



| Large System Update October 2007

IBM Health Checker for z/OS

“Stolen” from Riaz Ahmad at WSC

Uno Bengtrsson
IBM Sweden STG
Uno.bengtsson@se.ibm.com

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

DB2*	System z
DB2 Connect	Tivoli*
DB2 Universal Database	VM/ESA*
e-business logo	WebSphere*
GDPS*	z/OS*
Geographically Dispersed Parallel Sysplex	z/VM*
HyperSwap	zSeries*
IBM*	
IBM eServer	
IBM logo*	
Parallel Sysplex*	

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Intel is a registered trademark of the Intel Corporation in the United States, other countries or both.

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

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries.

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

Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.

SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput 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 improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Agenda

- Why did we need a health checker?
- Setting up IBM Health Checker for z/OS
- Check outputs
- Check exceptions – what they look like and what to do about them
- Creating your own checks for IBM Health Checker for z/OS
- What's new in z/OS 1.8 and 1.9

Why did we need a health checker?

- Analysis of outages showed:
 - Significant number were avoidable bad configurations with single points of failure
 - Configurations that were less than optimal with unnecessary performance bottlenecks
- Situation exacerbated by:
 - Complex configuration requirements
 - Experienced skills are limited
 - Rare failures mean less experience by operations staff

Why did we need a health checker?

- Many options for flexibility:
 - Sometimes, default values are best guesses
 - Best practices may not become known until good exposure in many environments
- Best practices are not widely known or implemented
 - Many sources of best practices: product pubs, WSC Flashes and White Papers, wizards, ...
 - Hard to determine applicability
 - May be out of date
 - Just providing documentation has a limited affect
- **In a nutshell, health checker is to help avoid outages!**

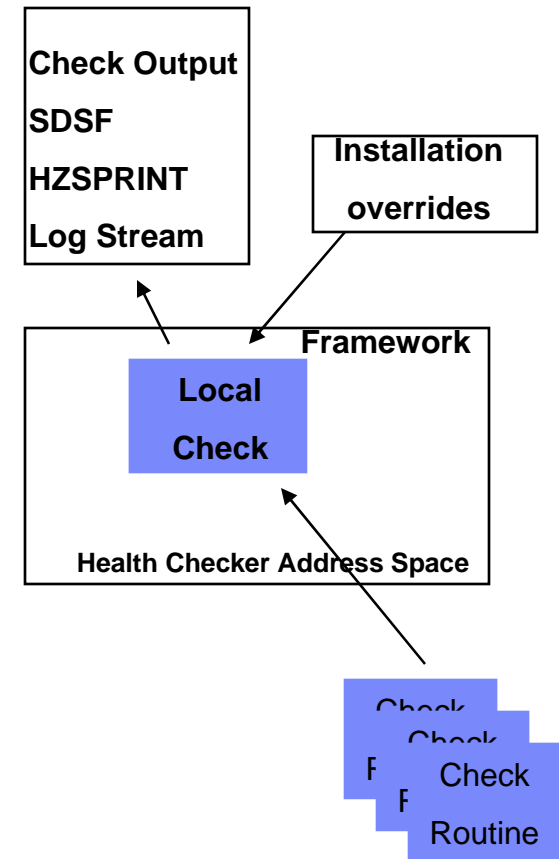
Health Checker Objectives

- Identify potential problems before they impact availability or, in worst cases, cause outages
- Check the current active z/OS and Sysplex settings and definitions
- Not a diagnostic *or* monitoring tool
- Continuously running preventative STC that finds potential problems
- Health Checker to produce output in the form of detailed messages
 - Potential problems and suggested actions to take

IBM Health Checker for z/OS

Integrated into z/OS

- Base element in z/OS 1.7 BCP
 - Available also as a web deliverable back to z/OS 1.4
- Consists of two parts:
 - **Framework** - interface that allows you to run and manage checks. The framework is a common and open architecture, supporting check development by IBM, independent software vendors (ISVs) and users
 - Provided as FMID HZS7720
 - **Checks** – look for component, element, or product specific settings, values, or definitions, checking for potential problems
 - Provided and owned by the component, element, or product themselves.
 - **Local Checks** – Run in the IBM health Checker for z/OS address space
 - Most IBM checks are Local
 - **Remote Checks** – Run in caller's address space



IBM Health Checker for z/OS

What is a check ?

- A **check** is a program or routine that identifies potential problems before they impact your system or sysplex availability
 - Checks are separate from the IBM Health Checker for z/OS framework
 - **Check Values:** contains pre-defined values such as: interval, severity, and routing and descriptor codes.
 - Modify values via: SDSF, HZSPRMxx parmlib member, or MODIFY command
 - **Check Output:** issued as messages. Exceptions produce WTO messages.
 - Can be viewed via: SDSF CK panel, HZSPRINT utility, or log stream
 - **Resolving Check Exceptions:** For best results, run the health checker continuously. Resolve exceptions by using the information in the check exceptions messages.
 - **Managing Checks:** print, display, activate/deactivate, add, refresh, run, update values temporarily or permanently
 - Permanent check updates can only be done via POLICY in HZSPRMxx

The Checks

- Checks are identified by:
(check-owner,check-name)
- Naming Conventions
 - Check-owner = IBM Component Name
 - Check-name= component-name_check-description
- Example
IBMASM,ASM_PAGE_ADD
- Wildcarding is allowed in check names
 - CHECK=(IBMCNZ,*)
 - All checks belonging to the console component

The Checks ...

