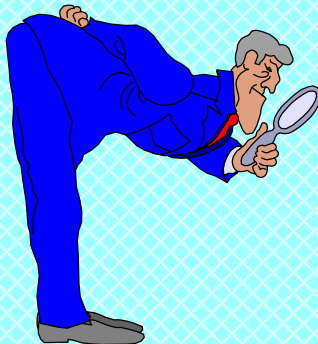


# ***Bit Bucket X'12'*** ***Adios 1900's***

**R.P. Shannon**  
**Programart Corporation**  
**124 Mt. Auburn Street**  
**Cambridge, MA 02138**  
**(617)498-2948**  
**shannon@programart.com**

**Carl Youngren**  
**State of California**  
**Health and Welfare Data Center**  
**Technology Division**  
**1651 Alhambra Blvd.**  
**Sacramento, CA 95816**  
**(916)739-7668**  
**cyoungre@hwdc.state.ca.us**



**SHARE 93**  
**Session 2817**  
**Chicago, IL**  
**August 27, 1999**

***Bit Bucket X'12'***



# Trademarks

**APPN**

**CICS/ESA**

**DATABASE 2**

**DB2**

**DFSMS**

**DFSMS/MVS**

**Enterprise System Connection Architecture**

**ES/9000**

**ESCON**

**Hardware Configuration Definition**

**IBM**

**CA-1**

**IDMS**

**IBM**

**IMS/ESA**

**MVS/ESA**

**NetView**

**OS/390**

**RACF**

**S/390**

**Sysplex Timer**

**System/390**

**VTAM**

## **PDS/E vs SYSRES**

**PDSEs and HFS are required to be SMS-managed**

**This has made it virtually impossible for IBM to ship products in PDSEs on the SYSRES volume:**

- Long standing recommendation not to SMS-manage SYSRES
- PDSE and HFS must be Catlg'd
- Can't have two PDSEs of the same name, much less manipulate them

**Sysprogs have been reluctant to utilize PDSEs:**

- SYSRES "cloning" is incompatible with SMS, PDSE, HFS, & VSAM
- SMS-management is an inhibitor
- Stigma of early data-sharing problems remains

**IBM has new function that requires PDSE and HFS**

- System datasets for C/C++ to ship in PDSEs
- Exploitation of compiler and Binder functions require Program Objects (PDSE or HFS exclusively)

## **PDS/E vs SYSRES**

**These conflicts are resulting in new function for PDSE and HFS and removal of the restriction that PDSEs and HFS be SMS-managed:**

- **Objective is to make use of PDSE vs PDS transparent - exceptions are NUCLEUS, LPALIB, PARMLIB**
- **PDSEs will not have to be SMS managed (allocate with VOL=SER)**
- **PDSEs will not have to be Cataloged to be referenced (use VOL=SER)**
- **Since not required to be catalogued can have multiple datasets with the same DSN (just like a standard PDS)**
- **SMS still must be active (ACS routines control SMS or non-SMS allocation)**
- **HFS files do not have to be SMS managed (just like PDSE)**
- **HFS files can be manipulated (copied, backed up, etc.) without being Cataloged**
- **USS requires a HFS to be Cataloged for Mount or access**

**This change affects many IBM programming labs who are working together**

## **PDS/E vs SYSRES**

**Changed components include: DFSMS, DFHSM, DFDSS, LISTDSI, FAMS, ISMF, ISPF, IEHLIST and ?**

**Support will be retrofitted back to DFSMS 1.4**

**All of this is the result of a "skunk works" project that began in mid-1998 by IBM**

**Satisfies SHARE Requirement SSMVS98022 "Allow PDSE to Reside on SYSRES Volume Maintaining Cloning"**

**Function to be delivered via service later this year:**

- **End of 4Q99**
- **Portions running today**
- **Testing delayed due to resource constraint**

