available to display automation flags, monitor times, trigger, or service periods. Use PF4 to jump between INGLIST and DISPSTAT panels. Use the DISPSTAT and INGLIST commands for detailed analysis of problems.

The SDF command shows agent status on customized panels. The displays are automatically updated whenever a resource status changes. The SDF command panels can be customized to display the status of resources such as applications, gateways, WTORS, and spool. The SDF command is for operator use only.

Graphical user interface alternatives to display Automation resource status:

- Tivoli Enterprise Portal, an OMEGAMON-based, explorer-like interface.
- Service Management Unite with its DASH-based user interface.

## INGLIST: Automation manager resource status display

- Displays resource information from the automation manager
  - Status: Compound, observed, automation, desired, startability, health
  - Resource name, jobname, and description
  - Resource type
  - Suspend
  - Group nature
  - Flags (automation, hold)
  - Start and stop type
  - Trigger name
  - Schedule name
  - Category and subcategory
  - Other information like runmode qualification and pacing gate and status
- Provides row-commands to manage resources

© Copyright IBM Corporation 2019

3-14

### INGLIST: Automation manager resource status display

The INGLIST command lists automation manager information about automated resources. Each row contains information about a single resource. The information includes all status values. You use row-commands that are displayed on the primary INGLIST panel to manage the listed resources.

Use PF10 to scroll left and PF11 to scroll right to see more columns of information. When PF4 is pressed the INGLIST panel, the same list of applications is shown on the DISPSTAT panel. The INGLIST command displays also application groups, REF resources, domains, and monitor resources, but the DISPSTAT command does not.

## INGLIST: Automation manager resource status display (continued)

- Extensive filter capability (PF5 or with parameters)
  - Resource name, description and jobname, including wildcards
  - Resource status: observed, desired, automation, compound, health
  - Automation flag
  - Resource category (CICS, DB2, USS, or others) and sub category
  - Runtoken
  - Group type
- A snapshot of status, not updated dynamically
  - Must use PF9 to refresh
- Command-line format examples
  - INGLIST
  - INGLIST RV\*/\*/MVS<sup>B</sup>
  - INGLIST COMPOUND=PROBLEM
  - INGLIST HEALTH=NORM

INGLIST: Automation manager resource status display (continued)

The INGLIST command supports display filters. You can filter the data by pressing PF5 or by using parameters with the command. The second example displays all resource types with RV in the first two characters of the name that are on system MVS<sup>B</sup>. The same filters are applied to the DISPSTAT if PF4 is pressed.

An important point to remember is that the INGLIST display is static and is not automatically updated. Use PF9 to refresh the INGLIST display as often as needed.

INGLIST also supports use of *NOT* when setting filters. For example, you can request to see all resources that have a health status of *NOT NORMAL*.

## INGLIST: An example of the primary panel

SA z/OS - Command Dialogs				Line 25 of 44	
Domain Id . : AOFDA		INGLIST		Date . . . : 10/25/18	
Operator Id : INGC102		Sysplex = SYSPLEX1		Time . . . : 11:01:26	
A Update	B Start	C Stop	D INGRELS	E INGVOTE	F INGINFO G Members
H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTR P INGPAC
R Resume	S Suspend	T INGTWS	U User	X INGWHY	/ scroll
CMD Name	Type	System	Sus	Compound	Desired Observed Nature
RACF	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RESOLVER	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RMF	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RMFGAT	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RVBASIC	APG	MVSA		INAUTO	AVAILABLE STARTING BASIC
RV01	APL	MVSA		AWAITING	AVAILABLE SOFTDOWN
RV02	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RV05	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
RV06	APL	MVSA		DIR SATISFACTORY	UNAVAILABLE SOFTDOWN
SDSF	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE
SYSPLEX	GRP			PROBLEM	AVAILABLE PROBLEM BASIC
SYSVILPC	APL	MVSA		SATISFACTORY	AVAILABLE AVAILABLE

Command ==> F1=Help F2=Edit F3=Return F4=DISPSTAT F5=Filters F6=Roll  
 F7=Backward F8=Forward F9=Refresh F10=Previous F11=Next F12=Retrieve

The positions of these columns are unchanged when scrolling right or left

© Copyright IBM Corporation 2019

3-16

### INGLIST: An example of the primary panel

This example shows the INGLIST primary panel. From left to right, it shows the following columns:

**CMD**, a column you can use to enter any of the commands listed at the top of the screen. These are known as line commands.

Resource **name**, a name that can be different from the job name.

Resource **type**, including applications (APL), application groups (APG), resource references (REF), remote domains (DMN), and monitors (MTR).

**System**, where the resource is active. Note, SA has an enterprise-wide view.

**Compound** status, a summary of all the statuses of a resource. It considers elements of four other statuses as well as a number of other values. The following values are the most important values:

- **Satisfactory**: The resource's desired and observed statuses are corresponding, no further automation or operator activity is required.
- **Degraded**: For a group it means that it is partially running, but not at full capacity. For normal resources, it can mean that the resource is *starting* or *stopping*, or that the application is suffering from a performance or throughput problem, which is the case here.
- **Problem**: This resource has a problem that automation cannot solve. It requires operator intervention.

**Desired** status: The status that the automation manager is trying to move the resource to. It can either be AVAILABLE or UNAVAILABLE.

**Observed** status: This represents the current status as observed by the automation agent. The automation agent also provides a more detailed agent status. The most important values are as follows:

- **AVAILABLE**: The resource is ready for use.
- **SOFTDOWN** (unavailable): The resource is unavailable and automation might restart it.
- **PROBLEM** (unavailable): The resource is unavailable and automation will not restart it.

You can enter commands on the command line directly above the PF keys.

Use the FIND (**F**) and RFIND (repeat find, **RF**) subcommands to search the displayed data for a specified string. When the list spans multiple panels, the TOP subcommand scrolls the display to the top of the first panel; the BOT subcommand scrolls to the top of the last panel.

The SORT subcommand sorts the list of items. Numbers in the SORT command correspond to the data columns on the panel (not the character columns). For example, SORT D 1 means sort in descending sequence on the first data column. (The CMD column is not counted as data and the automation flag settings are treated as a single column.). SORT can also be used for agent commands such as DISPSTAT. The slash (/) subcommand positions the selected row at the top of the display, like CSR in ISPF displays.

PF10 (Previous) scrolls horizontally to the left. PF11 (Next) scrolls horizontally to the right. PF12 (Retrieve) retrieves the last command that was entered and several commands that were entered before the last command. There is a limit to the number of commands that can be retrieved.

PF9 refreshes the display. When there is activity which can cause status changes, use PF9 regularly to ensure that current information is selected. In normal use, all resources in the SAplex are shown if INGLIST is entered with no filters and no operands. To get an explanation of the automation manager status values, on the command line, enter HELP or EXPLAIN followed by the status value. The left three columns (Name, Type, and System) remain fixed when scrolling left or right.

PF11 scrolls the display to the right. Some of the data that is shown in columns to the right of the Nature column, include automation and health states, description, Starttype, StopType, trigger, schedule, category, subtype jobname, inform list, pacing gates, and runmode (Qual).

Pressing PF2 ends all panels and returns you to the original point of entry.

## Row commands on the INGLIST panel

- A (Update): Updates status for the resource, sets the next start or stop type, and automation and hold flags
- B (INGREQ START): Makes the resource available
- C (INGREQ STOP): Makes the resource unavailable
- D (INGRELS): Displays the dependencies for the resource
- E (INGVOTE): Displays all requests and votes that are pending for the resource
- F (INGINFO): Displays details about the resource
- G (MEMBERS): Displays INGLIST panel with members of the selected application group
- H (DISPTRG): Displays the trigger details for the resource

### Row commands on the INGLIST panel

This slide shows some of the row commands available to manage Automation resources that are displayed on the INGLIST panel. The commands are not case-sensitive.

## Row commands on the INGLIST panel (continued)

- I (INGSCHED): Displays the resource schedule with overrides
- J (INGGROUP): Starts the INGGROUP command for the selected group
- K (INGCICS): Starts the INGCICS command for the selected resource
- L (INGIMS): Starts the INGIMS command for the selected resource
- M (DISPMTR): Displays monitor resource information
- P (INGPAC): Show pacing gates associated with resource
- R (INGSUSPD - RESUME): Resume resource (give back to automation)
- S (INGSUSPD - SUSPEND): Suspend resource (take out of automation)
- T (INGTWS): Starts the INGTWS command for selected resource
- U (USER): For customer specific line commands. Starts the AOFEXC04 user exit if it is installed
- X: (INGWHY): Show diagnostic information about resource
- /: make the selected line the first line of the display

*Row commands on the INGLIST panel part two*

This list shows the other row commands that are on the primary INGLIST panel.

## Setting view filters for INGLIST

- Select resources for display
  - Can be saved as defaults for later INGLIST calls
  - Lost at logoff
- Filter the following items:
  - Resource name, description and jobname, including wildcards
  - Resource status: observed, desired, automation, compound, health
  - Automation manager automation flag
  - Resource category (CICS, DB2, or others) and subtype
  - Runtoken, pacing gate
  - Group type
- Command INGFILT issued at logon to set defaults for operators
  - Example: **INGFILT REQ=SET RESOURCE=IMS\* OUTMODE=LINE**

### Setting view filters for INGLIST

Pressing PF5 on the primary INGLIST panel runs the INGFILT command. Use the INGFILT panel to set viewing filters for the INGLIST display. You can also use the INGFILT command-line form to specify filter criteria.

You can use the INGFILT command before Automation is initialized. The command sets NetView global variables. This function allows you to set default view filters in the initial log-on CLIST for operators. Filters can be set for individual operators. For example, one group of operators might view only CICS applications or resources from one system or SAplex. When setting viewing filters, you can use abbreviations and wildcard characters.

## INGFILT example

```

INGKYFLT          SA z/OS - Command Dialogs
Domain Id . . : AOFDA ----- INGFILT ----- Date . . : 10/25/18
Operator Id : INGC102                         Time . . : 11:06:47

Resources...      format: name/type/system or *am*/**/*
DB2**/*/*■

Observed status ==> *
Desired status ==> *
Automation status ==> *
Compound status ==> ^Satisfactory
Health status ==> *
Automation flag ==> _____ Yes or No
Category ==> *
Subcategory ==> *
Group type ==> *
Job name(s) ==> *
Description ==> *
RunToken(s) ==> *
Pacing gate(s) ==> *

AOF710A VERIFY/REVISE INPUT AND THEN PRESS ENTER
Command ==> _____
F1=Help   F2=End   F3=Return  F4=Clear   F5=Reset   F6=Roll
                           F9=Save    F12=Retrieve

```

Define filters and press ENTER to refresh INGLIST

© Copyright IBM Corporation 2019

3-20

### INGFILT example

This slide shows the INGFILT command panel. In this example, you see only resources with names that start with the string DB2. All other resources are filtered out of the display. Press ENTER to refresh the list of resources on the INGLIST panel.

PF9 (SAVE) allows the operator to define a set of default view filters for future INGLIST views. The filters remain in effect while the operator is logged on. PF5 (RESET) clears the filters.

Note: you have to press enter to refresh INGLIST.

You can specify more than one filter criteria. When you specify multiple criteria, Automation uses a logical AND to generate the combined view filter. You can use abbreviations, such as AV for Available. You can exclude statuses by prefixing the string with a backslash (\, hexadecimal code **E0**) or a not sign (hexadecimal code **5F**). For example, entering \Available in the Observed Status field returns all resources with an OBSERVED status of anything except AVAILABLE. The resource category (CICS, DB2, IMS) can be specified with the resource subtype; for example, IMSCTL.

## DISPSTAT: Display automation agent resource status

- Status of resources as maintained by the automation agent
  - Snapshot, not dynamically updated
  - Filtering and dependency selections
  - Row commands are available on the DISPSTAT
    - SETSTATE
    - INGAUTO
    - INGREQ
    - Other commands
- Run from INGLIST panel with PF4
- Run from the command line

Examples:

- DISPSTAT ALL
- DISPSTAT TARGET=MVSB
- DISPSTAT TARGET=\*ALL STATUS=\UP
- DISPSTAT \*KIDS VTAM

© Copyright IBM Corporation 2019

3-21

*DISPSTAT: Display automation agent resource status*

DISPSTAT shows the automation agent view of the applications. The DISPSTAT command displays a snapshot of applications and their status. This static view is manually refreshed by pressing PF9 on the panel. DISPSTAT also lists row-commands for displaying policy definitions for applications and initiating actions against applications in the list.

You can enter DISPSTAT from the NetView command line, or from any command area in a panel. You can also use PF4 from the INGLIST screen to run DISPSTAT. The default display is for the system or domain in which the command is entered. To display another system or domain, add the TARGET parameter.

When run from the command line, it allows filtering for target systems, resource agent status, and resource children or parents. For example:

- DISPSTAT \*KIDS VTAM lists the direct children of VTAM
- DISPSTAT \*ALLKIDS VTAM lists all children and children of children of VTAM
- DISPSTAT \*ALLPARENTS LLA lists all parents and parents of parents of LLA.

## DISPSTAT primary panel example

SA z/OS - Command Dialogs							Line 14	of 32
Domain Id . . . : AOFDA	DISPSTAT			System = MVSA			Date . . . : 10/25/18	Time . . . : 11:17:43
Operator Id : INGC102								
A dispflgs B setstate C ingreq-stop D thresholds E explain F info G tree								
H trigger I service J all children K children L all parents M parents								
CMD Resource	Status	System	Jobname	A	I	S	R	T
RS	Type			RS	Type			Activity
LLA	UP	MVSA	LLA	-	-	MVS	--none--	
OMVS	UP	MVSA	OMVS	-	-	MVS	--none--	
RACF	UP	MVSA	RACF	-	-	MVS	--none--	
RESOLVER	UP	MVSA	RESOLVER	-	-	MVS	--none--	
RMF	UP	MVSA	RMF	-	-	MVS	--none--	
RMFGAT	UP	MVSA	RMFGAT	-	-	MVS	--none--	
RV01	DOWN	MVSA	RV01	The positions of these columns are unchanged when scrolling right or left	-	MVS	--none--	
RV02	UP	MVSA	RV02		-	MVS	--none--	
RV05	UP	MVSA	RV05		-	MVS	--none--	
RV06	AUTODOWN	MVSA	RV06		-	MVS	--none--	
SDSF	UP	MVSA	SDSF		-	MVS	--none--	
SYSVIPLC	ENDED	MVSA	SYSVIPLC		-	ONE-SHOT	--none--	
TCPIP	UP	MVSA	TCPIP		-	MVS	--none--	

Command ==> \_\_\_\_\_

F1=Help F2=End F3=Return F4=INGLIST F5=Filters F6=Roll  
 F7=Backward F8=Forward F9=Refresh F10=Previous F11=Next F12=Retrieve

© Copyright IBM Corporation 2019

3-22

### DISPSTAT primary panel example

This slide shows an example of the DISPSTAT primary panel. If you enter the DISPSTAT command with no parameters, you see applications on the local system only. This slide is an example of output when the DISPSTAT command is entered with no parameters. DISPSTAT shows a list of all applications within a SAplex. Each row in the list contains information about one application. The status column is colored according to the status values. The status value colors are the same as the Status Display Facility colors. To refresh the static view of DISPSTAT, use PF9.

You can limit the applications that you see on the DISPSTAT panel. For example, enter the row-command K next to an application and you see a view of the application and its immediate children (dependent resources), only. Row-command J shows all children, the children of children until the end of the dependency chain. Row-command L does the opposite of row-command J. Again, only the start dependencies are consulted. Row-command M shows all immediate parents. You can use row commands J, K, L, and M repeatedly to alter views. PF3 returns to the previous panel. Use PF2 to exit DISPSTAT immediately.

Automation agent flag values (A, I, S, R, T, and RS) show only the flag settings that are modified with the INGAUTO command. You can use DISPSTAT row-command A (DISPFLGS) to display the effective automation flags. Automation flags are described later in this unit. The Activity column shows whether the resource is involved in a startup process (STARTING) or in a shutdown process (STOPPING).

## Setting view filters for DISPSTAT

A0FKSTA3 SA z/OS - Command Dialogs  
Domain Id . . : A0FDA ----- DISPSFLT ----- Date . . . : 10/25/18  
Operator Id : INGC102 Time . . . : 11:19:33

**Resources** ( \*, \*esour\*, resource or jobname )  
**RV\***

**Statuses** ( \*, status or choice, leading ^ or ö negates)  
**broken**

Precoded: 1 All resources that are currently UP or ENDED  
2 All resources that are in a normal status  
3 All resources that are in a down status  
4 All resources that are in a start transition state  
5 All resources that are in a stop transition state

**Targets** ( system name, domain ID, sysplex name or \* )  
**MVSA**

Command ==>  
F1=Help F2=End F3=Return F4=Clear F5=Reset F6=Roll  
F9=Save F12=Retrieve

Define your  
filters and press  
ENTER to  
refresh  
DISPSTAT

© Copyright IBM Corporation 2019

3-23

### Setting view filters for DISPSTAT

From the primary DISPSTAT panel, you can use PF5 (FILTERS) to set viewing filters. On the DISPLT panel, you specify filtering criteria. In a multisystem, SAplex environment, DISPSTAT supports displaying resources in one, several, or all NetView domains in that SAplex.

When monitoring a single system, enter the system name in the **Targets** field and press PF9. This command saves the values that are entered until either the filters are cleared with RESET command or the user logs off.

PF4 clears any filter values that are entered on the panel. PF5 sets filters back to the values shown when you entered the panel. Pressing PF12 retrieves commands that were entered while in Automation dialogs.

## Row-command E: Explanation of agent resource status

CNMPNL1.AOFH\$XPL      HELP EXPLAIN STOPPED      LINE 0 OF 17  
 ----- Top of Data -----\*

SA z/OS - STOPPED Help

The application has been shut down by an external agent, such as an operator cancel. SA for z/OS is not permitted to restart it and will not allow Automatic Restart Manager to restart it. This status is preserved across a recycle of SA for z/OS or a re-IPL of the processor, unless the application has its Restart after IPL option set to START or NONE.

An application remains in STOPPED status until an operator uses the SETSTATE command to change its status to either RESTART or CTLDOWN.

An application may also leave STOPPED status if it is restarted outside the control of SA for z/OS. In this case, it goes to either the ACTIVE or the UP status, depending on which message is

----- Bottom of Data -----

CMD	Resource	Status	System
---	---	---	---
—	RV04	STOPPED	MVS8
—	RV05	UP	MVS8
—	RV06	DOWN	MVS8

**DISPSTAT** row-command E or command: EXPLAIN STOPPED

Row command E: Explanation of agent resource status

Enter an E command to see a screen that explains the status of that application. You can also enter EXPLAIN status on any command line (also EXPLAIN *subsys\_name*).

EXPLAIN provides information for both agent and manager status values. You can also directly link to the EXPLAIN information for other status values, such as those for agents and subsystems.

## Status Display Facility (SDF) is a SysOps function

- Operates within Tivoli NetView for z/OS, independently of the automation agent and other monitoring
- Maintains and displays agent status
  - Applications and application groups
  - Monitor resources, tape, TWS, Configuration Refresh
  - Gateway sessions
  - Critical messages and WTORs
  - Processor Operations resources: processors, LPARs
- Status updates aggregate up using a tree structure
- Status updates are dynamic
- SDF focal point can display z/OS systems enterprise wide
  - Status is sent through gateway sessions using XCF and RMTCMD
- Is based on the Inform List policy
- Multiple concurrent users, displaying different panels
  - Same set of status
  - All displays updated immediately

### *Status Display Facility*

The SDF command starts the Status Display Facility. SDF consists of a hierarchy of panels that show messages, the status of system, application, monitor, processor operations resources, and other resources on a full screen, 3270 display.

The status conditions that can be displayed by SDF include those for:

- Applications and subsystems
- Monitor resources
- WTORs
- *Gateways (SA z/OS only): A gateway is a combination of a NetView to NetView task session and two automation operators (one on each of two systems) that allows communication of messages, commands, and responses between the two systems.*
- Application groups
- Configuration Refresh
- TWS
- Tape
- Exceptional messages
- Processors
- GDPS also uses SDF

SDF uses a tree-like hierarchy established during SDF customization to propagate status and related colors up to the highest level panel, the root panel. The color reflects the highest priority status of a status item representing for example a resource, a group or a system.

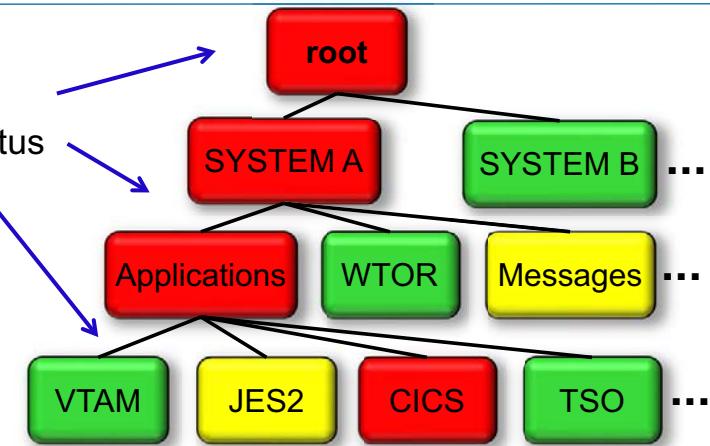
The display is automatically updated when the status of a resource changes.

The SDF focal point can display z/OS systems enterprise wide. The status is sent through gateway sessions using XCF and RMTCMD.

The Inform List policy can control SDF updates. For example, you can filter monitor resources from the SDF panels by not defining SDF in the Inform List policy for the monitor resources. Multiple users can access Status Display Facility at the same time. Although each user might be viewing a different Status Display Facility panel; the status that each user sees is the same. When a Status Display Facility status is updated, each user sees the update immediately. If one user deletes an entry, that entry disappears from the panels of all users.

## Status Display Facility panels

- Panel types below root:
  - Status and customized status
  - Detail status panels
- Use 3270 screens:
  - all screen sizes
  - full attributes
- Are customizable
- Can set PF keys to commands
- Provide dynamic display of resources, their status, and associated messages



© Copyright IBM Corporation 2019

3-26

### Status Display Facility panels

Status Display Facility is a 3270 application within the agent.

Root, status, customized status, and detail status panels can be navigated up and down. The panel hierarchy can be different from the status tree hierarchy.

A typical panel hierarchy is root showing all systems, second level are panels for each system and then from each system to its messages, resources, resource groups, and more. Status panels can display resources or messages sorted after priority.

The Automation administrator must customize the Status Display Facility panels for the installation. Panels can be tailored to display the status of resources across multiple systems, resources on a single system, or all resources of a specific type; for example, CICS resources. You can customize SDF to show group status, selected messages, and more. The Automation administrator customizes all SDF panels, except the Details view.

PF keys can be set to commands that are sent to the target system using the gateway sessions without the need for the operator to be logged on in the target system.

## SDF system status display example

```

MVSAMAIN          SUMMARY-STATUS
                  MVSA
!----- Resources -----!
>APPLS
>GROUPS
>MONITORS
!----- Messages -----!
>WTOR           >MESSAGES
!----- Special Items -----!
>GATEWAY        >Tape Devices
>TWS
10/25/18 11:24
====> █
1=Help 2=Detail      6=Roll  8=Zoom      12=Top

```

© Copyright IBM Corporation 2019

3-27

### SDF system status display example

The example on this slide shows a high-level view of resource categories that are defined for a single system. The panel title, SUMMARY STATUS MVSA and the panel ID, MVSAMAIN are two items that were changed on the supplied panel.

This example shows a summary (high-level) view of the state of a single system. The view divides the summary into three sections: resources, messages, and special items. By positioning the cursor over an item and pressing the PF8 (Zoom) or PF2 (Details), you can display more information about the item. SDF uses color to indicate status. Normal status is typically green.

The panel shows the shipped sample entry panel INGPMAIN.

The yellow color of Monitors shows that a Monitor resource has a health state that is not normal. If you want to have a deeper look into Monitors place the cursor on Monitors and press PF8 for Zoom. This invokes the defined “DOWN” panel.

The next slide shows the result of pressing PF2 (Details) with the cursor positioned over Monitors.

## Status details display example

```
1 of 3          ---- Detail Status Display ----      10/25/18 11:26:31
Component . . . : MTRNETV           System . . . . : MVSA
Color . . . . . : YELLOW            Priority . . . . : 450
Date . . . . . : 10/25/18           Time . . . . . : 11:21:50
Reporter . . . . : AUTWRK03        Node . . . . . : AOFDA

Info . . . . . : MTRNETV
Reference value : MTRNETV
User data . . . . : USER Data  
240 bytes

AOF550I STATUS OF MTRNETV MONITOR : OBSERVED=AVAILABLE HEALTH=WARNING -
MTRNETV: NETVIEW TOTL 254 62.66 100.00 0.00 0      35331      N/A

====> 1=HELP 3=RETURN 4=DELETE 6=ROLL 7=UP 8=DOWN 11=BOTTOM 12=TOP
```

© Copyright IBM Corporation 2019

3-28

### Status-details display example

This slide shows an example of an SDF Status Details panel which can not be customized with the exception of the PF Keys. The display includes the status component, priority and color, location, time stamp, user data and the message that caused the status.

The Detail Status Display Panel could have been invoked already on the previous panels using PF2 for the detail status on the status field. For showing the 'worst' status there is no need to dive down, just press PF2 and you see the propagated status information.

You can use the following PF keys:

- PF3 Returns to the previous panel
- PF4 Deletes the status you are currently viewing. (The record is physically deleted.)  
Note: You should not normally use this PF key.
- PF7 Scrolls back up the Detail Status Display panels in order of descending status priority.
- PF8 Scrolls down the Detail Status Display panels in order of descending status priority.
- PF11 Moves to the Detail Status Display panel for the resource with the lowest priority, that is, the last panel in the Detail Status Display stack.
- PF12 Moves to the Detail Status Display panel for the resource with the highest priority, the first panel in the Detail Status Display stack.

# Lesson 3 Other frequently used commands



## Lesson 3. Other frequently used commands



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson introduces some additional automation management commands. With the exception of INGHIST they can also be called from the INGLIST or DISPSTAT panels.

## Other frequently used commands

Command	Description
<b>INGINFO</b>	Automation manager display of automation details for a specific resource
<b>INGHIST</b>	Display historical information that the automation manager saved
<b>DISPINFO</b>	Automation agent display of automation details for a specific resource
<b>INGRELS</b>	Display relationships that are defined for a specific resource
<b>DISPTREE</b>	Display a tree view of related dependencies for the starting or stopping of a resource
<b>INGREQ</b>	Used to start and stop resources
<b>INGVOTE</b>	Manage requests and votes

### *Other frequently used commands*

**INGINFO:** Automation manager display of automation details for a specific resource

**INGHIST:** Display historical information that the automation manager saved

**DISPINFO:** Automation agent display of automation details for a specific resource

**INGRELS:** Display relationships that are defined for a specific resource

**DISPTREE:** Display a tree view of related dependencies for the starting or stopping of a resource

**INGREQ:** Used to start and stop resources

**INGVOTE:** Manage requests and votes

## INGINFO: Display automation details about a resource

- Automation manager details for a specified resource
  - Resource status: Compound, observed, desired, automation, startability, health
    - Compound status also includes time stamp of last change
  - Flags
  - Dependency status
  - Current order
  - Requests and votes
  - Trigger and service period information
  - Relationships
  - Desired available policy
  - Pacing gate
  - Runmodes
  - Start and stop types, plus commands
  - Group information
  - History
- If the resource type is APL, PF4 starts DISPINFO (agent) display

© Copyright IBM Corporation 2019

3-31

### INGINFO: Display automation details about a resource

The INGINFO command displays automation details about a selected resource. The automation manager shows these details. The command output is similar to the DISPINFO command, which shows the automation agent view of automation details for a resource. Pressing PF4 on the INGINFO panel, displays the DISPINFO (agent display) panel for the resource.

For a REF resource PF4 opens the INGDMN display.

You can use one of the following methods to run the INGINFO command:

- Entering the command INGINFO *resource\_name* on the command line
- Entering row-command **F** next to a resource on the INGLIST primary panel

## INGINFO: Example panel

```

INGKYINO          SA z/OS - Command Dialogs      Line 1  of 750
Domain Id . : A0FDA   ----- INGINFO ----- Date . . : 10/25/18
Operator Id : INGC102 Sysplex = SYSPLEX1 Time . . : 20:59:45

Resource    => RV06/APL/MVSA      format: name/type/system
Target      => _____ System name, domain ID or sysplex name

Resource     : RV06/APL/MVSA
Description   : RV06 Appl has service period  RV56SVP

Inform List   : SDF

Status...
Observed     : SOFTDOWN
Desired       : UNAVAILABLE
Automation    : IDLE
Startability  : DENIED
Compound      : SATISFACTORY   Last changed : 2018-10-25 10:09:26
Health Status : N/A

Dependencies...
PreStart      : Satisfied
Start         : Satisfied
PreStop       : Satisfied
Stop          : Satisfied
Startability  : Satisfied

Trigger       : -None-

Command ===>
F1=Help   F2=End   F3=Return   F4=DISPINFO
F8=For    d        F9=Refresh  F6=Roll
                                         F12=Retrieve

```

Row command on INGLIST panel

CMD	Name	Type	System
f	RV05	APL	MVSA
	RV06	APL	MVSA

© Copyright IBM Corporation 2019

3-32

### INGINFO: Example panel

This slide shows an example of the first INGINFO panel for a selected resource. The command displays the information in fields instead of columns. The upper right corner of the panel shows the number of lines of information that are available.

The FIND and RFIND subcommands can be used to find information. For example, the following command, scrolls the panel and positions the row which contains the Startability field, as the first row below the Target field.

```
find STARTABIL
```

You can also choose the direction of the search by adding to the subcommand **N**, (Next) or **P**, (Previous). Scroll forward to see more information about the resource. PF4 displays the agent view of the information.

## INGINFO: Example panel (continued)

### Showing history

```

INGKYINO          SA z/OS - Command Dialogs      Line 80   of 754
Domain Id . : AOFDA   ----- INGINFO ----- Date . . : 10/25/18
Operator Id : INGC102  Sysplex = SYSPLEX1    Time . . : 21:06:35

Resource      => RV06/APL/MVSA      format: name/type/system
Target        => _____ System name, domain ID or sysplex name

History       :
2018-10-25 21:00:00  Multiple schedule requests at 2018-10-25 21:00
HSAL6183I No votes present for desiredStatus
HSAL6308I New desiredStatus request recorded
HSAL6349I Group Desired Status Request changed
HSAL6348I Group Observer Update Requested
HSAL6251I Status/Desired is Available
HSAL6195I Resource should be prestarted
HSAL6198I Resource should not be prestopped
HSAL6138I Resource should be started
HSAL6141I Resource should not be stopped
HSAL6172I Group Observer update sent
2018-10-25 11:01:21  Status for RV06 set to IDLE by suspend process
HSAL6269I Status/Automation is Idle
HSAL6348I Group Observer Update Requested
HSAL6327I Resource is either down or not expected down
HSAL6172I Group Observer update sent
HSAL6525I Winning suspend request is satisfied
2018-10-25 11:01:21  Status for RV06 set to BUSY by suspend process
HSAL6271I Status/Automation is Busy
HSAL6348I Group Observer Update Requested

Command ===> █
F1=Help   F2=End   F3=Return   F4=DISPINFO   F6=Roll
F7=Backward F8=Forward F9=Refresh   F12=Retrieve

```

© Copyright IBM Corporation 2019

3-33

#### INGINFO: Example panel showing history

An important section of the INGINFO output is historical information. Scrolling forward or entering the following command displays the History section.

```
find history
```

Historical information can span several days. The data is displayed in an easier-to-read format than similar data in the NetView or z/OS log.

The INGINFO command shows information from the automation manager only. The history log does not show any detailed agent actions. For example, you might see that the last order sent to the automation agent was MakeUnavailable; but not the shutdown pass that the agent is processing.

History that the INGINFO command shows is stored as persistent data in the takeover file. A WARM or COLD start of the automation manager deletes the history. You can suppress display of resource history by adding the NOHIST parameter to the INGINFO command. The INGHIST command also displays resource automation history.

## DISPINFO: Agent display of automation details for a resource

- Detailed information about a specified resource
  - Everything that the automation agent uses to manage a resource
  - Run the command from these locations:
    - The DISPSTAT panel by entering row command F
    - The INGINFO panel by pressing PF4
    - The command line `DISPINFO resource_name`
- Examples:
- `DISPINFO resource_name`
  - `DISPINFO RV03` (displays a panel on which a target system is specified)

### *DISPINFO: Agent display of automation details for a resource*

The DISPINFO command displays automation details for a resource from the perspective of the automation agent. The command can be issued from the locations shown on the slide.

## DISPINFO: Example panel

```

AOFKINFO                               SA z/OS - Command Dialogs      Line 1      of 189
Domain Id . : AOFDA ----- DISPINFO ----- Date . . : 12/12/18
Operator Id : INGC102                  Time . . : 04:19:55

Subsystem  => JES2          Target  => MVSA      System name, domain ID
Subsystem   : JES2          on System : MVSA or sysplex name

Description : Job Entry Subsystem 2

Inform list : SDF

Class chain : C_JES2          Class for Job Entry Subsystem 2
Job Name    : JES2
ASID        : 0020
Job Type    : MVS
Category    : JES2

Command ==> _____
F1=Help   F2=End   F3=Return   F4=INGINFO
F8=Forward F9=Refresh
                                         Row command on DISPSTAT panel
                                         f_ JES2      UP      MVSA
                                         F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

3-35

### DISPINFO example

This slide shows an example DISPINFO output panel. The fields on this panel show information about a selected resource. Notice that the DISPINFO panel refers to the resource as a subsystem.

Similar to INGINFO panels, DISPINFO arranges information about the resource in fields instead of columns. DISPINFO displays the application classes to which an application is linked. In this example, RV04 is linked to RVCLASS, which in turn is linked to RV\_MSTR\_CLS.

Scrolling forward (PF8), displays more information such as agent automation flags, restart parameters, thresholds, start and stop phases, and captured messages.

## INGHIST: Display automation history for a resource

- Displays work item history
- Messages are written to the system logger
- Start and end time parameters can be specified
- **Find** and **Rfind** commands are supported
- Command examples
  - INGHIST
  - INGHIST RESOURCE=RMF/APL/MVSA
  - INGHIST RESOURCE=RV03 START=2019-04-01
  - INGHIST RESOURCE=RV03 MAX=5 WIMAX=10

### *INGHIST: Display automation history for a resource*

INGHIST displays history information that is recorded by the automation manager. The information history is for all work items that are processed within a specified time interval.

INGHIST also allows the display of messages that the manager issues to the system logger which requires that the System Logger has been configured.

If you enter the INGHIST command without parameters, you receive the output for the complete SAplex for the period starting from the last hour before you entered this command.

You can specify a certain resource that you want automation to display the work history for, or you can specify a time interval that you want to see all activities for, or you can combine both criteria. Additionally you can specify the maximum number of work items and the maximum number of work item records to be shown.

## INGHIST example

```

INGKYHIO          SA z/OS - Command Dialogs      Line 1   of 135
Domain Id . : AOFDA  ----- INGHIST ----- Date . . : 10/25/18
Operator Id : INGC102 Sysplex = SYSPLEX1    Time . . : 21:12:44

Start time ==> 2018-10-20 20:12:00      format: yyyy-mm-dd hh:mm:ss
End time ==> _____ Resource ==> RV05/APL/MVSA

Cmd: + Expand workitem (e) - Collapse workitem (c)
Cmd Date     Time       Footprint
--- -----
--- 2018-10-25 12:15:03 Termination processing for RV05/APL/MVSA completed
--- 2018-10-25 12:15:02 Agent status for RV05/APL/MVSA = AUTODOWN
--- 2018-10-25 12:14:50 Shutdown in progress for RV05/APL/MVSA
--- 2018-10-25 12:14:50 Agent status for RV05/APL/MVSA = AUTOTERM
--- 2018-10-25 12:14:50 No SHUTINIT commands to issue for RV05/APL/MVSA
--- 2018-10-25 12:14:50 MAKEUNAVAILABLE requested by INGC102 for RV05/APL/MVSA
--- 2018-10-25 10:55:15 Automation Agent is ready to receive start/stop orders
--- 2018-10-25 10:55:11 Automation Agent is suspended
--- 2018-10-25 10:55:09 SA z/OS V4R1 Logic Deck, 27 Jul 2017, 13:28:23, Last
APAR: OA52610
--- 2018-10-25 10:45:00 Startup for RV05/APL/MVSA completed
--- 2018-10-25 10:45:00 Agent status for RV05/APL/MVSA = UP
--- 2018-10-25 10:45:00 Agent status for RV05/APL/MVSA = ACTIVE
--- 2018-10-25 10:45:00 Agent status for RV05/APL/MVSA = STARTED
--- 2018-10-25 10:45:00 Startup for RV05/APL/MVSA in progress
--- 2018-10-25 10:45:00 No PRESTART commands to issue for RV05/APL/MVSA
--- 2018-10-25 10:45:00 Agent status for RV05/APL/MVSA = RESTART
--- 2018-10-25 10:45:00 Create schedule UP request with priority 02140000 for

Command ==> █
F1=Help   F2=End   F3=Return   F6=Roll
F8=Forward F9=Refresh   F12=Retrieve

```

© Copyright IBM Corporation 2019

3-37

### INGHIST example

Initially, for each work item, the following data is displayed:

- The timestamp (date and time) when processing of the work item started
- Footprint data that describes why the work item was submitted

As shown in this example of INGHIST, the most recent message is at the top of the list. To see more detailed information, expand the work item by typing either a plus sign (+) or E in the Cmd column.

Typical work items are status changes and begin and end of completion of startup and termination processing.

Initially, the work items are displayed in descending order. You can, however, use the SORT subcommand to rearrange the display in a specific order. For example, to display it in ascending sequence, enter SORT 1,2 A at the command line.

Use the FIND(F) and RFIND(RF) subcommand to search the displayed data for a specific string. The line containing the string is then displayed as the top line of the display.

## INGRELS: Display resource relationships

- Display all relationships that are defined for the specified resource:
  - Relationship type and direction
  - Partner resource name (supporting resource)
  - Relationship condition and condition status
- Start from INGLIST panel, option D
- Start as line command
  - Examples:
    - **INGRELS *resource\_name***
    - **INGRELS JES2/APL/MVSA**

### INGRELS: Display resource relationships

The INGRELs command displays all dependencies for a resource. It also shows whether those dependencies are currently SATISFIED. In some cases, it also shows whether the supporting resource is startable.

## INGRELS example

```

INGKYRL0          SA z/OS - Command Dialogs      Line 35   of 48
Domain Id . : AOFDA    ----- INGRELS -----  Date . . : 10/25/18
Operator Id : INGC102  Sysplex = SYSPLEX1  Time . . : 21:17:48

Resource      => VTAM/APL/MVSA
Target        => _____ System name, domain ID or sysplex name

Cmd: - Collapse + Expand E INGVOTE F INGINFO J INGGROUP S Focus

Cmd Name      Type System Dir Relationship
----- -----
JES2          APL MVSA  B PrepUnavailable/passive/WhenDown
JES2          APL MVSA  F HasParent
RV01          APL MVSA  B HasParent - Strong
RV02          APL MVSA  B HasParent - Strong
RV05          APL MVSA  B HasParent - Strong
RV06          APL MVSA  B HasParent - Strong
TCPIP         APL MVSA  B HasParent - Strong
               Cond: requiredAvailable
               Cond: stopBlocked
               Cond: stopDependency - UNSATISFIED
               Cond: reallyDown - UNSATISFIED
               Cond: allExpectedDown
               Cond: footprint
               B HasParent - Strong

+ TSO          MVSA

Use - or + to collapse or expand one relationship.
Use PF5 or PF11 to collapse or expand all relationships