- When checks execute, messages are written to message buffers
 - Messages may also be written to Logstream
 - SDSF CK panel displays results from message buffers
 - The print utility (HZSPRINT) displays messages from buffers and logstream
- Any exception messages are written as WTOs

IBM Health Checker for z/OS

Installation and Set up

- Satisfy software requirements for IBM Health Checker for z/OS
- Allocate the HZSPDATA data set to save check data between restarts
- Set up the HZSPRINT utility
- Define log streams to keep a record of the check output, as needed
- Create security definitions
- Set up customization and security for SDSF support for IBM Health Checker for z/OS
- Optionally create HZSPRMxx parmlib member
- Start IBM Health Checker for z/OS

IBM Health Checker for z/OS

Satisfy Software Requirements

- Software Requirements for IBM Health Checker for z/OS:
- **Framework: z/OS R4, z/OS R5, z/OS R6, or z/OS R7**
 - PTFs are necessary, see PSP HCHECKER, subset **HZS7720** (for ALL z/OS levels!)
 - z/OS R4, R5, and R6: RACF and SDSF PTFs
 - **Checks: may be dependent upon the level of z/OS you have**
 - Currently, checks are available for: Consoles, Contents Supervisor, RRS, XCF/XES, Logger, GRS, SDUMP, Security Server, RSM, VSM, ASM, TCP/IP, VTAM and z/OS UNIX System Services.
 - IBM will be adding more check periodically via APARs and integrated into z/OS releases
 - For obtaining the most recent appropriate checks for your z/OS level, use the Enhanced PSP Tool at http://techsupport.services.ibm.com/390/psp_main.html
 - You can identify available checks by retrieving the Extract File, and running the tool against it. Retrieve the Extract File by selecting a Type of *Function* and a Category of *Health Checker*.
 - Note that the names of some checks have changed since the prototype!

IBM Health Checker for z/OS

Allocate HZSPDATA data set

- Some checks use the HZSPDATA data set to save data required as part of their processing between restarts of the system or IBM Health Checker for z/OS
- Use the HZSALLCP sample JCL from SYS1.SAMPLIB
- Must have one HZSPDATA data set per image

```
//HZSALLCP EXEC PGM=HZSAIEOF,REGION=4096K,TIME=1440
//HZSPDATA DD DSN=SYS1.sysname.HZSPDATA,DISP=(NEW,CATLG),
//          SPACE=(4096,(100,400)),UNIT=SYSDA,
//          DCB=(DSORG=PS,RECFM=FB,LRECL=4096)
//SYSPRINT DD DUMMY
```

DS will be used in HZSPROC

Don't change these attributes

IBM Health Checker for z/OS

Set up HZSPRINT utility

- HZSPRINT utility allows you to see check output in the message buffer or the health checker logstream
- HZSPRINT writes the current message buffer for the target checks to SYSOUT
- Use the HZSPRINT sample JCL from SYS1.SAMPLIB
 - If LOGSTREAM is specified, then RACF authorization to the requested logstream is required
 - If LOGSTREAM is not specified, then RACF authorization to the requested check(s) is required

```
//HZSPRINT EXEC PGM=HZSPRNT,REGION=0M,
//      PARM= ( 'CHECK (check_owner,check_name) ' )
//*     PARM= ( 'LOGSTREAM(logstreamname) ', 'EXCEPTIONS' ,
//*     'SYSNAME (sysname) ' ,
//*     'CHECK (owner, name) ' )
//SYSOUT DD SYSOUT=A,DCB= (LRECL=256)
```

Defaults to *,*

Only show exceptions

When using a data set, it
must be seq, FB, LRECL 256

IBM Health Checker for z/OS

Define log stream

- Only the last iteration of a check is retained in the message buffer
- Define log streams to keep history of the check output
 - It's a good idea to retain historical data about your check results. To do that, you must define and connect to a log stream
 - Without logstream, only the results from the last execution each check are kept
- Plan for and set up the log stream. You may have one log stream per system or one log stream for multiple systems. Must begin with HZS
- Enable your log streams via MODIFY command or in HZSPRMxx parmlib member
 - F HZSPROC,LOGGER=ON,LOGSTREAM=*streamname*
 - For permanent specification, you must use HZSPRMxx parmlib member
 - LOGGER(ON) LOGSTREAM(*streamname*)
 - You can disable the log stream with a MODIFY command
 - F HZSPROC,LOGGER=OFF

IBM Health Checker for z/OS

CFRM and Logger Policies for logstream

Define CF structure for the logstream - IXCMIAPU

```
DATA TYPE(CFRM) REPORT(YES)
DEFINE POLICY(CFRMPOL) REPLACE(YES)
. . . . .
        STRUCTURE NAME(HZS_HCHECKER)
                SIZE(8000)
                PREFLIST(CF02,CF01)
```

Define logstream in the LOGR Policy

```
DATA TYPE(LOGR) REPORT(YES)
DEFINE STRUCTURE NAME(HZS_HCHECKER)
        LOGSNUM(1) AVGBUFSIZE(4096) MAXBUFSIZE(65532)
DEFINE LOGSTREAM NAME(HZS.HCHEK.HISTORY)
        STRUCTURENAME(HZS_HCHECKER) STG_DUPLEX(NO)
        AUTODELETE(YES) RETPD(7) HIGHOFFLOAD(80) LOWOFFLOAD(0)
```


