IBM Software Group

z/OS® V1R9 Communications Server

Overview - Other functions

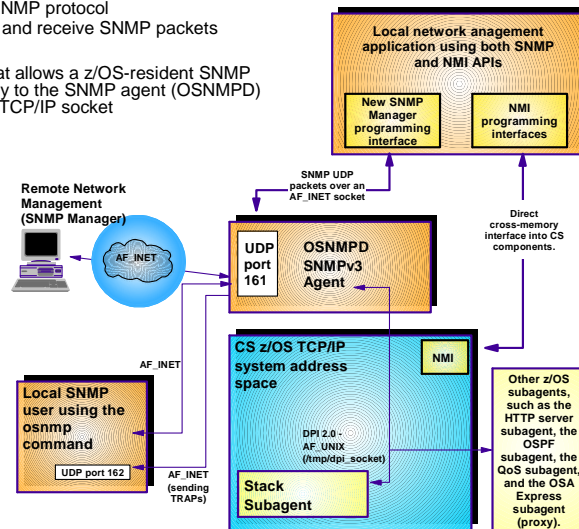This presentation discusses other enhancements for the z/OS V1R9 Communications Server.

Provide a programming interface for the SNMP manager

- The osnmp command is a user interface to the SNMP protocol
  - Uses a built-in set of functions to build, send, and receive SNMP packets

- z/OS V1R9 provides a programming interface that allows a z/OS-resident SNMP manager application to interface programmatically to the SNMP agent (OSNMPD) exchanging SNMP protocol packets over a local TCP/IP socket

- Local management applications that need frequent access to high-volume management data should use the NMI interfaces:
  - NMI provides low-overhead access to high-volume management data
  - Typically addresses performance monitoring applications

- Local management applications that need access to SNMP management data that is not provided with NMI can use this new API:
  - Resource monitoring
  - Availability management
  - Operations (using SNMP SET)

Local network anagement application using both SNMP and NMI APIs

New SNMP Manager programming interface

NMI programming interfaces

SNMP UDP packets over an AF_INET socket

Direct cross-memory interface into CS components.

Remote Network Management (SNMP Manager)

AF_INET

UDP port 161

OSNMPD SNMPv3 Agent

CS z/OS TCP/IP system address space

NMI

AF_INET

DPI 2.0 - AF_UNIX (/tmp/dpi_socket)

Other z/OS subagents, such as the HTTP server subagent, the OSPF subagent, the QoS subagent, and the OSA Express subagent (proxy).

Local SNMP user using the osnmp command

UDP port 162

AF_INET (sending TRAPs)

Stack Subagent

2

© 2007 IBM Corporation

SNMP is a standard's-based protocol for network management that is based upon the TCP/IP protocol (UDP). Part of the SNMP standards also includes a database structure specification for management objects called MIBs (Management Information Base). Use of SNMP is widespread, and work continues in the IETF mostly in the area of defining new MIBs. The SNMP protocol has been evolving for many years and has yielded several levels of the protocol, some of which were never adopted. Primarily, the supported protocol levels are: SNMPv1, SNMPv2c, and SNMPv3. SNMPv3 defines a user-based security model for SNMP, rather than the community-based model of SNMPv1 and SNMPv2c.

The key management entities of SNMP are:

- Agent - This entity implements the SNMP protocol stack and routes requests from managers to the appropriate subagents. Sometimes it is called the engine. It communicates with the subagents using the Distributed Protocol Interface (DPI) and with the Managers using the SNMP protocol. Subagents register their MIB objects with the Agent. For Comm Server, the agent is the osnmp daemon.

- Subagent - These entities are the providers of the MIB data. They communicate with the SNMP agents. In Comm Server, an example is the TCP/IP Subagent.

- Manager- These entities communicate using SNMP protocol requests with SNMP agents to retrieve management data. Comm Server only provides a command-line manager, the osnmp command. The manager function is typically part of management applications, such as Netview.

z/OS has not provided a formal, standards-based SNMP API for customer-driven manager applications. The osnmp command currently being shipped is, itself, a manager application, but customers have not been able to create their own application to manage SNMP, due to the absence of an external API.

z/OS Communications Server V1R9 provides a new application programming interface (API), the SNMP Manager API, for writing SNMP managers. Management application developers can use this API to build SNMP management functions to retrieve SNMP management data. This API provides the following functions:
- The ability to build and send SNMP messages for SNMPv1, SNMPv2, and SNMPv3 and receive responses
- The ability to decode the SNMP messages and retrieve the SNMP data

z/OS Communications Server also provides an extension of the SNMP Manager API, the SNMP Notification API, which leverages the functionality of the SNMP Manager API to send notifications to SNMP agents and/or SNMP Notification Receivers. Available notifications include Informs and both Version 1 and Version 2 Traps.