Command ==> F1=Help F2=End F3=Return F9=Refresh F5=Collapse* F6=Roll
F7=Backward   F11=Expand* F12=Retrieve

```

© Copyright IBM Corporation 2019

3-39

### INGRELS example

In this example, the plus sign (+) in the Cmd column, next to TCPIP expanded the view of the backward relationship, - B in the Dir column, between TCPIP and VTAM. The Dir column shows the direction of the relationship: forward (F) or backward (B). You see all dependencies for the resource, both dependent and supporting resources are shown. Forward dependencies are supporting dependencies for the selected resource. The forward relationships are usually the ones that are of interest. Backward relationships point to resources that are dependent on the selected resource.

The minus (-), and plus (+) signs, collapse and expand individual rows. PF5 collapses all rows; PF11 expands all rows. The Cond: item shows whether a condition is currently satisfied. HasMember relationships of group members are not shown on this panel.

The Cmd field lets you also specify command codes to show the requests and votes, detailed information, and to display the INGGROUP panel for the resource.

## DISPTREE:

### Display start or stop dependencies for a resource

- Show the start/stop dependency graph for a specified resource
- Start from these locations:
  - The DISPINFO panel, option G
  - The command line
    - DISPTREE resource\_name**

Example

**DISPTREE JES2**

- Useful to determine why the resource is not starting or stopping

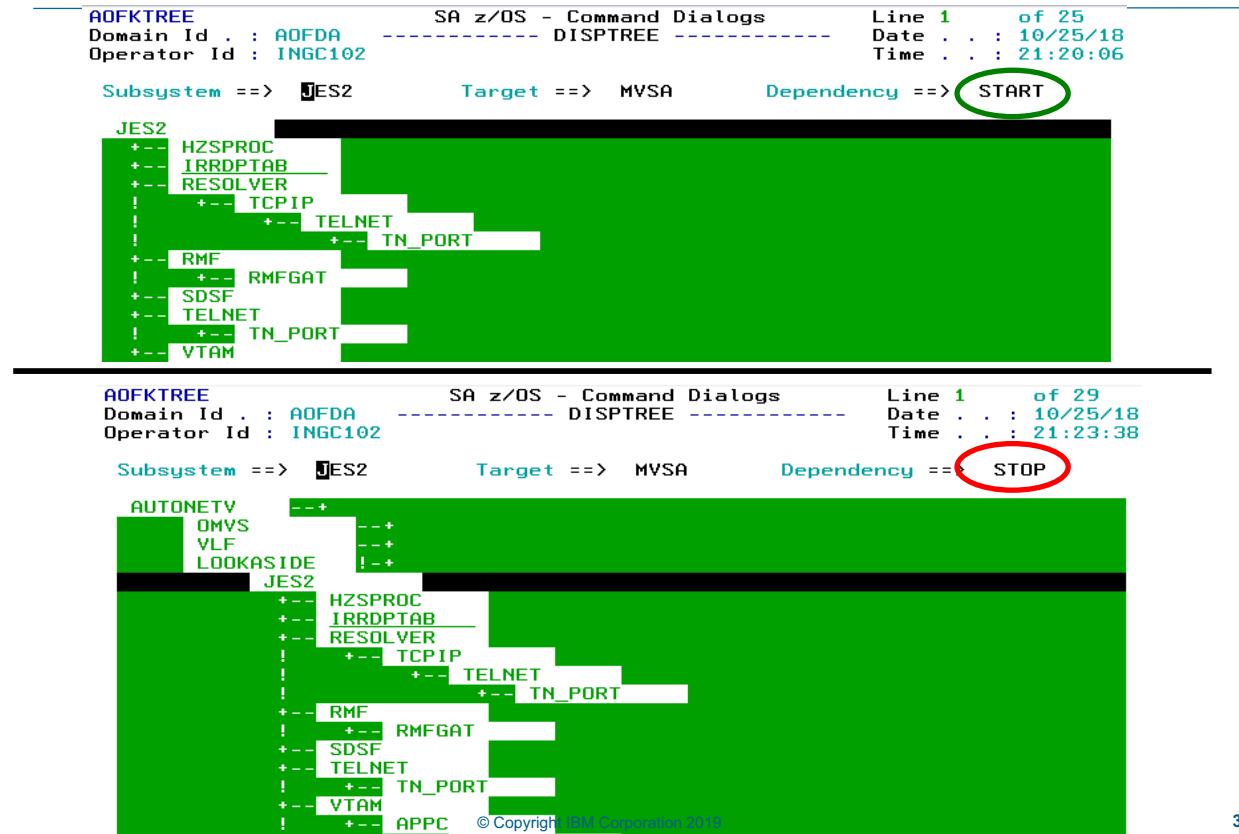
For example, the resource has a compound status of AWAITING or INHIBITED

#### *DISPTREE: Display start or stop dependencies for a resource*

The DISPTREE command can be issued from DISPINFO or the command line. The command displays a structured, tree-view of related dependencies for starting or stopping a resource.

The DEPENDENCY parameter specifies the type of dependency that should be used to draw the dependency graph: START, STOP, and GROUP.

## DISPTREE examples



3-41

### DISPTREE examples

This slide shows examples of the start and stop dependencies for JES2 on a selected system. The arrows point to JES2 which is the focus of the DISPTREE output. The start example shows the resource tree of dependencies for JES2. JES2 is the supporting resource for VTAM. The start order is from left to right, e.g. start JES2 and then VTAM or start TCPIP and then TELNET.

The reverse is true for the stop: The stop order is from right to left. VTAM is the supporting resource for JES2. The stop order is stop VTAM and then JES2 or stop TCPIP after TELNET.

To change the focus, position the cursor at an application in the tree and press PF4.

You can also see the groups to which an application belongs by making it the focus then selecting GROUP dependency. DISPTREE can help users to determine the reason applications are not starting or stopping. When an application has a compound status of awaiting or inhibited, the cause is usually an unsatisfied dependency. Dependency trees also show the agent status of each application, highlighting the possible cause. This view also shows any loops in parent-child relationships.

Place the cursor on an application and press PF5 to see a DISPINFO display for that application. If there is a dependency to a group, it is shown. The group name is displayed with a G next to the group name. If there is a dependency to a resource on another system, the resource is displayed with an X next to its name.

## INGREQ: Start or stop a resource

Domain Id . . . : AOFDA	-----	SA z/OS - Command Dialogs	-----	Page . . . : 1 of 2
Operator Id : INGC102	-----	INGREQ	-----	Date . . . : 10/25/18
				Time . . . : 21:27:16
<b>Resource Target</b>	=> RV02/APL/MVSA	format: name/type/system		
		System name, domain ID or sysplex name		
<b>Request Type</b>	=> STOP	Request type (START/UP, STOP/DOWN or CANCEL)		
<b>Type</b>	=> NORM	Type of processing (NORM/IMMED/FORCE/user) or ?		
<b>Scope</b>	=> ALL	Request scope (ONLY/CHILDREN/ALL)		
<b>Priority</b>	=> LOW	Priority of request (FORCE/HIGH/LOW)		
<b>Expire</b>	=>	Expiration date(yyyy-mm-dd), time(hh:mm) - see help for list		
<b>AutoRemove</b>	=>			
<b>Restart</b>	=> NO	Restart resource after shutdown (YES/NO)		
<b>Override</b>	=> NO	(ALL/NO/TRG/FLG/DPY/STS/SUS)		
<b>Verify</b>	=> YES	Check affected resources (YES/NO/WTOR)		
<b>Comment</b>	=>			
<b>Appl Parms</b>	=>			

AOF710A VERIFY/REVISE INPUT AND THEN PRESS ENTER

Command ==>			
F1=Help	F2=End	F3=Return	F6=Roll
F8=NextPnl		F12=Retrieve	

© Copyright IBM Corporation 2019

3-42

### INGREQ: Start or stop a resource

The INGREQ command creates a request to make a resource Available (start) or Unavailable (stop). The INGREQ panel is displayed when a user enters the start row-command (B), or stop row-command (C) next to a resource on the INGLIST primary panel. The command can also be run from the command line: INGREQ *resource\_name*.

On a panel similar to the one shown on this slide, you enter request information in the fields and press ENTER. Automation displays a verification panel if YES is specified in the **Verify** field on the primary INGREQ panel.

Advanced Automation Options (controlled by the settings of specific **AOF\*** and **ING\*** common global variables), are extended to support default values for the request. Valid parameters for the data fields are in the *IBM System Automation for z/OS: Operator's Commands*.

In the **Appl Parms** field, specify any modifications or parameters to the start or stop request. This setting applies primarily to CICS, DB2, and IMS subsystems.

The request and its attributes are persistent. For example, if you specify a start type and an override, Automation uses the same overrides when restarting, if it or the system fails. The **Expire**, **AutoRemove** and **Timeout** fields can be used to remove requests automatically.

Use the **Comments** field to document the reason for the request. The Comments field provides a mechanism to keep a record of the changes.

The **Interrupt** field, which is seen by pressing PF8 or PF11, is applicable only to shutdown requests. If this field is set to YES and the resource is starting, the automation manager interrupts the startup process and starts shutting down the resource. A typical use for Interrupt=YES is when escalating to a higher start type.



**Attention:** Before entering a request use INGVOTE to determine whether there is already a request or vote that could be canceled to achieve the same result.

## INREQ verification panel

- Displayed when the verify field is set to YES (default)
- Indicates success of request
  - If the affected resource is in the required state, then OK
  - If the affected resource is not in the required state, but the winning vote and the action are indicated, then OK
  - If there are no negative service period and trigger indications, then OK
- Not a guarantee that order will be sent
  - Dependencies: Satisfied or not satisfied
  - Manager or agent flag settings can be a delay

### INREQ verification panel

The verification panel gives the requester an opportunity to verify and confirm or cancel the request. All resources that are affected by the request are shown on the verification panel. The display includes the following information:

- Resources that are within the SCOPE specified for the request
- Supporting resources in the active dependency chain
- Each resource that receives a vote for the start or stop request. These resources are not necessarily the same resources that are checked before starting or stopping the resource.

For start requests of groups, or if a group is an affected resource, only the group resource name is shown. The members of the group are not shown. Groups are explained in another unit.

## INREQ verification panel example

SA z/OS - Command Dialogs						Line 1 of 11
INREQ						Date . . . : 10/25/18
						Time . . . : 21:28:45
<b>Verify affected resources for request STOP</b>						
<b>CMD: S show overrides T show trigger details V show votes</b> <b>Cmd Name Type System TRG SVP W Action Type Observed</b>						
APPCL	APL	MVSA		Y	STOP	NORM AVAILABLE
ASCH	APL	MVSA		Y	STOP	NORM AVAILABLE
RV01	APL	MVSA		Y	STOP	NORM SOFTDOWN
RV02	APL	MVSA		Y	STOP	NORM AVAILABLE
RV05	APL	MVSA		Y	STOP	NORM SOFTDOWN
RV06	APL	MVSA		Y	STOP	NORM SOFTDOWN
TCPIP	APL	MVSA		Y	STOP	NORM AVAILABLE
TELNET	APL	MVSA		Y	STOP	NORM AVAILABLE
TN_PORT	APL	MVSA		Y	STOP	NORM AVAILABLE
TSO	APL	MVSA		Y	STOP	NORM AVAILABLE
VTAM	APL	MVSA		Y	STOP	NORM AVAILABLE

Verification panel is displayed if VERIFY=YES (default)

Command ==> F1=Help F2=End F3=Return F9=Refresh F10=GO F11=CANCEL F6=Roll F12=Retrieve

### INREQ verification panel example

As shown in the example on this page, Automation lists the resources that are affected by the request. Y or blank in the W column shows that the resource received a winning vote. The votes are shown even if the status of the listed resource is not changed. What you see on this panel is each resource that received a vote. Pressing PF10 on this panel signifies a go-ahead for the requested action. Remember that inhibitors prevent orders from being sent to the automation agent.

If the W column for a resource is blank, it means the priority of vote for the resource is lower than an existing vote. Therefore, the request was not successful. Check the status of the resource to see whether it is already in the desired state.

The TRG column tells whether triggers are linked to the resource and if they are satisfied (S) or unsatisfied (U). If the trigger is inherited as a result of group membership, the value has a prefix of the letter G. For example, GU indicates that there is an unsatisfied group trigger. The SVP column contains data if there is a service period request for the resource.

## INGVOTE: Managing resource votes

- Display votes for a resource
  - Option to display winning or losing votes only
  - Command can be issued from these locations:
    - The INGLIST panel, row command E
    - The command line
- Example: **INGVOTE resource\_name,filter\_criteria**

### INGVOTE: Managing resource votes

While monitoring shutdown processing, you can use the INGVOTE command to see the request details for votes that are active for one resource or all resources.

# INGVOTE example

```

INGKYRQ0          SA z/OS - Command Dialogs      Line 1   of 10
Domain Id . : AOFDA    ----- INGVOTE -----  Date . . : 10/25/18
Operator Id : INGC102  Sysplex = SYSPLEX1  Time . . : 21:38:32

Resource  => RV05/APL/MVSA
Target    => _____ System name, domain ID or sysplex name

Desired Available...: Always

Cmd: C cancel request   K Kill request     S show request details
Cmd Action WIN Request/Vote Data
--- -----
-- START   Y Request : MakeAvailable_Only
           Created : 2018-10-25 21:36:38
           Originator : OPERATOR(INGC102)
           Priority : 01740000 Should Be Up - Operator
           Status : Winning/Satisfied
-- STOP    * Vote : MakeUnavailable
           From Req. : MakeUnAvailable for RV56SVP/SVP
           Created : 2018-10-25 21:37:00
           Originator : SCHEDULE
           Priority : 01120000 Should Be Down - Schedule

```

The MakeAvailable request from the operator received a higher priority than the MakeUnavailable request from the schedule. Although both requests were low-priority (first 2 characters are 01), the source value (7) and request type value (4) are higher than the source value and request type of the schedule request

Command ==> █  
F1=Help F2=End F3=Return F9=Refresh F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

3-46

## INGVOTE example

The example on this slide shows that there are two votes for the resource. The winning vote (WIN=Y) is a START (MakeAvailable) vote from an OPERATOR. The other vote is a STOP vote that is propagated to the resource by a service period.

The information has these indications:

- The time and date the request was submitted
- Internal priority
- Whether it is a winning vote
- Whether it is satisfied

Other details, such as start type, are seen only by looking at the actual request details. Details (option **S**) show whether Override, Timeout, Expire, Autoremove, Restart operands were specified. It also shows the start or stop type that is specified, and any application parameters.

You can set view filters to see specific votes; for example, winning only or from a certain source. Source is the value in the Originator field.

Entering row-command **C** in the command column of a request on an INGVOTE display means that you want to cancel the request. You cannot cancel a vote. Canceling a request cancels all the votes that the request created.

Before the request is canceled, you must respond with the GO command (PF10) on a confirmation panel. On the confirmation panel, you see a list of the votes and any automation policy items that are linked to the targeted resource.

If you want to cancel a request without knowing the details, you can use the **K(ill)** row-command. Use the **K** row-command only when you are sure of the consequences of this action.

The **K** row-command does not display a confirmation panel. There is no opportunity to cancel the command.



**Note:** Avoid removing requests from service periods. Instead, use the command INGSCHED to manage the service periods.

The INGVOTE command with no operands, lists all requests that are known to the automation manager. You see all requests for all systems in the SAplex. Normally, there are not many requests from the OPERATOR and AUTOOPS sources because these requests represent persistent changes to the normal policy. You might see requests from IBM Workload Scheduler (EXTERNAL requests) and from Automation service periods (SCHEDULE requests). The SOURCE and EXCLUDE parameters of the INGVOTE command filter the results of the command.

# Lesson 4 Starting and stopping resources



## Lesson 4. Starting and stopping resources



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to start and stop resources using requests and what the conditions are for a successful start and stop.

## Starting and stopping resources

- Conditions for a successful start
- Conditions for a successful stop
- Shutdown phase processing
- Display and managed non-automated jobs

### Starting and stopping resources

- Conditions for a successful start
- Conditions for a successful stop
- Shutdown phase processing
- Display and managed nonautomated jobs

## Conditions for a successful start

- Compound status of application is not PROBLEM
- Request must create highest priority vote for resource
- All inhibitors are satisfied
  - MakeAvailable dependencies
  - Start trigger, if any
  - Manager flags must allow automation
- Agent automation flags:
  - RESTART flag is YES when started from a RESTART status
  - INITSTART flag is YES when started from a DOWN status
  - Note: START flag only automates WTORs and messages that are issued during startup, not the start commands

### Conditions for a successful start

This list shows some of the conditions that must apply before a start request is successful.

A compound status of PROBLEM prevents further automation status changes. In such a case, status is an *inhibitor*.

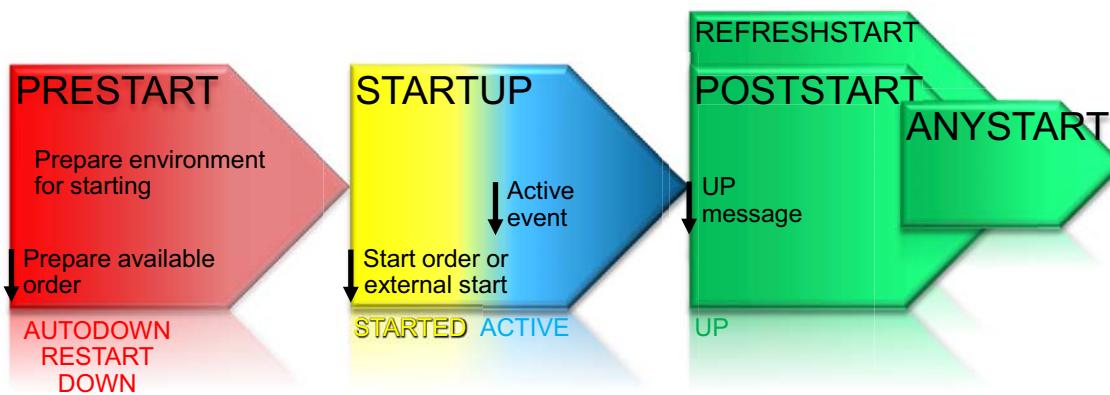
Before the manager sends an order, a number of possible *inhibitors* are checked. These inhibitors can delay the sending of the order until certain conditions are satisfied, or overridden. When all inhibitors are satisfied, the manager sends the order to the agent.

- MakeAvailable dependencies must be satisfied. If supporting resources are not up, Automation tries to start them for active dependencies. This is called vote propagation
- Start triggers, if any, must be satisfied
- Manager flags must allow automation. This includes the suspend flag
- Agent automation flags must allow automation. The flags are different for initial starts and restarts
- Agent must not be suspended

In addition to the conditions listed here, the start is successful only if the resource is not in a shutdown phase. See [Unit 8, “Troubleshooting”](#) for tips about debugging start failures.

## Starting applications

- Create a persistent MakeAvailable request
  - Override to the defined policy
  - Consider use of timeout, expire, or autoremove parameters to cancel the request
- A start has three major phases:
  - PRESTART - Prepare environment for starting, if required
  - STARTUP - The actual start commands
  - POSTSTART, REFRESHSTART, and ANYSTART
- A request might not be successful because of
  - Other requests, Start dependencies, Triggers, Status



© Copyright IBM Corporation 2019

3-50

### Starting applications with the INGREQ command

A request to start an application is a request to change its desired status. The INGREQ START command creates a MakeAvailable request for the application. Requests are persistent. The automation manager treats the request as the new automation goal for the resource, and the request stays in effect until it is canceled.

Consider any INGREQ START request as a deviation from the normal operation of the enterprise. Consider when that request must be canceled. Remember, a single request can generate multiple votes.

The start of an application is a three-phase process. The first is PRESTART which occurs as soon as there is a desired status of AVAILABLE. Also, if defined, the PrepAvailable condition must be satisfied. The PrepAvailable completion happens long before the application is ready to start. The second phase is the STARTUP phase when the application starts. POSTSTART, which occurs when the application is fully initialized (UP or ENDED), is the third phase.

Creating a request to set the desired status of an application to AVAILABLE might not be enough to start the application. Other requests can create MakeUnavailable votes. One of those votes can win, and the desired status remains UNAVAILABLE.

The desired status can be AVAILABLE, yet there might be these inhibitors that prevent the manager from sending the order to the agent:

- Unsatisfied dependencies
- Unset start-triggers
- Manager flags
- An observed status of HARDDOWN

You can use the INGMDFY command to change start phase commands and replies for the next start of an application. The INGREQ command can be initiated from the INGLIST display panel or on the command line. When used as a line command, INGREQ requires several parameters. These parameters are documented in Part 2 of the *IBM System Automation for z/OS: Operator's Commands*.

# Starting applications

If	Then
Desired state is UNAVAILABLE	Resolve the request that set the desired state.
SCHEDULE, normal automation, or IBM Workload Scheduler for z/OS set the desired state to UNAVAILABLE	Take one of these actions: <ul style="list-style-type: none"><li>▪ Use INGSCHED to modify SCHEDULE</li><li>▪ Use IBM Workload Scheduler for z/OS to modify its schedule</li><li>▪ Use INGREQ to enter a new operator request</li></ul>
Override is required to start resource because of these items: <ul style="list-style-type: none"><li>▪ Inhibitors</li><li>▪ Dependencies</li><li>▪ Triggers</li></ul>	Verify operational procedures with your administrator, for example: <ul style="list-style-type: none"><li>▪ Use INGREQ to enter a new operator request</li><li>▪ Optionally, cancel any other conflicting requests</li><li>▪ Cancel the operator request</li></ul>
Desired state is AVAILABLE and Observed Status is HARDDOWN	Use SETSTATE to set the agent status to AUTODOWN

## Starting applications

There are several ways to start a resource. This slide indicates the techniques for starting a resource. Remember that in Automation, a request to start a resource is really a request to make its desired status AVAILABLE. The actual start is implicit in the request.

Whenever an INGREQ START is issued for an application that is already available, an additional request is added to the request queue for the application. This situation is an exception to normal operating procedure. To override start inhibitors, use the INGREQ START command.

## Conditions for a successful stop

- Compound status of application is not PROBLEM  
Automation status of PROBLEM prevents shutdown
- The request must create the highest priority vote for the resource
- The application must have no inhibitors
- The application must not be suspended
- The agent TERMINATE (shutdown) flag must be set to YES
- The appropriate SHUTDOWN commands must be defined in the automation configuration file  
If the commands or replies are not specified for IMMED and FORCE shutdown, then NORM specifications are used

### Conditions for a successful stop

The target of the shutdown request and each supporting resource must have an acceptable COMPOUND status. Automation cannot shut down applications that have a COMPOUND status of PROBLEM. The problem must be resolved, and the status reset before the request is accepted. The request must generate the winning vote for the resource. Inhibitors must be satisfied or overridden. If necessary, you can override all inhibitors except status (STS) using INGREQ.

The effective value of the agent TERMINATE flag must be YES. Otherwise, the agent continues to retry until the value of the flag is YES. The COMPOUND status of the application is set to STOPPING.

There must be appropriate commands or replies to effect the shutdown. PRECHECK processing normally detects if the commands or replies are available. If no commands or replies are available, the request is not entered. If PRECHECK is set to NO and these commands are missing, the application immediately goes to an agent status of STUCK. See Lesson 1, "Solving problems with resources," on page 359 for tips on debugging stop failures.

## Considerations when stopping applications

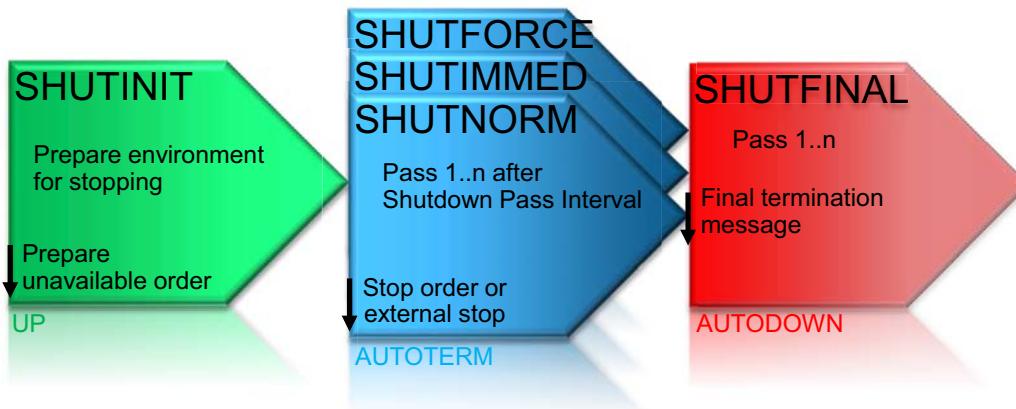
If	Then
Desired state is UNAVAILABLE <b>and</b> Observed status is PROBLEM	Resolve at the resource level so that the resource achieves UP status <b>or</b> Use SETSTATE to set the agent status to UP
SCHEDULE or IBM Workload Scheduler for z/OS set the desired state to AVAILABLE	Use INGSCHED to modify SA z/OS schedule (add a down window) <b>or</b> Use IBM Workload Scheduler for z/OS to modify its external schedule
Normal automation set the desired state to AVAILABLE	Use INGREQ to enter a new operator request

### Considerations when stopping applications

This chart shows the ways of stopping, or shutting down applications, based on their initial conditions.

## Stopping applications

- A stop has three major phases:
  - SHUTINIT - Prepare environment for stopping
  - SHUTxxxx - The actual stop commands
    - NORM - Normal shutdown
    - IMMED - Immediate shutdown
    - FORCE - Force a subsystem off the system
  - SHUTFINAL - After stop is completed
- Multiple passes for SHUTxxxx and SHUTFINAL, if required



3-54

### Shutdown phases

Similar to starts, stopping an application is a three-phase process. The SHUTINIT phase is scheduled when the desired status of the application is set to UNAVAILABLE and any PrepUnavailable dependency is satisfied. Commands to prepare the application for stopping can be issued, such as:

- Blocking further login requests
- Draining internal processes
- Reducing facilities
- Sending warning messages to users

The application is stopped in the SHUTxxxx phase. You can define three types of shutdown commands: NORM (*normal*), IMMED (*immediate*), and FORCE. Different commands can be specified for each stop type in each of the phases.

The application is inactive in the SHUTFINAL phase. This phase can include commands to stop monitoring, do clean up, or vary devices offline.

During the SHUTDOWN phase, the automation agent uses a series of defined passes. Each pass is a shutdown command. The commands can increase in severity, similar to the actions that a physical operator takes when trying to stop an address space.

For example, on a VTAM shutdown the passes can be as follows:

Pass 1: Z NET, QUICK  
Pass 2: Z NET, CANCEL

A first attempt is made to close the application - pass 1. In many cases, the command in pass 1 is enough to stop VTAM and set the agent status to AUTODOWN. The shutdown is then effectively complete, although SHUTFINAL commands can then be issued. However, in some cases the address space stays up. In such a situation, Automation issues the more severe command that is specified in pass 2.

The automation agent also uses a shutdown interval. It issues PASS1 commands, waits for the shutdown interval, checks for a continuing active address space, and then issues PASS2 commands, if necessary. The process is repeated until the application goes to the AUTODOWN status or there are no more passes.

You can specify multiple passes. On each pass, you can request that several commands be issued. In this context, the terminate command can include CLISTS. The interval between passes is site-dependent. See Chapter 5 in *System Automation for z/OS: Defining Automation Policy* for more details.

You can specify three types of shutdown, NORM, IMMED, and FORCE. These types provide variations to the shutdown processing. Many installations define the NORM shutdown only. In such installations, if IMMED or FORCE is requested, automation does a NORM shutdown.

When an automation configuration file loads, each application is checked for NORM shutdown commands. If the SHUNNORM commands are not specified, a warning message is issued and the load continues. No check is made for IMMED or FORCE definitions.

There is no automatic transition from one shutdown type to another. If the NORM shutdown does not work, the automation agent does not then run IMMED shutdown commands.

Use the INGMDFY command to change the stop phase (SHUTINIT, SHUTDOWN, and SHUTFINAL) commands and replies for the *next* stop of an application.

## Shutdown phase processing

- Commands in each pass can include replies to outstanding WTORs; for example, DSI802A for Tivoli NetView for z/OS
- Replies might be necessary for WTORs issued at shutdown
- Commands might be needed when specific messages occur
- Commands and replies vary according to shutdown type requested
- For IMMED and FORCE shutdown commands, if no commands or replies are specified, then NORMAL shutdown is used

### *Shutdown phase processing*

The commands that are issued for a shutdown pass can include entering a reply to an outstanding WTOR. Many applications have a permanently outstanding WTOR, which is used to enter commands to the application.

A pass can be made of several commands or it can be omitted. If it is omitted, the pass is effectively skipped. Skipping passes can be used to set different intervals between different sets of commands.

You might also need to take some of the following actions:

- Reply to WTORs that are issued after the shutdown process is started
- Issue commands when specific messages are issued during the shutdown process

All of these actions can be specified in the policy, and are done as part of an INGREQ STOP request.

## Shutdown pass examples

Pass	Action
TSO	
1	Issue the <b>MVS P TSO</b> command
2	Issue the <b>MVS C TSO</b> command
Tivoli NetView for z/OS	
1	Reply <b>CLOSE STOP</b> to the outstanding WTOR DSI802A
3	Issue the <b>MVS C CNMPROC</b> command

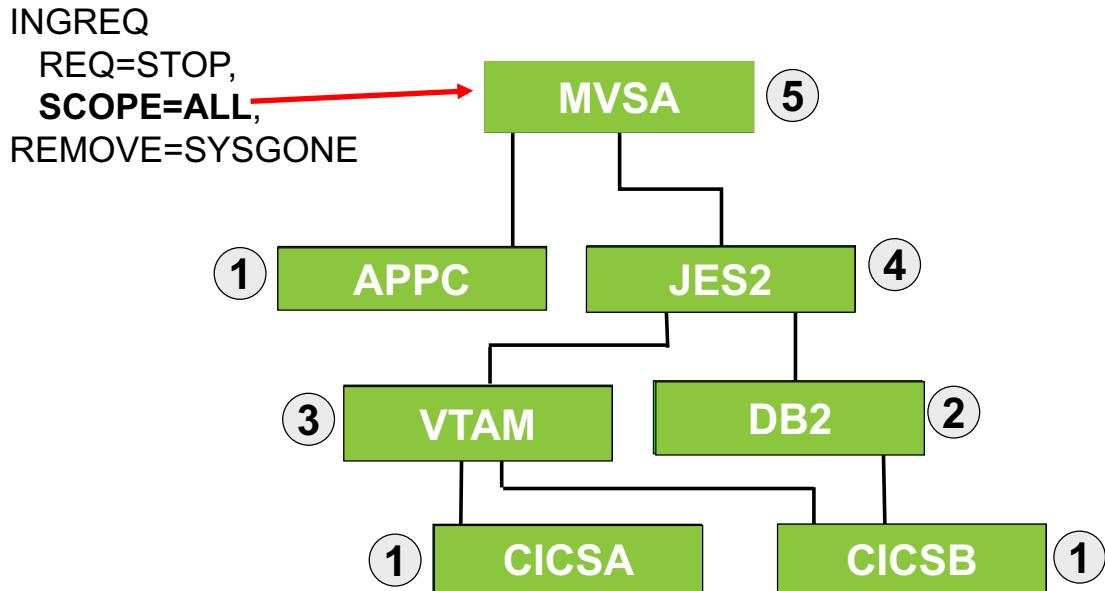
### Shutdown pass examples

This slide shows two examples of shutdown processing with multiple passes. It is not required that pass numbers be sequential. A pass can also be duplicated to issue multiple commands for the pass. If no action is defined, then the pass is skipped. The time interval between each pass is defined as the *shutdown pass interval*.

In the first example, TSO might issue a WTOR as it shuts down, asking whether to terminate the sessions of logged-on users. If the application is still up when the shutdown interval expires, the agent issues the cancel command in the second pass.

In the second example, the agent issues a CLOSE STOP command on the first pass, in response to an outstanding WTOR to shut down NetView. There is no second pass, therefore the agent executes the third pass after twice the shutdown pass interval. If the application is still up when the shutdown interval expires, the agent issues the cancel command that is defined in the third pass.

## Scope of a stop request



All dependencies are HasParent

No other votes

Each application is UP

© Copyright IBM Corporation 2019

3-57

### Scope of a stop request

A single INGREQ request can affect many resources. In this example, assume that there are no other votes for any application, and each application has the default desired status of AVAILABLE.

The INGREQ STOP request sets a desired status of UNAVAILABLE. The request is propagated to each resource. The applications are then shut down in a sequence that is determined by the dependency relationships.

Assuming the dependencies are as shown, the example shows the order in which individual applications might be shut down.

1. Shutdowns are initiated for APPC, CICSA, and CICSB because these applications have no dependencies. You cannot predict the order in which the applications end.  
Suppose APPC ends first. Its parent, MVSA, still has children active, and its shutdown is not started.
2. If CICSB ends second, one of its parents, DB2 now has no more active dependents, and DB2 starts to shut down. VTAM, which is the other parent of CICSB, still has one active dependent, CICSA; therefore, the shutdown of VTAM does not start.
3. CICSA ends next, and the shutdown of VTAM starts.
4. When VTAM and DB2 are both inactive, the shutdown for JES2 is started.

5. When JES2 is inactive, the shutdown for MVSA is started.

Applications within the request that are already inactive are not shut down, nor is their status altered. In other words, a BROKEN application remains in a BROKEN state. For system shutdown, you can specify INGREQ ALL REQ=STOP. The example on this slide shows the command-line form of the command.

# INGLKUP: 1 Manage non-automated jobs

## 2 Collect diagnostics

1. Display jobs that System Automation for z/OS does not control
  - Active jobs, Started tasks, APPC transactions (QUAL)
  - Exclude list can be defined using User Entry Type (UET) pairs
  - Stop, cancel, or force supported (COMMAND)
2. REQ=COLLECT obtains diagnostic information:
  - INGINFO, INGRELS, INGSCHED, DISPINFO, DISPTRG, INGVOTE, INGAMS, and INGHIST
  - DISPMTR (for MTR resources only)
  - DISPAPG (for APG resources only)
  - INGLIST, INGSESS, DISPMTR, DISPSTAT, INGVOTE if resource is unspecified

*INGLKUP: Display or stop non-automated jobs*

The INGLKUP command has three functions\_

The first is to **display and manage jobs**, tasks, and transactions that are not under the control of Automation. During a system shutdown, you can use the INGLKUP command in this way to stop, cancel, or force resources that are not defined to Automation. An exclude list can be defined using User Entry Type (UET) pairs. Use Entry INGLKUP and Type EXCLUDE and Keyword/Data JOB.

The second is to **generate and collect diagnostic information** about resources for debugging purposes. When used this way, the INGLKUP command triggers the running of several commands. The list of commands include:

- INGINFO, INGRELS, INGSCHED, DISPINFO, INGVOTE, INGAMS, DISPTRG, and INGHIST
- DISPMTR (for MTR resources only)
- DISPAPG (for APG resources only)
- INGLIST, INGSESS, DISPMTR, DISPSTAT, INGVOTE if the resource is unspecified

## INGLKUP Examples

- Started as line command
- TARGET, OUTDSN, and OUTMODE parameters
- REQ specifies the type of request to be performed:
  - JOB, COLLECT
- Display all jobs, started tasks, APPC programs:
  - INGLKUP REQ=JOB
- Collect diagnostics and write to data set or log
  - INGLKUP resource REQ=COLLECT OUTDSN=data\_set\_name

*INGLKUP: Display or stop non-automated jobs*

The INGLKUP command can be started as line command.

As most commands it supports TARGET, OUTDSN, and OUTMODE parameters.

The REQ parameter specifies the type of request to be performed: JOB, COLLECT.

When no OUTMODE is specified for parameter JOB a panel is displayed in operator sessions.

The SORT, FIND, and RFIND subcommands are supported.

## INGLKUP: Display jobs example

```

INGKYLUO          SA z/OS - Command Dialogs      Line 1   of 4
Domain Id . : AOFDA  ----- INGLKUP ----- Date . . : 04/01/15
Operator Id : INGC100  System = MVSA           Time . . : 22:34:34

Cmd: C cancel job    F force job     P stop job

Cmd Job      Step      Procedure Type ASID System
-----      ----      -----      ---  -----
— FTPSERVE  STEP1      SYSPROG  A0      MVSA
— RV00       RV00       STEP1      S       MVSA
— RV08       RV08       STEP1      S      0044  MVSA
— RV08       RV08       STEP1      S      0046  MVSA

Command ==> F1=Help   F2=End   F3=Return   F9=Refresh   F6=Roll   F12=Retrieve

```

© Copyright IBM Corporation 2019

3-60

### INGLKUP: Display jobs example

This example shows the INGLKUP REQ=JOB format.

The command has a QUAL parameter that can be used to display ALL, JOB, STC, or APPC resources.

The address space ID (ASID) is shown if there are duplicate job names. The **Cmd** field lets you specify the command codes to cancel, force or stop the job.

From left to right it shows:

- The job name
- The step name
- The procedure step name
- The job type
- The address space ID
- The name of the system that the job is running on

## Student exercise



© Copyright IBM Corporation 2019

3-61

### *Student exercises*

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you have completed this unit, you can perform the following tasks:

- Use the online help function
- Display resource status with SDF, INGLIST, and DISPSTAT
- Issue manager and agent commands to display resource information, relationships, and automation flags
- Use INGREQ to start and stop resources

### *Summary*

Now that you completed this unit, you can perform the following tasks:

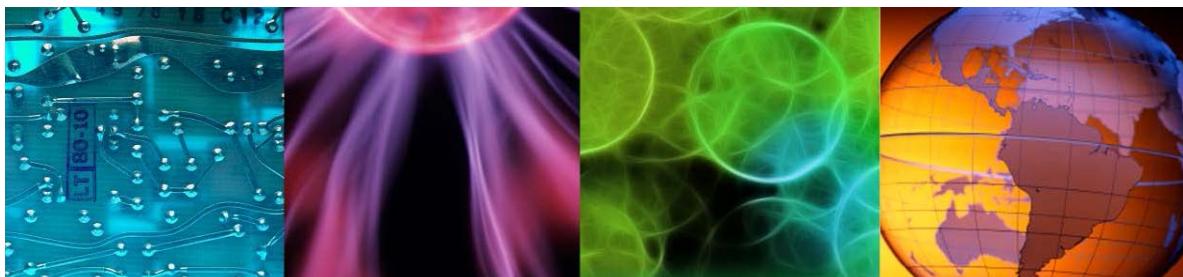
- Use the online help function
- Display resource status with SDF, INGLIST, and DISPSTAT
- Issue manager and agent commands to display resource information, relationships, and automation flags
- Use INGREQ to start and stop resources



## 4 More commands and operations



### Unit 4: More commands and operations



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit introduces more operator commands that are used to manage the automation environment. The lessons include topics on how to use commands such as INGAMS, INGRUN, INGRPT, DISPSYS, and INGPLEX.

# Objectives

---

When you complete this unit, you can perform the following tasks:

- Use System Automation for z/OS commands to:
  - Modify automation flags and resource status
  - Suspend and resume resources
  - Set resource start types and stop types
  - Refresh configurations
  - Modify the runmode for a system
  - Display and define Notify Operators
  - Display Service Periods, Triggers, and Events
- Use the INGPLEX command for sysplex functions

## *Objectives*

When you complete this unit, you can perform the following tasks:

- Use System Automation for z/OS commands to
  - Set resource start types and stop types
  - Modify automation flags and resource status
  - Refresh configurations
  - Modify the runmode for a system
  - Display and define Notify Operators
  - Display Service Periods, Triggers, and Events
- Use the INGPLEX command for sysplex functions

# Lesson 1 Commands to manage automation flags



## Lesson 1. Commands to manage automation flags



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson covers commands to manage automation agent and automation manager automation flags.

# Commands to manage automation flags

Command	Description
INGSET	View and change the values of the automation manager automation flags
DISPFLGS	View the settings of the automation agent automation flags
INGAUTO	Change the settings of the automation agent automation flags

## Commands to manage automation flags

Three commands are available to manage automation manager or agent flags:

- INGSET: View and change the values of the automation manager automation flags
- DISPFLGS: View the settings of the automation agent automation flags
- INGAUTO: Change the settings of the automation agent automation flags

## Changing resource settings at the automation manager

- Change value of automation manager flags
  - Set status
    - Observed
    - Automation
    - Agent
  - Clear or set the **hold** flag
  - Set the next start or stop type, including parameters
- Started by these commands:
  - INGLIST panel, row-command A (Update)
  - The INGSET command in a program
    - Example: REXX CLISTS

Note: INGSET for programmed use only

### *Changing resource settings at the automation manager*

To change the automation manager flags, or set agent status for a resource, enter row-command **A** next to the resource entry on the INGLIST panel. Setting the manager automation flag for a resource to a value of NO means that no orders are sent to the agent. However, the manager still makes all automation decisions.

You can also use the application programming interface (API) command INGSET to modify automation manager flags for a resource. The command is limited to programmed use in REXX CLISTS and cannot be used on the panel command line. For more information about the INGSET command, see *IBM System Automation for z/OS Operator's Commands*.

# INGLIST: Resource settings management panel

INGKYST1 SA z/OS - Command Dialogs  
Domain Id . : AOFDA INGLIST Date . . . : 10/25/18  
Operator Id : INGC102 Sysplex = SYSPLEX1 Time . . . : 21:55:10

Resource . . : RMF/APL/MVSA  
Description. : Resource Measurement Facility

Action to be performed ==>  Enter action number here

1. Set START Type  current setting

2. Set STOP Type

3. Set Automation Flag NO YES

4. Set Hold Flag YES NO

5. Set Group failed flag  (Yes or No)

6. Set Observed Status  AVAILABLE

7. Set Automation Status  IDLE

8. Set Agent Status  UP

Command ==> F1=Help F2=End F3=Return F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

4-6

INGLIST: Resource settings management panel

This example shows the automation flag setting and other manager and agent-related fields. You see this panel when you enter row-command **A** (update), on the INGLIST output panel. To set the next start-type for an application, enter **1** in the ACTION TO BE PERFORMED field and a *start\_type* in the START Type field. This value is reset to null after the next successful start. The same applies to the STOP type.

You can change one flag at a time by entering its corresponding number on the ACTION field. Some of the flags toggle, such as the Automation and Hold flags. Others require a parameter.

## Setting start-type or stop-type for a resource

- Set in the automation manager
  - Applies to only the next start or stop of the resource
  - Cleared if the start or stop is successful
  - Overrides are specified on INGREQ panel for the next successful start or stop request
  - If the resource fails, the start type for the restart reverts to normal unless start-type is set again
  - If the resource fails, the restart is done with the same start-type as used on the INGREQ panel
- Set from these places:
  - Panel that INGLIST row-command **A** displays
  - INGSET API command

### Setting start-type or stop-type for a resource

Use INGLIST row-command **A** or API command INGSET to change the start-type or stop-type for a resource. The life of the change expires after the next successful start or stop of the resource.

The start-type overrides any value that is specified in the TYPE field on the INGREQ panel. The change takes effect while the MakeAvailable or MakeUnavailable vote is active.

## DISPFLGS: Displaying automation agent flags

- Explicit agent automation flags
  - Agent flag settings from the automation control file
  - Actual settings versus effective settings
- Command can be issued in these ways:
  - The command line
  - Row-command **A** on the DISPSTAT panel

*DISPFLGS: Displaying automation agent flags*

The DISPFLGS command displays these items:

- All agent automation flag settings, which are specified in the automation configuration file
- Any that are set manually after automation configuration file load

SYSTEM, SUBSYSTEM, and MVSESA default values are always shown, even if they were not specified. The effective automation flags are also displayed. The INGAUTO command is used to manually set the automation flags.

## Determining agent flag settings for a resource

- Determine the effective automation flag:
  - Major resource (for example, subsystem)
  - Minor resource (for example, message ID or subsystem component)
  - SUBSYSTEM (default)
  - System DEFAULTS
  - If NO: Take no action
- Determine effective action flag:
  - If NO: Take no action
  - Else: Take automation action

### *Determining agent flag settings for a resource*

This slide shows the order of precedence that Automation uses when trying to determine the effective setting of an agent automation flag. For example, TSO issues an IKT010D message as it shuts down. In the automation policy, the administrator coded a response to the IKT010D message. In the initial stage, the effective AUTOMATE flag is determined by checking a sequence of flag settings. When one is found that is not null, Automation uses that value and checks no further. The flag settings are checked in the following order:

1. SUBCOMPONENT checking: Is there a subcomponent for the application and message ID combination; for example, TSO.IKT010D.AUTOMATE?  
*Note: Subcomponent checking is not performed for operator commands.*
2. APPLICATION checking: Is there a setting for TSO.AUTOMATE?
3. SUBSYSTEM default: Is there a setting for SUBSYSTEM.AUTOMATE?
4. SYSTEM default: Is there a setting for DEFAULTS.AUTOMATE?

If the policy search produces a value of NO, then no automated action is taken. However, if it produces a value of YES or null, the effective action flag is then determined. In this case, it is the TERMINATE flag. By setting a subcomponent flag for an application and message ID combination, you can disable automation that is driven by just that one message.

## DISPFLGS examples

AOFKAAAU		SA z/OS - Command Dialogs						Line 1 of 8						
Domain Id . : AOFDA		DISPFLGS						Date . . : 10/25/18-						
Operator Id : INGC102								Time . . : 22:00:51						
<b>Cmd:</b> A Add flags C Change flags		<b>R Reset flags</b>			<b>S Scheduled Overrides</b>									
		Actual			Effective									
<b>Cmd</b>	<b>System</b>	<b>Resource</b>	A	I	S	R	T	RS	A	I	S	R	T	RS
---	---	---	-	-	-	-	-	-	-	-	-	-	-	-
---	MVSA	DEFAULTS	Y	-	-	-	-	-	Y	Y	Y	Y	Y	Y
---	MVSA	MVSESA	-	-	-	-	-	-	Y	Y	Y	Y	Y	Y
---	MVSA	ENQ	-	-	-	-	-	-	Y	Y	Y	Y	Y	Y
---	MVSA	CMDFLLOOD	-	-	-	N	-	-	Y	Y	Y	N	Y	Y
---	MVSA	HUNGCMD	-	-	-	N	-	-	Y	Y	Y	N	Y	Y
---	MVSA	LONGENQ	-	-	-	N	-	-	Y	Y	Y	N	Y	Y
---	MVSA	RV06	S	-	-	-	-	-	S	S	S	S	S	S
---	MVSA	SUBSYSTEM	-	-	-	-	-	-	Y	Y	Y	Y	Y	Y

Command output for DISPFLGS

AOFKAAAU		SA z/OS - Command Dialogs						Line 1 of 1						
Domain Id . : AOFDA		DISPFLGS						Date . . : 10/25/18						
Operator Id : INGC102								Time . . : 22:02:04						
<b>Cmd:</b> A Add flags C Change flags		<b>R Reset flags</b>			<b>S Scheduled Overrides</b>									
		Actual			Effective									
<b>Cmd</b>	<b>System</b>	<b>Resource</b>	A	I	S	R	T	RS	A	I	S	R	T	RS
---	---	---	-	-	-	-	-	-	-	-	-	-	-	-
---	MVSA	RMF	-	-	-	-	-	-	Y	Y	Y	Y	Y	Y

Command output for DISPFLGS RMF

### DISPFLGS examples

The examples on this slide show the output of the command when entered without and with a resource name. The output in the lower half of the slide is also seen when the DISPFLGS row-command **A** is entered next to a resource name on the DISPSTAT panel.

The EFFECTIVE columns show the results of merging the actual values with values defined at the subsystem or defaults level. The EFFECTIVE columns in the second example show all agent automation flags on, even though nothing is specified in the ACTUAL columns. The effective values for the resource in the second example are the result of system resource defaults (DEFAULTS), shown on the first line in the first example.

## INGAUTO: Changing agent automation flags for a resource

- Agent flags can be set for these items:
  - Message ID (most commonly used)
  - Subcomponent
  - Subsystem
  - SUBSYSTEM default
  - SYSTEM default
- Row-command **C** on the DISPFLGS panel displays the INGAUTO panel
- These changes are temporary

© Copyright IBM Corporation 2019

4-11

### *INGAUTO: Changing agent automation flags for a resource*

Use the INGAUTO command to change the automation flags. Automation flag changes are temporary.

These changes are lost whenever the agent is restarted:

- NetView restart
- System IPL
- Automation manager COLD start

# INGAUTO panel

SA z/OS - Command Dialogs

Domain Id . : AOFDA ----- INGAUTO ----- Date . . . : 10/25/18  
Operator Id : INGC102 Time . . . : 22:04:07

Resource name => RV02 Specific or group resource or DEFAULTS

Sub-components => Note : Be sure to specify the correct sub-component name

Target => MVSA System name, domain ID, sysplex name, \*ALL

Scope => ONLY Change automation for:  
ONLY - only this resource  
CHILDREN - dependants  
ALL - resource and dependants

Specify new settings in these entry fields

Automation => \_\_\_\_\_ Specify new settings for automation flags  
Initstart => \_\_\_\_\_ Y - flag is always on  
Start => \_\_\_\_\_ N - flag is always off  
Recovery => \_\_\_\_\_ E - exits called if specified  
Terminate => \_\_\_\_\_ L - resulting commands are logged only  
Restart => \_\_\_\_\_ R - return to automation control file settings

Interval => \_\_\_\_\_ Specify duration of change (dd:hh:mm)  
or leave blank for an indefinite change

Command ==> F1=Help F2=End F3=Return F6=Roll F12=Retrieve

INGAUTO panel

This example shows the INGAUTO panel. Modify the automation flags and press Enter. When INGAUTO is run from a CLIST, the current settings are not displayed. If INGAUTO is called from the DISPLFGS panel, then the current settings are displayed. In the INTERVAL field, an expiration duration can be specified for automatic removal of changes that are made on this panel.

# Lesson 2 Suspend and resume resources



## Lesson 2. Suspend and resume resources



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to suspend and resume resources from automation.

## INGSUSPD panel: Suspend resource

Domain Id . . . : AOFDA	-----	SA z/OS - Command Dialogs	Page . . . : 1 of 1
Operator Id : INGC102	-----	INGSUSPD	Date . . . : 10/30/18
			Time . . . : 09:27:29
Resource Target	=> RVBASIC/APG/MVSA	format: name/type/system	
	=> _____	System name, domain ID or sysplex name	<b>Already filled in</b>
Request Scope	=> SUSPEND	Requested action (SUSPEND or RESUME)	
Expire	=> ONLY	Request scope (ONLY/ALL)	
AutoRemove	=> _____	Expiration date(yyyy-mm-dd), time(hh:mm)	
Verify	=> YES	Remove condition (SYSGONE/NO/*)	
Comment	=> _____	List affected resources (YES/NO/WTOR)	
Timeout	=> 0 / MSG	Interval in minutes / Option (MSG/CANCEL)	
Command S on INGLIST panel to start INGSUSPD			

AUFT Resource can be suspended by: THEN PRESS ENTER

Using the INGSUSPD command which produces a **suspend request**

- Entering resource in **suspend file** before a configuration refresh
- Suspend requests
  - One per resource for all sources, one priority, scope can be specified
  - Resource must complete its start or stop before suspend request succeeds

### INGSUSPD panel: Suspend resource

Normally, automation automates resources that are defined in the policy based on their desired status goals. In some situations, for instance, when maintenance activities require manual startup or shutdown of a resource, the involvement of automation is not wanted. In fact, it would be even counterproductive if automation "corrected" that. In such situations, automation allows the operator or administrator to suspend a resource. While it is suspended, automation does not attempt to start or stop this resource. Similarly, it does not react on messages that would normally trigger status commands or other commands that are defined in the policy for a message. Most importantly, it does not alert the operations team by exposing an unusual Automation Agent or observed status on any status display (for instance INGLIST or SDF), which operators are normally sensitive to.

Resource can be suspended by:

- Using the INGSUSPD command which produces a suspend request. There is only one suspend request per resource for all sources, only one priority. The scope of the suspend request can be specified. For a SCOPE=ALL request, suspend votes are propagated along the dependency chain to all dependents
- Entering resource in suspend file before a configuration refresh. The suspend file is processed during a manager COLD/WARM start or a configuration REFRESH.

### Suspend requests

- One per resource for all sources, one priority, scope can be specified
- Resource must complete its start or stop before suspend request succeeds

The INGSUSPD command suspends and resumes automation for one or more resources. With RESUME, a previously made SUSPEND request from the same source can be canceled.

The INGSUSP command creates a suspend request. The INGSUSPD panel INGKYPS0 is displayed when a user enters the start row-command (S) next to a resource on the INGLIST primary panel. The panel is already filled out as required.

The command can also be run from the command line: INGSUSPD *resource\_name*.

On a panel similar to the one shown on this slide, you enter request information in the fields and press ENTER. Automation displays a verification panel if YES is specified in the **Verify** field on the primary INGSUSPD panel.

Similar to INGREQ for start or stop requests, SA z/OS allows you to specify a SCOPE parameter of the INGSUSPD command to suspend just a single resource or all resources in the dependency chain. For a SCOPE=ALL request, suspend votes are propagated along the dependency chain to all dependents.

The request and its attributes are persistent. The **Expire**, **Timeout** and **AutoRemove** fields can be used to remove suspend requests automatically.

You can cancel the request using the INGVOTE, INGSUSPD, or INGSET command.

A suspend request from the same source replaces an existing suspend request.

Use the **Comments** field to document the reason for the request. The Comments field provides a mechanism to keep a record of the changes.

## INGSUSPD verification panel

AOFKVFY1	SA z/OS - Command Dialogs					Line 1	of 4					
Domain Id . . . : AOFDA	----- INGSUSPD -----					Date . . . :	10/30/18					
Operator Id : INGC102						Time . . . :	09:42:27					
<b>Verify affected resources for request SUSPEND</b>												
<b>CMD:</b> S show overrides <b>T</b> show trigger details <b>V</b> show votes <b>Cmd Name</b> <b>Type</b> <b>System</b> <b>W Sus</b> <b>Scope</b> <b>Desired</b> <b>Observed</b>												
---	---	---	---	---	---	---	---					
— RVBASIC	APG	MVSA	Y		AVAILABLE	AVAILABLE						
— RV01	APL	MVSA	Y		AVAILABLE	AVAILABLE						
— RV02	APL	MVSA	Y		AVAILABLE	AVAILABLE						
— RV03	APL	MVSA	Y		AVAILABLE	AVAILABLE						

Active group propagates request to members

Your suspend request is winning

Command ==> █

F1=Help

F2=End

F3=Return

F9=Refresh

F10=GO

F11=CANCEL

F6=Roll

F12=Retrieve

© Copyright IBM Corporation 2019

4-15

### INGSUSPD verification panel

The verification panel gives the requester an opportunity to verify and confirm or cancel the suspend request. All resources that are affected by the suspend request are shown on the verification panel. The display includes the following information:

- Resources that are within the SCOPE specified for the suspend request
- Supporting resources in the active dependency chain
- Each resource that receives a vote for the suspend request. These resources are not necessarily the same resources that are checked before starting or stopping the resource.

All suspend votes have the same priority, but a suspend with a SCOPE=ALL wins over a SCOPE=ONLY, so the suspend vote with greater impact will always win. If you have two suspend votes on a resource with the same scope, it doesn't matter which one is winning, because the resource is suspended anyway. The W column shows a Y if your request is winning.

## INGSUSPD: Suspend request completed