IBM Health Checker for z/OS

Create security definitions

- Set up security for the IBM Health Checker started task
 - Similar to other started tasks:
 - Create user ID for IBM Health Checker for z/OS that has superuser authority. Connect that superuser ID to a group.
 - Associate that HC ID to the started task, HZSPROC
 - Give HC ID access to HZSPDATA data set allocated on each system
 - Give HC ID read access to PARMLIB
 - If using log streams, access for HC ID to logstream
 - New HZS XFACILIT class
 - RACF APAR OA10774 provided this support
- If using HZSPRINT utility, follow instructions in the User's Guide
- If using SDSF CK panel, set up security for SDSF

IBM Health Checker for z/OS

Set up SDSF support

- Set up customization and security for SDSF support, if desired
- SDSF provides support to make management of your checks easier with the new CK panel for the IBM Health Checker for z/OS.
- You can use the CK panel to display checks, attributes, and status, taking advantage of standard SDSF sort, filter, and arrange support.
- The SDSF support has been made available in APAR PK00561 for z/OS R4, R5, and R6. It is integrated into z/OS R7.
- Use instructions in z/OS SDSF Operation and Customization:
 - Authorize users to the CK command using either the AUTH parameter of ISFPARMS or SAF.
 - Authorize users to action characters or overtypable fields with ISFPARMS or SAF
 - Customize columns on the panel for groups of users with ISFPARMS.

IBM Health Checker for z/OS

Optionally Create HZSPRMxx parmlib member

- Intended for permanent changes
- At first, don't modify your HZSPRMxx parmlib member. Check to see what you get as output from IBM Health Checker for z/OS. Then, make permanent changes
- HZSPRM00 is supplied in your DDDEF's PARMLIB data set:

```
LOGGER(ON) LOGSTREAMNAME(HZS.HCHECK.HIST)

ADDREPLACE POLICY STMT(WSC1) UPDATE
  CHECK(IBMxcf,XCF_SYSPLEX_CDS_CAPACITY)
  PARM(0,2,5) SEVERITY(NONE)
  DATE(20050909) REASON('It is intentional to keep only 4 systems')
```

IBM Health Checker for z/OS

Start IBM Health Checker for z/OS

- Copy HZSPROC from SAMPLIB into your PROCLIB
- Update to point to your HZSPDATA data set and HZSPRMxx parmlib member you previously set up
- Update your COMMNDxx parmlib member with COM='START HZSPROC'
- Start the started task: S HZSPROC,HZSPRM=00

Your parmlib member you set up

```
//HZSPROC  PROC  HZSPRM= ' 00 '
//HZSSTEP  EXEC   PGM=HZSINIT,REGION=0K,TIME=NOLIMIT,
//          PARM= 'SET  PARMLIB=&HZSPRM'
//HZSPDATA DD    DSN=SYS1.&SYSNAME..HZSPDATA,DISP=OLD
```

Your saved check data set you set up

IBM Health Checker for z/OS

Operlog View

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
Display Filter View Print Options Help
-----
SDSF OPERLOG DATE 12/02/2005 0 WTORs COLUMNS 52- 131
COMMAND INPUT ==> _ SCROLL ==> CSR
0210 larger size.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 073
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 068
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 961
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0010 HZS0002E CHECK(IBMcnz,cnz_syscons_pd_mode): 214
0010 CNZHF0010E System console SYSC is running in Problem Determination
0010 mode.
0010 *HZS0003E CHECK(IBMxcf,xcf_cds_separation): 215
0010 IXCH0240E Multiple PRIMARY couple datasets reside on volume XCFCD2.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 216
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0010 HZS0002E CHECK(IBMcnz,cnz_syscons_pd_mode): 074
0010 CNZHF0010E System console SYSD is running in Problem Determination
0010 mode.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 075
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0010 *HZS0003E CHECK(IBMxcf,xcf_cds_separation): 076
0010 IXCH0240E Multiple PRIMARY couple datasets reside on volume XCFCD2.
0010 HZS0002E CHECK(IBMcnz,cnz_syscons_pd_mode): 069
0010 CNZHF0010E System console HMC is running in Problem Determination
0010 mode.
0010 *HZS0003E CHECK(IBMxcf,xcf_cds_separation): 070
0010 IXCH0240E Multiple PRIMARY couple datasets reside on volume XCFCD2.
0210 HZS0004I CHECK(IBMxcf,xcf_sysplex_cds_capacity): 071
0210 IXCH0602E The sysplex couple dataset may need to be formatted to a
0210 larger size.
0210 LOGON
0201 $HASP100 AHMAD ON TSOINRDR
0010 $HASP373 AHMAD STARTED
0010 IEF125I AHMAD - LOGGED ON - TIME=08.16.20
-----
MA a 04/021
Connected to remote server/host 9.82.24.151 using lu/pool TCP15254 and port 23

```

SDSF CK Panel

The screenshot shows a terminal window titled "Session A - [43 x 80]". The window has a menu bar with "File", "Edit", "View", "Communication", "Actions", "Window", and "Help". Below the menu bar is a toolbar with various icons. The main content area displays the "SDSF PRIMARY OPTION MENU" with a list of options and their descriptions. The options are arranged in two columns. The first column includes DA (Active users), I (Input queue), O (Output queue), H (Held output queue), ST (Status of jobs), LOG (System log), SR (System requests), MAS (Members in the MAS), JC (Job classes), SE (Scheduling environments), RES (WLM resources), ENC (Enclaves), PS (Processes), and END (Exit SDSF). The second column includes INIT (Initiators), PR (Printers), PUN (Punches), RDR (Readers), LINE (Lines), NODE (Nodes), SO (Spool offload), SP (Spool volumes), RM (Resource monitor), CK (Health checker), and ULOG (User session log). The prompt "COMMAND INPUT ==> _" is visible, and the status "SCROLL ==> CSR" is shown in red on the right. At the bottom of the window, there is a status bar with "MA a" on the left and "04/02" on the right. A connection bar at the very bottom indicates the session is connected to a remote server.

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
  Display  Filter  View  Print  Options  Help
-----
HQX7720 ----- SDSF PRIMARY OPTION MENU -----
COMMAND INPUT ==> _                               SCROLL ==> CSR

DA      Active users
I       Input queue
O       Output queue
H       Held output queue
ST      Status of jobs

LOG     System log
SR     System requests
MAS    Members in the MAS
JC     Job classes
SE     Scheduling environments
RES    WLM resources
ENC    Enclaves
PS     Processes

END     Exit SDSF

INIT   Initiators
PR     Printers
PUN    Punches
RDR    Readers
LINE   Lines
NODE   Nodes
SO     Spool offload
SP     Spool volumes

RM     Resource monitor
CK     Health checker

ULOG   User session log
    
```