whatsnewOther.PPT

CICS® sockets enhancements
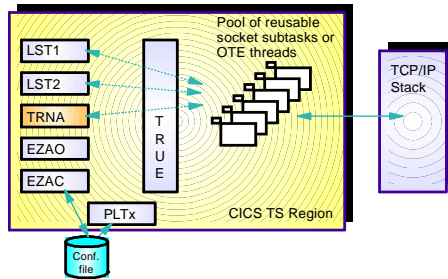
- The CICS Sockets listeners (two flavors) are part of CICS Sockets
  - A listener task accepts client connections and starts CICS transactions
- In prior releases if TCP/IP goes down, the listener task terminates
  - Operations personnel or automation need to restart them when TCP/IP comes back up
- In z/OS V1R9 the listener stays up when TCP/IP goes down

- In previous releases, the listener task ends if the socket table becomes full

- In z/OS V1R9, the listener stays up when this condition occurs
  - The listener stops accepting new connections for a period, and then resume normal processing

- The CICS Socket interface CICS/TS program load table (PLT) program, EZACIC20, supports an immediate shutdown method

The IP CICS Sockets listeners are part of CICS Sockets. A listener task accepts client connections and starts CICS transactions. In prior releases the IP CICS Sockets Listener task ends when the TCP/IP stack with which it has affinity is shutdown. A subsequent operation action is required to restart the listener task when the stack is reestablished.

In z/OS V1R9 the listener remains active when the TCP/IP stack with which it has affinity is recycled. The listener will try to reconnect to the stack.

The IP CICS Socket Listener is a concurrent listener executing as a CICS/TS task which creates socket descriptors. z/OS Unix System Services supports a configurable number of socket descriptors per process. The BPXPRMxx parameter, MAXFILEPROC, specifies the maximum number of descriptors for files, sockets, directories, and any other file system objects that a single process can have concurrently active or allocated. In prior releases when the socket table becomes full the listener ends. It is possible that MAXFILEPROC is incorrectly configured or is configured lower than the listener's NUMSOCK configuration. The operator was required to restart the listener transaction once the socket table full condition was resolved.

In z/OS V1R9 the listener will remain active when it reaches its maximum number of socket descriptors. The IP CICS Sockets Listener will not end when the EMFILE (24) errno condition is raised when accepting client connections. This problem can be avoided by configuring these values:

- Set the MAXFILEPROC value to be greater than or equal to the listener's NUMSOCK value or the listener's user ID's FILEPROCMAX value.

- Set the listener's NUMSOCK value to be less than or equal to the MAXFILEPROC value or the listener's user ID FILEPROCMAX value.
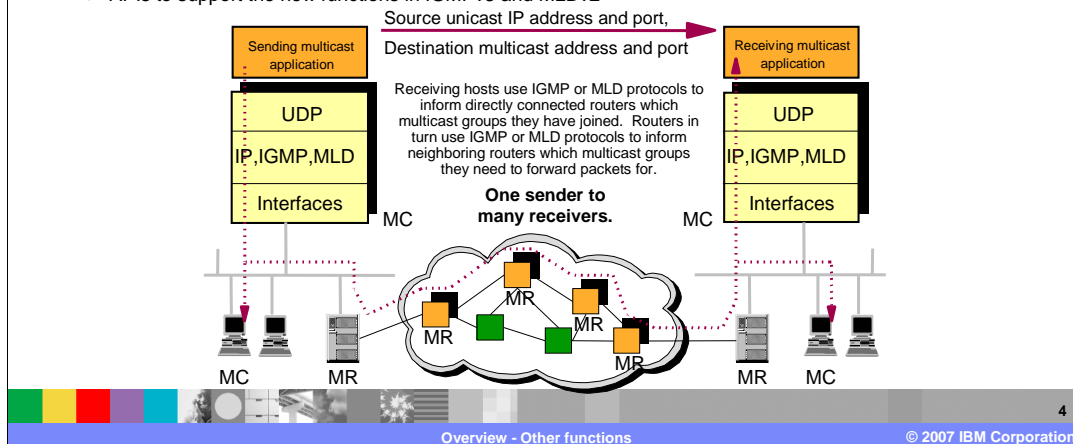
Before V1R9, the IP CICS Socket interface CICS/TS program load table (PLT) program, EZACIC20, only supported a deferred shutdown method. This forces CICS/TS shutdown to wait for all in-flight socket programs to end before the interface will shutdown. It was affected by blocking socket calls.

