IBM Washington Systems Center

IBM Healthchecker

Riaz Ahmad
IBM Washington Systems Center
Gaithersburg, Maryland

Trademarks

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

APPN*	IBM logo*	Virtual Image Facility
DB2*	IMS	VM/ESA*
e-business logo*	Magstar*	VSE/ESA
Enterprise Storage Systems	MVS	VTAM*
ESCON*	Netfinity*	WebSphere
FICON	OS/390*	z/Architecture
GDPS	Parallel Sysplex*	z/OS
Geographically Dispersed Parallel Sysplex	PR/SM	z/VM
HiperSockets	S/390*	zSeries
IBM*	S/390 Parallel Enterprise Server	

IBM eserver zSeries

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
LINUX is a registered trademark of Linus Torvalds
Penguin (Tux) complements of Larry Ewing
Tivoli is a trademark of Tivoli Systems Inc.
Java and all Java-related trademarks and logos are trademarks or registered 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.

IBM considers a product "Year 2000 ready" if the product, when used in accordance with its associated documentation, is capable of correctly processing, providing and/or receiving date data within and between the 20th and 21st centuries, provided that all products (for example, hardware, software and firmware) used with the product properly exchange accurate date data with it. Any statements concerning the Year 2000 readiness of any IBM products contained in this presentation are Year 2000 Readiness Disclosures, subject to the Year 2000 Information and Readiness Disclosure Act of 1998.

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.

Agenda

- Reasons for the Healthchecker
- Some Recent History
- Healthchecker overview and installation
- Healthchecker restructure in z/OS 1.7

Washington Systems Center

What is the problem?

- In depth analysis of outages show:
  - A significant number were avoidable
    - *bad* configurations
      - Single points of failure
  - Non-optimum configurations
    - Stressing key sysplex SW in unique ways
    - Unnecessary performance bottlenecks

© 2004 IBM Corporation

What is the problem?

- Parallel Sysplex design allows for elimination of single points of failure, but:
  - Complex configuration requirements
  - Skills are at a premium
  - Speed of recovery operations are critical
- Failures are rare
  - Operations and System Programmers caught off guard
- Sympathy Sickness can occur

What is the problem?

- Best practices are not widely known and implemented
  - Many sources of best practices materials
  - Multiple product publications, Redbooks, WSC Flashes, White Papers, Wizards, etc.
    - Voluminous, generic
      - Difficult to determine applicability
    - Static, point in time
    - Overwhelming
  - Documentation has limited affect

What is the problem?

- Designers and developers do not know what they do not know
  - Multiple tuning knobs for flexibility
  - Sometimes, default values are best guess
  - Some best practices not known until real customer experiences from multiple production environments
- Need the ability to improve availability characteristics

Washington Systems Center

What is the Objective of the Healthchecker?

- Provide a way to more easily and effectively assist installations to implement best practices
  - Proactive scan and identification of exceptions
- Environment Sniffer
  - Programmatically check various settings on the system
    - In storage checks - not a PARMLIB scan
    - Check against known best practices list
    - Notify when exceptions are found

© 2004 IBM Corporation

What is the Objective of the Healthchecker?

- Spread lessons learned from
  - Multiple environments
  - Installations
  - internal experiences
- This all boils down to:
  - ***Outage avoidance***

Washington Systems Center

Proof of Concept

- Developed a prototype that included a set of checks
  - Many checks from Parallel Sysplex Availability Checklist
  - From WSC experience of doing sysplex availability studies
- Run a batch job, get a report
- Added WTO of exceptions based on customer input

Proof of Concept

- Made available via web download
  - Large number of downloads
    - Customer interest shown
- Have learned many things:
  - Need ability to override supplied best practice value
    - But with strict controls
  - Need to expand the scope of components doing checks

Washington Systems Center

Proof of Concept

Over 2500
Downloads
Of prototype

- Made available via web download
  - Large number of downloads
  - Three updates to prototype
  - Active customer and IBMers defining new requirements
- Available on z/OS web site:
  - http://www.ibm.com/servers/eserver/zseries/zos/downloads/

