ibm.com

e-business ™

IBM

IBM Health Checker for z/OS V1R9 Enhancements

Redbooks

International Technical Support Organization

Trademarks

eNetwork	DFSMS/MVS	IMS	RMF
geoManager	DFSMSdfp	IMS/ESA	RS/6000
AD/Cycle	DFSMSdss	IP PrintWay	S/390
ADSTAR	DFSMShsm	IPDS	S/390 Parallel Enterprise Server
AFP	DFSMSrmm	Language Environment	SecureWay
APL2	DFSORT	Multiprise	StorWatch
APPN	Enterprise System 3090	MQSeries	Sysplex Timer
BookManger	Enterprise System 4381	MVS/ESA	System/390
BookMaster	Enterprise System 9000	Network Station	System REXX
C/370	ES/3090	NetSpool	SystemView
CallPath	ES/4381	OfficeVision/MVS	SOM
CICS	ES/9000	Open Class	SOMobjects
CICS/ESA	ESA/390	OpenEdition	SP
CICS/MVS	ESCON	OS/2	VisualAge
CICSPlex	First Failure Support Technology	OS/390	VisualGen
COBOL/370	FLowMark	Parallel Sysplex	VisualLift
DataPropagator	FFST	Print Services Facility	VTAM
DisplayWrite	GDDM	PrintWay	WebSphere
DB2	ImagePlus	ProductPac	3090
DB2 Universal Database	Intelligent Miner	PR/SM	3890/XP
DFSMS	IBM	QMFr	z/OS
	IBM System z	RACF	z/OS.e

Domino (Lotus Development Corporation)   **Tivoli (Tivoli Systems Inc.)**   **UNIX (X/Open Company Limited)**
DFS (Transarc Corporation)   **Tivoli Management Framework**   **Windows (Microsoft Corporation)**
Java (Sun Microsystems, Inc.)   **(Tivoli Systems Inc.)**   **Windows NT (Microsoft Corporation)**
Lotus (Lotus Development Corporation)   **Tivoli Manger (Tivoli Systems Inc.)**

Redbooks

Health Checker for z/OS Objectives

❏ Objective of IBM Health Checker for z/OS is to:

➤ Identify potential problems before they impact availability or cause outages

➤ Checks the current active z/OS and sysplex settings

➤ Checks definitions for a system and compares the values to those suggested by IBM or defined by you

❏ Not a diagnostic or monitoring tool

❏ Produces output in the form of detailed messages

➤ Indicates both potential problems and suggested actions

➤ Messages simply informs you of potential problems so that you can take action on your installation

---

Health Checks

❏ A check is actually a program or routine that identifies potential problems before they impact availability

❏ Checks are owned, delivered, and supported by:

➤ The component, element, or product that writes it

❏ You can override some check values:

➤ Statements in the HZSPRMxx parmlib member

➤ The MODIFY command - F HZSPROC,........

➤ SDSF interactive command panel

❏ Output is presented as messages

➤ You can save the check results using System Logger

Overview of Checks

❏ Each check has 3 parts

➤ The dynamic exit routine that identifies the check to the Health checker

➤ The check itself

➤ A message table to define messages that are issued by the check

❏ Each check includes the following:

➤ A set of pre-defined values, such as:

– Interval, or how often the check will run

– Severity of the check, which influences how check output is issued - (high, medium, and low)

– Routing and descriptor codes for the check

Health Checker for z/OS Address Space

Components, elements, or product checks

Checks loaded

Address Space

Check Routines

HZSPRMxx member
MODIFY command
SDSF commands

Installation overrides
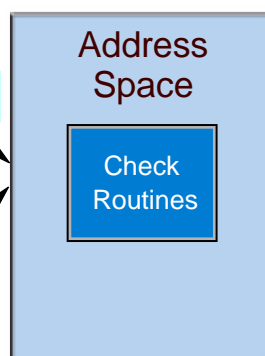
Potential problems

WTOs

Check output

SDSF (E)JES

HZSPRINT

Exception messages to SYSLOG

Logstream Archive

z/OS V1R8 Enhancement for Checks

❑ Checks can be add via HZSPRMxx parmlib member(s)