With z/OS V1R9, the IP CICS Socket interface supports two shutdown methods:

- A deferred shutdown allows all IP CICS Sockets tasks to end gracefully.

- An immediate shutdown directs all IP CICS Sockets tasks to be immediately terminated.

IGMPv3 and MLDv2 support

- Applications that want to receive multicasts, register with IGMP (for IPv4) or with MLD (for IPv6) using specific sockets API functions.
- IGMP and MLD both recently have been enhanced to support new capabilities:
  - Allow a receiving multicast application to specify a source IP address filter to limit which senders it is willing to receive multicast from on the specific multicast destination IP address (group)
- z/OS adds support for IGMPv3 (RFC 3376), MLDv2 (RFC 3810), and APIs (RFC 3678)
  - Protocol support as a host (not a multicast router)
  - APIs to support the new functions in IGMPv3 and MLDv2

Source unicast IP address and port,

Destination multicast address and port

Sending multicast application

Receiving multicast application

Receiving hosts use IGMP or MLD protocols to inform directly connected routers which multicast groups they have joined. Routers in turn use IGMP or MLD protocols to inform neighboring routers which multicast groups they need to forward packets for.

One sender to many receivers.

UDP

IP,IGMP,MLD

Interfaces

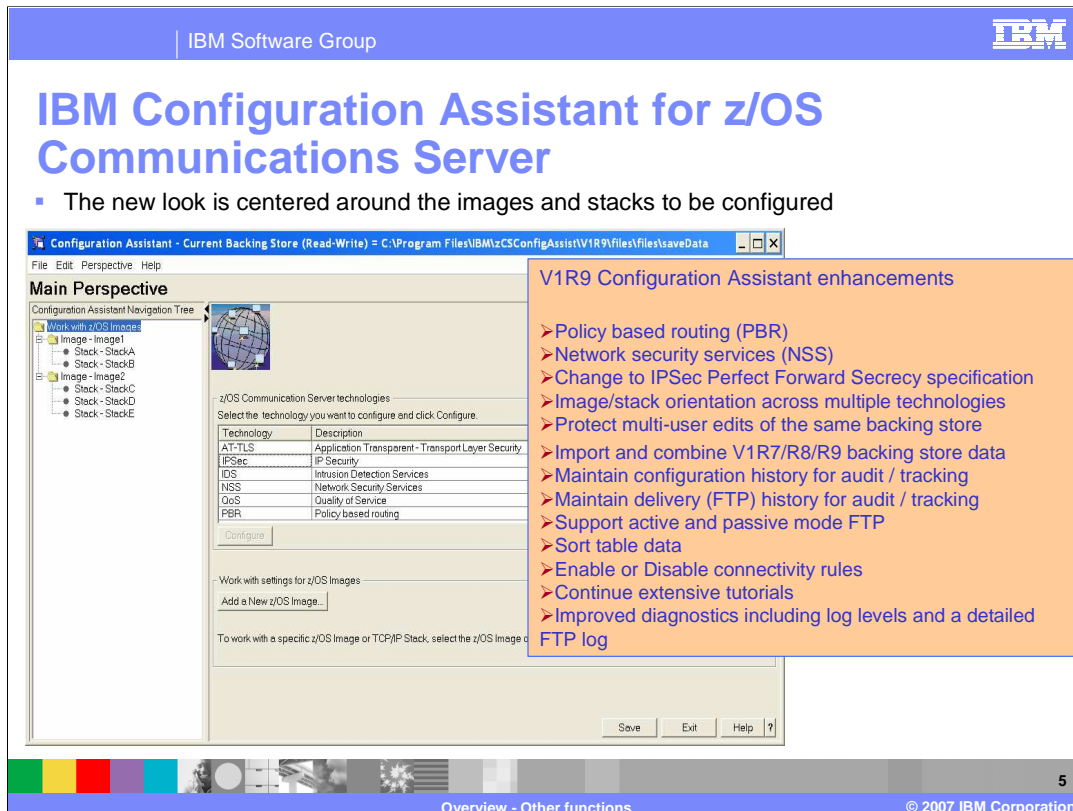MC

UDP

IP,IGMP,MLD

Interfaces

MC

MR MR MR MR MR MC

4

Applications that want to receive multicasts, register with IGMP (for IPv4) or with MLD (for IPv6) using specific sockets API functions. Multicast functions are available on UDP and RAW sockets. z/OS is an IPv4 and IPv6 multicast-enabled host (sending or receiving multicast datagrams). z/OS uses IGMP or MLD to report multicast group membership to directly connected multicast-enabled routers. The routers in the network infrastructure need to dynamically optimize how multicast datagrams are propagated throughout the entire network in order to reach all receiving membership host, with the least amount of datagram duplication effort.