MA a 04/02

Connected to remote server/host 9.82.24.151 using lu/pool TCP15223 and port 23

IBM Health Checker for z/OS

The ck panel display

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
Display Filter View Print Options Help
-----
SDSF HEALTH CHECKER DISPLAY  SYSB          DATA SET DISPLAYED
COMMAND INPUT  ==>
PREFIX=CAN*   DEST=(ALL)  OWNER=*   SYSNAME=*
NP   NAME                                     State
-----
CNZ_AMRF_EVENTUAL_ACTION_MSGS  ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_CONSOLE_MASTERAUTH_CMDSYS  ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_CONSOLE_MSCOPE_AND_ROUTCODE  ACTIVE(ENABLED)  EXCEPTION-LOW     SYS
CNZ_CONSOLE_ROUTCODE_11        ACTIVE(ENABLED)  EXCEPTION-LOW     SYS
CNZ_EMCS_HARDCOPY_MSCOPE        ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_EMCS_INACTIVE_CONSOLES      ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_SYSCONS_MASTER              ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_SYSCONS_MSCOPE              ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_SYSCONS_PD_MODE             ACTIVE(ENABLED)  EXCEPTION-MEDIUM  SYS
CNZ_SYSCONS_ROUTCODE           ACTIVE(ENABLED)  SUCCESSFUL        SYS
CNZ_TASK_TABLE                  ACTIVE(ENABLED)  SUCCESSFUL        SYS
CSV_APF_EXISTS                  ACTIVE(ENABLED)  EXCEPTION-LOW     SYS
CSV_LNKLST_NEWEXTENTS          ACTIVE(ENABLED)  SUCCESSFUL        SYS
CSV_LNKLST_SPACE               ACTIVE(ENABLED)  EXCEPTION-LOW     SYS
GRS_CONVERT_RESERVES           ACTIVE(ENABLED)  EXCEPTION-LOW     SYS
GRS_EXIT_PERFORMANCE           ACTIVE(ENABLED)  SUCCESSFUL        SYS
GRS_MODE                        ACTIVE(ENABLED)  SUCCESSFUL        SYS
GRS_SYNCHRES                   ACTIVE(ENABLED)  SUCCESSFUL        SYS
RACF_GRS_RNL                   ACTIVE(ENABLED)  SUCCESSFUL        SYS
RACF_SENSITIVE_RESOURCES       ACTIVE(ENABLED)  EXCEPTION-HIGH    SYS
RRS_DUROFFLOADSIZE             ACTIVE(ENABLED)  SUCCESSFUL        SYS
RRS_MUROFFLOADSIZE             ACTIVE(ENABLED)  SUCCESSFUL        SYS
RRS_RMDATALOGDUPLEXMODE        ACTIVE(ENABLED)  EXCEPTION-MEDIUM  SYS
RRS_RMDOFFLOADSIZE            ACTIVE(ENABLED)  SUCCESSFUL        SYS
RRS_RSTOFFLOADSIZE            ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_AFO                         ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_HVSHARE                    ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_MAXCADS                    ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_MEMLIMIT                   ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_REAL                       ACTIVE(ENABLED)  SUCCESSFUL        SYS
RSM_RSU                        ACTIVE(ENABLED)  SUCCESSFUL        SYS
SDUMP_AUTO_ALLOCATION           ACTIVE(ENABLED)  SUCCESSFUL        SYS
SDUMP_AVAILABLE                ACTIVE(ENABLED)  SUCCESSFUL        SYS
USS_AUTOMOUNT_DELAY            ACTIVE(ENABLED)  SUCCESSFUL        SYS
USS_FILESYS_CONFIG             ACTIVE(ENABLED)  SUCCESSFUL        SYS
USS_MAXSOCKETS_MAXFILEPROC     ACTIVE(ENABLED)  SUCCESSFUL        SYS
VSM_CSA_CHANGE                 ACTIVE(ENABLED)  SUCCESSFUL        SYS
-----
MA a
08/00
Connected to remote server/host 9.82.24.151 using lu/pool TCP15223 and port 23
    