   ➢ Easier to add home grown checks: Dynamic exit routine is not needed

   – Adding checks via ADDCHECK in HZSPRMxx parmlib

❑ Tasks running in their own address space can define themselves as REMOTE checks

   ➢ Check can be written to run in unauthorized address space - (outside of Health Checker address space)

   – Remote check uses HZSADDCK macro to register check

z/OS V1R9 Health Checker

❑ Today checks are written in assembler language

   ➢ Assembler skills outside IBM are a scarce resource

   ➢ Multiple requests to simplify check writing experience

❑ IBM Health Checker for z/OS has been enhanced to make it easier to write and view checks, as follows:

   ➢ System REXX check support

   ➢ Extended SDSF Health Checker support

   ➢ New health checks

IBM Health Checker for z/OS: User's Guide, SA22-7994-03

Write a System REXX Check

❑ Use System REXX services to write a REXX check to:

> Gather installation information

> Look for problems, most likely by reading data set(s)

❑ Use the AXRCMD function to issue a system command

> Look at the command output

> Then issuing the check results in messages

❑ IBM Health Checker for z/OS may also write check exception messages as WTOs

---

z/OS Health Checker and System REXX
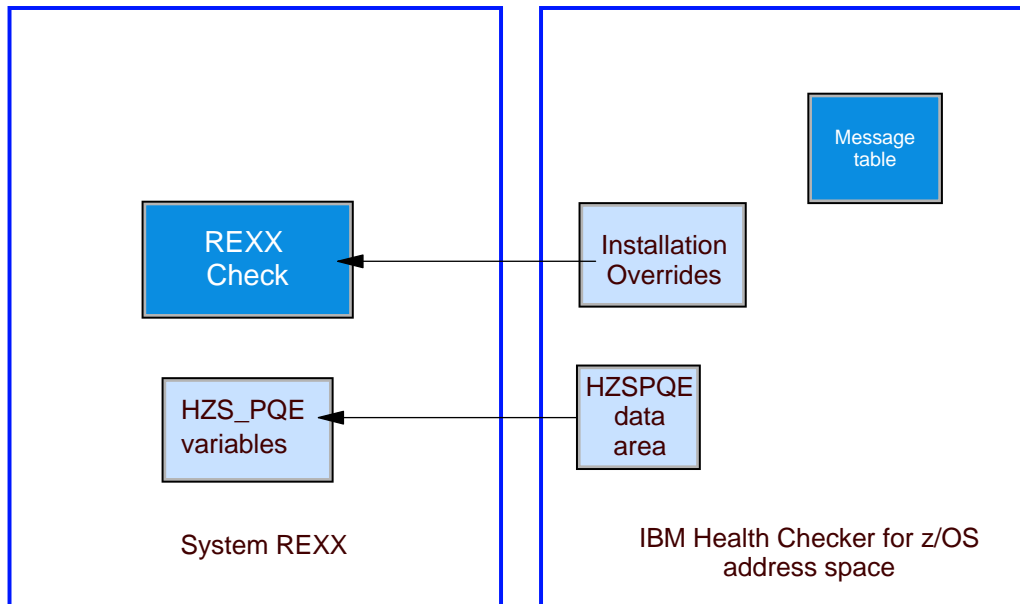
❑ Define a check to be an authorized REXX exec to run under System REXX

❑ Exec communicates with HC via calls to service routines

❑ A new flavor of "remote" check

> An exec gets told what "phase" this is, can access PQE, can add messages to check's message buffer

❑ 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

Blue Boxes Show the Parts the Check Developer Must Provide

System REXX		**IBM Health Checker for z/OS address space**

- REXX Check
- HZS_PQE variables
- Message table
- Installation Overrides
- HZSPQE data area

System REXX Check Support

❑ **Allows checks to be written in REXX**

➢ Available as R8 SPE (AO20393) for z/OS V1R8 and z/OS.e V1R8 and is identified by FMID HBB77SR