The current multicast model is referred to as any-source multicast (ASM). Multicast server programs send out datagrams using a multicast address as the destination address. Any client program on the network can choose to receive the multicast datagrams by joining the multicast group. This means that a client program which has joined a multicast group will receive multicast datagrams from any server, regardless of the source IP address.

It is possible for multiple multicast servers to be sending out datagrams for the same multicast address. An application could receive multicast datagrams from servers which it was not intending to receive. To avoid such collisions, an extension to the original ASM model was developed called Source-Filtered Multicast (SFM). SFM specifies a set of API extensions which allow an application to filter the datagrams it receives based on the source address. To complement these new source filtering API's, new versions of the Internet Group Management Protocol for IPv4 (IGMPv3) and the Multicast Listening Discovery protocol for IPv6 (MLDv2) are supported.
RFC3678 defines new socket options and functions to manage source filters. RFC 3376 and 3810 define the IGMPv3 (IPv4) and MLDv2 (IPv6) protocols used by systems to report their IP multicast group memberships to neighboring multicast routers. With the new versions of the protocols, multicast routers are informed of the source IP filtering of any applications on a system. This allows the multicast router to send only multicast datagrams, which the system has applications interested in receiving.

z/OS Communications Server supports the new API's to allow applications to specify source filter lists. This will allow the local system to filter on source addresses even if the system is not attached to a multicast router which supports source address filtering. It provides host support for IGMPv3 and MLDv2. The system will respond to queries from multicast routers and report the source filter state of each interface. z/OS Communications Server does not support any multicast routing protocols and therefore does not support any multicast routing functions of IGMPv3 or MLDv2.

IBM Configuration Assistant for z/OS Communications Server

- The new look is centered around the images and stacks to be configured

V1R9 Configuration Assistant enhancements

➢Policy based routing (PBR)
➢Network security services (NSS)
➢Change to IPSec Perfect Forward Secrecy specification
➢Image/stack orientation across multiple technologies
➢Protect multi-user edits of the same backing store
➢Import and combine V1R7/R8/R9 backing store data
➢Maintain configuration history for audit / tracking
➢Maintain delivery (FTP) history for audit / tracking
➢Support active and passive mode FTP
➢Sort table data
➢Enable or Disable connectivity rules
➢Continue extensive tutorials
➢Improved diagnostics including log levels and a detailed FTP log

Overview - Other functions   © 2007 IBM Corporation

5

Communications Server functions can be very complicated and time-consuming to configure by manually creating configuration files. The goal of the Configuration Assistant is to enable administrators to be able to configure these functions as easily as possible without having to understand the syntax of the configuration files.

Updates to the Configuration Assistant provide the solution for easier configuration of PBR and NSS. With the addition of these new technologies and to allow for expansion in the future, the Configuration Assistant was restructured to handle configuration of multiple technologies. Customers expressed concern about storing configuration information on the workstation as well as protecting against multiple administrators making configuration changes at the same time. These problems have been resolved by allowing the backing store files to be stored on z/OS using FTP and providing a locking mechanism to protect against multiple users making changes at the same time. The V1R8 Configuration Assistant allows for configuring one technology at a time. This prevents the ability to check for errors across technologies. The V1R9 Configuration Assistant solves this problem by allowing multiple technologies to be configured at the same time. Also it provides for the import of multiple backing store files from previous releases.

Customers have asked for a way to record changes made to configurations. The solution was to add the ability to enter comments to be stored in the backing store files whenever the files are saved. Also comments can now be added to the policy files when the files are delivered using FTP. To resolve problems with firewalls preventing the FTP of configuration files, the Configuration Assistant now supports both Active and Passive mode FTP. Most tables in the Configuration Assistant can now be sorted allowing for easier navigation within large tables. When customers wanted to make a quick configuration change temporarily and then revert back to the original configuration, they needed to remove configuration rules and then re-key them when reverting to the original configuration. In the V1R9 Configuration Assistant this is resolved by allowing rules to be disabled. When you want to revert to the original configuration, this is easily done by re-enabling the rules. Customers often comment about how much the tutorials have helped them. The Configuration Assistant includes new tutorials for PBR and NSS. To allow for faster resolution of customer problems, the V1R9 Configuration Assistant provides more detailed logging including a separate log for FTP connections.