```

IBM Health Checker for z/OS

Check details

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
Display Filter View Print Options Help
-----
SDSF OUTPUT DISPLAY CNZ_CONSOLE_MASTERAUTH_CMDSYS LINE 0          COLUMNS 02- 81
COMMAND INPUT ==> _                                           SCROLL ==> CSR
***** TOP OF DATA *****
CHECK(IBMCNZ,CNZ_CONSOLE_MASTERAUTH_CMDSYS)
START TIME: 11/21/2005 20:23:08.748411
CHECK DATE: 20040816 CHECK SEVERITY: LOW

CNZHS0002I At least one active console has MASTER authority and command
association to system SYSB.

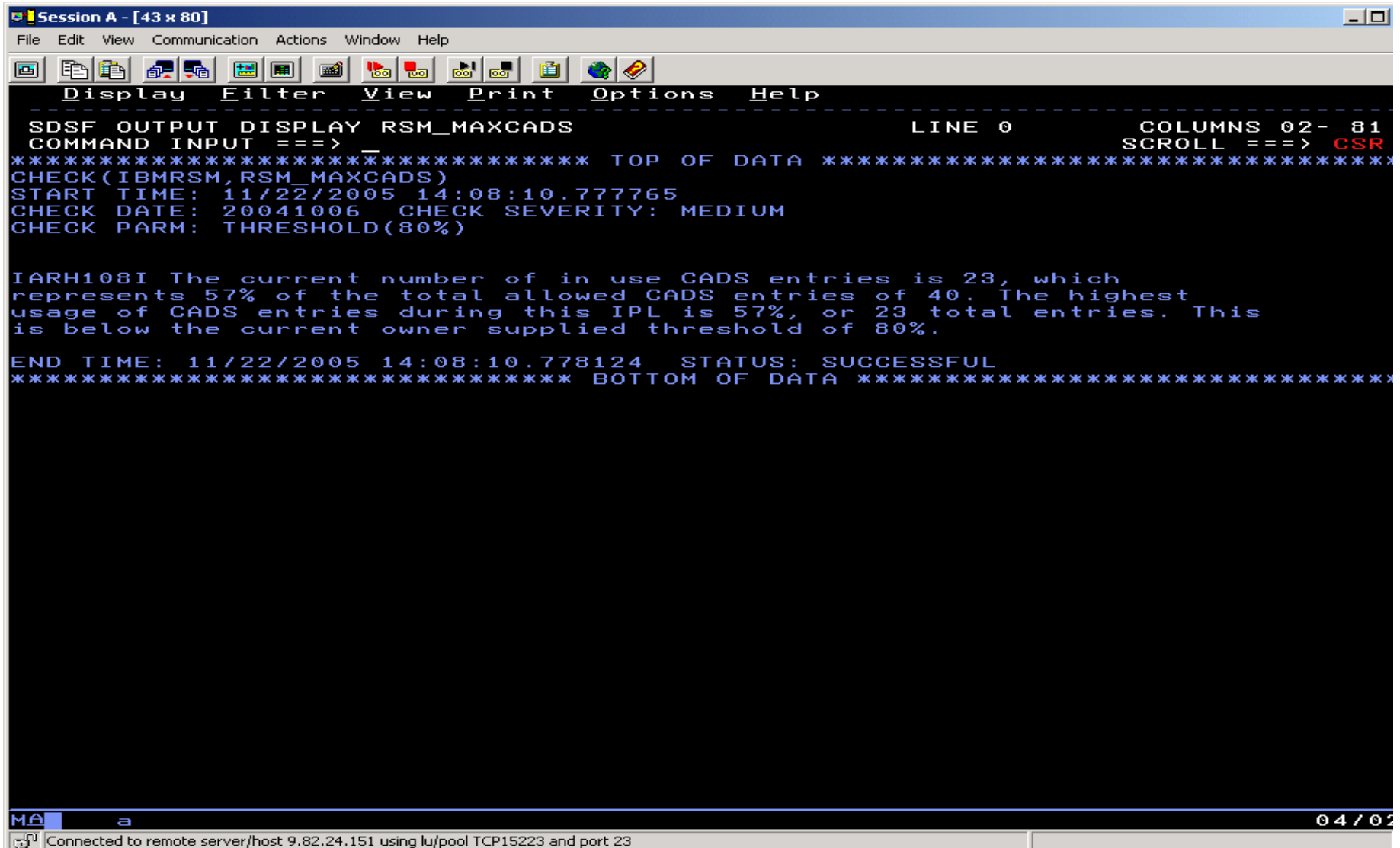
END TIME: 11/21/2005 20:23:08.751091 STATUS: SUCCESSFUL
***** BOTTOM OF DATA *****
    
```

MA a 04/02

Connected to remote server/host 9.82.24.151 using lu/pool TCP15223 and port 23

IBM Health Checker for z/OS

Check details ...



```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
Display Filter View Print Options Help
-----
SDSF OUTPUT DISPLAY RSM_MAXCADS                LINE 0                COLUMNS 02- 81
COMMAND INPUT ==> _                            SCROLL ==> CSR
*****
***** TOP OF DATA *****
CHECK(IBMRSM,RSM_MAXCADS)
START TIME: 11/22/2005 14:08:10.777765
CHECK DATE: 20041006 CHECK SEVERITY: MEDIUM
CHECK PARM: THRESHOLD(80%)

IARH108I The current number of in use CADS entries is 23, which
represents 57% of the total allowed CADS entries of 40. The highest
usage of CADS entries during this IPL is 57%, or 23 total entries. This
is below the current owner supplied threshold of 80%.

END TIME: 11/22/2005 14:08:10.778124 STATUS: SUCCESSFUL
*****
***** BOTTOM OF DATA *****
-----
MA a
04/02
Connected to remote server/host 9.82.24.151 using lu/pool TCP15223 and port 23