© 2004 IBM Corporation

Existing Healthchecker Overview

- Free “as-is” tool that can be downloaded from the Web
  - Upload to OS/390 R10 or z/OS system
  - Run as a batch job
  - View exception messages and reports
  - Make suggested changes manually
- Mostly a **Configuration Checker**
- Not an ***msys* for Operations** replacement
  - No automatic correction of problems
  - No online panel displays

Healthchecker Overview

- Version 1 was made available on February 6, 2003
  - Coupling facility structure attributes
  - XCF transport class verification and XCF cleanup values
  - CF Structure locations compared against defined preferences
  - Sysplex, CFRM, LOGR couple data set separation
  - Sysplex, EMCS, and MVS console definitions
  - z/OS UNIX Automove definitions
  - Available frame queue thresholds
  - Real storage settings
  - Reconfigurable storage settings
  - See WSC Flash 10213

Healthchecker Overview

- Version 2 was made available on April 29, 2003
  - Additional support and new checks
    - Write-to-operator (WTO) message support
    - Severity designation for checks
    - Virtual Storage checks and mapping
    - New data set (HCDATA) required
  - See WSC Flash 10225

Healthchecker Overview …

- Version 3 was made available October 4, 2004
  - Linklist checks
    - Data sets in the link list have secondary space allocated
    - New secondary extent created in a current linklist data set data set
  - APF data sets exist on the volume defined in the APF list
  - Synchronous reserve processing enabled
  - Duplicate member names in linklist and LPA list data sets
    - Disabled by default Disabled by default
  - Customer control over highlighting of exception messages
  - See WSC Flash 10261

Healthchecker Installation

- Obtain Resource Link ID
  - ibm.com/servers/resourcelink
- Go to z/OS Homepage and download
  - ibm.com/servers/eserver/zseries/zos/downloads
- What is in the download package
  - hchecker.mmddyy.load.bin
  - hchecker.mmddyy.samplib.bin
  - hchecker.mmddyy.readme.txt
  - hchecker.survey.txt
- z/OS and Sysplex Health Checker User's Guide (SA22-7931-03)

Healthchecker Installation …

- Upload the two binary files
- Use TSO RECEIVE INDATASET(dsn)
- APF-authorize the load library
- Use SAMPLIB job ALLOHCDA to allocate the HCDATA file on each system
- Modify and submit SAMPLIB job HCHECK

Healthchecker Execution Output

- Issues WTOs for each exception
  - *HZS003I High Severity: CDS_DATASET_SEPARATION: Multiple PRIMARY couple datasets reside on volume XCFCD2

    HZS002I Medium severity: XCF_SIGNALLING: GROUP 'UNDESIG' not assigned to transport class xyz on system (SYSA)

    HZS001I Low severity: CONSOLE_ROUTCODE_11: One or more console found with ROUTCODE (11)

- LOG DD
  - Information about each check performed
  - 'Health Checker For z/OS and Sysplex' is starting

    Operating System z/OS 01.06.00

    Check COUPLINGFACILITY_STRUCTURE started

    MSG00202 COUPLINGFACILITY_STRUCTURE The state of coupling facility CF2 is : NORMAL, VOLATILE

Washington Systems Center

Healthchecker Execution Output

- REPORT DD
  - Exception details

```
*Medium severity Exception: IBM Criteria not met*  XCF_SIGNALLING ---------------------------- (check #1)
  Group 'UNDESIG ' has not been assigned to transport class XYZ defined on system SYSA.
  IBM suggests that users explicitly assign the collection of undesignated groups to each transport class
  by coding the pseudo-group name UNDESIG in GROUP keyword on the CLASSDEF statement in the
  COUPLExx  member of PARMLIB.

  Action: Edit the COUPLExx member of PARMLIB and be sure that each CLASSDEF assigns at least the
  pseudo-group name UNDESIG to the transport class via the GROUP keyword.
  For example:
  CLASSDEF CLASS(ONE) CLASSLEN(956) MAXMSG(800) GROUP(UNDESIG)
  CLASSDEF CLASS(TWO) CLASSLEN(8K)
        MAXMSG(750) GROUP(UNDESIG,GROUP2)
  IBM Reason: Avoid problems with XCF signalling.
```