IBM Health Checker for z/OS enhancements

- CS z/OS V1R9 extends support for and use of the z/OS Health-checker infrastructure:
  - ▶ If a TCP/IP stack has DYNAMICXCF defined in combination with GLOBALCONFIG SYSPLEXMONITOR without the RECOVERY option, a warning will be issued since this is not a recommended best practice combination. In this case, the automated RECOVERY option should be used.
  - ▶ Various VTAM® checks added:
    - ✓ Check that the VIT SIZE is not lower than the default value of 999
    - ✓ Check that the VIT options PSS (VTAM's Process Scheduling Services) and SMS (VTAM's Storage Management Services) are turned on - they are always needed to service a problem
    - ✓ Check that DSPSIZE is at least 5 (50 Megabytes)
    - ✓ Check to see if someone has OPT=ALL specified (requesting all VIT trace options be turned on) - this may not be optimal, unless requested by VTAM services personnel
    - ✓ Check that the Enterprise Extender QDIO/iQDIO buffer pools (T1Buf and T2Buf pools) are of reasonable sizes depending on EE activation

Several health checks have been added to Communications Server in V1R9.

If IPCONFIG DYNAMICXCF or IPCONFIG6 DYNAMICXCF is specified in the TCP/IP profile, IBM recommends that GLOBALCONFIG SYSPLEXMONITOR RECOVERY be specified. Specifying the RECOVERY parameter will allow a TCP/IP stack in a sysplex to perform internal checks to determine if it needs to remove itself from the sysplex and allow a healthy backup stack to takeover the ownership of DVIPA interfaces. The TCP/IP check will determine if DYNAMICXCF has been specified either on the IPCONFIG or IPCONFIG6 statement, and, if so, will check if RECOVERY has been specified on the GLOBALCONFIG SYSPLEXMONITOR statement. If RECOVERY was not specified, an exception message will be issued suggesting that the configuration be changed to specify RECOVERY. **This check will be performed once at stack initialization**.

IBM recommends that the VTAM Internal Trace (VIT) table size be set to its maximum value (999 pages). This size allows maximal trace information to be collected, which assists in problem determination. The VIT SIZE check will determine if the VIT table size is less than the maximum value of 999. If so, an exception message will be issued suggesting that the size be set to 999. **This check will be performed once at VTAM initialization and any time the VIT parameters are modified.**

The VTAM Internal Trace SMS and PSS options should always be active. The Storage Management Services (SMS) and Process Scheduling Services (PSS) trace options are frequently needed for problem determination.   The VIT options check will determine if either the PSS or SMS trace option (or both) have been inactivated.   If so, an exception message will be issued suggesting that the configuration be changed to activate both PSS and SMS. **This check will be performed once at VTAM initialization and any time the VIT options are modified.**

IBM recommends that the VTAM Internal Trace (VIT) data space size be set to its maximum value (5, indicating 50 megabytes). This size allows maximal trace information to be collected, which assists in problem determination. The DSPSIZE check will determine if the VIT data space size is less than the maximum value of 5. If so, an exception message will be issued suggesting that the data space size be set to 5. **This check will be performed once at VTAM initialization and any time the VIT parameters are modified.**

Unless it has specifically been requested by IBM service, it is generally not optimal to run with all VIT options active, as this can impact system performance. The ALL options check will determine if all the VIT options are currently active.   If so, an exception message will be issued suggesting some VIT options be inactivated. **This check will be performed once at VTAM initialization and any time the VIT options are modified.**

When running without Enterprise Extender (EE) active, IBM recommends that the buffer pool allocations for the T1BUF and T2BUF buffer pools be set at their default values. The T1BUF and T2BUF pools are used during EE processing, and are not utilized when EE is not active.   This check will determine if EE is not in use (and not likely to be used) on this system and check if the T1BUF or T2BUF pool allocation is above its default value.   If so, an exception message will be issued suggesting that the pool allocations for these two pools be set to their default values.

This check will be performed once at VTAM initialization.

When running with Enterprise Extender (EE) active, IBM recommends that the buffer pool allocations for the T1BUF and T2BUF buffer pools be set above their default values. The default values for these pools are intended for systems that are not using EE. When using EE, IBM suggests that the T1BUF and T2BUF pool usage be monitored and the buffer pool allocations adjusted to minimize excessive pool expansions. This check will determine if EE is in use (or likely to be used) on this system and check if the T1BUF or T2BUF pool allocation is at its default value.   If so, an exception message will be issued suggesting that the pools be monitored to determine the optimal initial allocation values. **This check will be performed once at VTAM initialization and when the first EE line is activated.**

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?

- Did it help you solve a problem or answer a question?

- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_whatsnewOther.PPT

This module is also available in PDF format at: ../whatsnewOther.pdf

7

You can help improve the quality of IBM Education Assistant content by providing feedback.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

CICS   IBM   VTAM   z/OS

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

8

Overview - Other functions