```

IBM Health Checker for z/OS

Check details ...

```

Session A - [43 x 80]
File Edit View Communication Actions Window Help
-----
Display Filter View Print Options Help
-----
SDSF OUTPUT DISPLAY CNZ_SYSCONS_PD_MODE                LINE 0          COLUMNS 02- 81
COMMAND INPUT ==> _                                     SCROLL ==> CSR
***** TOP OF DATA *****
CHECK(IBM CNZ, CNZ_SYSCONS_PD_MODE)
START TIME: 11/22/2005 13:23:08.780464
CHECK DATE: 20040816 CHECK SEVERITY: MEDIUM

* Medium Severity Exception *

CNZHF0010E System console SYSB is running in Problem Determination
mode.

Explanation: The system console should not be running in Problem
Determination mode during normal operations.

System Action: The system continues processing.

Operator Response: Report this problem to the system programmer.

System Programmer Response: If the system console was automatically
placed in Problem Determination mode because all of the consoles in
the AUTOACT group were inactive (message IEA021I is issued), no
action is required. To take the system console out of Problem
Determination mode, issue the following command:
VARY CN(console_name),DEACTIVATE

Problem Determination: n/a

Source: Consoles (SC1CK)

Reference Documentation:
z/OS MVS System Commands
z/OS MVS Planning: Operations

Automation: n/a

Check Reason: Should only run in Problem Determination mode when
there is a problem

END TIME: 11/22/2005 13:23:08.782718 STATUS: EXCEPTION-MED
***** BOTTOM OF DATA *****
MA a 04/02
Connected to remote server/host 9.82.24.151 using lu/pool TCP15223 and port 23

```

IBM Health Checker for z/OS

So, you have exceptions !

- There are exceptions! You may have had your system configured a certain way for a long time, and now, IBM Health Checker for z/OS is telling you
 - Look over exceptions and make an evaluation. **YOU** need to decide what is appropriate for your system.
- Just because you get an exception, doesn't mean that there is a problem to report to IBM. Exceptions are meant to tell you about potential availability impacts.
- When you get an exception, it will be in:
 - Message buffer – you can view via SDSF, HZSPRINT or in log stream
 - WTO message – by default. Format is HZS message number then component message ID. Such as: *HZS0011 IXL0021*
- ***No automatic correction of exceptions is done by The Health Checker***

IBM Health Checker for z/OS

Resolving exceptions

- Resolving your exceptions is important, or they will be repeatedly reported. (And probably get mixed up with ones you don't want to ignore!)
- You can resolve exceptions in the following ways:
 - Make the suggested updates to your system. This will not result in an exception when the check runs again. This is the recommended way!
 - Look at the values that the check is using. If the values aren't appropriate, update the values for your system. This will not result in an exception when the check run again.
 - Make the check Inactive or delete the check. (Inactive is a user-controlled state.)

IBM Health Checker for z/OS

Using exception messages for automation

- Why automate with IBM Health Checker for z/OS
- Numerous ways you can automate Health Checker and its exception messages
 - Automate startup of the IBM Health Checker every time a system IPLs
 - Automate HZSPRINT to keep a record of check messages on each system

```
//HZSPRINT EXEC PGM=HZSPRNT,REGION=0M,  
//      PARM=( 'CHECK (*,*) 'EXCEPTIONS' )  
//SYSOUT DD SYSOUT=DSN=A.B,DISP=MOD
```

- Automate HZSPRINT on each system to send e-mail messages
 - Use SMTP FTP command to send e-mail

IBM Health Checker for z/OS

Using exception messages for automation ...

- A check exception message WTO consists of an HZS header message, followed by the check specific exception message text:

```
*HZS0003E CHECK(IBMxcf,XCF_CDS_SEPARATION): 531
  IXCH0240E Multiple PRIMARY couple datasets reside on volume XCFCD2.
  HZS0004I CHECK(IBMxcf,XCF_SYSPLEX_CDS_CAPACITY): 532
  IXCH0602E The sysplex couple dataset may need to be formatted to a
  larger size.
```

- HZS header messages issued with exception are:
 - HZS0001I Exception message: low severity
 - HZS0002E Exception message: medium severity
 - HZS0003E Exception message: high severity
 - HZS0004I Exception hardcopy message: informational

IBM Health Checker for z/OS

Using exception messages for automation ...

- Key automation off check severity
 - Checks are classified as HIGH, MEDIUM, or LOW
- Route exception alerts to either a generic on-call address or a product expert
- Automate by message using MPF exits
- Put check output in a central place for responders
- Keep it simple

IBM Health Checker for z/OS

Customization

- The Health Checker MODIFY Command (F HZSPROC) is useful for making dynamic, *temporary* changes to checks
- Run checks – Run all checks that have an owner beginning with IBM
 - F HZSPROC,RUN,CHECK=(IBM???,*)
- Activate checks
 - F HZSPROC,ACTIVATE,CHECK=(*,*)
- Deactivate checks
 - F HZSPROC,DEACTIVATE,CHECK=()
- Update a check – Change a check's attributes
 - F HZSPROC,UPDATE,CHECK=(xxxxx,xxxxxxx),SEVERITY=HIGH
- Activate a Policy
 - F HZSPROC,ACTIVATE,POLICY=*policyname*

IBM Health Checker for z/OS

Customization ...