❑ **REXX checks run in one of the AXR address spaces**

❑ **General rules and limitations of System REXX apply to REXX checks**

➢ Exec must reside in the library used by System REXX

– SYS1.SAXREXEC (Not recommended)

❑ **REXX checks use a traditional message table**

❑ **REXX checks are defined as TSO(YES|NO)**

System REXX Check Support

❏ ADD | ADDREPLACE command is enhanced to add a System REXX check via by using the following:
  ➤ HZSPRMxx parmlib
  ➤ IBM Health Checker for z/OS Modify command (F)
❏ HZSADDCK service is enhanced to add a System REXX check
  ➤ Used by the HZSADDCHECK dynamic exit routine to add checks
    – REXX=YES and other relevant parameters
❏ Sample System REXX exec is provided in:
  ➤ 'SYS1.SAMPLIB(HZSSXCHK)

SYS1.SAMPLIB Samples

❏ Start off by using the samples and follow these steps:
  ➤ Compile REXX sample HZSSXCHK into SYS1.SAXREXEC using the REXXC procedure in CPAC.PROCLIB
  ➤ Run job HZSMSGNJ in SYS1.SAMPLIB which invokes REXX procedure "HZSMSGEN" to build input asssembler code for the message table HZSSMSGT
  ➤ Run an assembler compile and link (ASMLKED) of HZSSMSGT from the library 'userid.TEMP.ASM' that was created by the REXX HZSMSGEN, into a linklist data set that is APF authorized

New Parameters - ADD|ADDREPLACE

```
{ADD|ADDREPLACE},
CHECK=(check_owner,check_name),
{   CHECKROUTINE=routinename
 |  EXEC=execname
    ,REXXHLQ(hlq)
[,REXXTIMELIMIT=timelimitvalue]
    {     [,REXXTSO=YES]|[,REXXTSO=NO
          [,REXXIN={NO | YES] ]
}
}
```

System REXX Parameters

❏ REXXHLQ - Optional parameter for a REXX exec check, specifies the high level qualifier for any input or output data set for the check

❏ REXXTIMELIMIT - Specifies number of seconds exec can run under AXREXX service before cancel - default=forever

❏ REXXTSO=YES|NO - Required for a REXX exec check, whether check runs in TSO or not default=yes

❏ REXXIN=YES|NO - Optional for a REXX exec check, if a non-TSO check requires an sequential input data set

➤ The REXXIN data set has exec name specified in the EXEC parameter, and an optional entry code specified in the ENTRYCODE parameter

System REXX Parameters

❏ REXXOUT DSN is applicable for all System REXX checks

❏ REXXOUT DSN allocated if check runs in DEBUG mode

❏ You can use SAY and TRACE to write to

❏ Null REXXOUT DSN is passed to System REXX when DEBUG(OFF)

❏ REXXOUT data set used to capture REXX output and messages

❏ To issue a message with check results in your REXX check, must use the HZSLFMSG function

---

Using the HZSADDCK Service

```
HZSADDCK  ...
  {  REMOTE=NO …
    |REMOTE=YES …
        [{  REXX=NO …
           |REXX=YES
               ,EXEC=xexec
               [,ENTRYCODE=.xentrycode]
               [,TIMELIMIT=.xtimelimit]
                ,REXXHLQ=xrexxhlq
                ,{  REXXTSO=YES
                   |REXXTSO=NO
                      ,{  REXXIN=NO
                         |REXXIN=YES …
```

SYS1.SAMPLIB Example

```
ADDREPLACE  ,CHECK(IBMSAMPLE,HZS_SAMPLE_REXXIN_CHECK)
        ,EXEC(HZSSXCHK)
        ,HLQ(IBMUSER)
        ,REXXTSO(NO)
        ,REXXIN(YES)
        ,MSGTBL(HZSSMSGT)
        ,ENTRYCODE(1)
        ,USS(NO)
        ,VERBOSE(NO)
        ,PARMS('LIMIT(047)')
        ,SEVERITY(LOW)
        ,INTERVAL(ONETIME)
        ,DATE(20070819)
        ,REASON('A SAMPLE CHECK TO DEMONSTRATE AN ' ,
               'EXEC CHECK USING REXXIN.')
```

System REXX Check Support

❏ To run a System REXX check, the System REXX address space (AXR) must be active and accepting new requests

❏ New and enhanced checks are only available when the corresponding program product/feature is installed

❏ Ask Health Checker to use the new check

➢ F hzsproc,ADD,PARMLIB=xx

Warning - (Gotcha)

❏ Do not make a typo when defining your REXX check!