```
AOFKMSG0          SA z/OS - Command Dialogs      Line 1      of 2
Domain Id . : AOFDA  ----- INGSUSPD -----  Date . . : 10/30/18
Operator Id : INGC102  Time . . : 09:46:49

Sel System   Message
----- -----
- MVSA       AOF302I  09:46:49 : REQUEST INGSUSPD SUSPENDSO BY INGC102 IS
                           COMPLETED FOR RVBASIC/APG/MVSA

Command ==>
F1=Help    F2=End    F3=Return
F6=Roll    F12=Retrieve
```

© Copyright IBM Corporation 2019

4-16

*Suspend request completed*

If your request is winning you will see message AOF302I indicating that your suspend request completed.

If suspend request succeeds and automation status is idle:

- Automation manager suspend flag is set to suspended
- Agent automation flag turned off. To differentiate between the turned off automation agent flag or a suspended APL, you can refer to the task global variables SUB\*SUSPEND provided with AOCQRY. All automation flags (Automation, Initstart, Start, Recovery, Terminate, and Restart) are set to 'S' and that setting is also propagated down to minor resources, if applicable.

While being suspended, DISPFLGS and INGAUTO will not allow you to change the status of any of these automation flags. You must first resume the resource before you can change the status of a flag.

Now you can use MVS commands to start or stop suspended resources outside of automation and use display commands (INGLIST, DISPSTAT, etc.) to track the status of suspended resources.

# INGLIST showing suspended

SA z/OS - Command Dialogs							Line 1 of 4
----- INGLIST -----							Date . . : 10/30/18
Operator Id : INGC102 Sysplex = SAPLEX							Time . . : 09:48:46
A Update	B Start	C Stop	D INGRELS	E INGVOTE	F INGINFO	G Members	
H DISPTRG	I TNRSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTTR	P INGPAC	
R Resume	S Suspend	T INGTWS	U User	X INGWHY	/ scroll		
CMD Name	Type	System	Sus	Compound	Desired	Observed	Nature
RVBASIC	APG	MVSA	DIR	SATISFACTORY	AVAILABLE	AVAILABLE	BASIC
RV01	APL	MVSA	IND	SATISFACTORY	AVAILABLE	AVAILABLE	
RV02	APL	MVSA	IND	SATISFACTORY	AVAILABLE	AVAILABLE	
E	RV03	APL	MVSA	IND	SATISFACTORY	AVAILABLE	

Legend:

- Display votes
- DIR(ectly) suspended
- IND(irectly) suspended
- Color indicates suspended
- Still monitored

Command ==>

F1=Help      F2=End      F3=Return      F4=DISPSTAT      F5=Filters      F6=Roll  
 F9=Refresh    F10=Previous    F11=Next    F12=Retrieve

© Copyright IBM Corporation 2019

4-17

## INGLIST showing suspended

Possible values for the automation manager suspend flag are:

- DIR The resource is **directly suspended** (got a suspend request) via INGSUSPD or the suspend file.
- IND The resource is **indirectly suspended** (because a supporting resource was suspended) via INGSUSPD or the suspend file.
- PEN The suspend request has been recognized but the resource is currently in a start or stop process. Therefore, the request is held back until the desired status has been reached.  
Note: As long as the suspend request on a resource is **pending**, the agent doesn't get a suspend order and the agent automation flags do not display an 'S' for the automation agent flag value.
- **UNSUSPENDED/blank** The resource is not suspended. In INGINFO, you see the value of UNSUSPENDED behind the manager suspend flag, but in INGLIST this is only shown as a blank field.

While a resource is suspended, SA z/OS still monitors whether it is active or inactive. However, unlike non-suspended resources, the lifecycle status model for suspended resources merely distinguishes two status values for Observed Status, Agent

Status, and Monitor Resource Status. INGLIST shows any status of suspended resources in turquoise.

Suspended group behaves like a passive group

- No votes are generated
- No members can be selected

## INGVOTE showing propagated suspend request

```

INGKYRQ0          SA z/OS - Command Dialogs      Line 1     of 5
Domain Id . : AOFDA   ----- INGVOTE -----  Date . . . : 10/30/18
Operator Id : INGC102    Sysplex = SYSPLEX1  Time . . . : 09:50:22

Resource  =>  RV03/APL/MVSA
Target    =>  _____ System name, domain ID or sysplex name
Desired Available...: Always

Cmd: C cancel request  K Kill request  S show request details
Cmd Action WIN Request/Vote Data
----- -----
SUSPEND Y Vote      : Suspend_Only
From Req. : Suspend_Only for RVBASIC/APG/MVSA
Created   : 2018-10-30 09:46:49
Originator: INGSUSPD(INGC102)
Priority  : 01220000 Should Be Suspended - Default

Command ===> █
F1=Help   F2=End   F3=Return   F9=Refresh   F6=Roll   F12=Retrieve

```

**Vote cannot be canceled here**

**Vote propagated from group**

Command S on INGLIST panel to start INGSUSPD

© Copyright IBM Corporation 2019

4-18

INGVOTE showing propagated suspend request

Using INGVOTE, suspend requests can be viewed, canceled, and killed. The latter two actions refer to resuming the resource. Here the vote cannot be canceled as the request was entered at the group and propagated to its members.

## Use INGSUSPD to resume

```
INGKYP$0          SA z/OS - Command Dialogs          Page  1 of 1
Domain Id . : AOFDA ----- INGSUSPD ----- Date . . : 10/30/18
Operator Id : INGC102                                Time . . : 09:52:56

Resource      => RVBASIC/APG/MVSA           format: name/type/system
Target        => _____ System name, domain ID or sysplex name

Request       => RESUME             Requested action (SUSPEND or RESUME)
Scope         => ONLY               Request scope (ONLY/ALL)
Expire        => _____ , _____ Expiration date(yyyy-mm-dd), time(hh:mm)
AutoRemove    => _____ Remove condition (SYSGONE/NO/*)
Verify        => YES                List affected resources (YES/NO/WTOR)
Comment       => _____
```

---

```
Timeout       => 0 / MSG   Interval in minutes / Option (MSG/CANCEL)
```

AOF710A VERIFY/REVISE INPUT AND THEN PRESS ENTER

Command ===> \_\_\_\_\_  
F1=Help F2=End F3=Return

F6=Roll

Command R on INGLIST panel to use INGSUSPD to resume resource

© Copyright IBM Corporation 2019

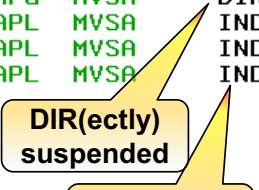
4-19

Use INGSUSPD to resume

With the RESUME request a previously made SUSPEND request can be removed from the same resource.

When the observed status differs from the desired status at the time when you resume a resource, SA z/OS immediately attempts to bring it to its desired status as usual.

## INGSUSPD verification panel for resume

AOFKVFY1	SA z/OS - Command Dialogs				Line 1	of 4				
Domain Id . : AOFDA	----- INGSUSPD -----				Date . . :	10/30/18				
Operator Id : INGC102					Time . . :	09:55:23				
<b>Verify affected resources for request RESUME Suspend_Only</b>										
<b>CMD: S show overrides T show trigger details V show votes</b> <b>Cmd Name      Type System Sus Scope Desired      Observed</b>										
---	---	---	---	---	---	---				
RVBASIC	APG	MVSA	DIR	ONLY	AVAILABLE	AVAILABLE				
RV01	APL	MVSA	IND	ONLY	AVAILABLE	AVAILABLE				
RV02	APL	MVSA	IND	ONLY	AVAILABLE	AVAILABLE				
RV03	APL	MVSA	IND	ONLY	AVAILABLE	AVAILABLE				
 										
Command ==>										
F1=Help	F2=End	F3=Return	F9=Refresh	F10=GO	F11=CANCEL	F6=Roll F12=Retrieve				

© Copyright IBM Corporation 2019

4-20

### INGSUSPD verification panel for resume

The verification panel gives the requester an opportunity to verify and confirm or cancel the cancellation of the suspend request. All resources that are affected by the suspend request are shown on the verification panel. The display includes the following information:

- Resources that are within the SCOPE specified for the cancellation of the suspend request
- Supporting resources in the active dependency chain
- Each resource that is affected by the cancellation of the suspend request. These resources are not necessarily the same resources that are checked before starting or stopping the resource.

When canceling a suspend request, a resource is affected when it holds a vote for the request to be removed. Only resources that are highlighted will be processed by the command. All other resources are already in the state satisfying the appropriate dependencies.

# Lesson 3 Other start/stop tools



## Lesson 3. Other start/stop tools



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to use other start/stop tools for start/stop pacing, for changing the automation agent status, so that resources can be automated again and for temporary modification of start or stop commands.

## Other start/stop tools

Command	Description
INGPAC	Display information about pacing gates
SETSTATE	Change resource status at automation agent
INGMDFY	Temporary modification of start or stop commands

*Other start/stop tools*

**INGPAC**

Display information about pacing gates

**SETSTATE**

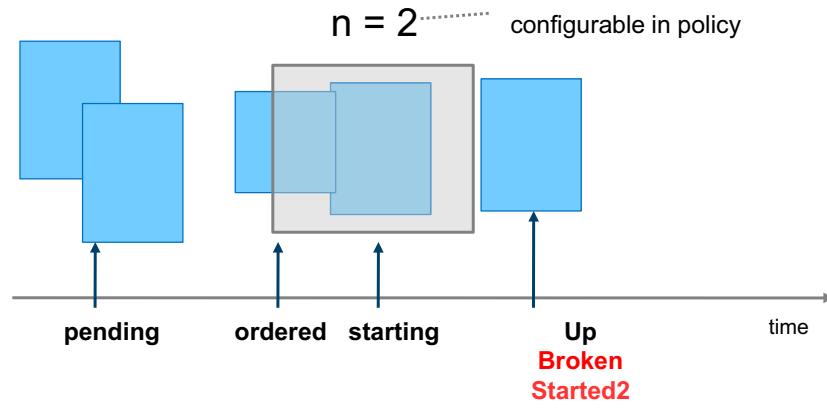
Change resource status at automation agent

**INGMDFY**

Temporary modification of start or stop commands

# Pacing Gates

- Pacing gates allow you to determine how many resources are started or stopped at the same time
- Resources waiting to be started have the following states:: Compound=AWAITING, Observed=SOFTDOWN, agent=RESTART
- Pacing gate is listed in INGLIST
- INGPAC allows to monitor progress through pacing gates, however no changing of gate supported although you can release waiting resources



© Copyright IBM Corporation 2019

4-23

## Pacing gates

SA z/OS pacing gates allow you to determine how many resources are started or stopped at the same time. Gate mechanism ensures that only up to n automation resources can pass the gate at any given time. The automation manager delays the order of the applications waiting in front of the gate. When the gate is open a start order is sent to the agent and the status becomes starting.

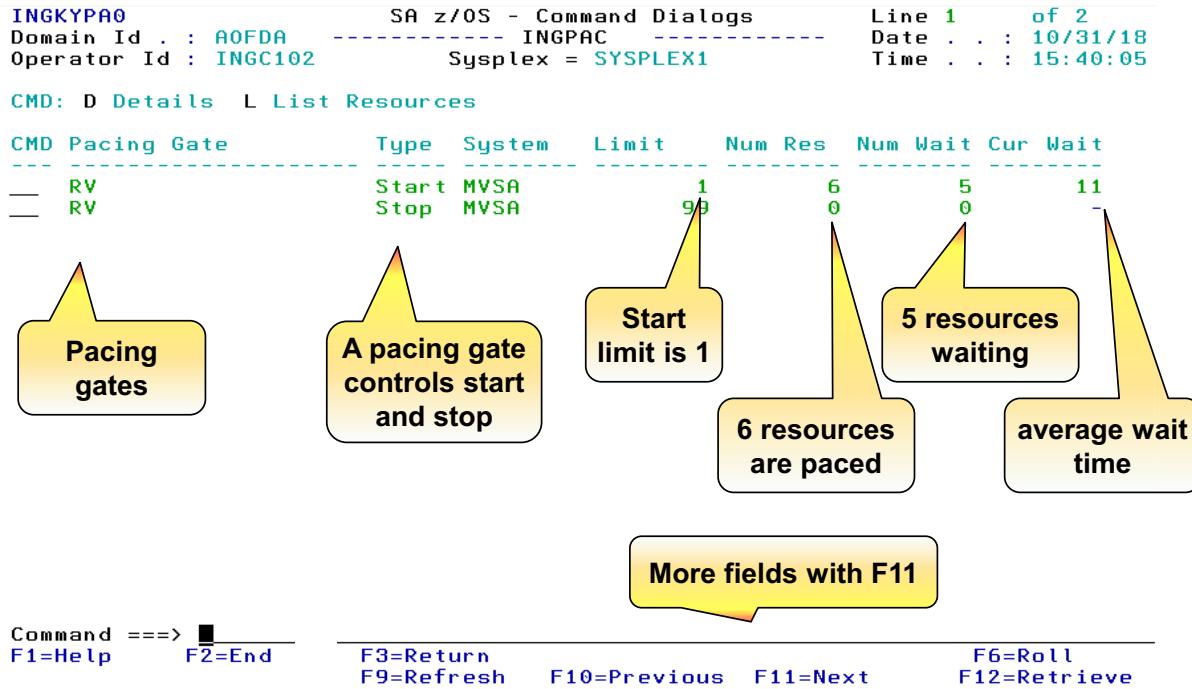
The gate will be opened again when the startup is done successfully, indicated by an UP status or unsuccessfully indicated by a STARTED2 or BROKEN agent status.

Waiting resources have the following states for start gates:

- Compound status AWAITING,
- Observed status SOFTDOWN
- Agent status RESTART

Pacing gate is listed in INGLIST and the INGPAC dialog allows to monitor progress through pacing gates, however no changing of gate supported although you can release waiting resources.

## Pacing gates dialog INGPAC



© Copyright IBM Corporation 2019

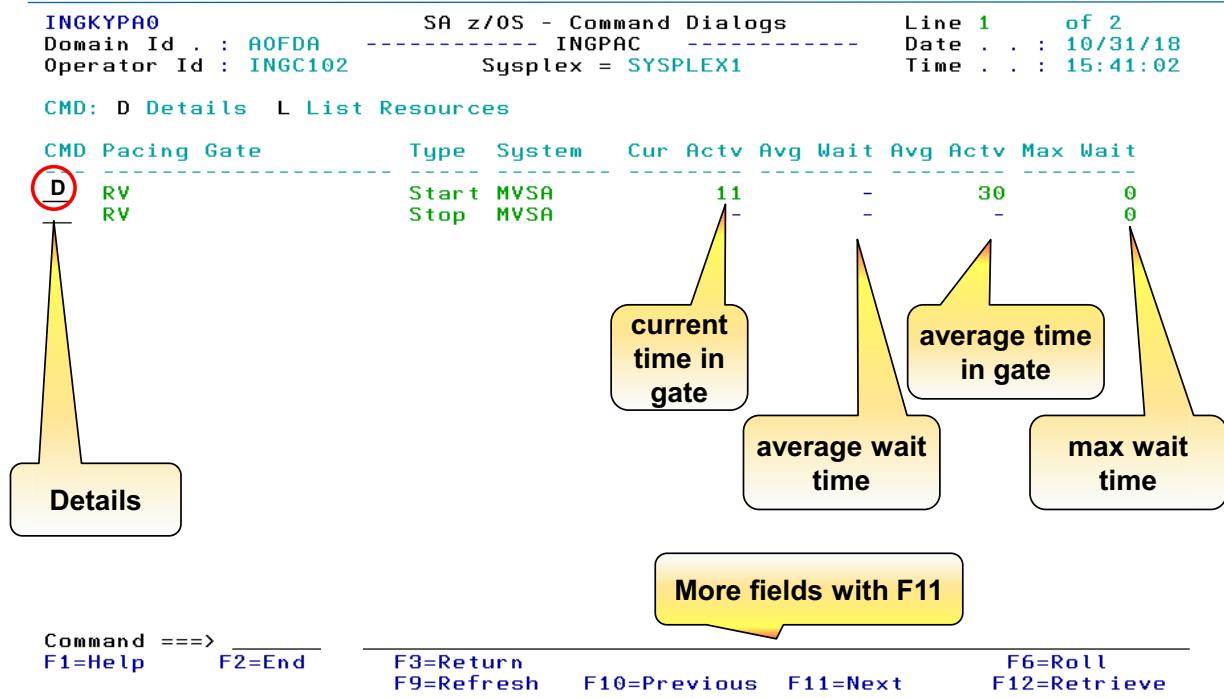
4-24

### Pacing gates dialog INGPAC

This panel displays details about the pacing gates that are defined in the SAplex. The INGPAC command displays information about pacing gates, their characteristics and runtime statistics. It also shows details about pacing gate contention with the resources currently granted access to the gate and those waiting for access to the gate. For each pacing gate the following information is shown:

- Pacing Gate: The name of the pacing gate.
- Type: The pacing gate type. The following types are distinguished:
  - Start: The pacing gate is effective when applications are started.
  - Stop: The pacing gate is effective when applications are stopped
- Limit: The number of applications that are allowed to transition this pacing gate at the same time. NOLIMIT is shown, if the pacing gate is effectively disabled
- Num Res: Total number of applications either waiting or transitioning through the gate
- Num Wait: Number of applications waiting for other applications leaving the gate
- Cur Wait: The average time in seconds applications currently are waiting to transition through the gate. If no application is currently waiting, '-' is shown.

## Pacing gates dialog INGPAC F11



© Copyright IBM Corporation 2019

4-25

### Pacing gates dialog INGPAC F11

- Cur Actv: The average time in seconds applications currently need to transition through the gate.
- Avg Wait: The average time in seconds an application waited to be allowed to transition through the gate.
- Avg Actv: The average time in seconds an application needed to transition through the gate. This is the average start or shutdown time of that application.
- Max Wait: The maximum time in seconds an application waited before it could transition through the gate.

On the next panel

- Max Actv: The maximum time in seconds an application needed to transition through the gate

Use one of the following command codes to invoke another command dialog:

- D Show the pacing gate details
- L Show all resources associated with this pacing gate

## Pacing gates dialog INGPAC details

INGKYPAA1	SA z/OS - Command Dialogs	Line 1 of 5			
Domain Id . : AOFDA	----- INGPAC -----	Date . . : 10/31/18			
Operator Id : INGC102	Sysplex = SYSPLEX1	Time . . : 15:38:22			
Pacing gate: RV	Type: Start	System: MVSA			
Description: Pacing Gate for RV APLs		Limit : 1			
Wait times (s)	Active times (s)	Number resources			
-----	-----	-----			
Current: 32	Current: 1	Waiting: 4			
Average: 30	Average: 30	Active: 1			
Maximum: 30	Maximum: 30	Other: 0			
CMD: I Resource Info R Release					
CMD Resource Name	Status	Wait	Active	Total	Req Date
---	-----	-----	-----	-----	-----
--- RV02/APL/MVSA	Starting	31	1	32	10/31/18
--- RV03/APL/MVSA	Waiting	32	0	32	10/31/18
--- RV04/APL/MVSA	Waiting	32	0	32	10/31/18
--- RV05/APL/MVSA	Waiting	32	0	32	10/31/18
--- RV06/APL/MVSA	Waiting	32	0	32	10/31/18

RV02 is in the gate

4 are waiting

More fields with F11

Command ==> █

F1=Help F2=End F3=Return F9=Refresh F10=Previous F11=Next F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

4-26

### Pacing gates dialog INGPAC details

If you select details on the previous panel you see a list of resources that are in or in front of the gate.

For each resources you see the Wait and Active and total time in seconds and at what date the request was entered.

## Pacing gates on INGLIST

SA z/OS - Command Dialogs							Line 1	of 6
INGLIST							Date . . .	: 10/31/18
Sysplex = SYSPLEX1							Time . . .	: 15:39:05
A Update	B Start	C Stop	D INGRELS	E INGVOTE	F INGINFO	G Members		
H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTTR	P INGPAC		
R Resume	S Suspend	T INGTWS	U User	X INGWHY	/ scroll			
CMD Name	Type	System	Sus	Compound	Desired	Observed	Nature	
RV01	APL	MVSA		SATISFACTORY	AVAILABLE	AVAILABLE		
RV02	APL	MVSA		SATISFACTORY	AVAILABLE	AVAILABLE		
RV03	APL	MVSA		INAUTO	AVAILABLE	STARTING		
RV04	APL	MVSA		AWAITING	AVAILABLE	SOFTDOWN		
RV05	APL	MVSA		AWAITING	AVAILABLE	SOFTDOWN		
RV06	APL	MVSA		AWAITING	AVAILABLE	SOFTDOWN		

RV03 is in the gate

3 are (A)waiting

SOFTDOWN

Command ==> █

F1=Help

F2=End

F3=Return

F9=Refresh

F4=DISPSTAT

F10=Previous

F5=Filters

F11=Next

F6=Roll

F12=Retrieve

© Copyright IBM Corporation 2019

4-27

### Pacing gates on INGLIST

INGLIST has a column showing the pacing gate for each resource. Only one pacing gate can be defined for each resource.

Pacing gates just delay sending the order to the agent, so the Compound status is AWAITING.

## Pacing gates on INGWHY

```

INGKYAN2          SA z/OS - Command Dialogs          Line 1      of 15
Domain Id . : AOFDA   ----- INGWHY -----  Date . . : 10/31/18
Operator Id : INGC102   Sysplex = SYSPLEX1  Time . . : 15:53:15

Analyzed Resource: RV03/APL/MVSA           Job Name: RV03
Status Compound: AWAITING      Desired: AVAILABLE  Observed: SOFTDOWN
-                                     More: F10/F11

SITUATION:
RV03/APL/MVSA is waiting to be started.

REASON 2 OF 2:
RV03/APL/MVSA is controlled by Pacing Gate RV.
RV03/APL/MVSA is 1 out of 5 resource(s) that is waiting to be started.

ACTION 2 OF 2:
Refer to your company's rules in order to take the appropriate action.          INGWHYSA(A0810700)

Wait until RV03/APL/MVSA is in its desired status.
If necessary, consider the following command to control the Pacing Gate
processing of RV03/APL/MVSA:
- INGPAC

```

**SOFTDOWN**

**WAIT...**

Command ==> █

F1=Help F2=En... F3=Return F9=Refresh F10=Previous F11=Next F6=Roll F12=Retriev...

© Copyright IBM Corporation 2019

4-28

### Pacing gates on INGWHY

INGWHY can identify that a resource is waiting for a gate.

There is nothing you can do, but wait...or use INGPAC to release the waiting resources.

## SETSTATE: Modify agent status for a resource

- Command can be run from these places:
  - Command line  
Example: **SETSTATE RMF**
  - Row command **B** on DISPSTAT panel
- Command is useful for these purposes:
  - Set an acceptable status after a manual shutdown
  - Prevent an automatic restart

### *SETSTATE: Modify agent status for a resource*

Use the SETSTATE command to change the agent status for a resource. As an example, an application goes to STARTED2 status, but it is UP. Use the SETSTATE command to apply the correct status.

Also, you must use SETSTATE to change an application from CTLDOWN to AUTODOWN when you are ready for Automation to resume control of the application.

## SETSTATE example

```
AOFKSAST          SA z/OS - Command Dialogs
Domain Id . . : AOFDA ----- SETSTATE ----- Date . . . : 10/25/18
Operator Id : INGC102                         Time . . . : 22:08:14

Specify subsystem and new state:

Subsystem    => RV02           Subsystem to be processed
Target        => MVSA
State         => autodown       Put subsystem in the following state:
                                         (RESTART / AUTODOWN / CTLDOWN / UP / ENDED /
                                         MOVED)

Scope         => ONLY          Change state for:
                                         ONLY   - only this subsystem
                                         CHILDREN - only dependant subsystems
                                         ALL    - resource and dependants

Command ===> _____
F1=Help      F2=End       F3=Return
                                         F6=Roll
                                         F12=Retrieve
```

© Copyright IBM Corporation 2019

4-30

### SETSTATE example

This example shows that the operator set the state of the application to AUTODOWN. This action allows System Automation z/OS to start the resource.

## INGMDFY:

### Temporary modification of start/stop phases

- Displays commands and replies for the start and stop phases
  - Start phases: PRESTART, STARTUP, and POSTSTART...
  - Stop phases: SHUTINIT, SHUTDOWN, and SHUTFINAL
- Can be used for these purposes:
  - Modify existing commands
  - Define new commands or replies
  - Delete existing commands or replies
- Attributes of changes
  - Valid for next start or stop
  - Removed after successful completion of start or stop
  - Changes are not persistent; lost at Tivoli NetView for z/OS recycle or system IPL
  - Changes expire after 24 hours

#### *INGMDFY: Temporary modification of start/stop phases*

The INGMDFY command displays start and stop phases, such as PRESTART, STARTUP, SHUTNORM, SHUTFINAL. Commands and replies can be modified for the *next* start or stop of the resource, and removed after the start or stop completes. Changes to the start and stop phases are lost when NetView or the system is recycled or when 24 hours elapses.

## INGMDFY STOP example

```

AOFKEDSS          SA z/OS - Command Dialogs          Line 1      of 19
Domain Id . : AOFDA  ----- INGMDFY          Date . . . : 10/25/18
Operator Id : INGC102  System = MVSA          Time . . . : 22:10:32

Subsystem: JES2      Category: JES2      Extstop : NEVER  Modified: No
                     Subcategory:           Jobtype : MVS

S/C Define Command S/R Define Reply

Cmd Phase      AutoFn/* Command Text
Pass      Retry Ct Reply Text
----- -----
SHUTINIT
-- JESOPER  DRAINJES &SUBSAPPL
-- JESOPER  MVS F BPXINIT,SHUTDOWN=FORKS
-- JESOPER  INGRCLUP AXR0*,*
-- JESOPER  PIPE NETV INGTJLM STATUS ! LOC /INGTJLM: ACTIVE/ ! EDIT

SHUTNORM
-- PASS4    MVS &SUBSCMDPFXP&SUBSJOB

SHUTIMMED
-- PASS1    MVS &SUBSCMDPFXP&SUBSJOB
-- PASS2    MVS &SUBSCMDPFXP&SUBSJOB, ABEND

SHUTFORCE
-- PASS1    MVS &SUBSCMDPFXP&SUBSJOB, ABEND

Command ==> █
F1=Help   F2=End   F3=Return  F4=Reset  F5=Save Chgs  F6=Roll
          F8=Forward F9=Refresh          F11=Next          F12=Retrieve

Command: INGMDFY JES2 STOP

```

© Copyright IBM Corporation 2019

4-32

### INGMDFY STOP example

This slide shows an example of output from INGMDFY command. This example shows the commands that are defined. Two commands are defined for the SHUTINIT phase. For the SHUTDOWN phase you see commands that are defined for three types of shutdowns: Normal (SHUTNORM), Immediate (SHUTIMMED) and SHUTFORCE. The Modified field on this panel indicates whether the policy has been modified.

To modify a command, follow these steps:

1. Place the cursor in the CMD column next to the PASS that you want to modify.
2. Enter **C** to modify the command or **R** to modify a reply. A new panel, AOFKEDS3, is displayed. You can modify the command definition on panel AOFKEDS3. Press Enter, and then PF3 to save the changes. You return to the INGMDFY panel.
3. Press PF5 to save your changes or PF4 to reset the command to the previous value. The Modified field changes to YES after you press PF5.

Automation uses the modified command or reply once only, at the next stop or start request, depending on your option you chosen at INGMDFY invocation.request for the resource.

# Lesson 4 Managing the automation environment



## Lesson 4. Managing the automation environment



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to display and manage the automation environment using the INGAMS command. You learn how to set the run mode for a system, how to display statistical information about the agent and manager and how to display and manage notify operators.

# Managing the automation environment

Command	Description
INGAMS	Show the status of managers and agents
INGRUN	Set the run mode for a system
INGRPT	Display statistical information about the agent and manager
DISPSYS	Display system
INGNTFY	Display and manage System Automation for z/OS notify operators

# INGAMS command functions

- Display all registered automation managers and automation agents
  - System name, status, role
  - SAplex name and XCF group name
  - Release level
  - Communication method in use – XCF
  - E2E automation indicator
  - Current runmode
- Change the operation mode (role) of automation managers
- Refresh configuration SAplex-wide
- Display automation manager details
  - Diagnostic information
  - Logic deck information
  - Work item statistics
- Enable and disable the takeover file
- Take diagnostic snapshots
- Suspend and resume all automation

## INGAMS command functions

The INGAMS command can be used to see a summary of all registered automation managers and automation agents in a SAplex. The command is primarily used to refresh the automation configuration. The command can also be used to control the recording of diagnostic information.

The next slide shows an example of the command output, when the command is entered without parameters. You can navigate directly to the other panels in the dialog by specifying more operands. See the *System Automation for z/OS: Operator's Commands* or the online help for complete command syntax.

The End-to-End (E2E) indicator is turned on when E2E functions are available for the agent or primary automation manager.

The agent shows yes in E2E column if the E2E automation adapter is running on the same system. The PAM shows yes in the E2E column if the E2E agent application is running in its SAplex. The agent with the E2E adapter is the agent that communicates with the E2E manager.

## INGAMS primary panel

SA z/OS - Command Dialogs				Line 1 of 4			
Domain Id . . . : AOFDB		Operator Id : INGC102		INGAMS Sysplex =		Date . . . :	10/29/18
Cmd: A Manage		B Show Details		C Refresh Configuration		D Diagnostic	
CMD	System	Member	Role	Status	Sysplex	XCF Group	Release Comm E2E
MVSA	MVSA\$\$\$\$\$2	SAM	READY		PLEX12	INGXSGSA	V4R1M0 XCF
MVSA	MVSA\$\$\$\$\$1	PAM	READY		PLEX12	INGXSGSA	V4R1M0 XCF
MVSA	MVSA	AGENT	READY		PLEX12	INGXSGSA	V4R1M0 XCF
MVSB	MVSB	AGENT	READY		PLEX12	INGXSGSA	V4R1M0 XCF

Choose option A (Manage) from the INGAMS main panel

This example shows a primary automation manager, a secondary automation manager, and two agents. All are System Automation for z/OS 4.1

Command ==> █

F1=Help	F2=End	F3=Return	F6=Roll
F9=Refresh	F10=Previous	F11=Next	F12=Retrieve

© Copyright IBM Corporation 2019

4-36

### INGAMS primary panel

The command lists all registered automation managers and agents in the SAplex controlled by the primary automation manager. Most of the fields on the panel are self-explanatory.

Each automation manager receives an internally generated name. The name which is nine-characters long consists of the eight-character system name and a single numeric digit (1-9). If the system name has less than eight characters, it is padded with dollar signs (\$). More than one automation manager can be running in a system. The maximum number of automation managers that are supported on a single system is nine.

The *Role* column shows the type of the member:

- AGENT Automation agent
- PAM The primary automation manager
- SAM A secondary automation manager

The *Status* column shows the status of the member. It can be one of the following:

- NOT READY: The automation agent or manager is initializing.
- PENDING: The automation manager is in the process of initializing as a primary automation manager (PAM).
- READY: The member is completely initialized.
- REFRESH: The automation manager is performing a configuration refresh.
- SELECTED: The automation manager is selected to become the next primary automation manager (PAM).
- STOPPING: The automation manager is terminating.
- SUSPENDED: The automation agent has been placed in suspended mode.

The *Comm* column shows the type of communication that the address space is using. This example is using XCF.

The *E2E* column shows that the E2E adapter is running on this system, if the system is an AGENT and this field contains YES. If the system is a PAM and this field contains YES, it shows that the PAM is also the end-to-end manager.

PF11(Next) scrolls the panel horizontally. You see the RUNMODE column in which active runmodes are displayed next to agents. If no defined runmode is active, you see \*ALL in the column. If there are no defined runmodes, the column is blank. (Runmodes are explained later in this lesson.)

Entering row-command **B** next to the automation manager displays detailed information about the manager.

Entering row-command **A** next to the automation manager shows the following panel:

# Suspending automation manager processing

```

INGKYAM1          SA z/OS - Command Dialogs
Domain Id . : AOFDA ----- INGAMS ----- Date . . : 10/25/18
Operator Id : INGC102      Sysplex = SYSPLEX1      Time . . : 22:17:50

Specify or revise the following data:

Action  =>  1. Make primary Automation manager - Start => _____
              2. Enable Takeover File
              3. Disable Takeover File
              4. Suspend System
              5. Resume System

Automation manager => MVSA$$$$$1
Target        => SYSPLEX1    System name, domain ID or sysplex name

Command ==> _____
F1=Help     F2=End      F3=Return
                                         F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

4-37

## Suspending automation manager processing

After entering row-command **A** (Manage) on the primary INGAMS panel, an operator can suspend automation manager processing. Action **4** suspends automation. Requests for resources on a suspended system are queued and are processed when automation is resumed. The compound status of requested resources is DENIED. The automation manager sends no orders to the agent on a system for which automation is suspended. However, requests are processed. The suspension is persistent across system IPLs. Multiple systems can be suspended or resumed with a single action. \***ALL** in the input field suspends or resumes all systems. Orders for pending requests are sent when automation is resumed.

The following is a sample scenario:

- Situation
  - Accidental request for a system shutdown
- Corrective action
  - Suspend automation processing
  - Delete the shutdown request
  - Verify
  - Resume automation processing

## Setting or changing runmodes

- At automation agent initialization

```
AOF767I AUTOMATION OPTIONS: 409
. STOP      - CANCEL AUTOMATION
. PAUSE     - SUSPEND AUTOMATION
. NOSTART   - DO NOT AUTOMATE SUBSYSTEM STARTUP
. RUNMODE=x - SET RUNMODE (CURRENT *ALL)
. ENTER     - CONTINUE
*0002 AOF603D ENTER AUTOMATION OPTIONS OR 'R' (RE-DISPLAY) - DOMAIN AOFDA
```

- By using the INGRUN command
  - Set the runmode for a system
  - Forcing runmode qualifications
    - Add qualifications to an active runmode
  - Delete selected resource qualifications from an active runmode
    - Remove qualifications for selected resources

### Setting or changing runmodes

Using runmodes is a flexible way to control the availability of resources without explicit START or STOP requests against them, or manipulating their automation flags. Runmodes are defined in the policy database during customization.

A runmode can be activated or changed for a system in one of two ways:

- By specifying the runmode when the AOF603D WTOR message is displayed at agent initialization
- By using the INGRUN command at any time while system automation is enabled and running

Along with changing the runmode, the INGRUN command can be used to add or delete (force) qualifications for resources. Forced runmode qualification is persistently stored in the automation manager. However, this should be an exception and you should remove unnecessary forced runmode qualifications using INGRUN.