- To make permanent changes how checks behave in your installation, you will create a policy in HZSPRMxx PARMLIB member
- All control statements for policy definitions are documented
- **An Example:** VSM_CSA_THRESHOLD check:

```
CHECK(IBMVSM,VSM_CSA_THRESHOLD)
CHECK SEVERITY: HIGH
CHECK PARM: CSA(80%),ECSA(80%)
```

You want to change the check values for CSA and CSA to 70% and MEDIUM severity

```
F hzsproc,UPDATE,CHECK=(IBMVSM,VSM_CSA_THRESHOLD),PARM=(70,70),SEVERITY=MEDIUM
```

To make permanent change in your environment, Update HZSPRMxx member

```
ADDREPLACE POLICY STMT(WSC2) UPDATE CHECK(IBMVSM,VSM_CSA_THRESHOLD)
      PARM(70,70) SEVERITY(MEDIUM)
      DATE(20060902) REASON('This is my shop')
```

IBM Health Checker for z/OS

What is new in z/OS 1.8

- **In z/OS 1.8, the Health Checker Framework is enhanced**
 - **The REMOTE Checks**
 - Check runs in caller's address space
 - Communicates with HC via IEAVPSE, HZSADDCK, HZSFMSG.....
- **Enabled for NLS**
- **PARMLIB Check Definition**
 - Define/add a check via a PARMLIB definition in HZSPRMxx instead of needing a program that does the definition
- **Enhanced Policy Support**
 - Support multiple policies
- **Verbose command output**
 - Check output be as verbose as possible
- **Parameter Parser**
 - Provide a service usable by check routines to parse parameters
 - HZSCPARS

IBM Health Checker for z/OS

New Checks in z/OS 1.8

- New checks are added in z/OS 1.8
 - z/OS Communications Server checks to alert customers to non-recommended TCP/IP configuration values
 - TCPIP Checks
 - CSTCP_TCPMAXRCVBUFSIZE_*tcpipstackname*
 - CSTCP_SYSTCPIP_*tcpipstackname*
 - VTAM Check
 - CSVTAM_CSM_STG_LIMIT
 - RACF Checks
 - RRS check to highlight when a configuration varies from the IBM recommendation
 - ASM
 - Check for unused PARTE entries
 - Check for COMMON page data set
 - Check the current TOTAL AUX SLOT usage

IBM Health Checker for z/OS

New Checks in z/OS 1.8

- **LOGGER (OA15593) – z/OS 1.7 and z/OS 1.8**
 - CHECK(IBMIXGLOGR,IXGLOGR_STRUCTUREFULL)
 - CHECK(IBMIXGLOGR,IXGLOGR_STAGINGDSFULL)
 - CHECK(IBMIXGLOGR,IXGLOGR_ENTRYTHRESHOLD)

- **GRS provides GRS_GRSQ_SETTING, which will examine the current GRSQ setting.**
 - IBM recommends having a GRSQ setting of CONTENTION
 - Which may significantly reduce dump time

- **Download sample checks**
 - <http://www.ibm.com/server/eserver/zseries/zos/integtst/>
 - Select IBM health Checker for z/OS: Sample Checks

- **In z/OS 1.8, sample checks are in SYS1.SAMPLIB**
 - HZSSADCK, HZSSCHKR, HZSSMSGT, HZSSRCHK, HZSSSYMD

z/OS 1.9 Usability

Health Checker checks to use REXX

- Prior to z/OS 1.9 HC checks had to be **written in assembler**
- 1.9 checks can be **written in REXX** to run under SYSTEM REXX
- New parameters specified when the check is defined to indicate the check is a System REXX check:
 - Name of exec.
 - **REXXTSO(YES|NO REXXIN (YES|NO))**
- Check uses a conventional check message table as specified at check definition.
- New IBM Health Checker for z/OS callable services for System REXX checks
 - **HZSLSTRT**: Used to indicate Check has started.
 - **HZSLFMSG**: Used to issue check messages.
 - **HZSLSTOP**: Used to indicate a check has completed
- Check is run with a REXXOUT dsn when the check is in DEBUG mode
- Sample REXX check will be included in SYS1.SAMPLIB

z/OS 1.9 Usability

Health Checker checks to use REXX- Extended SDSF CK support

- New columns added to panel CK in SDSF
 - **EInterval** (Exception Interval)
 - Interval check will run when it has raised an exception
 - **ExecName**
 - Name of exec to run
 - **Locale**
 - Where check is running (e.g. **HZSPROC**, **REMOTE**, **REXX**)
- New columns (continued)
 - Origin
 - Origin of check (e.g. **HZSADDCK**, **MODIFY**, **HZSPRMxx**)
 - Verbose mode for check, óvertypeable
 - Generates **f hc ,check= ,verbose=** command

z/OS 1.9 Usability

Health Checker checks - new and updated checks

- **Checks added/updated with z/OS 1.9**
 - **RACF**
 - **VTAM**
 - **System Logger**
 - **Parmlib**
 - **TSO/E**
 - **VSAM/RLS**

z/OS 1.9 Usability

Health Checker checks - new and updated checks

CHECK (IBMRACF , RACF_SENSITIVE_RESOURCES)

- New sensitive resource – `'SYS1.SAXREXEC'`

CHECK (IBMUSS , USSPARMLIB)

- This check will compare z/OS UNIX System Services current system settings with those specified in the `BPXPRMxx` parmlib members used during initialization. Reason:
- Reconfiguration settings should be kept in a permanent location so they are available the next time z/OS UNIX is initialized.
- Parameters: n/a
- Interval(1:00)

Special considerations:

z/OS 1.9 only, Remote Check

If check is deleted, z/OS UNIX must be restarted to re-add the check

z/OS 1.9 Usability

Health Checker checks - New and Updated Checks...