Healthchecker Customization

- Modify PARMS to override IBM check values
  - Checks are shipped as USERPARM in SAMPLIB data set
  - To modify copy IBM's USERPARM to a backup member
- Disable or modify any check
  - Specify NOCALL to disable
  - Modification requires a reason and date
- Create your own USERPARM with your modified or disabled checks

Healthchecker Customization …

- To Disable GRS_SyncRsv check

```
    CHECK(GRS_SyncRsv)
NOCALL
DATE(20040915)
REASON('Scheduled to implement in November);
```

Customers Have Asked For

- A more formal product
  - With formal support
- Checks from more z/OS components
- Checks from more IBM products
- Checks from ISV products
- Ability to write their own checks

Healthchecker Restructure

- Moving from self contained batch job to Started task providing services
  - Allow checks to be added dynamically
    - No previous knowledge of check required by the HC backbone
  - Log results to MVS Logger logstream
  - Provide check management services
  - Long running STC with ability to re-execute checks on interval basis
    - Intervals are unique to each check
    - Interval values from 1 minute to 43 days

IBM z/OS V1.6 Announcement (204-180)

- IBM Health Checker for z/OS and Sysplex will be a new base function in z/OS 1.7 (FMID HZS7720)
  - Checks delivered separately from the framework, can be added dynamically
  - Checks delivered by elements and components as PTFs
  - User overrides check defaults via HZSPRMxx parmlib updates or MODIFY command
- Framework and most checks intended to be made available as z/OS web download for z/OS releases V1.4, V1.5, and V1.6
- Initial support for most existing checks with plans for incremental delivery of new checks
- SDSF support for managing checks with CK panel

New Structure Overview

- Each check has 3 parts
  - The dynamic exit routine that identifies the check to the Healthchecker
  - The check itself
  - A message table to define messages that are issued by the check

Check Runs in Health Checker Subtask

Attached Check rtn
Setup Recovery
Call AnyCheckRtn(Pqe,Check)

Call AnyChkRtn(Pqe,Cleanup)
Write report to logstream
Delete recovery

The check is attached by the JOBSTEP TCB for execution

Recovery;
Call AnyChkRtn(Pqe,Cleanup)
Write report to logstream

- **AnyChkRt:**
  - **Function(Null)?**
    - **Return**
  - **Funtion(Init)?**
    - **.....**
  - **Funtion(Delete)?**
    - **.....**
  - **Funtion(Check)?**
    - **EntryCode(AnyChk1)?**
    - **Obtain storage for this iteration of AnyChk1**
    - **Execute**
  - **Funtion(Cleanup)?**
    - **EntryCode(AnyChk1)?**
    - **Release storage for this iteration of AnyChk1**

Dynamic Exits To Add Checks

- Each dynamic exit routine can either be added via operator command or via an API
- Any dynamic exit routine that is added prior to the start of the Healthchecker, will be invoked when Healthchecker is started
- Healthchecker must be told to run to pick up new checks that are added after Health Checker is started.
- The dynamic exit routine uses the HZSADDCK macro to define one or more checks

Check Structure

- 32 bytes check name and 16 bytes check owner (Company name, and component)
- Entry code (used by the check routine when a single check routine has multiple functions)
- The date the *best practice* values were recommended
- 126 bytes reason that summarizes why the check was written
- The severity of the problem(s) the check is looking for.
- Any default parameter values.
- The default Interval
- The name of the check load module
- The name of the check message table

Check Structure

- Each check is called with a check entry code as defined by HSADDCHK
- Function code:
  - Initialization - Initialization processing (once per life of check)
  - Verify installation parameters
  - Any processing that should be done one for the life of the check
  - Check - Normal check processing
  - Check_cleanup - free any storage obtained during the check.
  - Check_delete - cleanup for any processing done during check initialization

Check Message Table

- Each check has a message table
- Common look and feel
- Structured diagnostic message
- Each message is owned by the check.
- Exception messages contain the WTO text
- Message language based on XML/SGML
- Message source is converted to an assembler file that must be compiled and linked to create the message load module that is included with the check.