When you use INGAMS to load an ACF that does not contain the currently active runmode, the refresh is refused with error message:

```
AOF618I NO VALID ACF MEMBER FOUND FOR MVSA - CURRENT RUNMODE YES_RVAPPL
NO LONGER EXISTS
```

## Resources qualified for a runmode

IN GKYST0		SA z/OS - Command Dialogs				
Domain Id . :	AOFDA	INGLIST ----- Sysplex = SYSPLEX1				
Operator Id :	INGC102	A Update	B Start	C Stop	D INGRELS	E INGVOTE
		H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS
		R Resume	S Suspend	T INGTWS	U User	X INGWHY
CMD Name	Type System	Subcategory	Jobname	Qual		
— RVBASIC	APG	MVSA		--T		
— RV01	APL	MVSA	RV01	-R-		
— RV02	APL	MVSA	RV02	--T		
— RV03	APL	MVSA	RV03	--T		

Resource qualifies because of its runtokens

Resource qualifies because it was explicitly added

Qual indicates how a resource qualifies for the current runmode

- **G:** Resource is in one or more groups that qualify
- **R:** Resource was explicitly added via the INGRUN command
- **T:** Resource qualifies because of its runtokens

### Resources qualified for a runmode

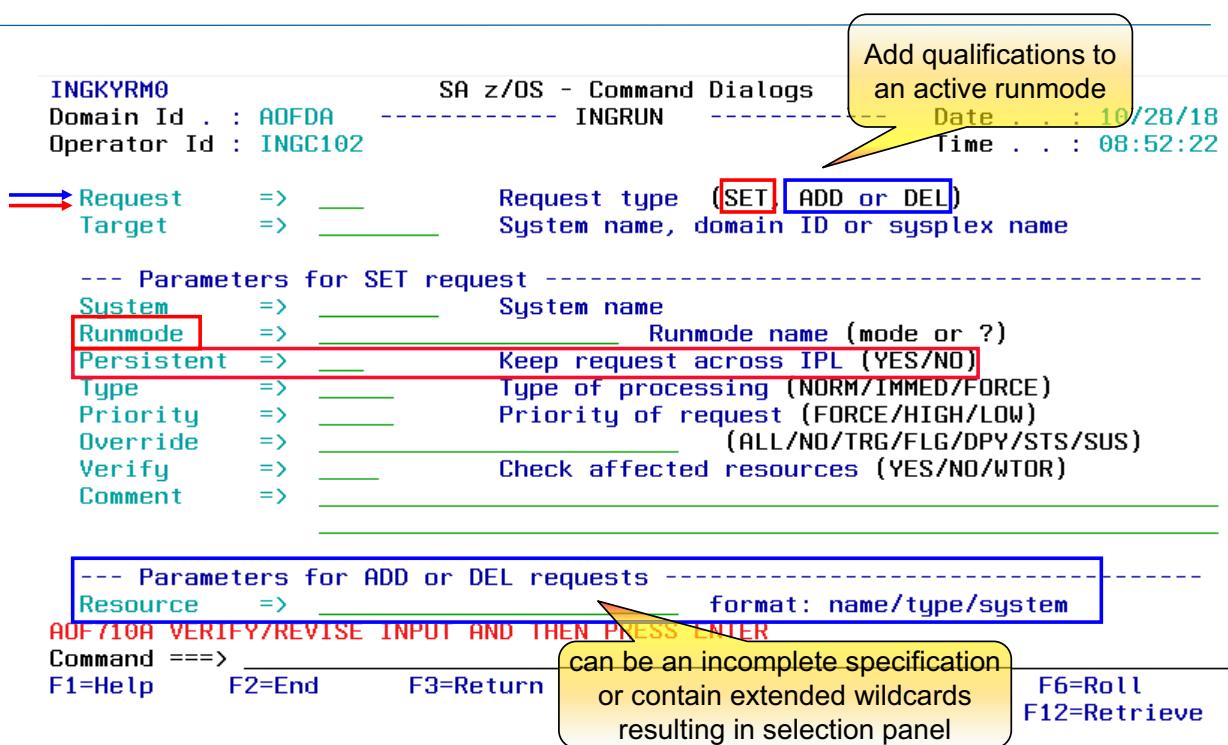
The resource qualification is displayed in INGLIST, INGVOTE, and INGINFO.

On a system with an active, defined runmode, resources qualify if one of the following is true:

- One of the runtokens that is linked to the resource matches a runtoken that is associated with the active runmode.
- The resource is a member of a group which is qualified.
- The resource qualification was added (forced) with the INGRUN command.

If no defined runmode is active on a system, by default, the active runmode becomes \*ALL which means that no qualification restrictions are in effect. No stop requests are generated. All resources which have a DESIRED status of MakeAvailable, and good shutdown status are started. Specifying \*ALL in the RUNMODE field on the INGRUN panel replaces an active, defined runmode.

## INGRUN command panel



© Copyright IBM Corporation 2019

4-40

### INGRUN command panel

This slide highlights the following key fields on the INGRUN panel:

- Request (choose one):
  - SET: Activate a runmode
  - ADD: Qualify one or more individual resources
  - DEL: Remove one or more force-qualified resources
- Runmode: Specify the name of the runmode to activate
- Persistent: Indicate your preference for persistence of the runmode
- Resource: Specify the names of resources for which you are adding (forcing) or deleting qualification

## INGRUN votes example

```

INGKXRQ0          SA z/OS - Command Dialogs      Line 1      of 15
Domain Id . : AOFDA ----- INGVOTE ----- Date . . : 10/28/18
Operator Id : INGC102     Sysplex = SYSPLEX1   Time . . : 13:48:44

Resource    => RV01/APL/MVSA
Target      => _____ System name, domain ID or sysplex name

Desired Available...: Always Runmode Qualification..: -R-
Cmd: C cancel request K Kill request S show request details
Cmd Action WIN Request/Vote Data

STOP * Vote : MakeUnavailable
From Req. : MakeUnavailable for MVSA/SYG/MVSA
Created : 2018-10-28 11:25:50
Usage : 2
Originator : INGRUN(INGC102)
Runmode : BASE
Runtokens : NORVAPPLS
Priority : 01220000 Should Be Down - Default
Priority : 01220000 Should Be Down - Default
STOP Vote : MakeUnavailable
From Req. : MakeUnavailable for RVBASIC/APG/MVSA
Created : 2018-10-28 11:29:10
Originator : GROUP RVBASIC/APG/MVSA: INGRUN(INGC102)
Runmode : BASE
Runtokens : NORVAPPLS
Priority : 01220000 Should Be Down - Default

Command ==> █
F1=Help F2=End      F3=Return      F6=Roll
F7=Backward F9=Refresh      F12=Retrieve

```

4-41

### INGRUN votes example

The highlighted areas on this slide summarize a few concepts about runmodes. Stop votes are issued for all resources in SYG. There are two stop votes against the resource.

5. The first vote is propagated from SYG. The \* in the WIN column means this vote is loosing but is being propagated. As RV01 has no MakeUnavailable forward relationship, this vote is actually not propagated.
6. The resource received a second vote because it belongs to group RVBASIC. The blank in the WIN column means this vote is loosing. The active runmode is BASE which has a runtoken of NORVAPPLS. The resource in the example does not have a matching runtoken for the active runmode, but it has forced runmode qualification indicated by R.

Despite two stop votes, the status of the resource is AVAILABLE. The reason is that an ADD action on another INGRUN request forced the qualification. Evidences of the force are that all INGRUN votes are loosing and the R in the RUNMODE QUALIFICATION field which is at the right of the DESIRED AVAILABLE field near the top of the panel.

## INGRPT: Statistics and base information

- Statistics counters for the current SA z/OS session because of SysOps initialization or last INGRPT RESET command
  - Not retained across SysOps sessions
- Summary or detail report
  - STATS=parameter
- Started as a line command
  - Examples:
    - **INGRPT STATS=SUMMARY**
    - **INGRPT STATS=DETAIL**
    - **INGRPT RESET**

### INGRPT: Statistics and base information

The INGRPT command produces two different types of reports: summary reports or detail reports by resource. Using the reports, you can determine the value of running System Automation for z/OS in your environment that is based on these items:

- The number of resources managed
- The number of messages automated and resulting commands
- The number of start and stop commands issued

During initialization, all statistics can be cleared by running the INGRPT RESET command through a user exit.

# INGRPT: Summary report

```

INGKYRPO          SA z/OS - Command Dialogs      Line 1   of 49
Domain Id . . . : AOFDA    ----- INGRPT ----- Date . . . : 10/28/18
Operator Id : INGC100       System = MVSA        Time . . . : 14:24:16

***** Summary Statistics *****
System : MVSA                         28 Oct 2018 14:24:15
-----
Begin of statistics : 10/21/18 16:40:34           Domain   : AOFDA
End   of statistics : 10/28/18 14:24:15           Sysplex : PLEX12
Elapsed time (hours) : 165:44 (9944 min)

Automation Agent Statistics
  CPU time used (seconds) : 108.23

  Total number of resources defined : 31
  Total number of resources managed : 22
  Total number of monitors defined : 3

  Total number of messages automated : 10
  Average number of messages per hour : 0.1
  Total number of resulting commands : 5
  Average number of commands per hour : 0.0

  Total number of StartUp commands issued : 28
  Total number of ShutDown commands issued : 8

  Total number of workitems sent : 1109

Command ===> _____
F1=Help   F2=End   F3=Return   F4=Forward   F5=Refresh   F6=Roll   F12=Retrieve
Press PF8 to display more statistics
-----
```

© Copyright IBM Corporation 2019

4-43

## INGRPT: Summary report

Issuing the INGRPT command with no parameters produces the Summary report shown here. You can scroll forward or backward to see the information.

INGRPT displays these automation statistics for each application:

- The number of messages that were automated
- The total number of automation commands that ran
- The number of START commands issued
- The number of STOP commands issued
- The number of critical threshold hits

The command INGRPT STATS=DETAIL,RESOURCE=resource\_name shows statistical details for the specified resource.

## DISPSYS: Display system information

```
AOKKADAE          SA z/OS - Command Dialogs          Line 1   of 58
Domain Id . : AOFDA  ----- DISPSYS -----  Date . . : 10/28/18
Operator Id : INGC100  System = MVSA             Time . . : 14:29:34

System           : MVSA      in Sysplex : PLEX12
Domain          : AOFDA
Sysplex Group   : SYSPLEX1
XCF Group name : INGXSGSA
XCF PlexID     : INGPX$00

Software
  Operating System : z/OS 02.02.00
  NetView          : IBM NetView for z/OS V6R2M1
  Tower(s)         : SA
  Automation       : System Automation for z/OS V4R1M0
  Tower(s)         : SYSOPS PROCOPS

Configuration
  Data set        : SAZOS.ACFS.OPSU4EX1(ACFZ999)
  Built by        : INGC102 08/30/18 18:45:26
  Activated       : 10/28/18 10:24:37
  CFG Token       : 20180830184526FF01F9672827

Runmode(s)
  Current : BASE
  Defined
    RVAPPLS : RVAPPLS
    BASE   : NORVAPPLS

Command ===> ■
F1=Help   F2=End   F3=Return   F8=Forward   F9=Refresh
                                         F6=Roll   F12=Retrieve

Press PF8 to see more
information, including
SDF focal points and the
history of captured
messages for the system
```

© Copyright IBM Corporation 2019

4-44

### DISPSYS: Display system information

The DISPSYS command panel shows automation settings on Automation managed systems. Included is information such as what NetView towers that are enabled, the current configuration data set, and defined runmodes. If data is not defined or a field is not relevant, for example, sysplex group name when not in a SAplex, the word NONE is shown in the field. Use PF8 to scroll forward and see more fields.

# INGNTFY: Managing notify operators

```

AOFKAASN          SA z/OS - Command Dialogs
Domain Id . : AOFDA ----- INGNTFY ----- Date . . : 10/28/18
Operator Id : INGC100                                     Time . . : 14:40:01

Status/Action => ADD          Operator status:
ON      - Set your notifications ON
OFF     - Set your notifications OFF
ADD     - Add or Modify an operator
DELETE  - Remove an operator
QUERY   - Look up an operator
Operator ID  => INGC100    Operator for ADD, DELETE or QUERY
Classes    => 40 80
Description => Rcv agent + mgr msgs

Held Messages - Information      => Y  Immediate Action  => Y
                  Eventual Action  => Y  Immediate Decision => Y
                  System Wait       => -
Target        => MVSA           System name, domain ID or sysplex name

```

Notify operators are defined to receive specific System Automation for z/OS messages, which are based on message class

```

AOF099I FUNCTION COMPLETED
Command ==> _____
F1=Help      F2=End      F3=Return
                                         F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

4-45

## INGNTFY: Managing notify operators

Notify Operators are logged-on users who receive Automation messages, which are based on message classes. Notify operators are identified in the automation configuration file. Use the INGNTFY command to manage notify operator assignments and add new notify operators at run time.

The messages are displayed on the user's NCCF session, with colors and attributes appropriate to their severity. The messages can also be held on the screen if required. The colors and attributes are defined as part of the NetView AOFMSGSY synonym definition. The Automation administrator can modify the colors and attributes.

On the primary INGNTFY panel, you can use row-command **C** to change existing entries. If you add a new notify operator or change an existing one, you see a panel similar to the example on this page. You can change the Action, Classes, or Held Messages settings.

The message classes that are used by Automation are documented in the *IBM System Automation for z/OS: Messages and Codes*.



**Note:** All changes that are made with this interface are temporary and are lost when the automation configuration file is reloaded.

# Lesson 5 Managing possible inhibitors



## Lesson 5. Managing possible inhibitors



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to find and remove possible inhibitors.

## Possible inhibitors

- Resource thresholds
- Schedules
- Triggers and events

### Managing possible inhibitors

- Resource thresholds
- Schedules
- Triggers and events

## An overview of resource thresholds

- Thresholds can be defined for resource recovery
- Commands, including a default command, can be defined for each threshold
- Thresholds can also be defined for minor resources
- System Automation for z/OS uses three thresholds:
  - INFR: Can be used as an initial indication of a possible problem
    - Issues a message and attempts recovery
  - FREQ: Can be used to indicate that the problem persists
    - Issues a more urgent message and attempts recovery
  - CRIT: The resource failed often enough to stop recovery
    - Recovery is suspended until the problem is corrected and the status of the resource is reset

### *An overview of resource thresholds*

Automation uses three thresholds when automating the restart of resources and minor resources. These thresholds can be used to identify the frequency of failures for an automated resource.

Restart automation is stopped when a defined CRIT (critical) threshold is met. When the critical threshold is met, operator intervention is required for Automation to restart the resource. Minor resource thresholds allow for System Automation z/OS administrators to define automated actions as reactions to each threshold, or a default action for all thresholds.

# INGTHRES: Display and manage thresholds

INGKYTH0		SA z/OS - Command Dialogs		Line 1 of 5	
Domain Id . : AOFDA		INGTHRES		Date . . . : 10/28/18	
Operator Id : INGC102				Time . . . : 14:42:59	
<b>Cmd:</b> A Add thresholds C Change thresholds D Delete thresholds					
Cmd	System	Resource	Critical	Frequent	Infrequent
---	---	---	---	---	---
—	MVSA	DEFAULTS	4 in 01:00:00	4 in 02:00:00	6 in 12:00:00
—	MVSA	MVSESA			
—	MVSA	MVSESA.MVSDUMP	6 in 00:20:00	4 in 00:20:00	2 in 00:20:00
—	MVSA	MVSESA.SMFDUMP	4 in 01:00:00	4 in 02:00:00	6 in 12:00:00
—	MVSA	SUBSYSTEM	3 in 12:00:00	3 in 24:00:00	2 in 24:00:00

Command ==> █

F1=Help F2=End F3=Return F9=Refresh F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

4-49

## INGTHRES: Display and manage thresholds

INGTHRES displays all thresholds that are defined in the automation control file, or set with the INGTHRES command. The example on this page shows the output of the INGTHRES command, when entered with no operands. Default thresholds can be defined at the SUBSYSTEM, MVSESA, or DEFAULTS level. The default threshold names are displayed in yellow in the Resource column. You cannot delete the threshold definitions for DEFAULTS, MVSESA, and SUBSYSTEM.

Use the (A)dd and (C)hange row-commands to create and modify new thresholds for a resource, or the (D)elete row-command to delete existing ones.

When used on the command line, INGTHRES can also set the defaults for all MVS subcomponents, subsystems, or MVSESA components.

To determine why a threshold is exceeded, you can use these commands:

- SDF to display the last message for the resource
- DISPINFO to display the captured messages for the resource
- BROWSE NETLOGA to find messages that are related to the resource

## INGSCHEDE: Managing schedules (service periods)

- List of defined schedules (service periods)
- List of resources that are linked to the service periods
- Base service period start and stop times:
  - UP time-slots and DOWN time-slots
  - Priority
- Display, create, change, or remove overrides to the service period
- Display, create, change, or remove overrides for a resource
  - Order of overrides display
    - Resource override
    - Schedule override
    - Base schedule

### INGSCHEDE: Managing schedules (service periods)

The INGSCHED command displays and updates schedules, also called service periods, that are known to the automation manager. You can use the INGSCHED command to:

- See which resources use (are linked to) a service period
- Display the service period windows of the base service period
- Display the service period overrides
- Add, update, or remove service period overrides
- Display resource overrides
- Add, update, or remove resource overrides

When displaying the definition for a particular day, Automation first looks to see whether there is a resource override for that date. If there is, the resource override is displayed; otherwise, it attempts to display a service period override for that date. If none is found, the service period definition for that day of the week is shown.

## INGSCHED: Display schedule list

```

INGKYSPI          SA z/OS - Command Dialogs
Domain ID = AOFDA ----- INGSCHED -----
Operator ID = INGC100           Sysplex = SAPLEX

Cmd: A Show Details   B Show Overrides   C Show Resources

Cmd Schedule Description
--- -----
  RV56SVP   Schedule for RV05 and RV06

```

```

INGKYSPI          SA z/OS - Command Dialogs
Domain ID = AOFDA ----- INGSCHED -----
Operator ID = INGC100           Sysplex = SAPLEX

Schedule. . : RV56SVP           Schedule for RV05 and RV06

Cmd: S Show Overrides
Cmd Resource Type System Description
--- -----
  RV05     APL  MVSA    RV05 Appl
  RV06     APL  MVSA    RV06 Appl

```

### INGSCHED: Display schedule list

This slide shows the panel that is displayed when an INGSCHED command is issued with no parameters. Automation displays all schedules that are known to the automation manager.

- Row-command **A** displays base schedule definitions (display only).
- Row-command **B** is used to display, add, or modify overrides to base schedule definitions.
- Row-command **C** displays the base schedule for the resources. For example, RV05 and RV06 on system MVSA are linked to RV56SVP. To understand the scope of any changes or overrides, you must know what resources the schedule controls.

## INGSCHED: Display base schedule details

```
INGKYS1          SA z/OS - Command Dialogs      Line 1  of 14
Domain Id . : AOFDA   ----- INGSCHED ----- Date . . : 10/25/18
Operator Id : INGC102  Sysplex = SYSPLEX1    Time . . : 08:27:31

Schedule. . : RV56SVP      Schedule for RV05 and RV06

Day     Pri  From-To    From-To    From-To    From-To
---  --  -----  -----  -----  -----
MON   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
TUE   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
WED   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
THU   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
FRI   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
SAT   UP   L  0100-2100  -----  -----  -----
DN   L  0100-2100  -----  -----  -----
```

Command ==> █  
F1=Help F2=End F3=Return F6=Roll  
F8=Forward F9=Refresh F12=Retrieve

© Copyright IBM Corporation 2019

4-52

### INGSCHED: Display base schedule details

Row-command **A** (Show Details) on the primary INGSCHED panel displays a panel similar to the example in this page. This schedule is defined in the manager configuration files. The display of the base schedule definition is read-only; however, you can create overrides to it, on other panels.

This example shows the weekly schedule for a base definition. No dates are shown because a day-of-the-week definition applies to all dates for that day.

# INGSCHED: Overriding a base schedule

```

INGKYS2          SA z/OS - Command Dialogs
Domain Id . . : AOFDA      INGSCHED      Date . . . : 10/25/18
Operator Id : INGC102      Sysplex = SYSPLEX1  Time . . . : 08:32:35

Resource . . . : RV05/APL/MVSA      Schedule . . . : RV56SVP
Starting date => 10 / 25 / 2018    (mm/dd/yyyy)

C Date          Pri From-To     From-To     From-To     From-To     From-To
-----+-----+-----+-----+-----+-----+-----+-----+
- 10/25/18 Thu UP  L 0833 0900  _____  _____  _____  _____
                  DN L 0100 0833  _____  _____  _____  _____
  10/26/18 Fri UP  L 0833 0900  _____  _____  _____  _____
                  DN L 0100 2100  _____  _____  _____  _____
  10/27/18 Sat UP  L 0833 0900  _____  _____  _____  _____
                  DN L 0100 2100  _____  _____  _____  _____
  10/28/18 Sun UP  L 0833 0900  _____  _____  _____  _____
                  DN L 0100 2100  _____  _____  _____  _____
  10/29/18 Mon UP  L 0833 0900  _____  _____  _____  _____
                  DN L 0100 2100  _____  _____  _____  _____

```

**AOF099I FUNCTION COMPLETED**

Command ==> \_\_\_\_\_

F1=Help	F2=End	F3=Return	F6=Roll
F7=Backward	F8=Forward	F9=Refresh	F12=Retrieve

Overrides are shown in yellow and pink

## INGSCHED: Overriding a base schedule

When row-command **S** (Show Overrides) is entered next to a resource on INGSCHED panel INGKYS3, you see details of the service windows on INGSCHED panel INGKYS2. The current date is shown first in the list of dates. To see a different set of dates, enter a new start date in the STARTING DATE field. Entering row-command **I** next to a resource on the INGLIST panel also displays panel INGKYS2. Scroll through the display to see any overrides and base schedules for the resource. Time slots that have already past are shown in blue. The overrides for past dates might be automatically deleted by the manager.

To change or override a schedule, type over the existing times. To remove all overrides on a line, enter the command **D** against that line. Overrides for a specific resource, *resource overrides*, are shown in pink. A resource override applies only to the selected resource. It does not apply to other resources that are linked to the same schedule definition.

To change a schedule so that it affects all resources that are linked to it, enter row-command **B** next to the schedule name on the list of schedules panel, INGKYS0. Overrides to a schedule affect all resources that are linked to the schedule, and are known as *schedule overrides*. Overrides are color-coded. Resource overrides are pink. Schedule overrides are yellow, and base definitions are green.

## Triggers and events

- Used to manage the start and stop of resources
- Event: Represents an external switch (set or unset)
- Trigger: Controls the starting or stopping of the resource
  - A combination of events
- INGTRIG: Displays automation manager triggers
- DISPTRG: Displays resource triggers
- DISPEVTS: Displays events
  - DISPEVT: Displays event details

### Triggers and events

Events are processes that occur outside of the automation environment. Triggers are connected to resources (applications, application groups, or monitor resources) and inhibit requested actions under certain conditions.

If you use INGLIST to manage your resources, you might notice that a start or stop request is not being processed. For example, desired status and observed status do not match. Use **PF11** to scroll the INGLIST display to the right to see service periods and triggers. You can use row-command **H** (DISPTRG) to display the triggers or row-command **I** (INGSCHED) to display the service periods.

# INGTRIG: Display triggers

```

INGKYTR0                               SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : AOFDA  ----- INGTRIG ----- Date . . : 10/31/18
Operator Id : INGC102           Sysplex = SYSPLEX1    Time . . : 12:32:03

Cmd: S show associated resources

Cmd Trigger   Description
s RV01TRIG trigger for RV01

```

Trigger

```

INGKYTR1                               SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : AOFDA  ----- INGTRIG ----- Date . . : 10/31/18
Operator Id : INGC102           Sysplex = SYSPLEX1    Time . . : 12:33:48

Trigger . . : RV01TRIG      trigger for RV01
Cmd: S show trigger details
Cmd Resource   Type System   Description
s RV01       APL MVSA     RV01 starts & has start trigger RV01TRIG

```

Resource having trigger

```

A0FLT000                               SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : AOFDA  ----- DISPTRG ----- Date . . : 10/31/18
Operator Id : INGC102           System = MVSA          Time . . : 12:35:38

Resource      => RV01/APL/MVSA
Target        => RV01TRIG          System name, domain ID or sysplex name
Trigger. . . : RV01TRIG          Observed status : SOFTDOWN
Cmd: S show details
Cmd Type   Events
s STARTUP   RV01EVT

```

Trigger details

© Copyright IBM Corporation 2019

4-55

## INGTRIG: Display triggers

Automation displays panel INGKYTR0, shown in the upper example, when you enter command INGTRIG with no operands. If defined, you see a list of triggers. The scope of INGTRIG is SAplex-wide.

The INGTRIG command can be used for these purposes:

- Display defined triggers
  - Trigger name
  - Trigger description
- List resources that are linked to a trigger
  - Resource name
  - Resource description
- Initiate the DISPTRG command
  - Conditions and events

Entering row-command **S** next to a trigger shows panel INGKYTR1, the second example on the slide. Resources that are linked to the trigger are listed. You can also get to the second panel directly by issuing the command, INGTRIG *trigger\_name*. A trigger can be satisfied for some resources and

not others because the INGEVENT command can set the trigger for one application or resource and not others. The automatic UNSET of an event applies at the application level.

Entering row-command **S** next to one of the resources on the second panel displays trigger information for that resource. Colors are used to indicate whether a trigger is satisfied. Satisfied triggers are displayed in white (on a normal black background), and unsatisfied triggers are displayed in red.

Entering row-command **S** next to a trigger displays the events that make up the trigger, and the status of each event. Each event can then be manually set or reset.

## DISPEVTS: List the events in a SAplex

```

AOFLT100          SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : A0FDA ----- Trigger Condition List ----- Date . . : 10/31/18
Operator Id : INGC102           System = MVSA        Time . . : 12:43:22

Trigger . . . : RV01TRIG
Condition type: STARTUP
Resource. . . : RV01/APL/MVSA          Observed status : SOFTDOWN

Cmd: D show resources  S set event for resource  U unset event for resource
Cmd Event Status     Unset Description
----- -----
d  RV01EVT  UNSET      UP

AOFLE000          SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : A0FDA ----- DISPEVTS             Date . . : 10/31/18
Operator Id : INGC102           Sysplex = SYSPLEX1  Time . . : 13:53:08

Event      => RV01EVT
Target     => MVSA      System name, domain ID or sysplex name
Description . . .
Unset condition. : UP

Resource    Type  System   Trigger   Status
----- -----
RV01       APL   MVSA     RV01TRIG

Event
----- -----
Resource having event
----- -----
```

```

A0FKMSG0          SA z/OS - Command Dialogs      Line 1   of 2
Domain Id . : A0FDA ----- DISPTRG             Date . . : 10/31/18
Operator Id : INGC102           Time . . : 13:55:33

Sel System  Message
----- -----
-  MVSA      A0F442I    SET EVENT RV01EVT DONE FOR RESOURCE RV01/APL/MVSA ON
                           SYSPLEX1.

Event set
----- -----
```

© Copyright IBM Corporation 2019

4-56

### DISPEVTS: List the events in a SAplex

The DISPEVTS command is used to display panel AOFLE100 on which you see all the events in a SAplex. Entering row-command **D** next to an event displays panel AOFLE000 (the panel in the second example). That panel shows the resources that are associated with that event. Row-command **S** command displays the INGEVENT panel on which you can set or reset events.



**Hint:** For the DESCRIPTION field in your environment, you use text that describes the event and how it relates to the resource.

# Lesson 6 Sysplex automation commands



## Lesson 6. Sysplex automation commands



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to use the INGPLEX command to manage and control your sysplex environment.

# INGPLEX

---

- Manage couple data sets: INGPLEX couple data set
  - Automatic creation and recovery of alternate couple data sets
  - Alternate couple data set is always available for these items:
    - Sysplex
    - Automatic restart manager
    - Coupling facility resource manager
    - Sysplex logger
    - Sysplex failure management
    - Workload manager
- Manage coupling facility: INGPLEX coupling facility
  - Drain: Prepares coupling facility for removal from the sysplex
  - Enable: Activates a coupling facility into a sysplex
  - Path: Controls the sender paths of a coupling facility
  - Structure: Displays, rebuilds, or deletes coupling facility structure
- Note: The **INGSTR** command can be used instead of INGPLEX coupling facility
- Manage the system logger

## *INGPLEX*

Automation provides functions to help in the management of sysplex environments. Management functions are available through commands such as INGPLEX and INGSTR. Other functions such as hardware validation are also provided.

# INGPLEX: Command options

```
INGLX000          SA z/OS - Command Dialogs
Domain Id . . : A0FDA      ----- INGPLEX ----- Date . . . : 10/28/18
Operator Id : INGCC102      Sysplex = PLEX12           Time . . . : 14:53:00

SAplex . . . . . : SYSPLEX1

Select the desired command:           INGPLEX . . .

1 Display systems (including ETR & signalling paths)   SYSTEM
2 Display consoles                                     CONsole
3 Control coupling facilities (INGCF)                CF
4 Control couple data sets                          CDS

6 Display IPL information                         IPL
7 Control dumps                                 DUMP

10 Control all structures (allocated and unallocated) (INGSTR) STR

Command ==> _____
F1=Help     F2=End     F3=Return           F6=Roll
                                         F12=Retrieve
```

## INGPLEX: Command options

Using the INGPLEX command helps you to manage and control your sysplex environment. This slide shows an example of the command output when it is entered with no parameters. From this panel, command options can be chosen.

The command can also be run with one of the following parameters:

- **SYStem:** Displays information about a system in the SAplex, such as its name, status, and global resource serialization (GRS) mode (STAR or RING).
- **CONsole:** Displays MVS console information such as the master name, WTO and WTOR buffer utilization, number of queued messages, and awaiting mounts and operator requests for the target SAplex.
- **CDS:** Displays couple data set information for all known couple data sets. Using this parameter with the command provides the following management functions:
  - Switch from primary to alternate couple data sets
  - Define new alternate couple data sets
  - Change the active policy

- **CF:** Runs the INGCF command to manage coupling facilities, including these supported functions:
  - Removal of all allocated structures
  - Disconnection of a coupling facility
  - Activating a coupling facility
  - Displaying sender paths
  - Rebuilding a selected coupling facility structure
- **IPL:** Displays IPL-related information for each system, such as the system name and when the last IPL was done. You can also compare data across IPLs to see what changed.
- **DUMP:** Provides management functions to control SDUMP options, initiate a supervisor call (SVC) dump, and control serviceability level indication processing (SLIP) trap settings.

The system logger (LOGR) provides a sysplex-wide logging facility. Applications write their data to a logging stream, which is usually associated with a coupling facility structure. As a result, LOGR is one of the couple data set types that are supported by Automation.

When a directory shortage is detected, Automation automatically reformats the primary and alternate LOGR couple data set with a larger DSEXTENT parameter.

Operators can use the INGPLEX CDS TYPE=LOGR command to display the details of the LOGR couple data set and the control card information, and list the log streams and structures.

## INGSTR command

You can use the INGSTR command to display and manage all of the coupling facility structures that are defined in your active policy. You can also use this command to display and manage all allocated coupling facility structures in the sysplex.

Options are available to include the rebuild condition with each structure (COND=YES) and to display all structures, including unallocated structures (ALL=YES). The options can also be specified on the INGSTR panel. You can rebuild or delete a selected structure or start and stop the duplexing of a structure, if applicable. INGSTR supports full screen mode and line mode. In line mode only, the display function is available.

# INGPLEX: List XCF couple data sets

```

INGKX300          SA z/OS - Command Dialogs           Line 1    of 15
Domain Id . : AOFDA      ----- INGPLEX CDS ----- Date . . : 10/28/18
Operator Id : INGC100     Sysplex = PLEX12        Time . . : 14:55:21

SApplex . . . . . ==> SYSPLEX1      COUPLExx . . . . . : COUPLEAB
Interval . . . . . : 165          OPNotify . . . . . : 168
Maxmsg . . . . . : 500          Cleanup . . . . . : 15
Retry . . . . . : 10           Classlen . . . . . : 1024
Max CFlevel . . . . : 20         Max SMlevel . . . . : 20
SMREBLD . . . . . : N/A

Cmds: A allocate alternate CDS / C display CHPIIDs
      D display CDS information / P switch alternate CDS to primary CDS

      Type      MS      Volume   Dev   Couple Dataset Name
      ----      --      -----   ---   -----
      — SYSPLEX
      PRIMARY..: 8      MVC206  0506  SYS2.PLEXA.CDS01
      ALTERNATE: 8      MVC207  0517  SYS2.PLEXA.CDS02
      — ARM
      PRIMARY..: 8      MVC206  0506  SYS2.PLEXA.ARMCPL01
      ALTERNATE: 8      MVC208  0518  SYS2.PLEXA.ARMCPL02
      — BPXMCDS
      PRIMARY..: 8      MVC207  0517  SYS2.PLEXA.OMVS1
      ALTERNATE: 8      MVC208  0518  SYS2.PLEXA.OMVS2
      — LOGR
      PRIMARY..: 8      MVC206  0506  SYS2.PLEXA.FDSS1
      ALTERNATE: 8      MVC208  0518  SYS2.PLEXA.FDSS2

Command ===>
F1=Help   F2=End   F3=Return   F6=Roll
          F8=Forward F9=Refresh   F12=Retrieve

```

© Copyright IBM Corporation 2019

4-60

## INGPLEX: List XCF couple data sets

Automation provides functions to automatically create and recover alternate couple data sets for each of the six supported couple data set types. The alternate couple data sets can be used as backups when the primary couple data set fails. The Automation administrator defines this automation in the customization dialogs.

This slide shows an example of the INGPLEX **CDS** command. In this case, a couple data set is defined for the sysplex and ARM. PF8 indicates that there is more data for display. Command actions are available to display more detailed data or allocate an alternate couple data set for a coupling facility and switch to it.

The couple data set can be one of the following types:

- **Sysplex:** Contains the policy and status for basic sysplex functions and points to the other couple data sets
- **ARM:** Contains the policy for automatic restart manager
- **CFRM:** Contains the policy for coupling facility resource manager
- **LOGR:** Contains an inventory for the sysplex logger function
- **SFM:** Contains the policy for sysplex failure management
- **WLM:** Contains the policy for the workload manager

# INGPLEX: IPL information

```

INGLX201          SA z/OS - Command Dialogs          Line 1      of 32
Domain Id . . . : AOFDA   ----- INGPLEX IPL ----- Date . . . : 10/28/18
Operator Id : INGC100     Sysplex = PLEX12           Time . . . : 15:17:10

System . . . . . : MVSA          IPL device/volume . : 0500      / ZOSOTO
Sysplex . . . . . : PLEX12        BCP release/FMID   . : SP7.2.2    / HBB77AO
BCP name . . . . : z/OS          LPAR/CPC name . . . :          / A01
Load parameters . . . : 050562M1  IPL timestamp/local : 2018-10-20 00:17
Node descriptor . . . : 002827.H43.IBM.02.00000003F967 / 10-20 00:17
Configuration id . . . : MVS1A       Active IODF . . . : SYS1.IODF01
MCAT volume/dsn . . . : MVC201 / ICFCAT.MVSC20.MASTCAT

----- Cmnds: C(S) compare (single) member(s) / D(S) display (single) member(s)

Param. Member Suffix(es)          CMNWKWND OUTPUT FROM SAZOS.AOFDA.INGC100.INGPIPLN LINE 0 OF 61
----- *----- Top of Data *----- IEASYS = 00 = *
----- APG=07, AUTOMATIC PRIORITY GROUP IS 7 DEFAULT
----- AUTOR=(00,01), SELECT AUTOR..
----- CATALOG=00, SELECT IGGCAT..
----- CLOCK=S0, S SELECT CLOCK00 DEFAULT
----- CLPA, CLEAR LINK PACK AREA
----- CMB=(UNITR,COMM,GRAPH,CHRDR), ADDITIONAL CMB ENTRIES
----- CMD=SO, S TOD PROMPT, SDUMP, TRACE ON AND RMF
----- CON=(00,NOJES3), S SELECT CONSOLO0 DEFAULT
----- COUPLE=AB, S COUPLING MEMBER
----- CSA=(4600,240M), CSA RANGE
----- DIAG=00, DEFINE DIAGXX MEMBER
----- DUMP=(DASD,00-02), DUMP TO DASD
----- FIX=00, FIX MODULES SPECIFIED
----- GRS=TRYJOIN, S NO COORDINATION OF GRS REQUESTS
----- GRSRNL=00, S NO DS ENQ IN SYSPLEX
----- LNKAUTH=LNKLST, DEFAULT, APFTAB IS ALT
----- LOGCLS=A, SYSOUT CLASS FOR LOG IS A
----- LOGLMT=020000, MUST BE 6 DIGITS, MAX WTL MSGS QUEUED
----- LOGREC=SYS2.&SYSNAME..LOGREC, NAME OF THE LOGREC DATASET

d IEASYM  IEASYM 00 62          INGPLEX details for IEASYSxx
d SYSPARM  IEASYS 00 62 62
d ALLOC    ALLOC
d APF      IEAPF
d CLOCK   CLOCK  S0
d CMD     COMMND 62
d CON     CONSOL 00
d COUPLE  COUPLE AB
d DEVSUP  DEVSUP
d DIAG    DIAG   00
d EXIT    EXIT

Command ===> F1=Help F2=End F3=Return F10>Show all F6=Roll F12=Retrieve
          F8=Forward
          
```

© Copyright IBM Corporation 2019

4-61

## INGPLEX: IPL information

INGPLEX IPL displays a list of system IPL records. Use row-command **D** to display detailed information, such as the COMMNDxx members used between system IPLs.

# Lesson 7 UNIX System Services automation



## Lesson 7. UNIX System Services automation



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn about UNIX System Services automation and the UNIX System Services add-on policy.

# UNIX System Services automation

- System Automation for z/OS provides monitoring and automation of UNIX System Services resources
  - Processes
  - Files
  - TCP/IP ports
- Managed like any other System Automation for z/OS resource
  - Can be monitored, started, or stopped
  - Can have dependencies; for example, to other IP-related services
  - STARTUP commands
  - SHUTDOWN commands

## *UNIX System Services automation*

UNIX System Services (USS) automation monitors and automates USS processes, files, and TCP/IP ports. Automation also provides an API to run z/OS UNIX commands: the INGUSS command. For example, you can automatically mount a file system when a USS process starts, and unmount the file system when the process ends. USS resources are applications with an application type of USS.

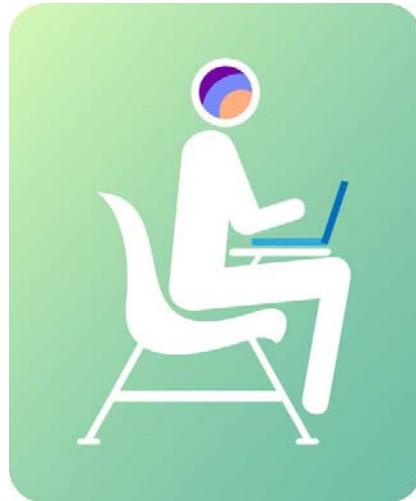
## UNIX System Services add-on policy

- UNIX System Services add-on policy to automate resources
  - SYSLOGD
  - PORTMAP
  - SSHD
  - FTPD
  - InetD
- UNIX System Services resources must run in a z/OS system that SA z/OS manages
  - The automation agent is responsible

### *UNIX System Services add-on policy*

Automation provides the \*USS add-on policy to manage and automate resources such as SYSLOGD, FTPD. Your system administrator can import the \*USS add-on and customize it for your environment.

## Student exercise



© Copyright IBM Corporation 2019

4-65

### Student exercises

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you completed this unit, you can perform the following tasks:

- Use System Automation for z/OS commands
  - Set resource start types and stop types
  - Modify automation flags and resource status
  - Refresh configurations
  - Modify the runmode for a system
  - Display and define Notify Operators
  - Display Service Periods, Triggers, and Events
- Use the INGPLEX command for sysplex functions

*Summary*

Now that you completed this unit, you can perform the following tasks:

- Use System Automation for z/OS commands
- Set resource start types and stop types
- Modify automation flags and resource status
- Refresh configurations
- Modify the runmode for a system
- Display and define Notify Operators
- Display Service Periods, Triggers, and Events
- Use the INGPLEX command for sysplex functions





# 5 Application groups



## Unit 5: Application groups



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit introduces application groups, group types, group natures, and their attributes. The unit highlights and explains the benefits of groups when managing automated business applications in an enterprise. The lessons include topics on group management commands such as INGGROUP, INGMOVE, INGVOTE, and INGLIST.

# Objectives

---

- When you complete this unit, you can perform the following tasks:
  - Describe application moves
  - Use application group management commands across the SAplex
    - Start
    - Stop
    - Display
    - Move
  - Describe application group preference value thresholds and calculation
  - Explain move group examples

## *Objectives*

When you complete this unit, you can perform the following tasks:

- Describe application moves
- Use application group management commands across the SAplex
  - ◆ Start
  - ◆ Stop
  - ◆ Display
  - ◆ Move
- Describe application group preference value thresholds and calculation
- Explain move group examples

# Lesson 1 Application moves



## Lesson 1. Application moves



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn about application moves, when they are initiated, what actions they cause and how the INGGROUP and INGMOVE commands support application moves as well as how you can speed up a move.

# Initiating application moves

- External events
  - System leaves the SAplex (SA z/OS only)
  - Application fails or has recovered
- Policy goals
  - INGGROUP command options
    - Exclude
    - Avoid
    - Include
- Manual actions (Targeted moves)
  - Manually move applications from one system to another using
    - INGMOVE: Select the target member to start the application
    - INGGROUP: Adjust preference values for application group members

## *Initiating application moves*

You can initiate application moves in one of three ways: externally, goal driven, or targeted.

Externally initiated moves can be a reaction to the loss of a system or when a member becomes unusable. Externally initiated moves can also be triggered when a preferred member becomes usable again, *or when a system rejoins the SAplex automated by System Automation for z/OS*.

In goal driven moves, the choice of members might vary. A member can be excluded, avoided, or included when deciding which members to start. Such actions are usually part of a planned move to accommodate a scheduled outage.

Targeted moves occur when group members are chosen manually at run time. The INGGROUP command can be used to modify member preference values.

Instead of the INGGROUP command, the INGMOVE command can be used to initiate immediate or planned moves. Using the INGMOVE command is simpler than calculating the results of adjusting preference values. Application moves apply to both move groups and server groups.

## Evaluating move actions

- Moving an application generates at least two votes
  - MakeUnavailable for the active member of the group
  - MakeAvailable for the replacement member
- The move mode policy determines when votes are generated
  - Parallel: Resources are started and stopped concurrently
  - Serial: Replacement resource is started after the active member is stopped

### Evaluating move actions

Moving a member application generates two votes. One vote stops the currently active member, and the second vote starts a different member. By default, both votes are generated at the same time. The result is that the replacement member starts at the same time that the active member is stopping. This result can cause a problem for some applications. To control the start of a replacement member in a server or move group, move mode policy must be specified in the customization dialogs.

The Move Mode defines when new group members are started in relation to the stopping of existing group members. A value of *PARALLEL* indicates that new members are started as old members are stopping. A value of *SERIAL* indicates that new members are started after the old members stop.

To ensure that only one copy of an application is running, specify serial move mode policy in the customization dialog. This technique replaces the need for an artificial MakeAvailable/WhenObservedDown relationship from the application to the group itself.

# Using INGGROUP for goal driven moves

Use INGGROUP to manage sysplex application groups

- Exclude systems
  - Immediately deselect resources on the target system so that they cannot be moved
  - Attempt to maintain availability goals by starting members on other systems in the group
  - These systems are disruptive
- Avoid systems
  - Receive reduced preference for the future
  - Active resources are not moved away
  - These systems are not disruptive
  - Moves resources away as scheduled outages allow
- Include systems
  - Undo previous exclude or avoid actions

© Copyright IBM Corporation 2019

5-6

Using INGGROUP for goal driven moves

Manual and automated actions can influence how a group is managed. Using the INGGROUP command, an operator can *EXCLUDE* a system from consideration when a group is evaluated. All active members in the system are immediately stopped. As resources are stopped, the group is reevaluated. Resources on the excluded system are considered non startable, and are not started. The group is set to *Recovery Mode* allowing *backup members* (with adjusted preference values less than 600) to be selected. Exclude is usually done immediately before a system is shut down.

On the INGGROUP command panel, an operator specifies the AVOID action for a system. The system is not considered when server groups are evaluated. All applications that are running on that system are left running. Any unavailable resource that requires the system receives a maximum effective preference value of 100. If applications are stopped, the group is reevaluated. However, applications on that system are not started unless necessary. The group is not put into RECOVERY MODE. An AVOID action is usually done well before a system is shut down. Using the AVOID action prevents applications from starting on a system just before the system is shut down.

An INCLUDE action voids the actions of EXCLUDE and AVOID. Server group members that are linked to the system are considered when the groups are being evaluated. An include action can trigger the starting and stopping of members in groups that are linked to the system which was the target of the include action.

With goal driven moves, you can pacify or activate a group. The PACIFY action sets an application group passive immediately. The PACIFY action withdraws all votes for the group. The ACTIVATE action sets an application group to active. This action allows for the propagation of group votes to members. When changing group policy (move and server groups), there can be new availability or satisfactory target (server group only). You can adjust preference values for members and use INGMOVE for move groups.

All actions are persistent. If a system is avoided for a group, that avoidance continues even after an IPL. Only a warm or cold start of the automation manager resets the AVOID action.

## INGMOVE: A command for move groups

- Start and stop the move group
- Manually move an application
  - Immediately (option M)
  - At the next recycle of the group (option P)
- Box the system
  - Set the current member preference value to 3200
  - Prevent the application from being moved to another system
- Reset the preference values to the defaults in the policy
- Display votes and relationships
- Display group members and their preference values
- Display service period definitions
- Display resource details

Move an application without changing a preference value

© Copyright IBM Corporation 2019

5-7

### *INGMOVE: A command for move groups*

INGGROUP can be used to manage all types of application groups. Typically, INGGROUP and INGLIST are used to accomplish moves.

INGMOVE combines INGLIST functions such as displaying votes, relationships, and service periods, with more functions specific to managing move groups. With INGMOVE, you can perform these tasks:

- Manually move an application without modifying preference values as you would from INGGROUP.
- Box the group; for example, to prevent the move of an application when you IPL again, you BOX the system before the shutdown. During the IPL, the application is not moved.
- Reset preference values back to the defaults that are defined in the move group policy.

## Prepare Move policy

If Prepare Move policy is set to YES:

- Move group member is not stopped until the replacement is ready to be started
  - For example, a parent must be started on a target system
- Server group member is not stopped until the replacement is running

### Prepare Move policy

Prepare Move is an application group policy attribute that specifies whether the downtime during the move or switch of a resource should be kept as short as possible:

For an application group of nature MOVE this means that the group member that is currently running will not be stopped until the replacement member is ready to be started, for example, all parents are available.

For an application group of nature SERVER this means that a group member will not be stopped until the replacement member that has just started is available.

To have the highest possible availability, it is recommended that your administrator sets this value to YES in the unless the application cannot support this.

## Move only if system is down or stopping vs. Move only after system failure

- Primary member has defined preference 2800
- Bonus value of -400 for stopping system
- Sticky bonus value of 175
- Effective preference calculated to 2575 ( $2800 - 400 + 175$ ) which is below **2600 threshold**
- **If backups preference is 500** no move will happen unless the group is in **recovery mode**, that is, the member was active when the system failure (SYSGONE) occurred
- **If backups preference is 700** a move will happen when an operator requests a shutdown
- If the effective preference is below **2400 threshold**, similar if the primary member is in HARDDOWN

*Move only if system is down or stopping and Move only after system failure*

*Primary member defined preference 2800, backups preference 500:*

As long as the primary is not SYSGONE it will be selected. If the primary is SYSGONE, its preference drops to 2575 ( $2800 - 400 + 175$ ), allowing an alternative viable member to be selected.

The backups will only be viable if the resource was previously active because of the groups recovery mode.

A shutdown of the primary system will not initiate a move of the application. Although the primary systems preference will be below the 2600 threshold, there will be no viable alternatives as the backups may only be used if the group is in recovery mode.

*Primary member defined preference 2800, backups preference 700:*

A shutdown of the primary system will initiate a move of the application during the shutdown as the backups are always viable.

If the effective preference is below the 2400 threshold, the move is similar if the primary member is in HARDDOWN.



## Lesson 2 Group management commands



### Lesson 2. Group management commands



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn about group management commands.

# Group management commands

- Commands to manage application groups
  - INGGROUP
  - INGMOVE
  - DISPAPG
- Commands to manage resources
  - INGLIST
  - INGVOTE
  - Other commands

## *Group management commands*

The most commonly used commands to manage application groups are INGGROUP, INGMOVE, INGVOTE, and INGLIST.

- INGGROUP: Displays group information and shows a menu of actions to modify defined policy for groups
- INGMOVE: Displays sysplex move groups, and provides a menu of actions to manage move groups
- INGVOTE: Displays votes for automated resources
- INGLIST: Displays status of resources, and command dialogs to manage and monitor resources on the list

DISPAPG displays basic information about application groups, from the perspective of the automation agent.

## **INGROUP: Primary panel**

**INGKYGRA** SA z/OS - Command Dialogs  
Domain Id . . . : AOFDA INGGROUP Date . . . : 10/28/18  
Operator Id : INGC102 Sysplex = SYSPLEX1 Time . . . : 18:53:38

Specify or revise the following data:

Target => \_\_\_\_\_ System name, domain ID or sysplex name

Action => \_\_\_\_\_ EXCLUDE-AVOID-INCLUDE or ACTIVATE-PACIFY or  
ADJUST or RESET DEFAULT or OVERRIDES, POLICY or  
RECYCLE-CANCEL

Opt.Parms => \_\_\_\_\_

Group(s) => RVSERVER/APG/MVSA

System(s) => \_\_\_\_\_

Command ==> \_\_\_\_\_

F1=Help F2=End F3=Return F4=Members F10=GO F6=Roll  
F12=Retrieve

Use PF4 to  
display group  
names or  
members

© Copyright IBM Corporation 2019

5-12

### *INGGROUP: Primary panel*

This slide shows an example of the INGGROUP primary panel which is displayed when row command **J** is entered next to an application group on the INGLIST panel. Use the INGGROUP command to display the application group members, their preference values, selectability, and their statuses. The group name is specified on the Group(s) field.

From this panel, you can perform these tasks:

- Exclude, Avoid, or Include a range of systems when managing one or more application groups. You enter the targeted systems in the SYSTEMS field. Display the systems that are defined for an application group by entering its name in the **Groups** field, and a question mark (?) in the SYSTEMS field.
  - Change the behavior of a group from ACTIVE to PASSIVE. This change causes the revocation of existing votes from the groups members and eliminates vote propagation. However, Automation still evaluates the status of the group. This change can also affect the

dependency relationship to other resources. Conversely, you can change the behavior of a group from PASSIVE to ACTIVE. Groups are normally active.

- Display OVERRIDES, such as any previous changes to preference values, satisfactory target, or availability target for a specified application group.
- Initiate or halt the *rolling recycle* of specified server and move groups. If the systems in a server group all have the same preference value, the active system at the end of a rolling recycle can be different than the active system before the action.

Specifying Action=OVERRIDES displays any changes that are made to preference values, satisfactory target, or availability target for the application group.

You must press PF10 to implement any changes that you make on this panel. Pressing PF4 on this panel displays panel INGKYGRB on which group members are listed. The next slide shows an example.

## INGGROUP members panel example