➢ When you define a REXX check in a HZSPRMxx parmlib member using the ADD | ADDREPLACE CHECK parameters, do it carefully, because it is a bad problem to delete check definitions created using parmlib members

❏ If you make a mistake

➢ Issue the following command, which will first delete all existing check definitions and then add the definitions found in the specified parmlib members:

– F hzsproc,REPLACE,PARMLIB=(suffix1..suffixn),CHECKS

Warning - (Gotcha)

❏ Example of error messages installing a REXX check

```
HZS0403I REPLACE PARMLIB PROCESSING HAS BEEN COMPLETED
HZS1002E CHECK(IBMSAMPLE,HZS_SAMPLE_REXXIN_CHECK):
AN ERROR OCCURRED, DIAG: 00000001_FFFFFFFD
HZS0001I CHECK(IBMSAMPLE,HZS_SAMPLE_REXXIN_CHECK):
HZSH0011E There are 0000000A (decimal 10) remaining available special
widgets. This is below the limit.
```

Problems Deleting Checks

❑ Defining a System REXX check in the HZSPRMxx parmlib member using the F hzsproc,ADD | ADDREPLACE

❑ Then, delete check using the DELETE command

❑ Now, to bring it back again with the ADD, CHECK command

➢ Command fails, with a message that the check already exists and it does not appear in SDSF

➢ The check definition is still in the HZSPRMxx parmlib member, and is still loaded in the system

❑ So, for check to run again, an ADDREPLACE, CHECK statement containing the check definition into parmlib

➢ Issue the F hzsproc,ADD,PARMLIB=

Comments from Early Experience

❑ Pending the availability of a solution from development

➢ Suggest that you effectively remove the SYS1.SAXREXEC data set from SMP/E control by cloning it to a non-system volume and cataloging that as the SYS1.SAXREXEC library

➢ You can still maintain the original SYS1.SAXREXEC with SMP/E using a specific volser

➢ There are currently only 3 execs provided by IBM and maybe they will not be often updated

➢ Put in place a process to copy the system SAXREXEC to your local one whenever you upgrade your system

SDSF CK Support

❏ **The support is invoked by - Health Checker (CK)**

❏ **New columns added to panel**

➢ EInterval (Exception Interval) - checks will run when it has raised an exception - (Overtypeable field)