- CHECK (IBMTSOE , TSOE_USERLOGS)
 - Verifies USERLOGS are in effect for SEND command
- CHECK (IBMTSOE , TSOE_PARMLIB_ERROR)
 - Verifies whether there were problems setting the groupings of settings (Authorized commands, Authorized programs, Send settings, etc.), when the **IKJTSOxx** parmlib members were processed.
- Check (IBMCS , CSTCP_SYSPLEXMON_RECOV_TCPIPStackname)
 - Verifies **IPCONFIG DYNAMICXCF** or **IPCONFIG6 DYNAMICXCF** parms and the **GLOBALCONFIG SYSPLEXMONITOR RECOVERY** parameter
- Check (IBMCS , CSVTAM_VIT_SIZE)
 - Verifies maximum VTAM Internal Trace (VIT) table size value
- Check (IBMCS , CSVTAM_VIT_OPT_PSSSMS)
 - Verifies the VIT PSS SMS options are active
- Check (IBMCS , CSVTAM_VIT_DSPSIZE)
 - Verifies a VIT dataspace table size of 5 (5 MB)

z/OS 1.9 Usability

Health Checker checks - New and Updated Checks...

- `Check (IBMCS , CSVTAM_VIT_OPT_ALL)`
 - Verifies all VIT options are NOT in effect
- `Check (IBMCS , CSVTAM_T1BUF_T2BUF_EE)`
 - Verifies T1BUF and T2BUF buffer pool allocations are not the defaults when Enterprise Extender is in use
- `Check (IBMCS , CSVTAM_T1BUF_T2BUF_NOEE)`
 - Verifies T1BUF and T2BUF buffer pool allocations defaults are in effect when the Enterprise Extender is not in use
- `CHECK (IBMPDSE , PDSE_SMSPDSE1)`
 - Verifies PDSE restartable address space is enabled
- `CHECK (IBMVSAMRLS , VSAMRLS_DIAG_CONTENTION) - OA17734`
 - Verifies there is no VSAMRLS latch contention
- `Check (IBMVSAMRLS , VSAMRLS_SINGLE_POINT_FAILURE) – OA17782`
 - Detects/flags single points of failure in the Share Control Data Sets (SHCDS).

z/OS 1.9 Usability

Health Checker checks - New and Updated Checks...

- CHECK (IBMIXGLOGR , IXGLOGR_STRUCTUREFULL)
 - Detects any logstreams that have encountered structure full conditions
- CHECK (IBMIXGLOGR , IXGLOGR_STAGINGDSFULL)
 - Detects any LOGGER Staging ds. that have encountered structure full conditions
- CHECK (IBMIXGLOGR , IXGLOGR_ENTRYTHRESHOLD)
 - Detects any logstreams that have encountered entry threshold problems

z/OS 1.9 Usability

Health Checker checks - New and Updated Checks...

- CHECK (IBMCSV , CSV_LNKLST_NEWEXTENTS)
 - Verifies that the number of extents used by each dataset in the LNKLST has not changed since the LNKLST was activated.
- CHECK (IBMCSV , CSV_LNKLST_SPACE)
 - Verifies that Partition Data Sets defined in any active LNKLST are allocated with only primary space.
- CHECK (IBMCSV , CSV_APF_EXISTS)
 - Verifies the data sets described by entries in the APF list are consistent with data sets that exist on the system.
- CHECK (IBMCSV , CSV_LPA_CHANGES)
 - Detects changes in LPA from IPL to IPL
- CHECK (IBMSUP , IEA_ASIDS)
 - Detects abnormal ASID usage, and detects/warns when a IPL may become necessary due to usage trends in ASIDs
- CHECK (IBMSUP , IEA_LXS)
 - Detects abnormal LX and ELX usage

z/OS 1.9 Usability

Health Checker *Appendix*

- IBM Health Checker for z/OS User's Guide (SA22-7994)
 - New IBM Health Checker System REXX Services
 - HZSLSTRT: Indicate check is running.
 - HZSLFMSG: IBM Health Checker for z/OS message service
 - HZSLSTOP: Indicate check completion
 - ADD/ADDREPLACE command is enhanced to add a System REXX check via HZSPRMxx parmlib, or IBM Health Checker for z/OS Modify command
 - HZSADDCK service is enhanced to add a System REXX check programmatically.

- Sample System REXX exec:
'SYS1.SAMPLIB(HZSSXCHK)'

Thinking of writing your own checks !

- Write a check routine that gathers information and compares and issues message with results.
- Create a message table for the check output.
- Create a HZSADDCHECK exit routine. This routine adds one or more checks and provides default values. It is called by IBM Health Checker for z/OS dynamic exit, HZSADDCHECK
- Add HZSADDCHECK exit routine to the HZSADDCHECK exit, and have system run the exit routine.
 - For testing, can do it via operator command or program.
- Provide documentation about check overrides, and when check was added
- Check will then run on its own, at the interval you specified for the check

IBM Health Checker for z/OS

Obtain checks

- As IBM adding more checks, periodically look for new ones
 - IBM intends to provide more checks both on release and non-release boundaries to respond quickly
- Reminder, for obtaining the most recent appropriate checks for your z/OS level, use the Enhanced PSP Tool
- The PTF letter for a check will describe how to add the new check.

Summary

- IBM Health Checker for z/OS integrated into z/OS R7
 - Available as a web deliverable back to z/OS R4
- Setting up IBM Health Checker for z/OS
 - Get to the right software levels
 - Allocate HZSPDATA data set for saving state data
 - Define logstream
 - Security definitions
 - SDSF customization
 - Create HZSPRMxx member and HZSPROC
 - Start the Health Checker
- Know what check exceptions are, and what to do about them
- Create your own checks, if you like!