```

INGKYGRB          SA z/OS - Command Dialogs      Line 1      of 4
Domain Id . : AOFDA   ----- INGGROUP ----- Date . . : 10/28/18
Operator Id : INGC102           Sysplex = SYSPLEX1 Time . . : 19:07:10

Group: RVSERVER/APG/MVSA       Nature: Server  Passive: NO    Suspend:
Description: Server Group for RV Applications
Excluded :
Avoided :
Mode : Normal     Availability Target: 2     Adjust: 0     Result=> 2
                           Satisfactory Target=> 2     Adjust: 0
Rolling Recycle: None

Name      Type System  Pref  Adj  Result  Avl  Eff  Stat  Act  Sus
-----  -----
RV01      APL  MVSA    500   0   500    No   0   Rcv   --   --
RV02      APL  MVSA    500   0   500    No   0   Rcv   --   --
RV03      APL  MVSA    700   0   700    Yes  950  Sel   --   --
RV04      APL  MVSA    700   0   700    Yes  950  Sel   --   --

Command ==>
F1=Help   F2=End   F3=Return   F9=Refresh   F10=GO   F11=Reset   F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

5-13

### INGGROUP members panel example

This slide shows an example on the INGGROUP members panel. All four members in this example of a server group are on system MVSA. The group mode is normal because all active group members have effective preference values greater than 600. On this panel, you can change preference values and target values to control the starting and stopping of members:

- SATISFACTORY TARGET: Change the value for the satisfactory target for the group.
- RESULT FIELD: Change the value for the availability target for the group.
- RESULT COLUMN: Change the effective preference value for a group member.

For each member, the columns on this panel show this information:

- PREF: Base preference value (specified in customization dialogs)
- ADJ: Adjusted preference value (runtime adjustment to base preference value)
- Result: Modified preference value
- EFF: Effective preference value (sum of the base preference value, adjusted preference value, and any bonus values assigned by the automation manager)
- STAT: Status of the application
- ACT: Shows the MakeAvailable and MakeUnavailable actions for the selected resources

Note: Changes take effect only after you press PF10.

## INGGROUP members panel example (continued)

INWKYGRB	SA z/OS - Command Dialogs	Line 1 of 4							
Domain Id . : AOFDA	----- INGGROUP -----	Date . . : 10/28/18							
Operator Id : INGC102	Sysplex = SYSPLEX1	Time . . : 19:08:42							
Group: RVSERVER/APG/MVSA	Nature: Server	Passive: NO							
Description: Server Group for RV Applications		Suspend:							
Excluded :									
Avoided :									
Mode : Normal	Availability Target: 2	Adjust: -1							
	Satisfactory Target=> 1	Adjust: -1							
Rolling Recycle: None		Result=> 1							
Name	Type System	Pref	Adj	Result	Avl	Eff	Stat	Act	Sus
RV01	APL MVSA	500	0	500	No	0	Rcv	--	--
RV02	APL MVSA	500	0	500	No	0	Rcv	--	--
RV03	APL MVSA	700	0	700	Yes	950	Sel	--	--
RV04	APL MVSA	700	0	700	Yes	950	Uns	MU	--

Modifying the preference result field for a member or the availability target of the group can cause the generation of MakeAvailable and MakeUnavailable votes

Command ===  
F1=Help F2=End F3=Return F9=Refresh F10=GO F11=Reset F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

5-14

### INGGROUP members panel example (continued)

This example shows that the availability target is changed from 2 to 1. The ADJUST field reflects the adjustment. There is a negative 1 (-1) in the field. The result of the change is the generation of a MakeUnavailable vote for one of the active resources.

The automation manager adds a bonus 25-point value to a selected member if the adjusted preference value for that member is less than or equal to 1000. If the adjusted preference value for a selected member is 1000 or more, the bonus point value is 175.

The automation manager assigns bonus value points as follows:

- +225 to a member with an observed status of Available or Stopping and the system is not stopping
- +25 more to the member if the adjusted preference value is <= 1000

OR

- +175 more to the member if the adjusted preference value is > 1000

See the Lesson 3, "Optional topics," on page 278 for more details about the automation manager bonus values.

## INGGROUP: Rolling recycle

```

INGKYINO          SA z/OS - Command Dialogs      Line 112  of 554
Domain Id . : AOFDA   ----- INGINFO -----    Date . . : 10/28/18
Operator Id : INGC100 Sysplex = SYSPLEX1     Time . . : 19:21:56

Resource  => RVSERVER/APG/MVSA      format: name/type/system
Target    => _____ System name, domain ID or sysplex name

HSAL6462I Rolling Recycle; Sequence complete
HSAL6172I Group Observer update sent
HSAL6427I Group requires evaluation
HSAL6172I Group Observer update sent
HSAL6457I Member Shutdown Held
HSAL6436I Selected member preparing to start
HSAL6427I Group requires evaluation
HSAL6172I Group Observer update sent
2018-10-28 19:13:38 Agent status for RV03/APL/MVSA = AUTODOWN
HSAL6427I Group requires evaluation
HSAL6172I Group Observer update sent
2018-10-28 19:13:26 Shutdown in progress for RV03/APL/MVSA
HSAL6427I Group requires evaluation
HSAL6172I Group Observer update sent
2018-10-28 19:13:26 Agent status for RV03/APL/MVSA = AUTOTERM
HSAL6427I Group requires evaluation
HSAL6172I Group Observer update sent
HSAL6262I Status/Observed is Degraded
HSAL6348I Group Observer Update Requested
HSAL6281I Status/Compound is Degraded
HSAL6172I Group Observer update sent

String found
Command ===>
F1=Help      F2=End      F3=Return      F4=DISPAPG      F6=Roll
F7=Backward  F8=Forward  F9=Refresh     F12=Retrieve

© Copyright IBM Corporation 2019

```

5-15

INGGROUP: Rolling recycle

This slide shows an example of the information that is logged by the automation manager when a rolling recycle is started for a server or move group. The RECYCLE action is entered on the INGGROUP primary panel. The action consists of sequentially stopping and then restarting each member of the group. At the end of the recycle all members that were active when the operation was requested are stopped and recycled, or stopped and left down. All the members of the group are started and stopped.

The recycle action can be done in one of two ways.

- In serial, which is stopping and restarting one member at a time
- In parallel, which is stopping and restarting several members concurrently

To use the serial method, specify the RECYCLE action with no parameters. To use the parallel method, specify the RECYCLE action with the keyword parameter CHUNK, and specify how many to recycle in parallel.

## INGGROUP: Rolling recycle notification

```

INGKGRB          SA z/OS - Command Dialogs      Line 1      of 4
Domain Id . : AOFDA  ----- INGGROUP ----- Date . . : 10/28/18
Operator Id : INGC102   Sysplex = SYSPLEX1    Time . . : 19:24:14

Group: RVSERVER/APG/MVSA      Nature: Server  Passive: NO   Suspend:
Description: Server Group for RV Applications
Excluded :
Avoided :
Mode : Normal   Availability Target: 2   Adjust: 0   Result=> 2
                           Satisfactory Target=> 2   Adjust: 0
Rolling Recycle: Complete 2018-10-28 19:13:38

Name  Type System  Pref  Adj  Result  Avl  Eff  Stat  Act  Sus
-----  -----
RV01  APL  MVSA   500   0   500   No   0   Rcv   --   --
RV02  APL  MVSA   500   0   500   No   0   Rcv   --   --
RV03  APL  MVSA   700   0   700   Yes  950  Sel   --   --
RV04  APL  MVSA   700   0   700   Yes  950  Sel   --   --

Command ==>
F1=Help   F2=End   F3=Return   F9=Refresh   F10=GO   F11=Reset   F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

5-16

### INGGROUP: Rolling recycle notification

The ROLLING RECYCLE field on the INGGROUP members panel shows the status, date, and time of the last rolling recycle.

## Display group OVERIDES example

```
INGKYGRC          SA z/OS - Command Dialogs      Line 1      of 6
Domain Id . . : AOFDA   ----- INGGROUP ----- Date . . . : 10/28/18
Operator Id : INGC102     Sysplex = SYSPLEX1    Time . . . : 19:27:53

Cmd: E INGVOTE   F INGINFO   G Show members   J INGGROUP

Cmd Name      Type System
-----      -----
RVSERVER      APG   MVSA

Details
-----
Server Group for RV Applications
Nature           : SERVER
Availability adjustment : 1
Satisfactory adjustment : 1
Members with adjusted preferences...
RV02/APL/MVSA      : 500

Command ===> _____
F1=Help   F2=End   F3=Return   F4=INGLIST   F6=Roll
F9=Refresh F12=Retrieve
```

© Copyright IBM Corporation 2019

5-17

### Display group OVERIDES example

Entering the OVERIDES action on the primary INGGROUP panel displays a panel similar to the one in this example. All overrides, which are changes to the base policy for the selected group, are listed in the Details column.

## INGMOVE example

```

INGKYM0          SA z/OS - Command Dialogs      Line 1   of 1
Domain Id . : AOFDB    ----- INGMOVE      Date . . : 10/29/18
Operator Id : INGC102  Sysplex = SYSPLEX1  Time . . : 20:55:09

B Start   C Stop    D INGRELS   E INGVOTE   F INGINFO   G Members   I INGSCHED
J INGGROUP M Move    P Prepare   R Reset    X Box Group / scroll
Cmd Group name Obs Status      ---- Systems ---- Move to
----- MVSA      MVSB

RVXMOVE  AVAILABLE  MVSA  MVSB  MVSB

The application is currently running
on the home_system MVSA
System MVSB is a viable target for the move
because the system is active in the sysplex
Observed status of the group is AVAILABLE
The active member is running on system MVSA

```

The application is currently running on the **home**\_system MVSA

System MVSB is a *viable* target for the move because the system is active in the sysplex

Observed status of the group is AVAILABLE  
The active member is running on system MVSA

Command ==> █

F1=Help F2=End F3=Return F9=Refresh F6=Roll F12=Retrieve

© Copyright IBM Corporation 2019

5-18

### INGMOVE example

This example of the **INGMOVE** command shows group, RVXMOVE with members that are linked on two systems, MVSA and MVSB. The application that is active on MVSA and MVSB is a viable target for a move of the application. The underscore (—) characters under system MVSA indicate that MVSA is the home system for the application. The group member on MVSA has a higher base preference value than the member on MVSB. The system with the highest base preference value is the home system. More than one home system can be defined for a move group. Unless it is a home system, the system with the highest adjusted preference value is shown in reverse video. Row command **M** is used for immediate moves. The target system for the move is specified in the Move to column. For delayed moves, use row command **P**. The systems are color-coded as follows:

- Green: The application is currently running on this system.
- Turquoise: Available systems to which the application can be moved. These group members are considered viable.
- Red: The system is unavailable. A move to the system is not possible.
- Yellow: Indicates that the group member is marked not startable. A move to that system is not possible.
- Pink: The system is excluded as a target for the move.
- White: The system is avoided as a target for the move.

## A summary of INGMOVE commands

Action	Command	Description
M	<b>Move</b>	Perform an immediate move
P	<b>Prepare</b>	Schedule a move for a later time
R	<b>Reset</b>	Reset member preferences
X	<b>Box</b>	Set member preference value to 2800

### A summary of INGMOVE commands

The following row commands are used exclusively for move groups:

- **M:** Perform an immediate move so that the application is started on a system other than the system on which it is currently active.
- **P:** Prepare a move so that the resource is moved the next time that the group is started.
- **R:** Reset the preference values to the initial values as defined in the group automation policy.
- **X:** Box the group. The currently active member is given an adjusted preference value of 2800. It can also receive other bonus values from the automation manager. A value of 2800 means that the resource is locked to the system on which it is currently started. The effect of this action is if the active member terminates, no other group member is activated. The application remains down and the application group becomes unavailable.

Consider boxing a group when initializing the system that hosts the primary group member, if moving to another system or to a home system increases the length of time to IPL.

# Lesson 3 Optional topics



## Lesson 3. Optional topics



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how preference values and bonuses that the automation manager adds are used during member selection and group recovery. Scenarios include sticky move, disruptive versus non disruptive moves.

# Optional topics

- Move group examples
  - Restart (disruptive versus non disruptive)
- INGGROUP scenarios
  - Changing the availability target
  - Adjusting preference values

## *Optional topics*

This lesson is optional. Some of the previous topics are covered in more detail:

- Move group operational options:
  - Disruptive moves
  - Non disruptive moves
- INGGROUP scenarios
  - Changing the availability target
  - Adjusting preference values

## Disruptive versus non disruptive moves

- Move groups ensure that only one instance of an application is available
  - If it fails, a backup is started on another system
- What do you want to happen when the original application is available to be restarted:
  - Move back immediately (can be disruptive)
  - Move after the next scheduled outage (nondisruptive)

### *Disruptive versus non disruptive moves*

When an application fails on one system, it is automatically started on an available system. What happens when the application is restarted? Do you move the application as soon as it can be restarted, or do you wait for a planned outage? Moving immediately might be as disruptive as the original failure and can take several minutes to complete. Waiting for a planned outage might not be as disruptive. Can the backup system handle the workload for a sustained time?

The next few slides show that the choice depends on your selection of base preference values for members of the move group. If the base preference value for the backup member is less than the base preference value of the primary member by 250 or more, the move is disruptive. The base preference value for the primary member is always higher than the effective preference value for the backup member, if you use a base preference value of less than 1000.

## Example of a disruptive move

### Environment

- Move group that is defined with two members
  - One with a preference value of 700 (A)
  - The other with a preference value of 200 (B)

### Scenario

- The group starts
- Member A is activated
- Member A fails
- A MakeAvailable vote is generated for member B  
Effective preference value now  $200 + 250$
- Eventually, member A is fixed and is eligible to be restarted
- A MakeAvailable vote is generated for member A because it has a higher effective preference value
- A MakeUnavailable vote is generated for Member B

Because A was restarted, there is an additional disruption of service to the application

### *Example of a disruptive move*

In this example, a move group with two members is defined. One has a preference value of 700, and the second has a preference value of 200. The first instance is started when the move group is activated. If it fails, then the second member is activated.

When the first member (preference value of 700 originally) can be restarted, it has a higher effective preference value (700 versus  $200+250$ ) and is restarted automatically. In this case, the restart causes another disruption of service.

## Example of a non disruptive move

### Environment

- Move group that is defined with two members
  - (A) with a preference value of 700
  - (B) with a preference value of 500

### Scenario

- The group starts
- Member A is activated
- Member A fails
- MakeAvailable vote is generated for member B (EPV now 500+250)
- Member A is fixed and is eligible to be restarted
- Member B has a higher effective preference value (750); it stays active
- Member A is restarted after the next scheduled outage

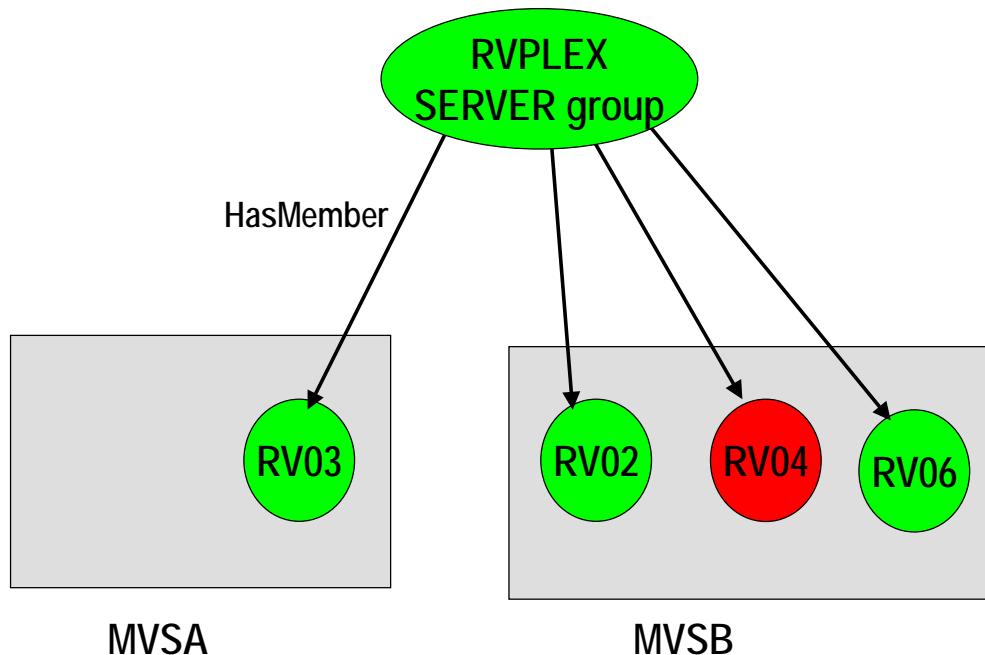
There is no additional disruption of service to the application

#### *Example of a non disruptive move*

In this example, a move group is defined with two members. One has a preference value of 700, and the second has a preference value of 500. The first instance is started when the move group is activated. If it fails, then the second member is activated.

When the first member (defined preference value of 700) can be restarted, it has a lower effective preference value (700 versus 500+250). It is not restarted until the next planned outage. In this case, there is no additional disruption of service to the application users.

## Configuration for INGGROUP examples



© Copyright IBM Corporation 2019

5-25

### Configuration for INGGROUP examples

This example shows a simple sysplex server application group configuration. The scenarios in this section use the configuration for the INGGROUP example. The sysplex application server group, RVPLEX has four members:

- RV03 on system MVSA
- RV02 on system MVSB
- RV04 on system MVSB
- RV06 on system MVSB

The status of the group is dependent on its availability and satisfactory targets. The group is active even though RV04 is down.

The INGGROUP scenarios in this section cover these adjustments:

1. Adjusting the availability target from 2 to 3 to start an additional member
2. Adjusting the preference value for a member to have it started

## Example 1 scenario: Changing the availability target

- Sysplex server group (RVPLEX)
- Four applications (RV02, RV03, RV04, and RV06)
- Two systems (MVSA and MVSB)
- Scenario
  - RV03 is active on MVSA
  - RV02, RV04, and RV06 are active on MVSB
  - Availability target for RVPLEX is 4
  - Change the availability target for RVPLEX to 2
  - Two members are stopped

### *Example 1 scenario: Changing the availability target (1 of 2)*

To change the availability target, type over the existing number in the RESULT field on the INGGROUP member panel. The change is not applied until you press PF10. The resulting actions of MakeAvailable (MA) or MakeUnavailable (MU) are displayed to show the effect of your change.

## Example 1 scenario: Changing the availability target (continued)

```

INGKYGRB          SA z/OS - Command Dialogs      Line 1   of 4
Domain Id . . : AOFDA  ----- INGGROUP ----- Date . . : 10/28/18
Operator Id : INGC102    Sysplex = SYSPLEX1     Time . . : 20:11:33

Group: RVSERVER/APG/MVSA      Nature: Server  Passive: NO  Suspend:
Description: Server Group for RV Applications
Excluded :
Avoided :
Mode : Normal   Availability Target: 2   Adjust: 0   Result=> 2
          Satisfactory Target=> 2   Adjust: 0
Rolling Recycle: Complete 2018-10-28 19:13:38

Name      Type System  Pref  Adj  Result  Avl  Eff  Stat  Act  Sus
-----  -----  -----  -----  -----  -----  -----  -----  -----  -----  -----
RV01      APL  MVSA    500  200  700  Yes  950  Sel  --  --
RV02      APL  MVSA    500  200  700  Yes  950  Sel  --  --
RV03      APL  MVSA    700  0    700  Yes  950  Uns  MU  --
RV04      APL  MVSA    700  0    700  Yes  950  Uns  MU  --

```

Command ==> \_\_\_\_\_

F1=Help F2=End F3=Return F9=Refresh F10=GO F11=Reset F6=Roll F12=Retrieve

The current availability target is 4. Change the **Result** field to **2** and press Enter

© Copyright IBM Corporation 2019

5-27

Example 1 scenario: Changing the availability target (2 of 2)

Pressing PF4 on the primary INGGROUP panel (INGKYGRA) displays the members of the group. The following command also displays this panel:

```
INGGROUP RVPLEX, ACTION=MEMBERS
```

You see the status of the server group, RVPLEX, and its members, RV03 on MVSA, RV02, RV04, and RV06 on MVSB. No systems are excluded or avoided.

The Availability (AVL) column shows that all members are available. The possible values for this column are Yes, No, or Nst (not startable).

- **Yes** in the AVL column indicates that the member can start. At the time of the display, the status of the group can be Available, Starting, or Degraded.
- **Nst** indicates that the automation manager cannot start the member.
- **No** in the AVL column covers all other cases.

The Status (STAT) column shows if a member is selected to start. A value of Sel indicates that the member is selected and is included in the availability target evaluation. Sel also indicates that a MakeAvailable request is submitted for the resource. Uns indicates that the member is not selected and is not considered when the group is evaluated. However, the member is a candidate for selection. Rcv (recovery) indicates that the adjusted preference value for the member is 600 or less

## 5 Application groups

### *Example 1 scenario: Changing the availability target (2 of 2)*

and the group is in Normal mode. Therefore, the resource is not eligible for selection. If the group goes to Recovery mode, resources that were in Rcv status become candidates for selection. Resources that have Uns and Rcv status values also have MakeUnavailable requests issued for them.

The changes that can be made on this panel are based on the type of group. Availability target can be changed for server groups only. To change the availability target, you must change the RESULT field. Preference values of members can be changed for move and server groups.

As changes are made, the adjusted values (RESULTS) are shown. The actions are not taken until the GO command (PF10) is pressed.

Example 1 shows MakeUnavailable requests are generated for RV04 and RV06 on MVSB. The satisfactory target is also 2. The compound status of the RVPLEX server group is SATISFACTORY.

## Example 2 scenario: Adjusting preference values

- Sysplex server group (RVPLEX)
- Four applications (RV02, RV03, RV04, and RV06)
- Two systems (MVSA and MVSB)
- Scenario
  - RV02 is active on MVSB
  - RV03 is active on MVSA
  - Adjust preference value for RV04 on MVSB

### Example 2 scenario: Adjusting preference values

To change the preference values on the INGGROUP member panel, type over the existing number in the RESULT column on the row of the member that you want to change. The adjusted and effective preference values change. This action is valid for server and move groups only. Resulting actions are not immediate.

Pressing PF9 refreshes the panel, and shows the effect of the changes. The ACT column contains the MakeAvailable and MakeUnavailable requests that occur when PF10 is pressed. Changes to the availability target and preference values can be done at the same time.

## Example 2 scenario: Adjusting preference values (continued)

```

INGKYGRB          SA z/OS - Command Dialogs      Line 1   of 4
Domain Id . : AOFDB  ----- INGGROUP ----- Date . . : 10/29/18
Operator Id : INGC102           Sysplex = SYSPLEX1 Time . . : 20:46:05

Group: RVXSERVER/APG          Nature: Server  Passive: NO    Suspend:
Description: RV Sysplex Server Group
Excluded :
Avoided :
Mode     : Normal   Availability Target: 4   Adjust: -2    Result=> 2
           Satisfactory Target=> ████ Adjust: 0

Rolling Recycle: None

Name      Type System  Pref  Adj  Result  Avl  Eff  Stat  Act  Sus
-----  ----  -----  ---  ---  -----  ---  ---  ---  ---  ---
RV02     APL  MVSB    700   0   700   Yes  950  Sel  --  --
RV03     APL  MVSA    700   0   700   Yes  950  Sel  --  --
RV04     APL  MVSB    700   0   700   No   700  Uns  --  --
RV06     APL  MVSB    700   0   700   No   700  Uns  --  --

```

Command ==> F1=Help F2=End F3=Return F9=Refresh F10=GO F11=Reset F6=Roll F12=Retrieve

Four applications in the group two are active

© Copyright IBM Corporation 2019

5-29

Example 2 scenario: Adjusting preference values

In this example, RV02 and RV03 are active and included in the count for the availability target of the group. The automation manager can generate start orders for RV04 and RV06, if they are needed. The STAT column for both members show that they are unselected (Uns).

## Example 2 scenario:

### Adjusting preference values (continued)

INGKYGRB SA z/OS - Command Dialogs Line 1 of 4  
 Domain Id . : AOFDB INNGROUP Date . . : 10/29/18  
 Operator Id : INGC102 Sysplex = SYSPLEX1 Time . . : 20:50:33

Group: RVXSERVER/APG Nature: Server Passive: NO Suspend:  
 Description: RV Sysplex Server Group  
 Excluded :  
 Avoided :  
 Mode : Normal Availability Target: 4 Adjust: -2 Result=> 2  
 Satisfactory Target=> 2 Adjust: 0  
 Rolling Recycle: None

Name	Type	System	Pref	Adj	Result	Avl	Eff	Stat	Act	Sus
RV02	APL	MVSB	700	0	700	Yes	950	Sel	--	--
RV03	APL	MVSA	700	0	700	Yes	950	Uns	MU	--
RV04	APL	MVSB	700	1000	1700	No	1700	Sel	MA	--
RV06	APL	MVSB	700	0	700	No	700	Uns	--	--

RV04 on MVSB has an effective preference value of 1700. Two requests are generated: a MakeAvailable for RV04 and a MakeUnavailable for RV03

Command ==>  
 F1=Help F2=End F3=Return F9=Refresh F10=GO F11=Reset F6=Roll F12=Retrieve

#### Example 2 scenario: Adjusting preference values (continued)

The slide shows a preference value adjustment of +1000 for RV04. The RESULT column contains 1700. Two votes were generated: a MakeAvailable (MA) for RV04 and a MakeUnavailable for RV03.

The availability target stays at 2. The effective preference value for RV04 is 1700. The example does not show any bonus points because PF10 was not pressed.

## Example 2 scenario: Adjusting preference values (continued)

```

INGKYGRB          SA z/OS - Command Dialogs      Line 1   of 4
Domain Id . : AOFDB  ----- INGGROUP ----- Date . . : 10/29/18
Operator Id : INGC102           Sysplex = SYSPLEX1 Time . . : 20:48:18

Group: RVXSERVER/APG          Nature: Server  Passive: NO    Suspend:
Description: RV Sysplex Server Group
Excluded :
Avoided :
Mode     : Normal   Availability Target: 4   Adjust: -2   Result=> 2
           Satisfactory Target=> █   Adjust: 0

Rolling Recycle: None

Name      Type System  Pref  Adj  Result  Avl  Eff  Stat  Act  Sus
-----  -----  -----  ---  ---  -----  ---  ---  ---  ---  ---
RV02     APL  MVSB    700   0   700   No   700  Uns  --   --
RV03     APL  MVSA    700   0   700   Yes  950  Sel  --   --
RV04     APL  MVSB    700  1000 1700  Yes  2100 Sel  --   --
RV06     APL  MVSB    700   0   700   No   700  Uns  --   --

```

The effective preference value for RV04 on MVSB increased to 2100

Command ==> \_\_\_\_\_  
F1=Help F2=End F3=Return F9=Refresh F10=GO F11=Reset F6=Roll F12=Retrieve

Pressing PF10 saved the adjusted preference value, and the actions were completed. RV03 is stopped and RV04 is started. This move is a *targeted move* because an operator requested the move.

The effective preference value is 2100 because the automation manager added a total of 400 bonus points. The automation manager added the base bonus of 225 points, and 175 sticky points because the combined base preference value and adjusted preference value are greater than 1000.

## Student exercise



© Copyright IBM Corporation 2019

5-32

### Student exercises

Open your *Student Exercises* book and perform the exercises for this unit.

# Summary

- Now that you completed this unit, you can perform the following tasks:
  - Describe application moves
  - Use application group management commands across the SAplex
    - Start
    - Stop
    - Display
    - Move
  - Describe application group preference value thresholds and calculation
  - Explain move group examples

## *Summary*

Now that you completed this unit, you can perform the following tasks:

- Describe application moves
- Use application group management commands across the SAplex
  - ◆ Start
  - ◆ Stop
  - ◆ Display
  - ◆ Move
- Describe application group preference value thresholds and calculation
- Explain move group examples



# Unit 6: Application monitoring

---



## Unit 6: Application monitoring



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

The primary objective of this unit is performance monitoring of automation managed resources. The lessons introduce and explain automation monitor resources, exception-based, and event-based monitoring, OMEGAMON XE, and the Tivoli Enterprise Portal.

## Objectives

---

When you complete this unit, you can perform the following tasks:

- Use DISPMTR to manage monitor resources
- Use OMEGAMON XE for event-based monitoring
- Use Tivoli Enterprise Portal workspaces to monitor the status of System Automation for z/OS resources

### *Objectives*

When you complete this unit, you can perform the following tasks:

- Use DISPMTR to manage monitor resources
- Use OMEGAMON XE for event-based monitoring
- Use Tivoli Enterprise Portal workspaces to monitor the status of System Automation for z/OS resources

# Lesson 1 Use DISPMTR to manage monitor resources



## Lesson 1. Use DISPMTR to manage monitor resources



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how use the DISPMTR command to manage monitor resources (MTR).

## Display health status

INGKYST0 SA z/OS - Command Dialogs Line 1 of 1						
Domain Id . . . : AOFDA	-----	INGLIST	-----	Date . . . : 10/28/18		
Operator Id : INGC102	Sysplex = SYSPLEX1			Time . . . : 20:28:32		
A Update	B Start	C Stop	D INGRELs	E INGVOTE	F INGINFO	G Members
H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTR	P INGPAC
R Resume	S Suspend	T INGTWS	U User	X INGWHY	/ scroll	
CMD Name	Type	System	Sus	Compound	Desired	Observed Nature
— AUTONETV	APL	MVSA		DEGRADED	AVAILABLE	AVAILABLE

PF11 to scroll right to see health status

INGKYST0 SA z/OS - Command Dialogs Line 1 of 1						
Domain Id . . . : AOFDA	-----	INGLIST	-----	Date . . . : 10/28/18		
Operator Id : INGC102	Sysplex = SYSPLEX1			Time . . . : 20:30:07		
A Update	B Start	C Stop	D INGRELs	E INGVOTE	F INGINFO	G Members
H DISPTRG	I INGSCHED	J INGGROUP	K INGCICS	L INGIMS	M DISPMTR	P INGPAC
R Resume	S Suspend	T INGTWS	U User	X INGWHY	/ scroll	
CMD Name	Type	System	Automation	Startable	Health	Auto Hold
— AUTONETV	APL	MVSA	IDLE	YES	WARNING	YES NO

Health-status influences compound status

© Copyright IBM Corporation 2019

6-4

### Display health status

The top panel in the example shows DEGRADED for the compound status of resource AUTONETV. The lower panel shows that a monitor resource assigned a health status of WARNING to the application. The health status is WARNING indicates that the application has a possible performance problem. Use row-command **M** (DISPMTR) to display further details.

## DISPMTR command output example

INGKYM00	SA z/OS - Command Dialogs					Line 1	of 4
Domain Id . . : A0FDA	----- DISPMTR -----					Date . . . :	10/28/18
Operator Id : INGC102	Sysplex = SYSPLEX1					Time . . . :	20:31:28
A Reset      B Start      C Stop      D Details      E INGVOTE      F INGINFO      I INGSCHED							
R Resume      S Suspend      / scroll							
CMD Monitor	System	Sus	Status	Health	Last monitored		
---	---	---	---	---	---	---	---
— DEMO_1MON	MVSA		INACTIVE	UNKNOWN	2018-10-28 18:53:08		
— JES2MON	MVSA		ACTIVE	NORMAL	2018-10-27 14:52:01		
— JES2SPOOL	MVSA		ACTIVE	NORMAL	2018-10-28 20:27:08		
— MTRNETV	MVSA		ACTIVE	WARNING	2018-10-28 20:28:10		

Command ==> \_\_\_\_\_

F1=Help      F2=End      F3=Return      F9=Refresh      F10=Previous      F11=Next      F6=Roll      F12=Retrieve

The health status of MTRNETV is still WARNING  
Press PF11 to scroll right and see the last message for the resource

© Copyright IBM Corporation 2019

6-5

### DISPMTR command output example

This slide shows an example of the DISPMTR panel for the MTRNETV monitor. MTRNETV is running; its status is ACTIVE. The LAST MONITORED column shows the time at which the health status was assigned. Use PF11 to scroll right to see messages in the STATUS MESSAGE column. You use the INGREQ command to start and stop the monitor.

If the monitor is not active, you can use the INGVOTE command to check for stop votes against it. The monitor might not be active because of a MakeUnavailable vote that is generated for a service period.

You can use row-command **F** to see what the automation manager knows about the resource. For example, the monitor might not be active because the resource that it is monitoring is down. The automation manager information shows the relationships between a monitor and monitored resources. You define relationships between applications and monitors in the automation policy. Monitor relationship definitions are optional.

## DISPMTR command output example (continued)

```

INCKYMO0          SA z/OS - Command Dialogs      Line 1      of 4
Domain Id . : AOFDA  ----- DISPMTR ----- Date . . : 10/28/18
Operator Id : INGC102   Sysplex = SYSPLEX1 Time . . : 20:33:30

A Reset    B Start    C Stop     D Details   E INGVOTE  F INGINFO  I INGSCHED
R Resume   S Suspend  / scroll
CMD Monitor   System   Status message
-----
— DEMO_1MON  MVSA
— JES2MON   MVSA
— JES2SPOOL  MVSA
— MTRNETV   MVSA
                               $HASP9302 JES2 CHECKPOINT LOCK RELEASED
                               2.0883% SPOOL UTILIZATION
                               MTRNETV: NETVIEW TOTL 254 118.68 100.00 0.00 0 ...
                               ...
The first 48 characters of any message are shown in the Status message column.

Select option D to see more details about the monitor resource and message history.