Messages

- Checks issue both verbose ('configuration is good' messages) and exception messages.
- Check output
  - WTOs – exception messages are written as a HSZ (Healthchecker) message number and the component message ID follows HZS msg: **HZS001I IXL002I . . .**
    - Output: All messages are written to the REPORT file (last instance of check)
    - Check history via MVS Logger logstream
    - When an exception message is written, a summary WTO is written to outline the problem

External Interfaces

- Parmlib Support HZSPRMxx
  - Concatenation of members supported
  - Cross Component support
  - User overrides to:
    - Severity, WTO descriptor codes, intervals, active or inactive, categories, parameter values
- Categories
  - Installations can group multiple checks
    - Perform actions against categories
    - One check can be in up to 16 categories
- Operator Interfaces
  - Command interface
    - Display Command
    - Modify Checks
      - Run now, pause, refresh, etc.
  - SDSF CK panel

SDSF CK Panel

- Display checks, attributes, and status, taking advantage of standard SDSF sort, filter, and arrange support
- Alter check attributes
  - status, interval, severity, category, and WTO descriptor
- Browse check output for the most recent check
- Print check output or sent it to a data set

SDSF: Sample CK Display

```
SDSF HEALTH CHECKER DISPLAY  (ALL)                LINE 1-3 (3)
COMMAND INPUT ===>                               SCROLL ===> CSR
NP   OWNER        Name                      Category
     XCF          COUPLINGFACILITY_STRUCTURE FIRSTSHIFT
     XCF          COUPLINGFACILITY_STRUCTURE GRS_STAR
     GRS          GRS_SYNCRES                IPL
```


Washington Systems Center

SDSF: Sample CK Display

```
SDSF HEALTH CHECKER DISPLAY  (ALL)                LINE 1-3 (3)
COMMAND INPUT ===>                               SCROLL ===> CSR
NP   OWNER        Status       Interval Severity Start-Time
     XCF          Initializing 48:00    MEDIUM   22:00:00
     XCF          Initializing 48:00    MEDIUM   22:00:00
     GRS          Running      48:00    MEDIUM   22:00:00
```

SDSF: Sample CK Display, Browse a Check

```
 SDSF OUTPUT DISPLAY ALTERNATE_CONSOLE_GROUPS  LINE 0       COLUMNS 02- 81
 COMMAND INPUT ===>                                          SCROLL ===> CSR
******************************** TOP OF DATA ********************************
*Medium severity Exception: IBM Criteria not met*
   ALTERNATE_CONSOLE_GROUPS
   The following consoles have no alternate group (ALTGRP) defined:
    Console     Console    Console        Active
      ID         Name       Type          System
      1         POSIXCON   MCS            SY1
      2         PLEXSY2    MCS            (Inactive)
      3         BARCON1    MCS            (Inactive)
IBM suggests that alternate groups be defined to increase availability if
   there is a console failure. MVS can then switch to another console. MVS
   searches for the first available console based on the order of the
   console members defined for the alternate console group. Alternate
   groups help to avoid single points of failure. Note that IBM does NOT
   suggest use of the ALTCONS facility.

   Action: To define an alternate group, use the ALTGRP keyword of the
```


Washington Systems Center

IBM Healthchecker for z/OS and Sysplex Documentation

- User's Guide
- Check Developer's Guide
- Will ship a sample check
  - Including sample message table