– Generates f hc,check=,einterval= command

➢ ExecName - Name of exec to run

➢ Locale - Where check is running

– HZSPROC, REMOTE, or REXX

➢ Verbose - Specifies where the check is running

➢ Origin - Specifies the origin of the check

– HSZADDCK, MODIFY, or HZSPRMxx

REXX Columns for Health Checker

```
  Display  Filter  View  Print  Options  Help
 -----------------------------------------------------------------------
 SDSF HEALTH CHECKER DISPLAY   SC63                      LINE 27-53 (97)
 COMMAND INPUT ===> _                                  SCROLL ===> HALF
 NP    NAME                       EInterval ExecName Locale   Origin    Verbo
       CSVTAM_VIT_OPT_ALL         SYSTEM             HZSPROC  HZSADDCK  NO
       CSVTAM_VIT_OPT_PSSSMS      SYSTEM             HZSPROC  HZSADDCK  NO
       CSVTAM_VIT_SIZE            SYSTEM             HZSPROC  HZSADDCK  NO
       GRS_CONVERT_RESERVES       SYSTEM                                NO
       GRS_EXIT_PERFORMANCE       SYSTEM             HZSPROC  HZSADDCK  NO
       GRS_GRSQ_SETTING           SYSTEM             HZSPROC  HZSADDCK  NO
       GRS_MODE                   SYSTEM                                NO
       GRS_RNL_IGNORED_CONV       SYSTEM                                NO
       GRS_SYNCHRES               SYSTEM             HZSPROC  HZSADDCK  NO
       HZS_SAMPLE_REXXIN_CHECK    SYSTEM    HZSSXCHK REXX     HZSPRMPR  NO
       IEA_ASIDS                  SYSTEM             HZSPROC  HZSADDCK  NO
       IEA_LXS                    SYSTEM             HZSPROC  HZSADDCK  NO
       IXGLOGR_ENTRYTHRESHOLD     SYSTEM             HZSPROC  HZSADDCK  NO
       IXGLOGR_STAGINGDSFULL      SYSTEM             HZSPROC  HZSADDCK  NO
       IXGLOGR_STRUCTUREFULL      SYSTEM             HZSPROC  HZSADDCK  NO
```
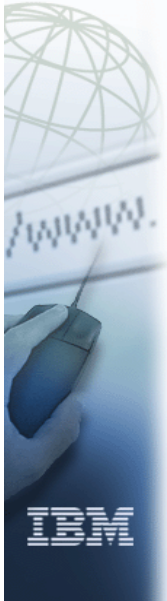
ibm.com

e-business

IBM

New Health Checks in z/OS V1R9

Redbooks
International Technical Support Organization

CHECK(IBMUSS,USSPARMLIB)

❑ This check compares z/OS UNIX System Services current system settings with BPXPRMxx parmlib members used during initialization

➢ Reconfiguration settings should be kept in a permanent location so they are available the next time z/OS UNIX is initialized

❑ z/OS V1R9 only

❑ Remote check

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

Redbooks

New TSO/E 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

---

Communications Server Checks

❑ Check(IBMCS,CSTCP_SYSPLEXMON_RECOV_TCPIPStackname)

➤ Verifies IPCONFIG, DYNAMICXCF, and IPCONFIG6 DYNAMICXCF parameters and the GLOBALCONFIG, SYSPLEXMONITOR, and RECOVERY parameter are in effect for the specified TCP/IP stack

❑ 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)

Communications Server 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

---

VSAM RLS Checks

- ❏ CHECK(IBMVSAMRLS,VSAMRLS_DIAG_CONTENTION) - OA17734

  - ➤ Verifies there is no VSAM RLS latch contention

- ❏ Check(IBMVSAMRLS,VSAMRLS_SINGLE_POINT_FALURE) – OA17782

  - ➤ Detects/flags single points of failure in the Share Control Data Sets (SHCDS)

With the APARs - applies to z/OS V1R8

System Logger Checks

❏ CHECK(IBMIXGLOGR,IXGLOGR_STRUCTUREFULL)

➢ Detects any logstreams that have encountered structure full conditions

❏ CHECK(IBMIXGLOGR,IXGLOGR_STAGINGDSFULL)

➢ Detects any LOGGER Staging dataset that have encountered structure full conditions

❏ CHECK(IBMIXGLOGR,IXGLOGR_ENTRYTHRESHOLD)

➢ Detects any logstreams that have encountered entry threshold problems

With OA15593 - applies to z/OS V1R7 and above

Contents Supervisor Checks

❏ CHECK(IBMCSV,CSV_LNKLST_NEWEXTENTS)
➢ Verifies number of extents used by each data set in LNKLST has not changed since 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

PTFs applicable for R1.4 – R1.8 (First 3 checks only)
CHECK(IBMCSV,CSV_LPA_CHANGES) – R1.9 only
APAR OA12777

New Supervisor Checks

❏ 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

Migration Considerations

❏ There are 2 REXX exec samples in SYS1.SAMPLIB:

➤ HZSSXCHK - Sample REXX exec check routine

➤ HZSSMSGT - Sample message input

❏ IBM Health Checker for z/OS SPE APAR for System REXX will be available with V1R9 GA for z/OS V1R8