Command ===> _____
F1=Help   F2=End   F3=Return   F9=Refresh   F10=Previous   F11=Next   F6=Roll
                                         F12=Retrieve

```

The first 48 characters of any message are shown in the Status message column.

Select option D to see more details about the monitor resource and message history.

© Copyright IBM Corporation 2019

6-6

This example of the DISPMTR panel shows the most recent message for the MTRNETV monitor. The STATUS MESSAGE column shows only the first 48 characters of any message. Three dots at the end of the message line indicate that the message is truncated. To see the complete message, use option D (Details) to display the message and more information.

## DISPMTR command output example (continued)

```

INGKYM01          SA z/OS - Command Dialogs      Line 1      of 123
Domain Id . : AOFDA   ----- DISPMTR -----  Date . . : 10/28/18
Operator Id : INGC100  Sysplex = SYSPLEX1    Time . . : 20:36:07

Monitor          : MTRNETV/MTR/MVSA
System           : MVSA
Description       : MTR to monitor NetView utilization

Monitored Object : MTRNETV

Inform List       : SDF

Commands...
  Activate        :
  Deactivate      :
  Monitoring      : MTRNETV

Interval         : 00:10

Last termination : 21:00:09 on 08/30/18
Last start       : 16:40:45 on 10/21/18

Monitor Status   : ACTIVE at 2018-10-28 20:28:10
Health Status     : WARNING
                    MTRNETV: NETVIEW TOTL 254 118.68 100.00 0.00 0 40560 N/A

Monitoring History (maximum is 50) ...
  2018-10-28 12:14:48 - ACTIVE      HEALTH=WARNING

Command ===> █
F1=Help   F2=End   F3=Return   F6=Roll
          F8=Forward F9=Refresh  F12=Retrieve

```




© Copyright IBM Corporation 2019

6-7

DISPMTR details show more detailed information about the monitor:

- Monitor resource information
- The monitored object name defaults to the monitor resource name
- Activate command, if defined
- Deactivate command, if defined
- The monitoring routine (MTRNETV)

All messages from the monitor routine are shown under the History section. The oldest message is displayed at the top.



## Lesson 2 Using OMEGAMON XE for event-based monitoring



### Lesson 2. Using OMEGAMON XE for event-based monitoring



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson covers the functions in automation for event-based monitoring that uses the OMEGAMON XE monitoring infrastructure.

# Using OMEGAMON XE for event-based monitoring

## Using OMEGAMON XE for event-based monitoring

- Attributes
- Situations
- Situation events

© Copyright IBM Corporation 2019

6-9

The following OMEGAMON XE topics are described

- Attributes:  
Attributes define the data that a Tivoli Enterprise Monitoring Agent collects.
- Situations:  
Situations test the value of one or more attributes.
- Situation events:  
a situation event can be generated when predefined conditions are detected

## Event-based monitoring

- Uses OMEGAMON XE, which is a passive monitor
  - Monitors raised situation events
  - No monitor routine is needed
  - Uses the Tivoli Enterprise Portal
- Also known as reflex automation
  - Triggered by message ING150I
  - Automated actions are run on the managed system where the System Automation for z/OS agent is running
- OMEGAMON XE also uses a SOAP server connection to a monitoring server
  - XML-based interface for commands and responses
  - Provides access to more monitoring data

### Event-based monitoring

Passive monitoring uses monitor resources that are driven when situation events are generated from OMEGAMON XE monitors. This type of monitoring is called *event-based monitoring*. Within the IBM Tivoli Monitoring architecture, reaction to this type of monitoring is called *reflex automation*. Event-based monitoring involves the automation of message ING150I. *Situations* are defined with the *Situation Editor* function of the Tivoli Enterprise Portal.

## Attributes, situations, and situation events

- **Attributes** are characteristics of objects monitored by a monitoring agent. Attributes are queried to collect data for workspaces and situations
  - Related attributes can be organized into attribute groups
- **Situations** test one or more conditions (values of attributes) and can trigger an event when conditions are met
- **Situation events** indicate a problem or potential problem
  - Critical 
  - Warning 
  - Information 

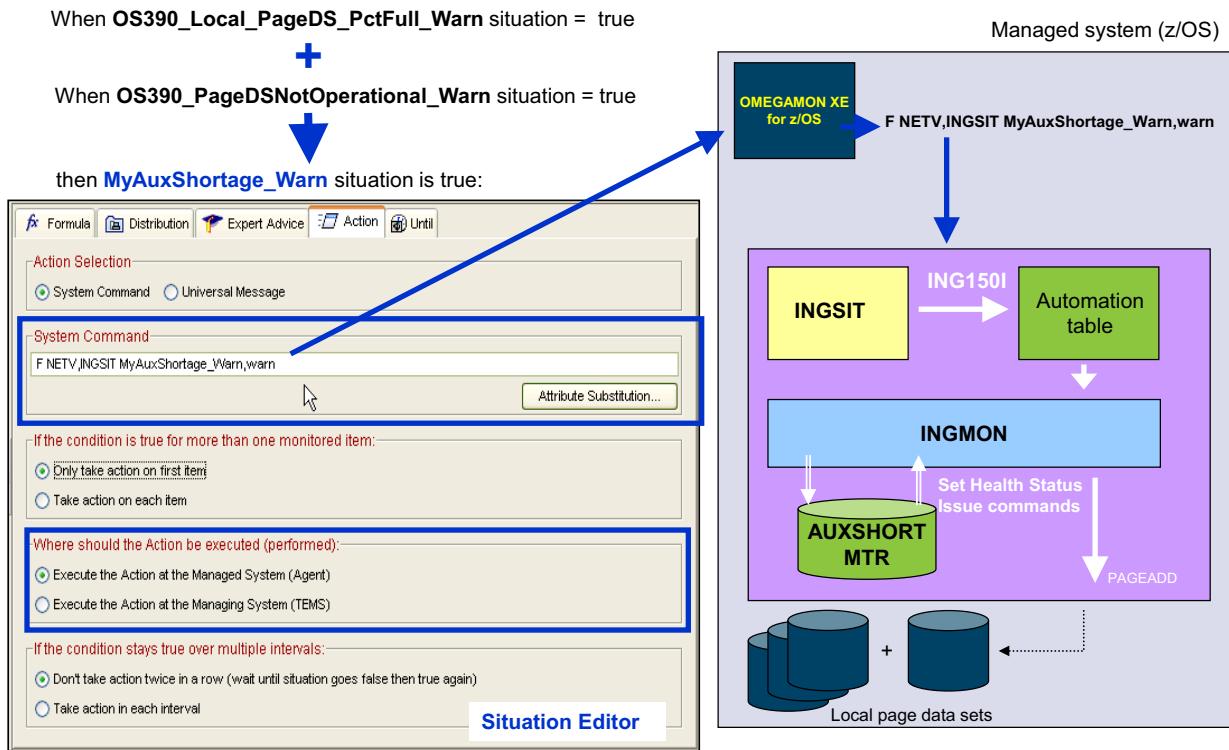
The System Automation for z/OS monitoring agent collects SAplex-wide information from the primary automation manager and system-specific information from the automation agents (local and remote)

### Attributes, situations, and situation events

*Attributes* define the data that a Tivoli Enterprise Monitoring Agent collects. *Situations* test the value of one or more attributes. During the test, a *situation event* can be generated when predefined conditions are detected; for example, when the observed status of a resource is HARDDOWN.

Automation provides several pre-defined situations. For example, KAH\_RSRC\_NOT\_SATISFACTORY\_CRIT generates a situation event if a resource compound status is PROBLEM.

## Event-based automation example



© Copyright IBM Corporation 2019

6-12

### Event-based automation example

The example on this slide shows a user-defined situation named *MyAuxShortage\_Warn*. The situation tests two OMEGAMON XE for z/OS situations. If the conditions are true, Automation routines issue a PAGEADD command to add another page data set to the z/OS system.

Use the following tabs in the Situation Editor:

- The **Formula** tab, if selected, shows the OS390\_Local\_PageDS\_Pct\_Full\_Warn situation plus the OS390\_PageDSNotOperational\_Warn. The situation evaluates as True. These two situations are defined in OMEGAMON XE for z/OS.
- The **Action** tab shows the Automation **INGSIT** command that is driven when the *MyAuxShortage\_Warn* situation generates a situation event. Automation issues the command the first time that a situation event occurs at the managed system.

The INGSIT command generates an ING150I message. The ING150I message drives the automation table and runs the Automation INGMON command to set the health status of the AUXSHORT monitor resource. Commands that are defined for the monitor resource trigger a PAGEADD command.

# Lesson 3 Using Tivoli Enterprise Portal



## Lesson 3. Using Tivoli Enterprise Portal



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn how to use the Tivoli Enterprise Portal for displaying automation resources and status.



**Note:** The announcement of IBM Service Management Suite for z/OS, V1.5.0 contains a statement of general direction that IBM intends to discontinue System Automation for z/OS support for collecting automation data using the Monitoring Agent and feeding that data into the Tivoli Enterprise Portal (TEP).

# Using Tivoli Enterprise Portal

- Use Tivoli Enterprise Portal for event-based monitoring
  - Monitoring agent
  - Workspaces
  - Situation events
  - Take Action commands

## *Using Tivoli Enterprise Portal*

This lesson introduces Tivoli Enterprise Portal workspaces and explains how to use them to monitor Automation resources. Automation generates situation events to inform of possible problems. You can use the Take Action function of the Tivoli Enterprise Portal to react to situation events.

## Monitoring agent overview

- Use the System Automation for z/OS monitoring agent for these purposes:
  - Monitor and collect data from the SAplex
  - Display the following information in workspaces:
    - Resource overview and detail
    - Resource topology
    - Resource request information
    - Automation environment information
    - Automation statistics
    - System and application health information
    - User-defined status items

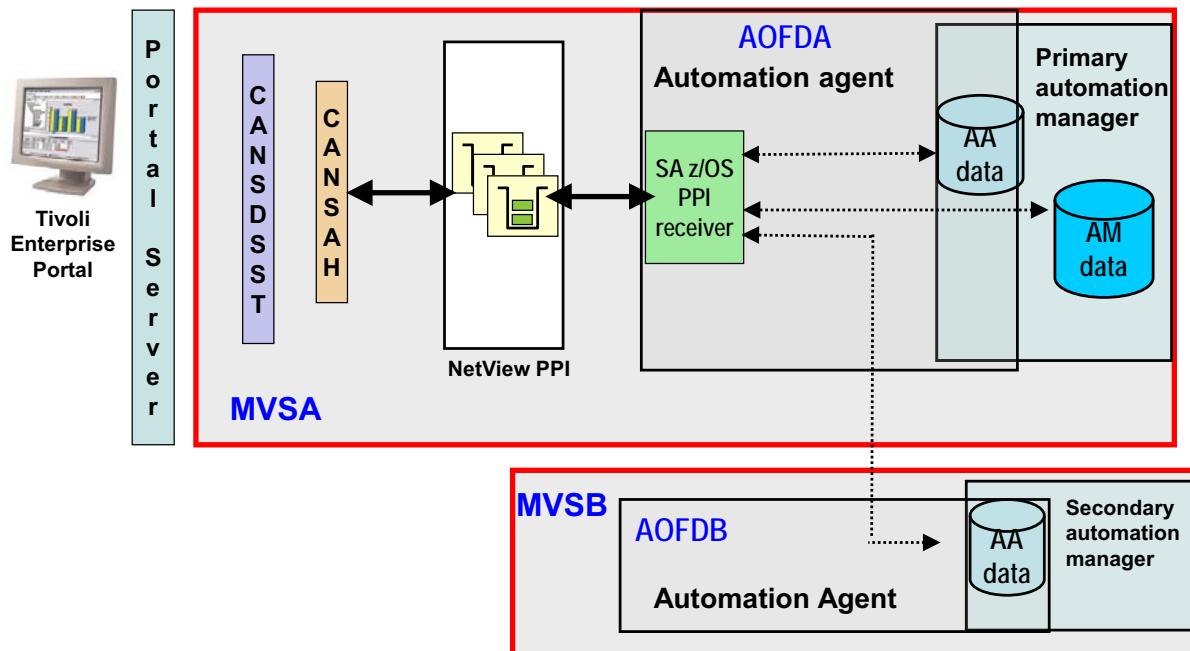
### *Monitoring agent overview*

The Automation monitoring agent is a Tivoli Enterprise Monitoring Agent.

Using the Tivoli Enterprise Portal workspaces, users can perform these tasks:

- Monitor the system resources and automation environment. Integration on the Tivoli Enterprise Portal means integration across different types of resources.
- View situations and situation events
  - Predefined by Automation
  - Customizable
- View expert advice
- Use defined Take Action commands to initiate corrective action.

# System Automation for z/OS monitoring agent with z/OS monitoring server



**CANSAH:** Default monitoring agent address space

**CANSDSST:** Default monitoring server address space

© Copyright IBM Corporation 2019

6-16

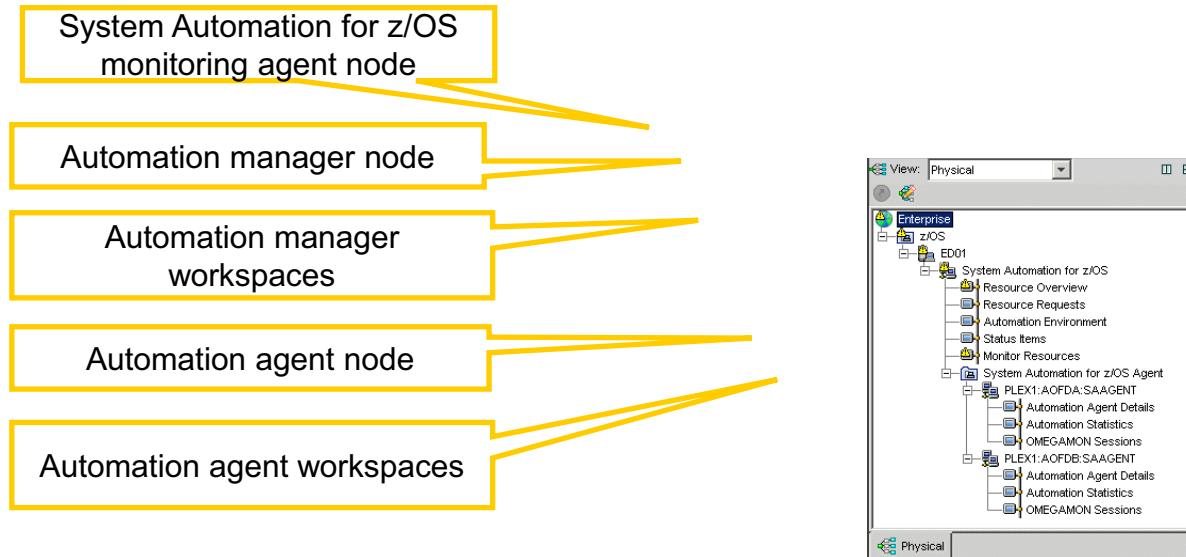
SA z/OS Tivoli Enterprise Monitoring Agent with z/OS monitoring server

This example shows a monitoring environment in which a SA z/OS Tivoli Enterprise Monitoring Agent and a z/OS Tivoli Enterprise Monitoring Server are installed. The SA z/OS Tivoli Enterprise Monitoring Agent and the automation agent run on the same z/OS system. A Tivoli Enterprise Monitoring Agent connects to a Tivoli Enterprise Monitoring Server.

Tivoli Enterprise Monitoring Servers can run on z/OS or distributed systems. Multiple Tivoli Enterprise Monitoring Servers can be set up in a radial configuration. One server is configured as a hub, and the other servers as remote servers. The remote servers connect to the hub. Tivoli Enterprise Monitoring Servers connect to Tivoli Enterprise Portal Servers that run on distributed systems. You can use a web browser or desktop client to connect to the Tivoli Enterprise Portal, which connects to the Tivoli Enterprise Portal Server.

In this example, an OMEGAMON Run Time Environment (RTE) is configured with the SA z/OS Tivoli Enterprise Monitoring Agent (CANSAH) in one address space. The z/OS Tivoli Enterprise Monitoring Server (CANSDSST) in a different address space. Both are configured to run on system MVSA. Automation agents from MVSA and MVSB, and the primary automation manager send data to the SA z/OS Tivoli Enterprise Monitoring Agent. They use the Tivoli NetView for z/OS program to program interface (PPI) buffers to send information to the monitoring agent. The SA z/OS Tivoli Enterprise Monitoring Agent then forwards the information to the Tivoli Enterprise Monitoring Server. The SA z/OS Tivoli Enterprise Monitoring Agent, CANSAH, runs on one system only.

# System Automation for z/OS Physical navigator example



## Automation Physical navigator example

This slide shows the physical navigator view with the Automation workspaces. A workspace is the work area of the Tivoli Enterprise Portal application window and is made up of one or more views. A view is a pane in the workspace (typically a chart, graph, or table) showing data that a monitoring agent collects.

The physical navigator shows a hierarchical view of the enterprise with all monitored resources. In the physical navigator view, you see the monitored resources as nodes (objects) in the left window pane. Each node has one or more workspaces. When you click an item in the Navigator, its default

*workspace* displays in the application window - *the right pane*. In the application window, you see groupings of information that the monitoring agent collected.

- Each primary automation manager node has a set of workspaces that can include the following information:
  - Resource Overview
  - Resource Requests
  - Resource Topology
  - Automation Environment
  - Status Items
  - Monitor Resources
  - Outstanding WTOR replies
- Each automation agent node has a set of workspaces that can include the following information:
  - Automation Agent Details
  - Automation Statistics
  - OMEGAMON Sessions
  - Gateway Connection Overview
  - Critical Messages Display
  - Message Events

This example shows two automation agents, one in domain AOFDA and the other in AOFDB.

## The Resource Topology workspace

- A predefined workspace which contains the following views
  - Resource List
    - Lists all resources in the Resource Overview workspace
  - Resource Topology
    - Shows a graphical view of dependencies for the resource in focus
      - Start dependencies
      - Stop dependencies
      - Group dependencies
      - All dependencies

### *The Resource Topology workspace*

The Resource Overview workspace shows information equivalent to the INGLIST command. The Resource Topology workspace displays dependencies between the resource in focus and other resources.

Four modes are available:

- Start Dependencies: Displays all resources that need to be up so the resource in focus can be started.
- Stop Dependencies: Displays all resources that need to be down so the resource in focus can be stopped.
- Group Dependencies: Displays all groups to which the resource in focus belongs. If this resource is a group, all of its members and submembers are also shown.
- All Dependencies: Displays all resources that have direct dependencies with the resource in focus.

## Resource Details workspace example

The screenshot shows the Tivoli Enterprise Portal interface with three main windows:

- Resource Votes for RV01/APL/TIVED1:** This window displays resource votes. A yellow circle highlights the title bar, and a yellow box contains the status message: "Status (P): Vote propagated from RV\_MOVE APG."
- Manager Information about RV01/APL/TIVED1:** This window shows detailed manager information. A yellow circle highlights the title bar.
- Agent Information about RV01/APL/TIVED1:** This window shows detailed agent information. A yellow circle highlights the title bar.

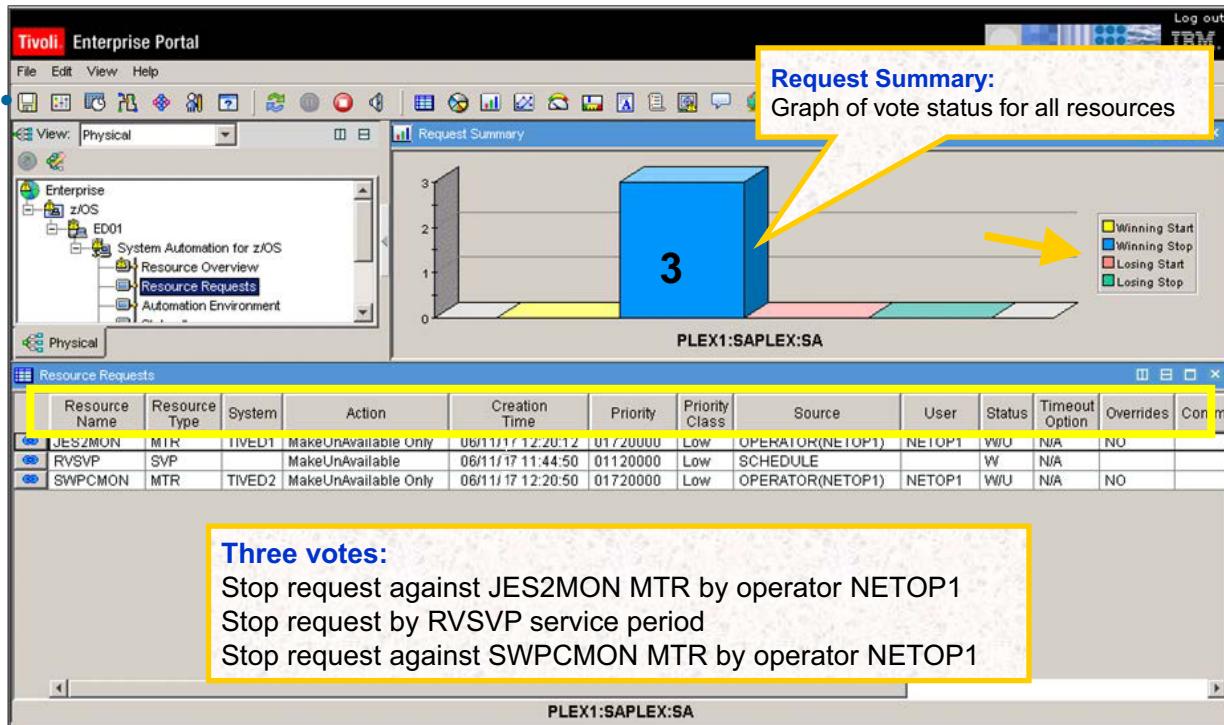
Resource Details workspace accessible from links on Resource Overview and Resource Requests

### Resource Details workspace example

The Resource Details workspace is accessible only from a link. It is not available in the navigator physical view. The workspace has these tables:

- **Resource Votes:** Shows all requests for the resource, including votes that are propagated to the resource from dependencies. This information is similar to the output from an INGVOTE command.
- **Manager Information:** Shows detailed resource information from the automation manager. This information is similar to the output from an INGINFO command.
- **Agent Information:** Shows detailed resource information from the automation agent. This information is similar to the output from the DISPINFO and DISPMTR commands.

# Resource Requests workspace



© Copyright IBM Corporation 2019

6-20

## Resource Requests workspace

The Resource Requests workspace shows data that equals the output of the INGVOTE command. The workspace has these charts:

- **Request Summary** bar chart with requests by type: Start, stop, winning, and losing
- **Resource Requests** table with current requests:
  - Resource name
  - Resource type
  - System
  - Request type (for example, MakeUnavailable)
  - Source of request
  - Numeric priority value
  - Status (for example, winning/satisfied)

A link is provided to start the Resource Details workspace for a selected resource.

# Automation Environment workspace

**Automation manager details:**  
Equal to **INGAMS DETAILS**  
command against the primary  
automation manager

**Two System Automation for z/OS  
automation agents:**

- One primary automation manager
- One secondary automation manager

© Copyright IBM Corporation 2019

6-21

## Automation Environment workspace

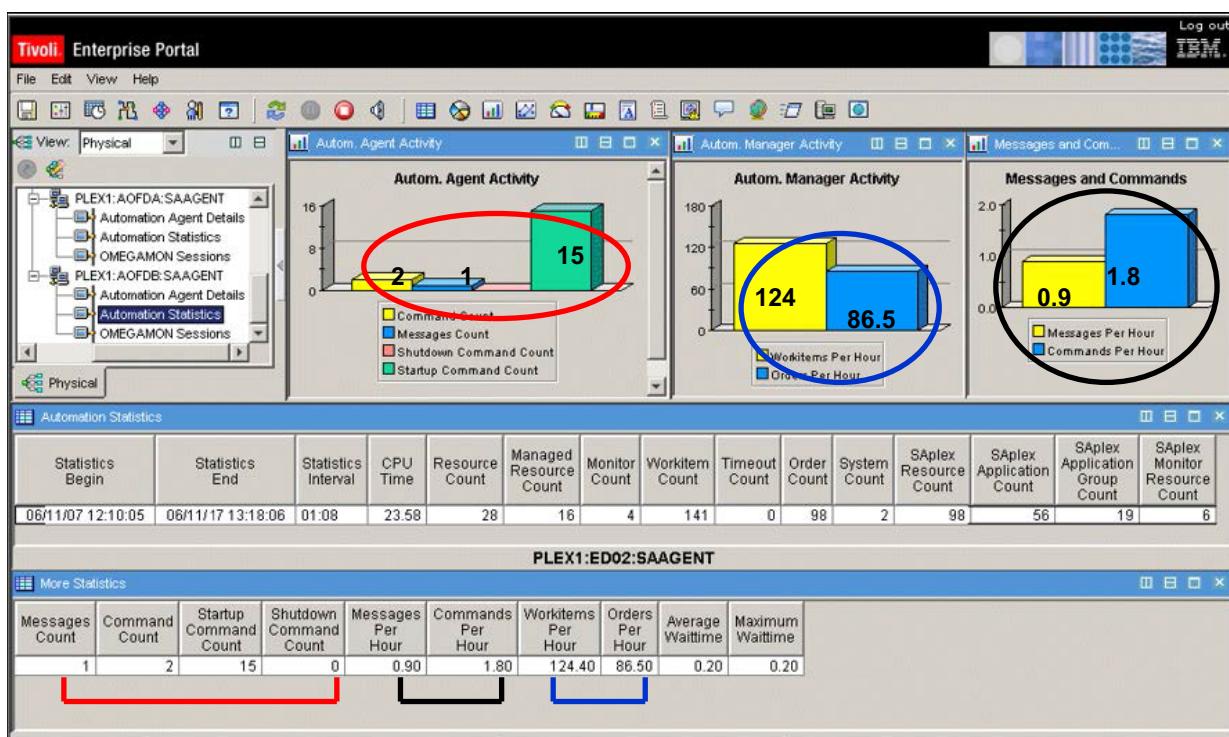
The Automation Environment workspace has these two tables:

- Automation Environment Members: Shows a list of active automation managers and automation agents, similar to output from the INGAMS command.
- Automation Manager Details: Shows detailed information for the selected manager; for example, takeover file, configuration file details. The information in this display is like output of the INGAMS DETAILS command for an automation manager.

Links are provided to LAUNCH the following workspaces:

- Automation Agent Details workspace for a selected agent
- OMEGAMON Sessions workspace
- Automation Statistics workspace

## Automation Statistics workspace



© Copyright IBM Corporation 2019

6-22

### Automation Statistics workspace

The Automation Statistics workspace shows information similar to the INGRPT command.

- Automation Agent Activity bar chart:** Shows the number of commands, messages, and startup and shutdown commands since the last reset.
- Automation Manager Activity bar chart:** Shows the hourly rate of work items and orders since the last reset.
- Messages and Commands bar chart:** Shows the hourly rate of commands and messages since the last reset.
- Automation Statistics table:** Displays detailed statistics for manager and agent.

Automation agent statistics:

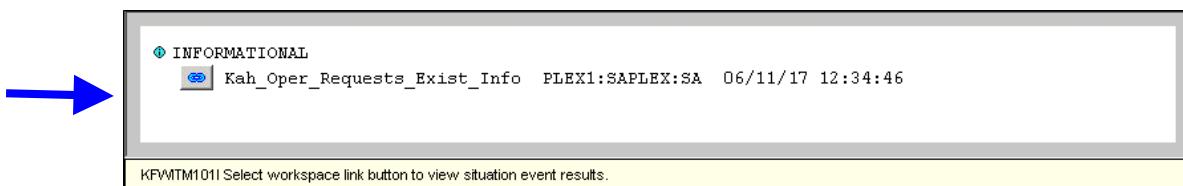
- CPU time that is used (seconds)
- Total number of resources that are defined and managed
- Total number of monitors defined
- Total number of messages automated
- Average number of messages and commands per hour
- Total number of resulting commands
- Total number of Startup and Shutdown commands issued
- Total number of work items sent
- Average number of orders per hour

Automation manager statistics:

- Number of systems in SAplex
- Total number of resources: Applications, application groups, and monitor resources

## Situations and situation events

- Example: The Kah\_Oper\_Requests\_Exist\_Info situation generates an event if an operator takes an action against a resource such as INGREQ STOP
- When true, an informational event is raised on Tivoli Enterprise Portal
  - Can be more than one operator request



- Click link to display all operator requests

© Copyright IBM Corporation 2019

6-23

### Situations and situation events

The Kah\_Oper\_Requests\_Exist\_Info situation monitors for outstanding operator requests such as INGREQ STOP. When one or more operator requests are detected (situation evaluates as True), a situation event is generated.

*Situation events* indicate a problem or potential problem. They can be classified as critical, warning, or informational. Situations generate situation events.

*Situations* test one or more conditions (values of attributes) and can trigger an event when conditions are met. *Attributes* are characteristics of objects that a monitoring agent monitors. Attributes are queried to collect data for workspaces and situations. Related attributes can be organized into attribute groups.

The Automation monitoring agent collects SAplex-wide information from the primary automation manager and system-specific information from the automation agents (local and remote). The information is stored in attributes. Automation provides situations to monitor the values of key attributes.

# Viewing operator requests

The screenshot shows the Tivoli Enterprise Portal interface. In the left pane, under 'z/OS' and 'ED01', 'System Automation for z/OS' is expanded, with 'Resource Requests' selected. A yellow box highlights this selection. In the center, the 'Initial Situation Values' table shows two entries:

Operator Request	Managed System	Resource Name	Resource Type	System	Action	Creation Time	S
1	PLEX1:SAPLEX:SA	SWPCM0N	MTR	TIVED2	MakeUnavailable Only	06/11/17 12:20:50	OPERATC
1	PLEX1:SAPLEX:SA	JES2MON	MTR	TIVED1	MakeUnavailable Only	06/11/17 12:20:12	OPERATC

A yellow box with the text 'An operator issues a stop request against two monitored resources, SWPCM0N and JES2MON' is positioned above the table. In the bottom right, a 'Command View' window titled 'Take Action' contains a 'Action' section with 'Name: <Select Action>' and 'Command:'. A yellow box with the text 'You can define the INGREQ REQ=CANCEL as a Take Action command, and then issue it here' is positioned below this. To the right, a 'Expert Advice' window titled 'Kah\_Oper\_Requests\_Exist\_Info' provides 'Suggested Actions' and 'Suggested Commands'. A yellow box with the text 'Expert advice suggests removing the request with an INGREQ resource REQ=CANCEL command' is positioned above the 'Expert Advice' window.

© Copyright IBM Corporation 2019

6-24

## Viewing operator requests

This example shows the Kah\_Oper\_Requests\_Exist\_Info situation information. The Initial SITUATION VALUES table shows that an operator issued the INGREQ STOP command against two resources. The Expert Advice describes how to remove the requests with the INGREQ CANCEL command. You can issue The INGREQ CANCEL command as a Take Action command.

The OPERATOR REQUEST column shows the attribute that generated the situation event. The operator request attribute is binary:

- 0 if there are no operator requests
- 1 if one or more operator requests exist

## Take Action commands

- Optionally, define Take Action commands
- Examples of Take Action commands:
  - INGSESS START or STOP
  - INGREQ START or STOP
  - INGREQ CANCEL to cancel a vote
  - OMEGAMON commands
  - Commands for situations (event-based automation)
  - REXX EXECs
- Command responses are logged in DSLOG

### *Take Action commands*

Automation does not provide any Take Action commands. For example, if you want to issue an INGREQ CANCEL to cancel an operator request, then you must create the Take Action command. The command response is sent to the NetView log.

The default is to route MVS commands. If you want to issue NetView commands, you can route the command to NetView with the MVS MODIFY command; for example:

```
F AUTON,INGREQ resource_name REQ=CANCEL
```

A BNH806I message is recorded in the NetView log for each command sent. The command responses are not displayed at the Tivoli Enterprise Portal. You must browse the NetView log for the response.

## Student exercise



© Copyright IBM Corporation 2019

6-26

*Student exercises*

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

Now that you completed this unit, you can perform the following tasks:

- Use DISPMTR to manage monitor resources
- Use OMEGAMON XE for event-based monitoring
- Use Tivoli Enterprise Portal workspaces to monitor the status of System Automation for z/OS resources

### Summary

Now that you completed this unit, you can perform the following tasks:

- Describe a monitor resource and its effect on the health status of linked resources
- Describe active and passive monitoring
- Use DISPMTR to manage monitor resources
- Use OMEGAMON XE for event-based monitoring
- Use Tivoli Enterprise Portal workspaces to monitor the status of System Automation for z/OS resources

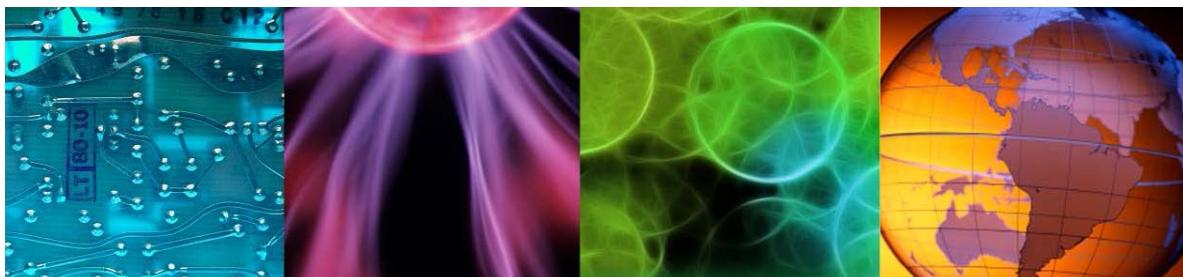




# 7 Centralized operations



## Unit 7: Centralized operations



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit reinforces topics that are covered in previous lessons in the course. You learn to use Automation to manage resources in a multisystem environment. The lessons in this unit include topics on managing gateway sessions, and managing enterprise automation from a *focal point*. You learn to use the Status Display Facility (SDF), Single Point of Control (SPOC) techniques, and the TARGET parameter to route commands.

## Objectives

---

When you complete this unit, you can perform the following tasks:

- Use System Automation for z/OS to manage resources in a multisystem environment
- Describe the purpose of the gateway connections
- Use Status Display Facility (SDF) at a focal point system
- Use System Automation for z/OS as a single point of control (SPOC)

### *Objectives*

When you complete this unit, you can perform the following tasks:

- Use System Automation for z/OS to manage resources in a multisystem environment
- Describe the purpose of the gateway connections
- Use Status Display Facility (SDF) at a focal point system
- Use System Automation for z/OS as a single point of control (SPOC)

# Lesson 1 Gateways



## Lesson 1. Gateways



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson about gateway sessions, what they are used for and how to use the dispgw command to display information about gateway sessions for a domain.

# Gateways

---

- Overview of gateway sessions
- Understanding the structure of gateway sessions
- Verifying gateway sessions

## *Gateways*

- Overview of gateway sessions
- Understanding the structure of gateway sessions
- Verifying gateway sessions

## An overview of gateways

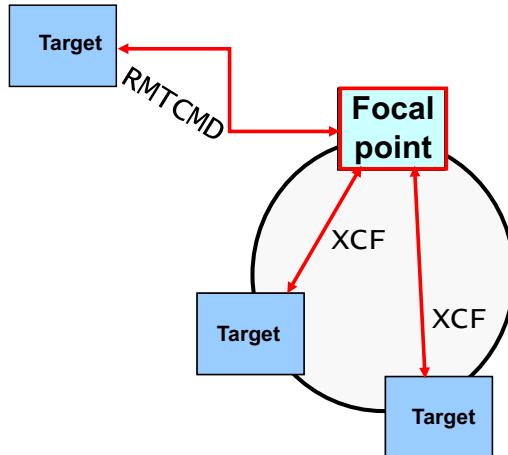
- Definition of a gateway
  - A link between two Tivoli NetView for z/OS domains that run System Automation for z/OS automation agents
  - Communication occurs through inbound and outbound sessions
  - Sessions start when System Automation for z/OS initializes
  - Are typically kept active
- Purpose of a gateway
  - Base for Status Display Facility focal point operation
  - Base for non-sysplex SPOC
    - Used to determine active systems
    - Enables commands and messages to flow

### An overview of gateways

A *gateway link*, more commonly called a *gateway*, is a link between two NetView domains. The link consists of an inbound session and an outbound session between the two domains. A gateway link enables commands, status, and messages to flow between the domains. Such a link forms the basis for the Status Display Facility (SDF) *focal point*. Gateways can be set up between domains even when neither domain is a focal point.

Automation uses z/OS cross-system coupling facility (XCF) for communication between agents within the same SAplex. Gateway sessions between agents in different SAplexes use the NetView RMTCMD function. RMTCMD is a NetView function that uses TCP/IP or VTAM/SNA for communication between domains. Examples are shown later in this unit.

# Gateway communication

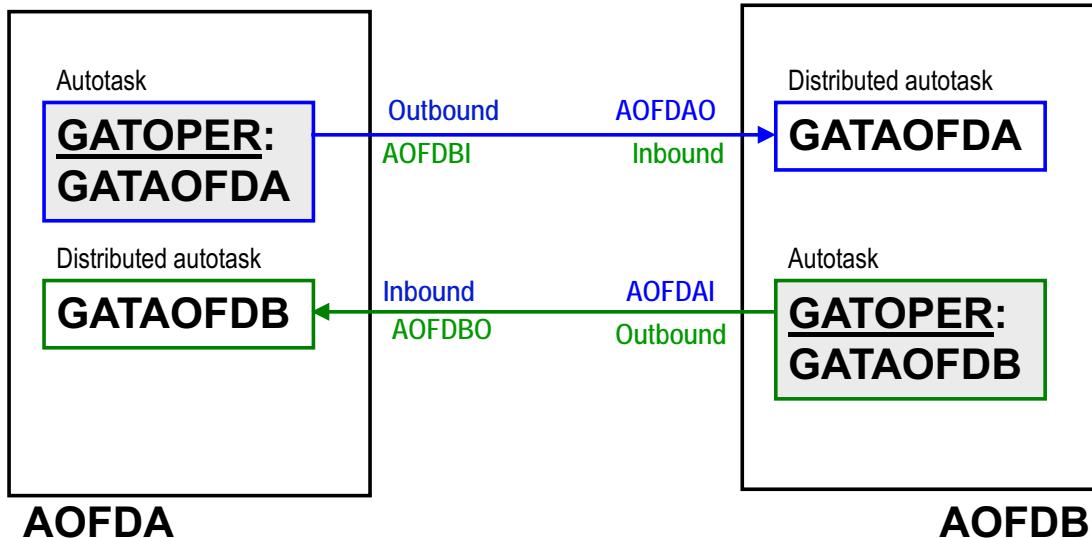


- Communication between systems
  - XCF within the sysplex
  - RMTCMD (SNA, IP) cross-sysplex
- Instant detection when communication is lost

## Gateway communication

Gateway links connect Automation domains. Gateway sessions use XCF within SAplexes and NetView RMTCMD (SNA or TCP/IP) for communication between different SAplexes. Automation provides automatic detection and recovery of gateway sessions.

# Gateway sessions



**Outbound task:** Supports outbound communication to all other systems

**Inbound task:** One for each remote system, established when remote system connects

## Gateway sessions

This slide shows a simple configuration of two domains, AOFDA and AOFDB. When Automation initializes in AOFDA, the GATOPER autotask GATAOFDA starts. After GATAOFDA logs on, it initiates outbound sessions to each active, defined domain and logs on to each domain.

By default, the name of the GATOPER autotask is the combination of the letters GAT concatenated to the NetView domain name. For example, if the domain ID is AOFDA, then GATAOFDA is the default autotask name. The default autotask name can be changed in the customization dialogs. Internal commands and messages are sent to the GATOPER, which forwards them to other domains over the outbound session.

The outbound session from AOFDA to AOFDB is displayed on Status Display Facility panels on MVSA as **AOFDBO** (the suffix O, marks the session as outbound to AOFDB). The same outbound session is displayed on Status Display Facility panels on MVSB as **AOFDAI** (the suffix I, marks the session as inbound from AOFDA).

Similar processing occurs in AOFDB for its GATOPER, GATAOFDB, to establish the inbound session to AOFDA and complete the gateway link between the two domains. The outbound session from AOFDB is the inbound session to AOFDA.

The GATOPER autotasks must be defined in member AOFOFGW and SAF, if SAF is used for autotasks.

## Displaying gateway connections

```
AOFK2GL                               SA z/OS - Command Dialogs      Line 1      of 1
Domain Id . : AOFDB                   DISPGW                         Date . . : 10/31/18
Operator Id : INGC102                 System = MVSB                  Time . . : 10:42:41
                                         Focal Point = AOFDA

Domain   SDF Root Status    Comm   Release level In Status Out Status
-----+-----+-----+-----+-----+-----+-----+-----+-----+
AOFDA   MVSA     ACTIVE      *XCF*  SA z/OS 4.1.0 ACTIVE   ACTIVE

Domain   In Req   Out Req   System   SMF Id   SApex          XCF Group
-----+-----+-----+-----+-----+-----+-----+-----+
AOFDA   CONNECT  CONNECT  MVSA     MVSA     SYSPLEX1        INGXSGSA
Domain   Net Id   SYSPLEX  PrimaryFP BackupFP
-----+-----+-----+-----+-----+-----+-----+
AOFDA   USIBMES PLEX12   AOFDA   AOFDB

Domain   Description
-----+-----+
AOFDA   Connection from target to SDF backup FP

AOFDB gateway connection to AOFDA is XCF
```

© Copyright IBM Corporation 2019

7-8

### Displaying gateway connections

The DISPGW command displays information about gateway sessions for a domain. The information includes method of communication, the status of inbound and outbound connections, and the level of Automation running in the target domain.

The second column in the first example shows the communication method between each connected domain. The values that you can see in this column are:

- \*XCF\*: Transport mechanism is XCF for systems in the same sysplex only
- \*SNA\*: Transport mechanism is NetView RMTCMD over SNA
- \*IP\*: Transport mechanism is NetView RMTCMD over TCP/IP
- \*RPC\*: Transport mechanism is the remote procedure call technique

In this example, AOFDB has an active XCF gateway connection to AOFDA. The following NetView command can be used to check the status of gateway sessions outside of a sysplex:

```
rmtcmd query rmtdoms
```

The examples below show some additional information that is displayed by pressing PF11 (Next).

# Lesson 2 Status Display Facility focal point



## Lesson 2 Status Display Facility focal point

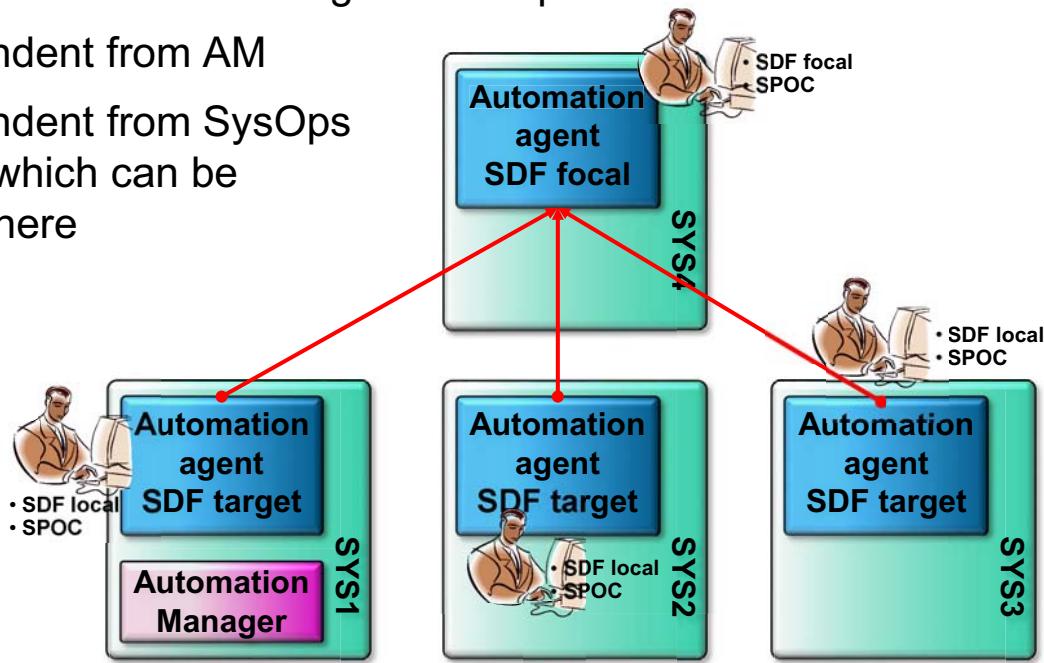


Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn about the Status Display Facility (SDF), the SDF focal point, Enterprise monitoring with SDF, and panel examples.

## Status Display Facility focal point and Single Point of Control (SPOC)

- SDF is an agent function and needs to be implemented on all systems with one being the focal point
- Independent from AM
- Independent from SysOps SPOC which can be everywhere



© Copyright IBM Corporation 2019

7-10

### Status Display Facility focal point

This slide shows a simple System Automation for z/OS implementation across four systems. SYS4 is the *SDF focal point*. SYS1, SYS2, and SYS3 are the *SDF target systems*.

Each system is running an automation agent and Status Display Facility (SDF). If an agent in a target system detects a problem, SDF is updated in the agent's SDF, and the status is propagated over the outbound gateway session to the SDF focal point.

The SDF focal point in SYS4 is customized to display the status of resources across all four systems on an Enterprise Status panel. An operator in SYS4 can then use SDF as a status focal point.

An operator on any system can also access SysOps automation functions to control resources across all four systems if they are in the same SAplex or have gateway sessions to each other like shown in an example later in this unit which shows four domains that are interconnected. This type of management is called *Single Point of Control (SPOC)*.

In this example, the SDF focal point is in SYS4. The location of the primary automation manager is independent from the SDF focal point. The primary automation manager and the SDF focal point can be on different systems. For example the primary automation manager can run in SYS1.

## An overview of SDF focal points

- Status changes are forwarded to the Status Display Facility focal point
  - Primary focal point, if active
  - Backup focal point otherwise (if defined)
  - To both in parallel (new)
- Requires gateway connections
- If no focal point is available, then no forwarding occurs
- Status Display Facility status updates require the Inform List policy definition

### *An overview of SDF focal points*

Typically, the Automation system administrator sets up SDF and the focal points and gateway sessions. SDF status information flows across the gateway sessions (XCF or RMTCMD). To see SDF status updates, your system administrator must customize the Inform List policy definition.

## Enterprise monitoring with SDF

- Status Display Facility uses a set of customized panels
- Focal point uses an Enterprise Status panel
- All other domains use panels to display local status

### *Enterprise monitoring with SDF*

The supplied Status Display Facility panels, SDF definitions, and gateway session definitions must be tailored for the installation. They can display the resource status for multiple systems.

## Determining the focal point

```
AOFKADAE          SA z/OS - Command Dialogs           Line  30   of  53
Domain Id . : AOFDA  ----- DISPSYS ----- Date . . : 10/31/18
Operator Id : INGC102    System = MVSA             Time . . : 11:06:26

Automation      : Yes
Init Start     : Yes
Start          : Yes
Recovery        : Yes
Terminate       : Yes
Restart         : Yes

Scheduling Subsystem : JES2
Type            : JES2

Root for SDF updates : MVSA
SDF actual focal point : AOFDB
primary focal point : AOFDB
backup focal point : AOFDA

USS path       : /usr/lpp/ing/ussauto/lib/
Inform list    : SDF

IPL Complete Considerations
Time Limit     : - None -
Expected Statuses :
                  AVAILABLE
Important Resources :
                  - None -

Command ===> _____
F1=Help   F2=End   F3=Return
F7=Backward F9=Refresh
                                         F6=Roll
                                         F12=Retrieve
```

Current focal point is the primary Status Display Facility (SDF) focal point, AOFDB

© Copyright IBM Corporation 2019

7-13

### Determining the focal point

The DISPSYS command displays information about Automation. The information includes domain names of the primary and backup focal points, and identifies the current focal point. In this example, AOFDB is the primary focal point and the current focal point.

## SDF enterprise view panel example

MVSA

SA z/OS 4.1 Education

05/22/19 19:54:25

Environment >Configuration Refresh

SAplex  
sysplex  
VM guest      SYSPLEX1  
PLEX12  
mvsch0d      >MVSA      >MVS<sup>B</sup>

STANDALONE SYSTEMS      >MVSC

Hardware >Processors

>Ensembles

====> \_\_\_\_\_  
1=Help 2=Detail 3=Return      6=Roll 8=Zoom

© Copyright IBM Corporation 2019

7-14

*SDF enterprise view panel example*

The "SDF" command will initially show the top panel. From there customers can use the standard, well known SDF navigation techniques and PF Keys to view data for specific systems and underlying resources. You place the cursor on a system name and use the F8 key to go to the Summary status for a system panel.

This slide shows the customized root panel *INGPOTOP* that is provided as a sample in the SINGNPRM data set. The systems have been adapted and the panel title has been changed. This is an enterprise-wide panel that can be used as the top panel defined for an SDF focal point. Also system symbols are used to display the sysplex name and the z/VM guest the system runs in.

The panel was designed to show the SA for z/OS controlled systems MVSA and MVS<sup>B</sup> in SAplex SYSPLEX1. System MVSC is displayed in the standalone systems area.

Configuration Refresh Monitoring displays the status of the configuration refreshes.

The hardware area has not been removed but is not implemented here.

The next slide shows an example of a System Status Summary panel.

## System status summary example

```
MVSAMAIN          SUMMARY-STATUS
                  MVSA
!----- Resources -----
>APPLS
>GROUPS
>MONITORS
!----- Messages -----
>WTOR          >MESSAGES
!----- Special Items -----
>GATEWAY        >Tape Devices
>TWS
10/25/18 11:24
==> █
1=Help 2=Detail      6=Roll  8=Zoom      12=Top
```

Press the Tab key to APPLS name and press PF8 (Zoom) to see the applications on MVSA

© Copyright IBM Corporation 2019

7-15

### System status summary example

In this example, the System Status Summary panel shows a high-level view of resources, messages, and special items. By placing the cursor on an item and pressing ZOOM (PF8) again, you can see more information about the item. The next slide shows the status of each application (>APPLS) on this system.

The Automation administrator defines SDF panel content and navigation paths. Customizing SDF is covered in the *IBM System Automation for z/OS: Implementation and Administration* course.

MVSB	MVSB: SUBSYSTEM-STATUS	1/32 of 32
VLF	RV06	
RESOLVER	SDSF	
VTAM	ASCH	
AUTONETV	TSO	
ZFS	RMFGAT	
JES2	TCPIP	
LLA	TELNET	
RACF	FTP	
AUTOSSI	TN_PORT	
OMVS	FTP_PORT	
APPC	<u>BLSJPRMI</u>	
DLF	<u>IRRDPRTAB</u>	
HZSPROC	<u>SYSVIPLC</u>	
RMF	<u>RV05</u>	
RV02	<u>AUTOMGR2</u>	
RV04	AUTOMGR	
08/31/18 11:53		
====> ■		
1=Help 2=Detail	6=Roll	10=Previous 11=Next 12=Top
13=EXPLAIN 17=SETSTATE 18=INGVOTE 19=INGREQ		23=INGLIST 24=INGINFO
© Copyright IBM Corporation 2019		
7-16		

Subsystem status panel example

PF keys on the subsystem status panel can be used to issue commands against any subsystem in the list.

# Lesson 3 Managing resources in a multisystem environment



## Lesson 3. Managing resources in a multisystem environment



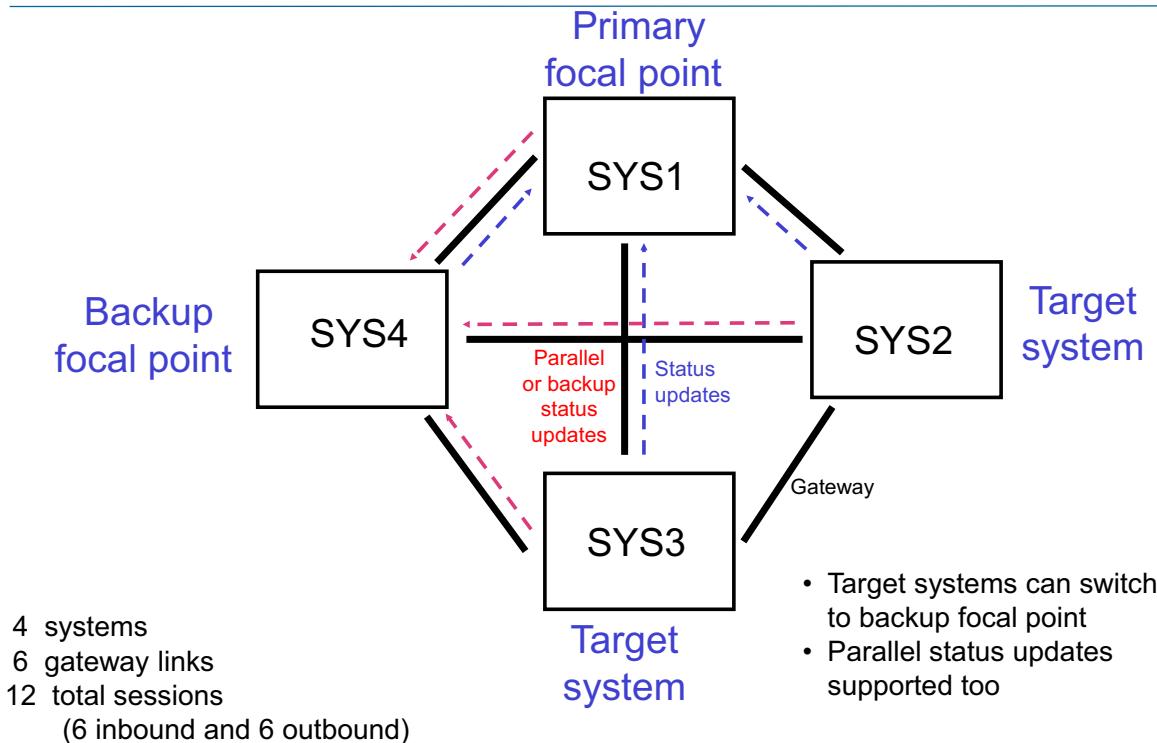
Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

In this lesson you learn to manage resources in a multisystem environment using the automation single point of control and how to route messages and commands in a multisystem environment.

## Managing resources in a multisystem environment

- Gateways, focal point, and SPOC
- SPOC examples
- Message notification and forwarding
- Routing commands

# Gateways, focal point, and SPOC



© Copyright IBM Corporation 2019

7-19

## Gateways, focal point, and SPOC

This slide illustrates three concepts:

- SDF focal points
- SPOC
- Gateway links between the domains to support the focal point and SPOC

In this example, gateway links, represented by the solid lines, connect all four domains. There are 6 gateway links, 12 sessions. This type of configuration enables any domain to be used as the single point of control. AOFA is the primary focal point and AOF1 is the backup focal point.

The dashed lines represent status, which flows to the focal point from domains, AOF1, AOF2, and AOF3. If all of these systems are in the same sysplex, the gateway sessions use XCF for communication.

## A summary of SPOC

- System Automation for z/OS commands can be issued to any domain
- The target of a command, domain, system, or sysplex name is specified on most panels
  - You can also use the TARGET parameter for command line entry

```
>--.-.---.----.--->
  '-TARGET=- .-| Destination | -.-'
    |-| Dest_list | ---|
      '--*ALL-----'
  | -OUTDSN=dsname-----|
  | -OUTMODE=-. -LINE---.-----|
  | -AUTO---|
  | -NETLOG-'
```

- Communication methods
  - XCF if both domains are within the sysplex
  - RMTCMD to target domains outside of a sysplex

### A summary of SPOC

Single point of control (SPOC) enables Automation commands to be routed to remote or targeted domains from any system in the enterprise. Most Automation commands can route requests to one or more systems in an enterprise. The output can be displayed on a panel, sent to a data set, to the NetView log, or back to the caller.

Command routing is accomplished by specifying a target domain, system name, or SAplex name as a parameter or input field on a panel. The DISPSTAT, SETSTATE, and INGREQ command panel all have fields in which to specify the target of the command.

Commands are routed to target domains through XCF or private RMTCMD sessions. The transport mechanism is invisible to the person or task that issues the command. You can create more complex automation routines with OUTMODE=LINE. SPOC also uses your gateway connections initially to collect topology information about the Automation domains and how to communicate with them.

## Centralizing operations

- Use Status Display Facility for status monitoring
- Route System Automation for z/OS commands
  - From panels (if supported)
  - Line mode (using TARGET parameter)
- Remotely browse NetView log, data set, or members
  - Base NetView function
  - Requires RMTCMD over SNA or TCP/IP
  - Example: **BROWSE LU=AOFDB NETLOGA**

### Centralizing operations

Browsing the NetView log, a data set, or members of a data set is a basic NetView function called *remote browsing*. Remote browsing uses RMTCMD connections, which can be SNA (NetView 5.1 or later) or TCP/IP (NetView version 5.2 or later).

NetView also provides a panel interface command BLOG for searching the NetView log. The following example shows one command format that is used to browse remotely the NetView log of a domain AOFDB. This example uses the SNA form of RMTCMD.

```
BROWSE LU=AOFDB NETLOGA
```

In a multisystem environment, Automation commands can be routed from operator panels or command line.

## The TARGET parameter Used in RV scenario

- The TARGET parameter is used in the following commands:
  - DISPSTAT: Display agent status for applications
  - SETSTATE: Change agent status of applications
  - DISPTRG: Display triggers
  - DISPEVT: Display events
  - INGEVENT: Set or unset events
- Almost all SA z/OS commands can specify the TARGET= parameter to force the command to execute on the target system

### *The TARGET parameter*

The slide lists several Automation commands that are used during a scenario that is shown in the next few slides. In the scenario, a search is done for inactive applications with RV as the first 2 characters in the automation name. The inactive application that is found in this case RV01. The application is inactive because an event is not set. The unset event is preventing a trigger from being satisfied. When the event is set, the application starts.

The TARGET parameter is used with Automation commands to route the commands from domain AOFDA to domain AOFDB. These commands can be used in REXX execs. It is more efficient to place a frequently used sequence of commands in a REXX exec, which can be scheduled or triggered automatically.

Almost all SA z/OS commands can specify the TARGET= parameter to force the command to execute on the target system.

# Using the TARGET parameter: Example

## 1. DISPSTAT RV\* TARGET=AOFDB OUTMODE=LINE

SUBSYSTEM	STATUS	SYSTEM	A	I	S	R	D	RS	Monit
RV01	CTLDOWN	MVSB	Y	Y	Y	Y	Y	Y	08:46
RV02	UP	MVSB	Y	Y	Y	Y	Y	Y	08:46
RV03	STOPPED	MVSB	Y	Y	Y	Y	Y	Y	08:46
RV04	STOPPED	MVSB	Y	Y	Y	Y	Y	Y	08:46

## 2. SETSTATE RV01 AUTODOWN TARGET=AOFDB OUTMODE=LINE

AOF571I 09:19:04 : RV01 SUBSYSTEM STATUS FOR JOB RV01 IS AUTODOWN -  
REQUESTED BY OPERATOR NETOP1 - FROM=MVSB

## 3. DISPSTAT RV01 TARGET=AOFDB OUTMODE=LINE

SUBSYSTEM	STATUS	SYSTEM	A	I	S	R	D	RS	Monit
RV01	AUTODOWN	MVSB	Y	Y	Y	Y	Y	Y	09:16

### Using the TARGET parameter: Example

- Issue a DISPSTAT command to AOFDB to display all RV\* applications. RV01 is down, with a status of CTLDOWN.
- Issue a SETSTATE command to AOFDB to set the status of RV01 to AUTODOWN. This action starts the process to activate the application if it can be started.
- To verify that the SETSTATE was successful, issue another DISPSTAT command for RV01.  
The SETSTATE command worked, but RV01 is still not UP.

## Using the TARGET parameter: Example (continued)

### 4. DISPTRG RV01/APL/MVSB TARGET=AOFDB OUTMODE=LINE

```

Resource = RV01/APL/MVSB
Trigger = RV01TRIG          System = MVSB
Type      Events      (events prefixed by a '*' have the status 'SET')
-----
STARTUP   RV01EVT
*** End of Display ***

```

### 5. DISPEVT RV01EVT TARGET=AOFDB OUTMODE=LINE

```

Event = RV01EVT
Unset condition = UP
Description = EVENT FOR RV01
Resource    Type   System     Trigger   Status
-----
RV01        APL    MVSA       RV01TRIG
RV01        APL    MVSB       RV01TRIG   UNSET
*** End of Display ***

```

The SETSTATE command worked, but RV01 is still inactive. The INGVOTE command can be used to show if there are any MakeUnavailable votes against RV01. Instead, assume that a trigger is unsatisfied and it is the inhibitor that is blocking the start of RV01.

4. Use command DISPTRIG to check whether any triggers are linked to application RV01. DISPTRG shows that a startup trigger, RV01TRIG, is linked to application RV01, and RV01TRIG is linked to a startup event, RV01EVT. Startup trigger RV01TRIG is unsatisfied because event RV01EV is unset.

The full resource name RV01/APL/MVSB must be used in the command. The automation name of the resource is RV01. The resource type is APL (application), and the name of the target system is MVSB.

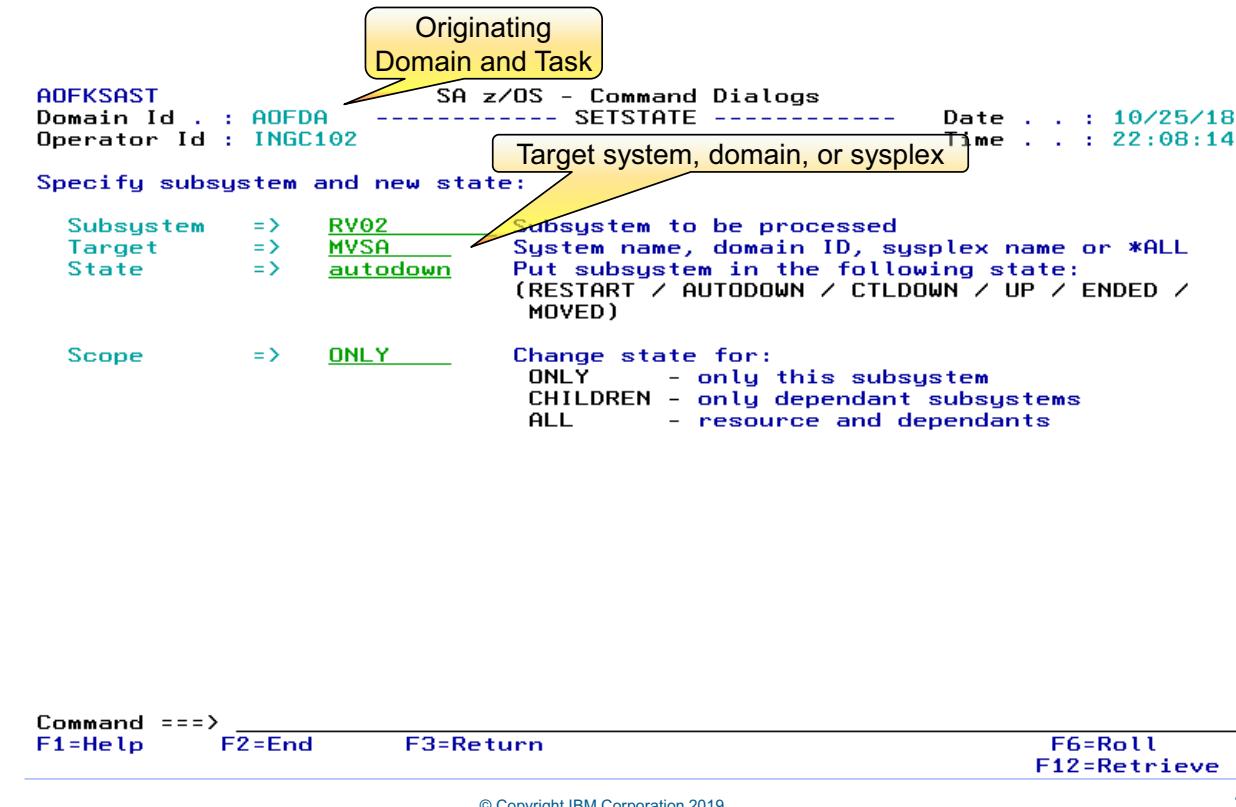
5. The DISPEVT command shows that trigger RV01TRIG is unset.

The INGEVENT command can be used to set the event.

```
INGEVENT RV01EVT REQ=SET TARGET=MVSB WAIT=YES
```

In this example, the TARGET parameter is used to illustrate routing requests to remote domains. The TARGET parameter is not required when the target system is in the same SAplex as the source of the command.

# The SETSTATE command panel



*The SETSTATE command panel*

This example shows how to specify a target domain on the SETSTATE panel to change the status of an application in a remote domain. As seen in the example on this slide, you can change the STATUS field for resources on any Automation managed system. If a SAplex name is specified in the SYSTEM field, then the SETSTATE command is sent to all RV03 resources in the System Automation for z/OS SAplex. This process is similar to issuing the command-line version of the SETSTATE command.

When there are multiple instances of an application in a SAplex and ENTER is pressed while the SYSTEM field is blank, a selection panel is displayed. The next slide shows an example of the selection panel.

## The system selection panel

```

AOFKSEL1          SA z/OS - Command Dialogs      Line 1   of 2
Domain Id . : AOFDA ----- SETSTATE ----- Date . . : 10/31/18
Operator Id : INGC102                         Time . . : 12:53:04

Multiple instances found for TSO in sysplex(es) PLEX12
Select one or more items to be processed, then press ENTER.

Sel.  System    Domain
----  -----    -----
  MVSA    AOFDA
  MVSB    AOFDB

```

Similar for other dialogs when resource is not unique

```

Command ==> _____
F1=Help     F2=End     F3=Return
                                         F6=Roll
                                         F12=Retrieve

```

© Copyright IBM Corporation 2019

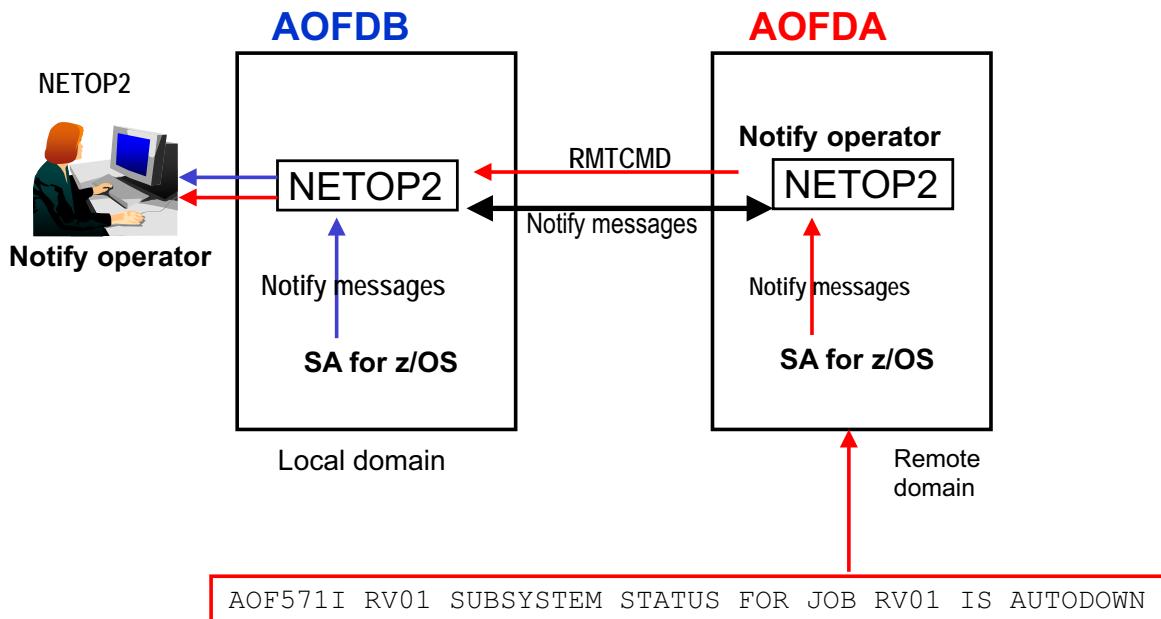
7-26

### *The SETSTATE system selection panel*

If there are multiple instances of a selected application in a SAplex and the SETSTATE command is issued without a fully qualified resource name, a selection panel is displayed. On the panel that is shown in the previous slide, the ENTER key was pressed while the TARGET field was blank with TSO specified as Subsystem and a State specified as well.

Most automation commands that provide fields to route requests to other domains also display a pop-up selection panel if the command is entered with an empty TARGET field.

# Routing messages in a multisystem environment



## Routing messages in a multisystem environment

Centralization of notify operators for forwarding messages between Automation domains is supported.

To route messages across NetView domains, perform these tasks:

- Define Automation notify operators either in the customization dialogs or the INGNTFY command
- Based on Automation message classes, assign messages to notify operators
- Establish RMTCMD sessions between local and remote domains with notify operator IDs

Messages flow across RMTCMD sessions to the local domain. When messages arrive at the local domain, they can be displayed and automated. For example, NETOP2 is defined as a notify operator in all NetView domains. NETOP2 in AOFDB can start a RMTCMD session to AOFDA. The definition log NETOP2 is on AOFDA by default. Messages that are destined for NETOP2 in AOFDA flows across the RMTCMD session to NETOP2 in AOFDB also.

The net effect is that NETOP2 in AOFDB receives notification messages for both systems. This technique can be applied to the entire enterprise to provide centralized notification and message forwarding. For example, you can define NETOP2 to receive only messages that are related to CICS from across the enterprise.

In this example, message AOF571I is sent to NETOP2 who is defined as a notify operator in domain AOFDA. The message flows across the RMTCMD session to domain AOFDB.

SYSOP is a special operator ID that is assigned to z/OS system consoles. SYSOP can also be defined as a notify operator. When SYSOP is a notify operator, Automation messages are displayed on the z/OS system console and logged in the z/OS system log.

## Notify operator example view

```

SA V4R1 Education      Tivoli NetView    AOFDA INGC100 10/28/18 21:40:18 A
! AOFDA   HSAL6051I MVSA/SYS/MVSA; SATISFACTORY
! AOFDA   HSAL6051I APPC/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I ASCH/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I AUTOMGR2/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I AUTOSSI/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I BLSJPRMI/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I DLF/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I FTP/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I FTP_PORT/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I HZSPROC/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I IRRDPTAB/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I JES2/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I LLA/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I OMVS/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I RACF/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I RESOLVER/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I RMF/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I RMFGAT/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6051I SDSF/APL/MVSA; SATISFACTORY
- AOFDA   DS1151I MORE HELD MESSAGES EXIST AND ARE NOT SHOWN
! AOFDA   HSAL6030I DEMO_1MON/MTR/MVSA; IN AUTOMATION
! AOFDA   HSAL6051I DEMO_1A/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6040E DEMO_1/APG/MVSA; DEGRADED
! AOFDA   HSAL6041E DEMO_1MON/MTR/MVSA; DEGRADED
-----
U AOFDA   AOF571I 21:40:18 : DEMO_1A SUBSYSTEM STATUS FOR JOB DEMO1A IS UP -
UP MESSAGE RECEIVED
! AOFDA   HSAL6051I DEMO_1B/APL/MVSA; SATISFACTORY
! AOFDA   HSAL6050I DEMO_1/APG/MVSA; SATISFACTORY
???

```

Notify operator in domain AOFDA receives messages from domain AOFDA

### Notify operator example view

In this example, the notify operator who is logged on in domain AOFDA receives messages from system MVSB.

# Routing commands

- Tivoli NetView for z/OS
  - RMTCMD
    - Connection is TCP/IP or SNA
    - Supports line-mode command output only
    - Operator is logged on to target domain
    - Default remote ID is the same operator ID
- System Automation for z/OS
  - INGSEND
    - Uses Tivoli NetView for z/OS RMTCMD
  - INGEXEC
    - Routes commands across the System Automation for z/OS sysplex
    - Supports the use of filters
    - Command output can be automatically returned

## Routing commands

The Automation INGSEND and INGEXEC commands, and the NetView RMTCMD can all be used to route commands to remote NetView domains. INGSEND, which uses the NetView RMTCMD, provides a panel interface for issuing non-Automation commands to other domains.

The INGEXEC and INGSEND commands require a fully initialized Automation environment. The commands interrogate the automation manager to determine what resources are affected. The scope of the INGEXEC command is SAplex-wide, and it supports the use of filters. The command-line output restrictions of the other commands do not apply.

## RMTCMD example

```

SA V4R1 Education          Tivoli NetView      AOFDA INGC102 10/30/18 16:28:27 A
* AOFDA RMTCMD SEND LU=AOFDB,MVS D A,L
" AOFDB
IEE114I 16.28.27 2018.303 ACTIVITY 610
JOBS   M/S   TS USERS    SYSAS   INITS   ACTIVE/MAX VTAM     OAS
00001  00021  00001  00032  00010  00001/00025  00009
LLA    LLA   LLA    NSW S VLF    VLF    VLF    NSW S
JES2   JES2  IEFPROC  NSW S NET    NET    VTAM    NSW S
RACF   RACF  RACF   NSW S TSO    TSO    STEP1   OWT S
AUTOSSI AUTOSSI NETVIEW NSW S AUTONETV AUTONETV NETVIEW NSW SO
APPC   APPC  APPC   NSW S DLF    DLF    DLF    NSW S
HZSPROC HZSPROC HZSTEP  NSW S RV08   RV08   STEP1   OWT S
RMF    RMF   IEFPROC  NSW S RV03   RV03   STEP1   OWT S
SDSF   SDSF  SDSF   NSW S TCPIP  TCPIP  TCPIP  NSW SO
ASCH   ASCH  ASCH   NSW S RMFGAT RMFGAT IEFPROC NSW SO
TELNET TELNET TN3270  NSW S FTPSERVE STEP1   SYSPROG OWT AO
RV09   RV09  STEP1   OWT S RV02   RV02   STEP1   OWT S
LLCOX  OWT

```

???

Output from MVS D A,L command by task on AOFDA to domain AOFDB

© Copyright IBM Corporation 2019

7-30

### RMTCMD example

This slide shows the result of routing z/OS command and parameters MVS D A,L from domain AOFDA to domain AOFDB.

RMTCMD SEND LU=AOFDB,OPERID=TSOST01,MVS D A,L

RMTCMD requires setup in the style sheet. Browse member CNMSTGEN and search for RMT.

You can use a simplified syntax for RMTCMD, called a *labeled* RMTCMD. With a labeled RMTCMD, you enter the target domain with a colon as the label, followed by the command you want to route.

#### Example 1:

Route the command to AOFDB and run under your ID.

AOFDB: MVS D A,L

#### Example 2:

Route the command to AOFDB and run under ID, OPER1.

AOFDB/OPER1: MVS D A,L

## INGEXEC example

```

SA V4R1 Education          Tivoli NetView    AOFDA TNGC102 10/30/18 16:40:20 A
* AOFDA INGEXEC MVSB/SYS/MVSB CMD='MVS D A,L' SELECT=FIRST
I AOFDA
SYSTEM=MVSB MVSB/SYS/MVSB RC=0 CMD='MVS D A,L'
IEE1141 16.40.20 2018.303 ACTIVITY 636
JOBS   M/S   TS USERS     SYSAS   INITS   ACTIVE/MAX VTAM      OAS
00001  00021  00000   00032  00010  00000/00025  00009
LLA    LLA    LLA    NSW  S  VLF    VLF    VLF    NSW  S
JES2   JES2   IEFPROC  NSW  S  NET    NET    VTAM   NSW  S
RACF   RACF   RACF   NSW  S  TSO    TSO    STEP1  OWT  S
AUTOSSI AUTOSSI NETVIEW  NSW  S  AUTONETV AUTONETV NETVIEW  NSW  SO
APPC   APPC   APPC   NSW  S  DLF    DLF    DLF    NSW  S
HZSPROC HZSPROC HZSSTEP  NSW  S  RV08   RV08   STEP1  OWT  S
RMF    RMF    IEFPROC  NSW  S  RV03   RV03   STEP1  OWT  S
SDSF   SDSF   SDSF   NSW  S  TCPIP  TCPIP  TCPIP  NSW  SO
ASCH   ASCH   ASCH   NSW  S  RMFGAT  RMFGAT  IEFPROC NSW  SO
TELNET TELNET  TN3270  NSW  S  FTPSERVE STEP1  SYSPROG OWT  AO
RV09   RV09   STEP1   OWT  S  RV02   RV02   STEP1  OWT  S
-----
```

???

© Copyright IBM Corporation 2019

7-31

### INGEXEC example

This example shows the format of the INGEXEC command.

```
INGEXEC MVSB/SYS/MVSB CMD='MVS D A,L' SELECT=FIRST
```

For more information about the INGEXEC command, see the *System Automation for z/OS Programmer's Reference*.

## Student exercise



© Copyright IBM Corporation 2019

7-32

### Student exercises

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you completed this unit, you can perform the following tasks:

- Use System Automation for z/OS to manage resources in a multisystem environment
- Describe the purpose of the gateway connections
- Use Status Display Facility (SDF) at a focal point system
- Use System Automation for z/OS as a single point of control (SPOC)

### *Summary*

Now that you completed this unit, you can perform the following tasks:

- Use System Automation for z/OS to manage resources in a multisystem environment
- Describe the purpose of the gateway connections
- Use Status Display Facility (SDF) at a focal point system
- Use System Automation for z/OS as a single point of control (SPOC)



## 8 Troubleshooting

---

IBM System Automation for z/OS 4.1



## Unit 8: Troubleshooting



© Copyright IBM Corporation 2019

Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This unit is building on what you have learned to determine why resources won't start or stop. Collecting information, using trace and debug and solving problems with messages.

## **Objectives**

---

When you complete this unit, you can perform the following tasks:

- Determine why resources won't start or stop
- Collecting information
- Using trace and debug
- Solving problems with messages

### *Objectives*

When you complete this unit, you can perform the following tasks:

- Determine why resources won't start or stop
- Collecting information
- Using trace and debug
- Solving problems with messages

# Lesson 1 Solving problems with resources



## Lesson 1. Solving problems with resources



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson focuses on solving problems with resources. You will learn problem analysis and resolution using the INGWHY command to speed up problem resolution



## Solving problems with resources

---

- Design policy to make problem detection easy
- What's available for problem detection
- Problem states and solution
- Analysis and resolution using the INGWHY command

### *Lesson agenda*

- Design policy to make problem detection easy
- What's available for problem detection
- Problem states and solution
- Analysis and resolution using the INGWHY command

## Implement policy that is correct, can detect problems, can filter and can avoid false alerts

- Use correct state messages, dependencies, start and stop commands... (IBM best practices recommended)
- Error Thresholds for recovery and messages
- Capture exception messages using
  - Specific messages
  - Generic patterns using descriptor codes or last byte of message ID=D,A
- Automatic collection of important error information:
  - Message: AOF313I START FOR resource WAS NOT ATTEMPTED ...
  - INGLKUP resource REQ=COLLECT
- Use monitor resources & health states for proactive automation
  - Integrate with monitoring, add your own e.g. for NetView
- Implement alerting using built-in alert points and INGALERT
  - Filtering of unimportant resources, messages or codes
- Record KPIs from INGRPT, AUTOCNT, CPU...

### *Implement policy that is correct*

As you have learned, the automation product gives you the policy functions to model your configuration including dependencies, goals, thresholds, triggers, restart options and a lot more.

So downside is that there could be many reasons why resources won't start or stop, but the automation product offers a lot of status information and insights that you will learn to use.

A prerequisite for easier problem detection is implementing policy that is correct, can detect problems, can filter and can avoid false alerts. Otherwise the operators have problems detecting problems or even become annoyed and stop paying attention to red alerts.

IBM best practices are recommended for new installations, but also long time users should follow IBM best practices as they are being updated.

Error Thresholds for recovery and messages can stop automation and flooding operators with redundant alerts.

The policy allows to capture exception messages using specific messages or generic patterns using MVS descriptor codes or the last byte of message ID which indicates errors, decisions or immediate actions.

Automatic collection of important error information can reduce problem resolution time. A good candidate is message: AOF313I START FOR resource WAS NOT ATTEMPTED ...reason. You could add it to the policy and collect important error information using

```
INGLKUP resource REQ=COLLECT
```

Use monitor resources & health states for proactive automation by integrating automation with monitoring or add your own health monitoring like we did in the course with the monitor resource for NetView.

Implementation of alerting is easy when using the built-in alert points and the alerting infrastructure with INGALERT.

Filtering of unimportant resources, messages or codes can be done with CODE processing, like suppressing alerts from certain jobs or transactions or by changing the INFORM policy to send an alert to different targets or not at all.

Recording key performance indicators (KPIs) provided by INGRPT, AUTOCNT, or some CPU monitor on a regular base allows you to analyze trends or spot spikes and to take corrective actions before they become worse.

#### Command

```
INGRPT OUTMODE=LINE
```

outputs some useful metrics like:

Total number of messages automated	:	66
Average number of messages per hour	:	0.6
Total number of resulting commands	:	130
Average number of commands per hour	:	1.1

Total number of StartUp commands issued	:	79
---	---	----

Total number of ShutDown commands issued	:	57
--	---	----

\*\*\*\*\* Summary Statistics \*\*\*\*\*

OBSERVER	641
----------	-----

OTHERS	118
--------	-----

RESUME	14
--------	----

START	460
-------	-----

STOP	273
------	-----

SUSPEND	17
---------	----

Average number of orders per hour	:	15.9
-----------------------------------	---	------

#### Automation Manager Statistics

Number of systems in sysplex	:	1
------------------------------	---	---

Total number of resources	:	51
---------------------------	---	----

Number of APL resources	:	27
-------------------------	---	----

Number of APG resources	:	8
-------------------------	---	---

Number of MTR resources	:	4
-------------------------	---	---

## Command

```
AUTOCNT REPORT=MSG NAME=INGMSG01
```

outputs some useful metrics like:

```
DSI#0031 COMPLETED INSERT FOR TABLE #3: INGMSG01 AT 08/31/18 19:54:50
STATISTICS STARTED = 08/31/18 19:54:50
TOTAL MSGS PROCESSED = 26345
MSGs MATCHED = 26345
MSGs RESULTING IN COMMANDS = 229
TOTAL COMMANDS EXECUTED = 230
TOTAL ROUTES EXECUTED = 0
AVERAGE COMPARES/MSG = 27.53
AVERAGE MSGS/MINUTE = 4
MINUTES ELAPSED = 5813
```

# Detecting problems

- Watch out for problem compound statuses:
  - Start at aggregated resources, like sysplex, system or application groups
  - Unsatisfactory: Resource is not reaching its goal
  - Degraded: not reaching group satisfactory target, health state is not Normal...
- Watch out for problem (captured) messages
  - Use INGMSGSS or SDF, DISPINFO or DISPSYS
- Analyze KPI trends and identify spikes
  - Not available out of the box

© Copyright IBM Corporation 2019

8-6

## *Detecting problems*

Watch out for **problem compound statuses**. This status is a summary of all the statuses of a resource. It considers elements of the four other statuses as well as a number of other values.

Start at aggregated resources, like sysplex, system or application groups.

Unsatisfactory means the resource is not reaching its goal.

Degraded for a group indicates the group is not reaching the group satisfactory target.

For normal resources, it can mean that the resource is Starting or Stopping.

Degraded can also indicate that a health state is not Normal.

Watch out for **problem (captured) messages** by using INGMSGSS or SDF, DISPINFO or DISPSYS.

Analyzing KPI trends and identifying spikes is highly recommended, but not available out of the box.

## Analysis approach

- Check dependencies (INGRELS or DISPTREE)
- Check resource information (INGINFO or DISPINFO)
- Check votes (INGVOTE, INGSCHED)
- Check flags (DISPFLGS or SETSTATE)
- Check APG details – Pref.Values – Sat.Tgt – Avail.Tgt (INGGROUP)
- Check Triggers ....
- Check ...

A good scenario is described in *System Automation for z/OS: User's Guide, Chapter 9. Solving Problems with Resources, Availability Scenario*

### Analysis approach

A typical analysis approach checks all goals, all inhibitors and resource information of the resource having the problem as well as of supporting resources as follows:

- Check dependencies (INGRELS or DISPTREE)
- Check resource information (INGINFO or DISPINFO)
- Check votes (INGVOTE, INGSCHED)
- Check flags (DISPFLGS or SETSTATE)
- Check APG details – Preference values – Satisfactory target – Availability target (using INGGROUP)
- Check Triggers ....
- Check ...

A good scenario is described in *System Automation for z/OS: User's Guide, Chapter 9. Solving Problems with Resources, Availability Scenario*

## Finding the reason for start problems

### ***Check Compound & Observed status with INGLIST***

- IF compound\_status = **SATISFACTORY** and desired\_status = **AVAILABLE** (no problem - Exit)
- IF compound\_status = **AWAITING**
  - Check with INGRELS if relationships conditions are unsatisfied
  - Check with INGINFO if start dependencies are unsatisfied
  - Check with INGTRIG/DISPTRG if start triggers are unsatisfied
  - Check whether inherited from group or why supporting resources are not in the required condition
- IF compound\_status = **SATISFACTORY** & desired\_status = **UNAVAILABLE**
  - Check with INGVOTE if stop requests exists
  - Check with INGSCHED if down window exists
- IF compound\_status = **DENIED** & observed\_status = **STARTING** & agent\_status = **RESTART**,
  - Check with DISPINFO / DISPSTAT if RESTART agent flag is OFF
- IF compound\_status = **DENIED**
  - Check with INGINFO/INGLIST if automation flag is OFF
- IF compound\_status = **PROBLEM** & observed\_status = **PROBLEM** & agent\_status = **STARTED2**
  - Check with DISPINFO for valid start commands and/or missing UP msg
- IF compound\_status = **PROBLEM** & observed\_status = **SOFTDOWN** & agent\_status = **INACTIVE**
  - Address space has not been started - check definitions with DISPINFO
- IF compound\_status = **PROBLEM** & observed\_status = **HARDDOWN**
  - Check with DISPSTAT / DISPINFO for agent status

© Copyright IBM Corporation 2019

8-8

#### *Finding the reason for start problems*

If you have resources with a start problem you can find the reason by checking the compound & observed statuses with INGLIST using the following list:

A compound status of AWAITING means the resource is waiting for any supporting resources to reach the specified condition. Check with INGRELS if relationships conditions are unsatisfied.

If the relationship is active, votes are propagated to the supporting resources that hopefully will put the supporting resources into the specified condition.

If the relationship is passive, the resource is waiting for any supporting resources but votes are not propagated, so it could wait forever.

A compound status of DENIED means automation is unable to proceed because either the automation flag has been turned off or the automation manager hold flag for the resource is set to YES.

A compound status of PROBLEM means a harddown agent status and that operator intervention is required.

## Finding the reason for start problems with **Compound status = INHIBITED, Degraded, InAuto**

**Inhibited** means:

- Supporting resource has a compound status of Problem or Denied or is suspended
- Operators should try to fix the supporting resources
- For suspended resources, find out whether the resource is suspended directly or indirectly and then, if resource maybe resumed, do a resume at the right level

**Degraded** means:

- For a group it means that it is partially running, but not at full capacity. This means the satisfactory target is not reached
- For normal resources, it can mean that the resource is Starting or Stopping, or
- The application health state is not Normal

**InAuto** means:

- SA z/OS is in the process of starting or stopping the resource
- If the resource stays in InAuto longer than it would take for starting or stopping, you can expect a follow on problem state, like STUCK
- For groups, if you see InAuto combined with Observed status of STARTING, at least one required member is not available

*Compound status = INHIBITED, Degraded, InAuto*

**Inhibited means:**

Supporting resource has a compound status of Problem or Denied or is suspended. Operators should try to fix the supporting resources. For suspended resources, find out whether the resource is suspended directly or indirectly and then, if resource maybe resumed, do a resume at the right level.

**Degraded means:**

For a group it means that it is partially running, but not at full capacity. This means the satisfactory target is not reached. For normal resources, it can mean that the resource is in agent status HALTED or HALFDOWN, is starting or stopping, or the application health state is not Normal.

**InAuto means:**

Automation is in the process of starting or stopping the resource. If the resource stays in InAuto longer than it would take for starting or stopping, you can expect a follow on problem state, like STUCK.

For groups, if you see InAuto combined with observed status of STARTING, at least one required member is not available.

# Problem states and finding the reason

## **Observed Status = HARDDOWN**

- IF compound\_status = PROBLEM & observed\_status = HARDDOWN & agent\_status = CTLDOWN
  - Check why the resource has been brought into CTLDOWN with SETSTATE
  - Use SETSTATE to bring agent status into AUTODOWN
- IF compound\_status = PROBLEM & observed\_status = HARDDOWN & agent\_status = STOPPED
  - Check why the resource has been stopped outside of System Automation
  - Use DISPINFO to check "Restart when Stopped": ABENDONLY
  - Use SETSTATE to bring agent status into AUTODOWN
- IF compound\_status = PROBLEM & observed\_status = HARDDOWN & agent\_status = BROKEN
  - Check whether the resource has reached the thresholds. Why?
  - Check whether resource had a non-restartable ABEND
  - Fix problem
  - Use SETSTATE to bring agent status into AUTODOWN

### *Observed Status = HARDDOWN*

The resource is unavailable and automation will not restart it. The agent status is one of:

1. CTLDOWN

Check why the resource has been brought into CTLDOWN

2. STOPPED

The automation agent has determined that the resource has been stopped normally outside the control of automation and its policy indicates that an automated restart is not allowed.

3. BROKEN

the application has failed in a nonrecoverable fashion. Proposed actions:

a. Check whether the resource has reached the thresholds. Why?

b. Check whether resource had a non-restartable ABEND

After fixing the ABEND problem or clarifying the manual actions, put resource into AUTODOWN using SETSTATE.

## Finding the reason for stop problems

### Check Compound & Observed status with INGLIST

- IF compound\_status = **SATISFACTORY** and desired\_status = **UNAVAILABLE**
  - No problem - Exit
- IF compound\_status = **AWAITING**
  - Check with INGRELS if relationships conditions are unsatisfied
  - Check with INGINFO if stop dependencies are unsatisfied
  - Check with INGTRIG/DISPTRG if stop triggers are unsatisfied
  - Check whether inherited from group or why supporting resources are not in the required condition
- IF compound\_status = **SATISFACTORY** & desired\_status = **AVAILABLE**
  - Check with INGVOTE if start requests exists
  - Check with INGSCHED if up window exists
- IF compound\_status = **DENIED** & observed\_status = **STOPPING** & agent\_status = **AUTOTERM**
  - Check with DISPINFO / DISPSTAT if shutdown agent flag is OFF
- IF compound\_status = **DENIED** & observed\_status = **AVAILABLE**
  - Check with INGINFO/INGLIST if automation flag is OFF
- IF compound\_status = **PROBLEM** & observed\_status = **STOPPING** & agent\_status = **STUCK**
  - Check with DISPINFO for valid stop commands and/or missing DOWN msg
  - Consider to cancel stop request or use another INGREQ STOP with another shutdown type

#### *Finding the reason for stop problems*

If you have resources with a stop problem you can find the reason by checking the compound & observed statuses with INGLIST using the following list:

A compound status of AWAITING means the resource is waiting for any supporting resources to reach the specified condition. Check with INGRELS if relationships conditions are unsatisfied. If the relationship is active, votes are propagated to the supporting resources that hopefully will put the supporting resources into the specified condition.

If the relationship is passive, the resource is waiting for any supporting resources but votes are not propagated, so it could wait forever.

A compound status of DENIED means automation is unable to proceed because either the automation flag has been turned off or the automation manager hold flag for the resource is set to YES.

## INGWHY - What does it provide?

- Initial analysis of a given situation
- Initial analysis of:
  - Why is a resource in a given desired status
  - Why is a resource in a given compound status
- Tries to get the operator an initial idea of where to look at
- If no sufficient help is given, detailed manual analysis must be performed
- Multiple reasons may exist why a resource does not reach its desired status
- Shows potential reasons

### *INGWHY - What does it do?*

The INGWHY command queries the automation manager and the various agents for the reasons why a resource is in the displayed status or why it cannot be brought into the desired status.

The INGWHY command tries to get the operator an initial idea of where to look at.

If no sufficient help is given, detailed manual analysis must be performed. There are situations where no reason can be found. If that situation appears, it is recommended to use the existing product commands to analyze manually.

Multiple reasons may exist why a resource does not reach its desired status.

The INGWHY command shows potential reasons.

## INGWHY - What is the cause?

- PROBLEM states
  - Agent state: STOPPED, BROKEN, CTLDOWN, STARTED2, STUCK,...
- MTR health states
- Automation flags (Mgr & agent)
- Server & Move APGs
  - Changed preference values, satisfactory target, availability target
  - Avoided or excluded Systems
- Active RUNMODEs and qualified resources
- Start / Stop Processes controlled by 'Pacing Gates' or 'Rolling Recycle'
- Triggers (TRG)
- Start/stop requests from other sources
  - Operator, External, Schedule, Autooperator
- Suspended resources
- A dependency to a 'misbehaving' resource ....
- Conflicting votes

### *INGWHY - What is the cause?*

If a resource is not in a status as you expect, you can start the analysis by using the INGWHY command. INGWHY is an operator support function that helps operators to initially analyze situations to find why automation took a resource into the displayed status, or why automation was unable to take a resource into the desired status.

INGWHY analyzes PROBLEM states:

- Agent state: STOPPED, BROKEN, CTLDOWN, STARTED2, STUCK,...
- MTR health states

INGWHY looks at:

- Automation flags (Mgr & Agent)
- Server & Move APGs
  - Changed preference values, satisfactory target, availability target
  - Avoided or excluded Systems
- Active RUNMODEs and qualified resources
- Start / Stop Processes controlled by 'Pacing Gates' or 'Rolling Recycle'
- Triggers (TRG)
- Start/stop requests from other sources
  - Operator, External, Schedule, Autooperator
- Suspended resources
- A dependency to a 'misbehaving' resource ....
- Conflicting votes

## INGWHY – Does not...

INGWHY gets its information mainly from the Automation Manager. Some data (like Agent Automation Flags) is received from the Automation Agent

INGWHY does not ...

- ... issue commands to z/OS,
- ... search any log,
- ... search through history data,
- ... analyze a cause in situations, where resources just seem to be slow at startup or shutdown
  - (i.e.: Compound State is ,INAUTO'),
- ... analyze policy configuration error

If INGWHY cannot identify a causing resource, it will indicate this and ask users to press F1=HELP

Further instructions are given at the end of the panel help

*INGWHY – Does not...*

INGWHY gets its information mainly from the automation manager.  
Some data (like Agent Automation Flags) is received from the automation agent.

INGWHY does not ...

- ... issue commands to z/OS,
- ... search any log,
- ... search through history data,
- ... analyze a cause in situations, where resources just seem to be slow at startup or shutdown
  - (i.e.: Compound State is ,INAUTO'),
- ... analyze policy configuration error.

If INGWHY cannot identify a causing resource, it will indicate this and ask users to press F1=HELP.  
Further instructions are given at the end of the panel help.

## Using INGWHY: The situation

```
INGKYAN2          SA z/OS - Command Dialogs          Line 1  of 17
Domain Id . : A0FDR  ----- INGWHY -----  Date . . : 08/31/18
Operator Id : INGC102 Sysplex = SYSPLEX1  Time . . : 21:29:41

Analyzed Resource: A01B/APL/MVSA      Job Name: A01B1
Status Compound: PROBLEM             Observed: HARDDOWN
Desired: AVAILABLE

SITUATION:          The situation
A01B/APL/MVSA is unavailable.

REASON 1 OF 2:      Desired Availability
The 'Desired Availability' of A01B/APL/MVSA is set to 'Always'.
A01B/APL/MVSA is started by default once its
dependencies are satisfied.
Owner: Paul Newman / Ext.12-3456

ACTION 1 OF 2:      Recommended actions
Refer to your company's rules in order to take the appropriate
action.
No action required.           are careful, but can be
                               customized

Command ===>          INGWHYSA(A0106900)
F1=Help   F2=End   F3=Return   F9=Refresh   F10=Previous   F11=Next   F6=Roll
                           F12=Retrieve

Press F11 for next reason
```

© Copyright IBM Corporation 2019

8-15

### Using INGWHY: The situation

In the header is the most important information about the resource that is being analyzed.

SITUATION: The current situation of the analyzed resource

REASON: Potential reason(s) why the analyzed resource is in its current situation.

ACTION: The action(s) that might be considered to change the situation.

If the situation is not caused by the analyzed resource 'A' itself, then another resource 'B' will be referenced in the reason description. There may be many other resources within the dependency path from resource 'A' to resource 'B' that do not influence the situation. These resources are not mentioned in the reason explanation. The phrase 'A has a dependency on B' is used, that reflects the analyzed and the causing resource only. That implies, that you might not find any relationship defined in the automation policy, that is defined on 'A' which refers to 'B'.

If there is more than one potential reason that causes the situation, you will find the reason header line showing 'REASON x OF y'. 'F10=Previous' and 'F11=Next' let you roll through the individual reasons. Accordingly, the related actions 'ACTION x OF y' are shown.

Actions need special considerations. Actions might differ from company to company or even from resource to resource, but INGWHY allows the automation administrators to adapt the proposed actions to their needs. A DSIPARM member INGWHYU is provided for user-defined actions.



**Hint:** If no reason can be identified, analysis data is written to the netlog as yellow text. This data is important to enhance the capabilities of this function. Please store this data for later use.

Use this web link to get further instructions how you can contribute to increasing the capabilities of this function: [http://ibm.biz/sazos\\_ingroup](http://ibm.biz/sazos_ingroup)

In this scenario, the situation is that A01B is unavailable but its desired status is available.

Reason 1 of 2 explains where the desired status comes from, it comes from the Desired Availability policy which is set to ALWAYS.

## Using INGWHY: Scroll to next reason

```

INGKYAN2          SA z/OS - Command Dialogs      Line 1      of 17
Domain Id . : AOFDA ----- INGWHY ----- Date . . . : 08/31/18
Operator Id : INGC102 Sysplex = SYSPLEX1 Time . . . : 21:30:24

Analyzed Resource: A01B/APL/MVSA           Job Name: A01B1
Status Compound: PROBLEM             Observed: HARDDOWN
Desired: AVAILABLE

SITUATION:          The situation
A01B/APL/MVSA is unavailable.          (repeat)

REASON 2 OF 2:          More: F10/F11
A01B/APL/MVSA is in a PROBLEM status and requires operator
intervention.
A01B/APL/MVSA is in the agent status 'STOPPED'.
Owner: Paul Newman / Ext.12-3456

ACTION 2 OF 2:          INGWHYSA(A0209500)
Refer to your company's rules in order to take the appropriate
action.
Clarify, whether the resource is allowed to be restarted.
Consider the following commands to apply to A01B/APL/MVSA:
- EXPLAIN
- SETSTATE

```

- EXPLAIN can explain the status
- Use SETSTATE to set agent status to AUTODOWN after checking with Paul

© Copyright IBM Corporation 2019

8-16

*Using INGWHY: Scroll to next reason*

After pressing F11 you see the next reason:

It can not be started as it has been stopped outside of SA. You could contact the owner or search the log who stopped it.

If the stop was illegal you can follow the actions:

Use SETSTATE to set agent status to AUTODOWN after checking with Paul.

# Lesson 2 Collecting information



## Lesson 2. Collecting information



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson teaches what is recommended for collecting information:

- INGLKUP resource REQ=COLLECT
- NVINFO
- NETLOG and syslog
- Joblogs

## NVINFO collects NetView diagnostic information

- LISTVAR environment variables
- RESOURCE command to show system resources
- LISTA to show NetView Dataset information
- Task information - TASKMON ALL,LIST STATUS=TASKS....
- TASKUTIL to show Task information
- LIST OVERRIDE
- Security Information - LIST SECOPTS
- CNMSTYLE information as found in BR CNMSTYLE
- Provides an option to take a dump

### *NVINFO collects NetView diagnostic information*

The NVINFO exec is a single command that can collect diagnostic information as well as dumping the NetView trace dataspace. NVINFO collects the following NetView diagnostic information and write it to the netlog or a dataset:

- LISTVAR environment variables
- RESOURCE command to show system resources
- LISTA to show NetViewdata set information
- Task information -TASKMON ALL,LIST STATUS=TASKS....
- TASKUTIL to show Task information
- LIST OVERRIDE
- Security Information -LIST SECOPTS
- CNMSTYLE information as found in BR CNMSTYLE
- Provides an option to take a dump

NVINFO will write output to the netlog (BR NETLOGA) by default. There is also a DUMP option on the NVINFO exec to take a dump of the address space.

# NETLOG, syslog and joblogs

- NETLOG is an automation log
  - Message **CNM493I** shows match in an automation table
  - Messages from any source can be copied to netlog
  - Output from REXX, commands, pipes...
  - Command responses
  - Trace and debug messages
  - Can be archived
- Syslog contains everything unless removed by MRT
  - Use CANZLOG filters
- Joblogs
  - SA joblog automation
  - SDSF
  - Can be archived

© Copyright IBM Corporation 2019

8-19

## NETLOG, syslog and joblogs

### NETLOG is an automation log

- Message **CNM493I** shows match in an automation table
- Messages from any source can be copied to netlog
- Output from REXX, commands, pipes...
- Command responses
- Trace and debug messages
- Can be archived

### Syslog contains everything unless removed by MRT

- Use CANZLOG filters

### Joblogs

- SA joblog automation
- SDSF
- Can be archived

# INGLKUP resource REQ=COLLECT

REQ=COLLECT obtains diagnostic information:

- INGINFO, INGRELS, INGSCHED, DISPINFO, DISPTRG, INGVOTE, INGAMS, and INGHIST
- DISPMTR (for MTR resources only)
- DISPAPG (for APG resources only)
- INGLIST, INGSESS, DISPMTR, DISPSTAT if resource is unspecified
- Collect diagnostics and write to data set or log

```
* INGLKUP F01B REQ=COLLECT
! ****
! * INGINFO F01B/APL/MVSA ALL OUTMODE=LINE TARGET=MVSA
! ****
! Resource      : F01B/APL/MVSA
! Description    : # AAAZSSEM WTO definitions only
! Inform List   : SDF
! Status...
!   Observed     : HARDDOWN
!   Desired      : AVAILABLE
!   Automation   : IDLE
!   Startability : NO
!   Compound     : PROBLEM      Last changed : 2018-08-31 21:20:41
!   Health Status: N/A
! Dependencies...
!   PreStart     : Satisfied
!   Start        : Satisfied
```

© Copyright IBM Corporation 2019

8-20

## INGLKUP resource REQ=COLLECT

INGLKUP can obtain diagnostic information about a particular resource if you want to perform debugging:

- INGINFO, INGRELS, INGSCHED, DISPINFO, DISPTRG, INGVOTE, INGAMS, and INGHIST
- DISPMTR (for MTR resources only)
- DISPAPG (for APG resources only)
- INGLIST, INGSESS, DISPMTR, DISPSTAT if resource is unspecified

INGLKUP collects diagnostics and writes to data set or log.

Parameter resource specifies the name of the resource to be processed. The format is name{/type{/system}}. Wildcard is supported. If more than one resource matches the wildcard specification, a selection panel is displayed.

# Lesson 3 Using trace and debug



## Lesson 3. Using trace and debug



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson covers how to use trace and debug

- Automation manager State Trace Table
- Trace services for the automation manager and agent
- AOCTRACE

## Automation manager State Trace Table

The event handler trace back table is written to SYSLOG

```
F AM,$TRACESTATE
HSAM1399I EVENT HANDLER TRACE BACK TABLE:
HSAM1399I 24319393341 Sta(1 ,1 ) Evt=2 Act=2 , Joinself_Complete
HSAM1399I AM INGXSGSA MVSA$$$$$1
HSAM1399I 24319393341 Sta(1 ,1 ) Evt=5 Act=6 , AMMode_Lock_Posted
HSAM1399I 24319393344 Sta(1 ,6 ) Evt=7 Act=8 , Set_PAM_Slctd_resp
HSAM1399I , ReadCfgOp=INITIAL
HSAM1399I 24319393481 Sta(6 ,6 ) Evt=0 Act=0 , Ignore_event
HSAM1399I 24319393481 Sta(6 ,6 ) Evt=0 Act=0 , Ignore_event
HSAM1399I 24319393591 Sta(6 ,6 ) Evt=9 Act=10, Read_Config_resp
HSAM1399I , ConnSys=MVSA
HSAM1399I , RegComms=MVSA
HSAM1399I , RegComms=MVSA$$$$$1
HSAM1399I , AcfOp=Load
HSAM1399I 24319393893 Sta(6 ,7 ) Evt=10 Act=11, Set_PAM_Pendg_resp
HSAM1399I 24319393893 Sta(7 ,7 ) Evt=11 Act=12, Override_Updt_resp
HSAM1399I , ApplyCfgOp=CONTINUE
HSAM1399I 24319394384 Sta(7 ,7 ) Evt=12 Act=13, Apply_Config_resp
HSAM1399I 24319394385 Sta(7 ,10) Evt=15 Act=20, Set_PAM_ready_resp
HSAM1399I , StartProcessing
HSAM1399I 24319400982 Sta(10,10) Evt=22 Act=37, ACF_Load_response
HSAM1399I 24319401033 Sta(10,10) Evt=47 Act=56, AA_Status_Posted
```

© Copyright IBM Corporation 2019

8-22

Automation manager State Trace Table

The event handler trace back table is written to the SYSLOG.

## Trace services for the automation manager and agent

Uses the MVS Component Trace Facility

1. Trace data set must be allocated using sample HSACTWR

2. Start external writer:

```
TRACE CT,WTRSTART=HSACTWR
```

3. Activate the trace:

```
TRACE CT,ON,COMP=HSAAM
```

4. Respond WTOR with trace options

```
xx,WTR=HSACTWR,OPTIONS=(ALL|set_of_options),  
END
```

5. Deactivate the trace with the command:

```
TRACE CT,OFF,COMP=HSAAM
```

6. Stop external writer:

```
TRACE CT,WTRSTOP=HSACTWR
```

7. Analyze trace with IPCS or print using sample HSAJPTRC

© Copyright IBM Corporation 2019

8-23

*Trace services for the automation manager and agent*

Uses the MVS Component Trace Facility

- Trace data set must be allocated using sample HSACTWR
- Start external writer:TRACE CT,WTRSTART=HSACTWR
- Activate the trace:TRACE CT,ON,COMP=HSAAM
- Respond WTOR with trace optionsxx,WTR=HSACTWR,OPTIONS=(ALL|set\_of\_options),END
- Deactivate the trace with the command:TRACE CT,OFF,COMP=HSAAM
- Stop external writer:TRACE CT,WTRSTOP=HSACTWR
- Analyze trace with IPCS or print using sample HSAJPTRC

## AOCTRACE dialog

```
AOFKAANL          SA z/OS - Command Dialogs          Line 1  of 3
Domain Id . : AOFDA ----- AOCTRACE ----- Date . . : 09/01/18
Operator Id : INGC102                         Time . . : 19:27:40

Enter ADD in the Command line to create an entry
Cmd: A Add   C Change   D Turn off Debug   R Reset   T Turn off Trace

Cmd CLIST      System      Dbg Level      T Subroutines/Messages being traced
-----      -----
*GLOBAL* MVSA
- INGREQ MVSA      Y
- INGRYR00 MVSA      Y MAX
```

```
AOFKAAND          SA z/OS - Command Dialogs          Date . . : 09/01/18
Domain Id . : AOFDA ----- AOCTRACE ----- Time . . : 19:07:59
Operator Id : INGC102

Specify or revise the following parameters:

Clist name
debug mode      OFF      (ON or OFF)
debug detail level      (NORM, MAX, /AM, /SLAVE)
REXX trace option      -      (A, R, I, C, E, F, L, O, N)

Subroutines to be traced: (* for all)
_____
_____
_____
- or -
Message id _____
```

© Copyright IBM Corporation 2019

8-24

### AOCTRACE dialog

The AOCTRACE command turns the SA z/OS system operations debugging feature on or off. AOCTRACE allows you to activate:

1. Debugging globally
2. Debugging for specific REXX script
3. Debugging at a certain level of detail
4. Tracing for a particular message
5. REXX trace for a specific REXX script

## AOCTRACE for automation debugging

Enable or disable the automation debugging facility

- Debugging globally
  - AOCTRACE ON
- Debugging for specific REXX script. Must use real name, not synonym. Implement same debugging for your REXX
  - AOCTRACE rexx\_exec ON
- Debugging at a certain level of detail
  - AOCTRACE rexx\_exec ON level  
Level=NORM | MAX | /AM | /SLAVE
- Tracing for a particular message
  - AOCTRACE MSG/id ON
- REXX trace for a specific REXX script
  - AOCTRACE rexx\_exec Trace

### AOCTRACE for automation debugging

Enable or disable the automation debugging facility using:

- Debugging globally:  
AOCTRACE ON
- Debugging for specific REXX script. Must use real name, not synonym. Implement same debugging for your REXX:  
AOCTRACE rexx\_exec ON
- Debugging at a certain level of detail:  
AOCTRACE rexx\_exec ON level (Level=NORM | MAX | /AM | /SLAVE)
- Tracing for a particular message:  
AOCTRACE MSG/id ON
- REXX trace for a specific REXX script:  
AOCTRACE rexx\_exec Trace

# Lesson 4 Solving message automation problems



## Lesson 4. Solving problems with messages



Course materials may not be reproduced in whole or in part without the prior written permission of IBM.

This lesson teaches solving problems with message automation.

## NetView message types, AT, MRT, CRT

Messages in NetView are of different type (HDRMTYPE), indicated in netlog in front of message, and can come from various sources:

- E Unsolicited single line messages come in through the SSI (" for MLWTOs)
- > WTOR
- Command responses are solicited messages coming in through your autotask's EMCS
- \* Command echo and output of REXX say, to avoid them use pipe logto netlog
- C NetView commands
- - NetView messages (Z)
- ' MLWTO message originating from NetView itself
- " MLWTO message
- ! NetView message from an immediate command processor.
- (HEX'10') Management services units (MSUs) are data structures that carry alert major vectors and other management-services data for network management
- ...
- All messages in NetView are processed by the automation table(s) (AT)
- Message Revision Table (MRT) can intercept MVS messages before they are displayed, logged, automated, or routed through your sysplex
- The command revision table (CRT) can intercept MVS commands before they are processed

© Copyright IBM Corporation 2019

8-27

### NetView message types, AT, MRT, CRT

Messages in NetView are of different type (HDRMTYPE), indicated in netlog in front of message, and can come from various sources:

- E Unsolicited single line messages coming in through the SSI (" for MLWTOs).
- > WTOR.
- Command responses are solicited messages coming in through your autotask's EMCS.
- \* Command echo and output of REXX say. To avoid them use pipe logto netlog.
- C NetView commands.
- - NetView messages (Z).
- ' MLWTO message originating from NetView itself.
- " MLWTO message.
- ! NetView message from an immediate command processor.
- (HEX'10') Management services units (MSUs) are data structures that carry alert major vectors and other management-services data for network management.
- ...

All messages in NetView are processed by the automation table(s) (AT).

Message Revision Table (MRT) can intercept MVS messages before they are displayed, logged, automated, or routed through your sysplex.

The command revision table (CRT) can intercept MVS commands before they are processed.

## Does message get to your task, operator, or REXX?

- WTO or WTOR must be sent to NetView.  
Indicated by bit 10 of the suppression flag (bytes 7 and 8)
  - MPF: AUTO(Y) or MRT REVISE("1" AUTOMATE)
  - z/OS Message Flood Automation can „stop“ flooding messages
  - Use z/OS message analysis program
  - Recommendation: Use generated MPF or MRT to not miss any messages
- CNM493I message in the network log indicates a match in a certain automation table and command execution routed to autotask(s)
  - ISSUEACT, ISSUECMD, and ISSUEREP are called to execute your commands defined in policy
  - Message DWO032E indicates that autotask was not logged on
- Message can be stopped by AT THRESHOLD condition
- Autotask could be waiting or suffering from queueing > Use TASKUTIL
- Command responses are usually solicited messages coming back to your autotask's EMCS and waiting REXX, but if not:
  - Unsolicited messages come in through the SSI and need to be routed to your autotask

*Does message get to your task, operator, or REXX?*

In order to be automated, a WTO or WTOR must be sent to NetView which is indicated by bit 10 of the suppression flag (bytes 7, 8).

To send a WTO or WTOR to NetView, specify AUTO(Y) for this message ID in MPF or specify REVISE("1" AUTOMATE) for this message ID in MRT.



**Note:** The recommendation is to use the generated MPF or MRT in order to not miss any messages.

Messages can also be missing as z/OS Message Flood Automation can „stop“ flooding messages, however the MRT can override z/OS Message Flood Automation.

Use the z/OS message analysis program to analyze whether messages are sent to NetView.

The CNM493I message in the network log indicates a match in a certain automation table and what command are routed to autotask(s):

- ISSUEACT, ISSUECMD, and ISSUEREP are called to execute your commands from policy.
- Message DWO032E indicates that autotask was not logged on.

## 8 Troubleshooting

*Does message get to your task, operator, or REXX?*

Other reasons for messages not being automated:

- Message can be stopped by AT THRESHOLD condition.
- Autotask could be waiting or suffering from queuing. You can use the TASKUTIL command for analysis.
- Command responses are usually solicited messages coming back to your autotask's EMCS and waiting REXX, but if they are unsolicited messages they come in through the SSI and need to be routed to your autotask which could be done by dynamically creating an automation table.

# CNM493I shows match in an automation table

```
STATMON.BROWSE      ACTP   NETWORK LOG FOR 08/31/18 (18243) COLS 008 085 20:15 A
                                         DOMAIN: AOFDA   SCROLL ==> CSR
--1-----+---2-----+---3-----+---4-----+---5-----+---6-----+---7-----+---8-----+
AUTWRK03 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001078 : MSGID=AAA008I :
AUTWRK03 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001082 : MSGID=AAA011I :
AUTWRK02 AOFDA 16:00:00 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK03 AOFDA 16:00:00 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK20 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001078 : MSGID=AAA008I :
AUTWRK20 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001082 : MSGID=AAA011I :
AUTWRK10 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001078 : MSGID=AAA008I :
AUTWRK10 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001082 : MSGID=AAA011I :
AUTWRK20 AOFDA 16:00:00 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK10 AOFDA 16:00:00 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK10 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001078 : MSGID=AAA008I :
AUTWRK01 AOFDA 16:00:00 - CNM493I INGMSG02 : 00001082 : MSGID=AAA011I :
AUTWRK01 AOFDA 16:00:00 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK04 AOFDA 16:00:41 - CNM493I INGMSG02 : 00001078 : MSGID=AAA008I :
AUTWRK04 AOFDA 16:00:41 - CNM493I INGMSG02 : 00001082 : MSGID=AAA011I :
AUTWRK04 AOFDA 16:00:41 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
AUTWRK18 AOFDA 16:10:56 - CNM493I INGMSG02 : 00002106 : MSGID=HSAM1010I :
AUTWRK18 AOFDA 16:10:56 - CNM493I INGMSG02 : 00000121 : MSGID=IEF450I :
AUTLOG AOFDA 17:01:36 - CNM493I CNMSDCA : #0000023 : MSGID=DSI200I :
AUTLOG AOFDA 17:01:36 - CNM493I INGMSGSA : #0001146 : MSGID=DSI200I :
INGC102 AOFDA 19:38:48 - CNM493I INGMSGSA : #0001097 : MSGID=DSI020I :
AUTLOG AOFDA 19:39:33 - CNM493I INGMSG02 : 00002090 : MSGID=HSAM1000I :
AUTLOG AOFDA 19:39:43 - CNM493I INGMSG02 : 00002119 : MSGID=HSAM1308I :
AUTWRK19 AOFDA 19:41:29 - CNM493I INGMSG02 : 00000069 : MSGID=IEF403I :
AUTWRK16 AOFDA 19:41:30 - CNM493I INGMSG02 : 00000069 : MSGID=IEF403I :
AUTWRK19 AOFDA 19:41:30 - CNM493I INGMSG02 : 00000092 : MSGID=IEF404I :
```

Autotask name

Domain

AT name

AT sequence number

Message ID

© Copyright IBM Corporation 2019

8-29

*CNM493I shows match in an automation table*

You can see the

- Autotask name under which the AT is processed for this message
- Domain
- AT name
- AT sequence number
- Message ID

# CNM493I lists commands executed

```
STATMON.BROWSE      ACTP   NETWORK LOG FOR 08/31/18 (18243) COLS 037 114  19:42 F
                     DOMAIN: AOFDA  SCROLL ==> CSR
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
--4----+---5---+---6---+---7---+---8---+---9---+---10---+---11---+
CNM493I INGMSG02 : 00001078 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA011I   : TERMMSG FINAL=YES
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=A0E
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=B0E
CNM493I INGMSG02 : 00001078 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA011I   : TERMMSG FINAL=YES
CNM493I INGMSG02 : 00001078 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA011I   : TERMMSG FINAL=YES
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=A02
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=A0E
CNM493I INGMSG02 : 00001078 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA011I   : TERMMSG FINAL=YES
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=A0E
CNM493I INGMSG02 : 00001078 : MSGID=AAA008I   : TERMMSG FINAL=NO
CNM493I INGMSG02 : 00001082 : MSGID=AAA011I   : TERMMSG FINAL=YES
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=D01
CNM493I INGMSG02 : 00002106 : MSGID=HSAM1010I  : TERMMSG ABEND=YES,FINAL=YES
CNM493I INGMSG02 : 00000121 : MSGID=IEF450I   : TERMMSG JOBNAME=AUTOMGR,CODE1
CNM493I CNMSDCA : #00000023 : MSGID=DSI200I  : CNME8251
CNM493I INGMSGSA : #00011146 : MSGID=DSI200I  : A0FR1ILL
CNM493I INGMSGSA : #0001097 : MSGID=DSI202I   : A0FR1ILL
CNM493I INGMSG02 : 00002090 : MSGID=HSAM1000I  : ACTIVMSG UP=NO
CNM493I INGMSG02 : 00002119 : MSGID=HSAM1308I  : ACTIVMSG UP=YES
CNM493I INGMSG02 : 00000069 : MSGID=IEF403I   : ACTIVMSG JOBNAME=IRRDPTAB
CNM493I INGMSG02 : 00000069 : MSGID=IEF403I   : ACTIVMSG JOBNAME=BLSJPRMI
CNM493I INGMSG02 : 00000092 : MSGID=IEF404I   : TERMMSG FINAL=YES,JBNAME=IRR
```

AT name

AT sequence number

Message ID

Command executed with parameters

© Copyright IBM Corporation 2019

8-30

*CNM493I lists commands executed*

You can see the

- AT name
- AT sequence number
- Message ID
- Command executed with parameters

## Tracing and simulating messages

- Tracing for a particular message
  - AOCTRACE MSG/id ON
- Simulating message by generating WTO
  - Caution, SA checks jobname (can be simulated too) and does ASSIGN by jobname to work operator autotask
- Tracing automation table with TRACE action
- Testing an Automation Table using recorded messages or automation internal function requests (AIFR)
  - Start recording: AUTOTEST RECORD=TESTRECS
  - Stop recording: AUTOTEST RECORD=OFF
  - Replay using test AT: AUTOTEST SOURCE=TESTRECS, REPORT=TESTRPT
  - Report: AUTOCNT REPORT=BOTH, FILE=TESTTBL, TEST
- Use test application and/or test system for the real scenario

### Tracing and simulating messages

The following is available for tracing and simulating messages:

- Tracing for a particular message:  
AOCTRACE MSG/id ON
- Simulating messages can be done by generating WTOs. Caution, SA checks jobname (can be simulated too) and does ASSIGN by jobname to work operator autotask.
- Tracing automation table with TRACE action.
- Testing an Automation Table using recorded messages or automation internal function requests (AIFR):
  - Start recording: AUTOTEST RECORD=TESTRECS
  - Stop recording: AUTOTEST RECORD=OFF
  - Replay using test AT: AUTOTEST SOURCE=TESTRECS, REPORT=TESTRPT
  - Report: AUTOCNT REPORT=BOTH, FILE=TESTTBL, TEST
- Use test application and/or test system for the real scenario.

# Finding SA message automation

Messages and their automation are visible in netlog:

- Status messages (any source) (can also issue commands or replies)
  - Active or up message: ACTIVMSG UP=YES | NO
  - Termination message: TERMMMSG FINAL=NO ABEND=... BREAK=... CODEx=
  - Final termination message: TERMMMSG FINAL=YES ABEND=... BREAK=... CODEx=
  - Halted message: HALTMSG
- Captured messages show up in DISPINFO, DISPMTR, DISPSYS, SDF
  - Capturing: AOFCPMMSG
  - Deleting with DOM: AOFCPMMSG DOM=YES MSG=IKT008I
- Automated messages defined in MESSAGES/USER DATA policy
  - Issue command for job message: ISSUEACT MSGTYPE='HASP099'
  - Issue command for system msg: ISSUEACT SYSTEMMSG=YES
- Health state messages of MTR resources
  - Set health state: INGMON SPOOL,MSGTYPE=HASP050,STATUS=WARNING
- Builtin message automation uses one of the SA prefixes: AOF, ING, EVE...
- WTOR messages
  - Store WTOR: OUTREP

## Finding SA message automation

This chart can help you to find SA message automation. Messages and their automation are visible in the netlog.

*Status messages of any source which can also issue commands or replies are one of:*

- Active or up message:  
ACTIVMSG UP=YES | NO
- Termination message:  
TERMMMSG FINAL=NO ABEND=... BREAK=... CODEx=
- Final termination message:  
TERMMMSG FINAL=YES ABEND=... BREAK=... CODEx=
- Halted message:  
HALTMSG

- *Captured messages* show up in DISPINFO, DISPMTR, DISPSYS, SDF panels. Capturing is done by AOFCPMSG. Deleting is done with DOM:

AOFCPMSG DOM=YES MSG=IKT008I

- *Automated messages* which are defined in MESSAGES/USER DATA policy
  - Issue command for a message issued by a job:

ISSUEACT MSGTYPE='HASP099'

- Issue command for system message:

ISSUEACT SYSTEMMSG=YES

- *Health state messages* of MTR resources
  - Set health state:

INGMON SPOOL,MSGTYPE=HASP050,STATUS=WARNING

- Builtin message automation uses one of the SA prefixes: AOF, ING, EVE...

- *WTOR messages* are stored with OUTREP

# List Automation Table commands

```

SA V4R1 Education          Tivoli NetView    AOFDA INGC102 09/02/18 13:30:54
! AOFDA IF MSGID = 'DFHLG0740' THEN EXEC(CMD('TERMMMSG FINAL=NO,ABEND=YES')) R
! AOFDA IF MSGID = 'DFHNC01011' THEN EXEC(CMD('ACTIVMSG UP=NO')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHNC01021' THEN EXEC(CMD('ACTIVMSG UP=YES')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHNC01111' THEN EXEC(CMD('TERMMMSG FINAL=NO')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHNC01121' THEN EXEC(CMD('TERMMMSG FINAL=YES')) ROUTE(ONE %A
! AOFDA EXEC(CMD('TERMMMSG ABEND=YES,CODE1=' CODE1 ',CODE2=' CODE2))
! AOFDA IF MSGID = 'DFHRM01301' THEN EXEC(CMD('TERMMMSG FINAL=NO')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHRM01341' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHRM01361' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHRM01441' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHRM04011' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHSI15021' THEN EXEC(CMD('AOFCPMMSG')) ROUTE(ONE %AOFOPGS
! AOFDA XEC(CMD('ACTIVMSG UP=YES')) ROUTE(ONE %AOFOPGSSOPER%));
! AOFDA IF MSGID = 'DFHSR06031' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHSR06051' THEN EXEC(CMD('TERMMMSG FINAL=NO,BREAK=YES')) R
! AOFDA IF MSGID = 'DFHTM17971' THEN EXEC(CMD('TERMMMSG FINAL=NO,ABEND=YES')) R
! AOFDA IF MSGID = 'DFHXQ01011' THEN EXEC(CMD('ACTIVMSG UP=NO')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHXQ01021' THEN EXEC(CMD('ACTIVMSG UP=YES')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHXQ01111' THEN EXEC(CMD('TERMMMSG FINAL=NO')) ROUTE(ONE %A
! AOFDA IF MSGID = 'DFHXQ01121' THEN EXEC(CMD('TERMMMSG FINAL=YES')) ROUTE(ONE %A
! AOFDA EXEC(CMD('TERMMMSG ABEND=YES,CODE1=' CODE1 ',CODE2=' CODE2))
! AOFDA IF IFRAUWF1(6) = '1' THEN EXEC(CMD('OUTREP'))ROUTE(ONE %AOFOPSYOPER)
! AOFDA IF MSGID = 'DSNB3091' & TEXT = . 'FAILURE' . THEN EXEC(CMD('HALTMSG
! AOFDA EXEC(CMD('ISSUEACT AUTOTYP=RECOVERY 'CODE))ROUTE(ONE %AOFOPGSSOPER%))
! AOFDA EXEC(CMD('ISSUEACT AUTOTYP=RECOVERY 'CODE))ROUTE(ONE %AOFOPGSSOPER%));
! AOFDA EXEC(CMD('ISSUEACT AUTOTYP=RECOVERY 'CODE))ROUTE(ONE %AOFOPGSSOPER%))
! AOFDA EXEC(CMD('ISSUEACT AUTOTYP=RECOVERY 'CODE))ROUTE(ONE %AOFOPGSSOPER%))
???
PIPE NETVIEW (NOPANEL) BR INGMMSG01 ! SEP ! CHOP 72 ! LOCATE /CMD(/ ! STRIP BOTH!
CHOP 68 ! CONS■

```

© Copyright IBM Corporation 2019

8-33

## List Automation Table commands

Here is a pipe that can list Automation Table commands:

```

pipe netview (nopanel) br ingmsg01 ! sep ! chop 72 ! joincont not trailing
;/! locate /cmd(/ ! strip both ! chop 68 ! cons

```

The pipe uses the *netview* stage to browse Automation Table *ingmsg01* without displaying it in a panel. The output lines are separated with the *sep* stage and chopped at column 72.

Then lines that do not have a trailing ; are joined. This puts an entire AT statement into 1 line.

The *locate* stage filters all AT statements that contain the string cmd, which means a command is called.

The *strip both* stage strips both leading and trailing blanks.

To make it fit on a 80 column console, all lines are chopped at column 68.

The *cons* stage writes all of its input messages to the console.

# INGMSGs to display exceptional messages

```
INGKYMSG          SA z/OS - Command Dialogs          Line  81   of 497
Domain Id . : AOFDA  ----- INGMSGs -----      Date . . : 09/03/18
Operator Id : INGC102    System = MVSA           Time . . : 22:14:25

CMD: D Delete     F INGINFO/DISPSYS

CMD Timestamp      S Message
----- -----
HZA0002E CHECK(IBMDAE,DAE_SUPPRESSING):
ADYH005E The DAE SVC_DUMP suppression option is expected to be
SUPPRESSALL but is SUPPRESS
— 2018-08-31 01:37:42 C SYSTEM=MVSA RESOURCE=HZSPROC/APL/MVSA
HZA0003E CHECK(IBMRACF,RACF_SENSITIVE_RESOURCES):
IRRH204E The RACF_SENSITIVE_RESOURCES check has found one or
more potential errors in the security controls on this system.
— 2018-08-31 01:42:29 C SYSTEM=MVSA RESOURCE=HZSPROC/APL/MVSA
HZA0003E CHECK(IBMVSM,VSM_CSA_THRESHOLD):
IGVH104I Common Storage Tracker is Inactive
— 2018-08-31 01:42:50 C SYSTEM=MVSA RESOURCE=FTP/APL/MVSA QUAL=SUBSYSTEM/FTP
AOF577E 01:42:50 : RECOVERY FOR SUBSYSTEM FTP (JOB FTPSERVE) HALTED ...
    CRITICAL THRESHOLD EXCEEDED
— 2018-08-31 01:47:29 C SYSTEM=MVSA RESOURCE=HZSPROC/APL/MVSA
HZA0003E CHECK(IBMVSM,VSM_CSA_THRESHOLD):
IGVH104I Common Storage Tracker is Inactive
— 2018-08-31 01:52:29 U SYSTEM=MVSA RESOURCE=HZSPROC/APL/MVSA
HZA0002E CHECK(IBMVSM,VSM_SQA_THRESHOLD):
IGVH104I Common Storage Tracker is Inactive
— 2018-08-31 01:52:29 C SYSTEM=MVSA RESOURCE=HZSPROC/APL/MVSA

— .
An exceptional message is a message whose severity is either Unusual, Important, or Critical.
    Messages can be sorted, filtered by resource and deleted
```

© Copyright IBM Corporation 2019

8-34

## INGMSGs to display exceptional messages

The INGMSGs command displays all important (that is, exceptional) messages that currently exist for a given system. The command is also used to delete exceptional messages.

An exceptional message is a message whose severity is either Unusual, Important, or Critical. Messages can be sorted, filtered by resource and deleted.

## Finding status messages

HASP085	Description	JES2 has terminated	<p>Status messages can not be seen in DISPINFO or DISPACF, only in the policy report or the Automation Table</p>	
	Defined for MVS Components	+SA PREDEFINED MSGS		
	<i>SA Predefined Data</i>			
	AT ignore message id characters	LEADING		
	Message component	JES2		
HASP095	AT application status	TERMINATED	The related AT entries are already predefined, but can be overruled	
	Description	JES2 Catastrophic ABEND ERROR	<p>See also Status Message Report</p>	
	Defined for Applications	C JES2		
	Defined for MVS Components	JES2 +SA PREDEFINED MSGS		
	<i>SA Predefined Data</i>			
HASP098	Message component	JES2	<p>See also Status Message Report</p>	
	AT override	IF MSGID = . 'HASP095' & TEXT = . 'OPHIC 'CODE1 . !='CODE2 . THEN EXEC(CMD('TERMMSG MSGTYPE="HASP095" CODE1='CODE1 'CODE2='CODE2) ROUTE(ONE %AOFOPGSSOPER%));		
	Description	Enter termination option		
	AT ignore message id characters	LEADING		
	Defined for Applications	C JES2 JES2		

© Copyright IBM Corporation 2019

8-35

### Finding status messages

Status messages can not be seen in DISPINFO or DISPACF panels, only in the policy report or the Automation Table.

The related AT entries are already predefined, but can be overruled by defining the same message ID.

See also the Status Message Report.

# DISPACF for JES2 messages

```

Command = ACF ENTRY=JES2,TYPE=*,REQ=DISP
SYSTEM = MVS AUTOMATION CONFIGURATION DISPLAY - ENTRY= JES2
-----  

TYPE IS HASP095
CODE      = (*,$PJ*,,"STOPPING")
CODE      = (ERROR*,$K02,,,"BREAKING")
CODE      = (ERROR*,$K03,,,"BREAKING")
CODE      = (ERROR*,$K08,,,"BREAKING")
CODE      = (ERROR*,$K15,,,"STOPPING")
CODE      = (ERROR*,$V01,,,"BREAKING")
CODE      = (ABEND*,SA22,,,"STOPPING")
CODE      = (*,*,,"ABENDING")  

TYPE IS HASP098
REPLY     = (PASS1,,,'DUMP')
REPLY     = (PASS2,,,'END')  

TYPE IS HASP099
CMD       = (,,,'MVS &SUBSCMDPFXP&SUBSJOB')  

TYPE IS HASP405
REPLY     = (,,,'Y')  

TYPE IS HASP420
REPLY     = (,,,'Y')  

TYPE IS HASP426
REPLY     = (COLD,,,'COLD,NOREQ')
REPLY     = (NORM,,,'WARM,NOREQ')  


```

Code processing for termination message HASP095 Catastrophic ABEND|ERROR

Reply dump for pass 1, end for pass 3

Execute command. Command prefix and jobname substituted

Reply Y

Replies for COLD and NORM starts

IF MSGID = . 'HASP095' & TEXT = . 'OPHIC ' CODE1 .'=' CODE2 .

## DISPACF for JES2 messages

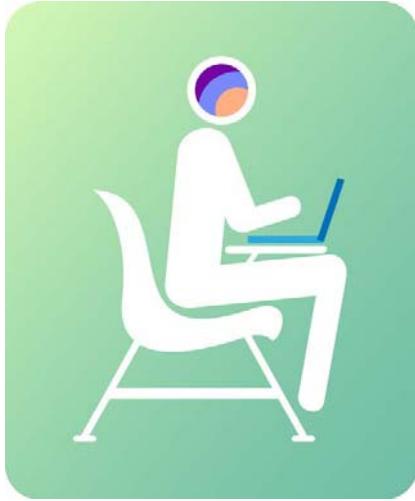
The DISPACF command displays resource information and automation policy settings for a specific entry or entry-type pair in the automation control file.

For JES2 you see:

- Code processing for termination message HASP095 Catastrophic ABEND|ERROR
- For message HASP095 the replies dump for pass 1 and end for pass 3
- For message HASP095 a command is executed. Command prefix and jobname are substituted
- Message HASP405 and HASP420 are replied with Y
- Replies for COLD and NORM starts are defined for message HASP426

Most of this information can also be displayed using the INGINFO / DISPIINFO commands. However DISPACF works for any entry in the automation control file.

## Student exercise



© Copyright IBM Corporation 2019

8-37

### *Student exercises*

Open your *Student Exercises* book and perform the exercises for this unit.

## Summary

---

Now that you completed this unit, you can perform the following tasks:

- Determine why resources won't start or stop
- Collecting information
- Using trace and debug
- Solving problems with messages

### *Summary*

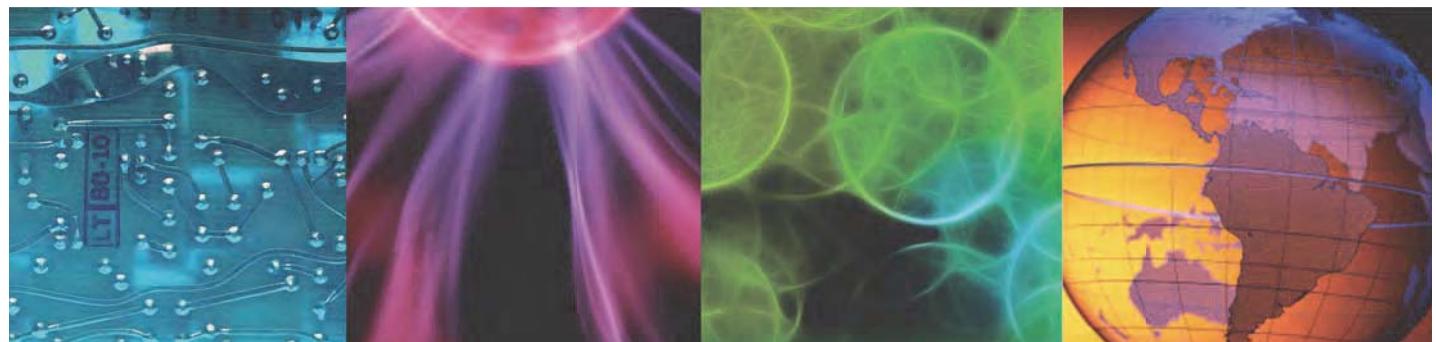
Now that you completed this unit, you can perform the following tasks:

- Determine why resources won't start or stop
- Collecting information
- Using trace and debug
- Solving problems with messages





SM927 1.1



[ibm.com/training](http://ibm.com/training)

Authorized  
**IBM | Training**