**APAR numbers not available yet - watch OW35441 and children**

## **APARs of Interest**

### **OW37566 (F907) ABEND0C4 IN ISRDDNL**

**Applicable to all releases**

**Fixes a minor problem and provides general functional and reliability improvements**

**Function dependent on release level**

**Most apply to release 4.5 (OS/390 2.6)**

**Use SDB for CLIST data set**

**Pattern matching on Member command (e.g.: M \*X\*)**

**CLIST command to create CLIST of current allocations**

**Dynamic linklist support for Linklist command**

**Global, System, Systems, Step and Reserve to ENQ**

**Includes new undocumented LPA Compare command**

# APARs of Interest

## Current Data Set Allocations

Line 21 of 41

Command ===>

Scroll ===> PAGE

Volume	Disposition	Act	DDname	Data Set Name	List Actions: B
SYS001	SHR,KEEP	>	SYSPROC	SYS3.CLIB	
OS39R6	SHR,KEEP	>		ISP.SISPCLIB	
SCPMV5	SHR,KEEP	>		SYS1.CLIST	
OS39R6	SHR,KEEP	>		ISP.SISPEXEC	
OS3R6A	SHR,KEEP	>		ISF.SISFEXEC	
OS39R6	SHR,KEEP	>	SYSTCPD	TCPIP.TCPIP.DATA	
SCPMV5	SHR,KEEP	>	SYSUADS	SYS1.UADS	
SYS001		>	LINKLIST	SYS3.LINKLIB	
OS3R6A		>		SYS1.MIGLIB	
OS39R6		>		SYS1.CSSLIB	
OS3R6A		>		SYS1.LINKLIB	
<snip>					
SYS001		>	LPALIB	SYS3.LPALIB	
OS39R6		>		SYS1.LPALIB	
OS39R6		>		ASMA.V1R2M0.SASMMOD1	
OS39R6		>		REXX.V1R3M0.SEAGALT	
<snip>					

# APARs of Interest

ISRDDN Duplicates list

Userid: COY1

Time: 18:42 99/08/16

Member	DD Name	Data Set Name
-----	-----	-----
DFSMRCL0	--LPA---	Loaded at location 00CC9590 Size: 000005F8
DFSMRCL0	LINKLIST	IMS.RESLIB
DFSMRCL0	LINKLIST	SYS1.P390STUB.LINKLIB
DFSMRCL0	LPALIB	SYS1.ADCD.LPALIB
DXRRL186	LINKLIST	DSN510.SDXRRESL
DXRRL186	LINKLIST	IMS.SDXRRESL
DXRRL186	LINKLIST	IMSRLM61.SDXRRESL
E	LINKLIST	SYS1.CMDLIB
E	SYSHELP	SYS1.HELP
EAGIVVER	--LPA---	Loaded at location 83E6A000 Size: 00005D88
EAGIVVER	LPALIB	SYS1.SEAGLMD
EAGKCPT	--LPA---	Loaded at location 8458CBD8 Size: 00000058
EAGKCPT	LINKLIST	REXX.V1R3M0.SEAGALT
EAGKCPT	LPALIB	REXX.V1R3M0.SEAGALT
FLMQ	ISPEXEC	ISP.SISPEXEC
FLMQ	SYSEXEC	ISP.SISPEXEC
FLMQ	SYSPROC	ISP.SISPEXEC

## **APARs of Interest**

### **Compares content of every library and LPA**

- **Only/Exclude commands limit library selection - no cascading Exclude**
- **Cute warning message and confirmation before proceeding**
- **You need READ authority to all libraries; if not, processing continues without offending library**

**Not the only software available, but definitely the cheapest**

**P.S.**

**Another Doug Nadel tool is TASID (a Bit Bucket favorite)  
TASID (5.05K) has moved from the IBM ISPF home page to  
<http://www.mindspring.com/~somebody/>**

## **APARs of Interest**

### **OW37807 (F903) Cannot delete all messages**

**Applicable to OS/390 1.1 +**

**Messages issued with Route Code 11 could not be completely suppressed by MPF exits or IEAVMXIT**

**Message exit parameter list (CTXT, mapped by IEZVX100)**

- **CTXTRDTM bit "deleted" hardcopy and console display**
- **CTXTESJL bit "suppressed" joblog**

**However Route Code 11 forced through WTP to either SYSMSG or TSO user; "delete" and change of route code was not allowed**

**New bit, CTXTNWTP avoids WTP**

**Use in conjunction with CTXTRDTM and CTXTESJL**

**Message will still be shown to the message SSI**

**IBM advises extreme caution as there will be no record of the message if needed for problem diagnosis**

**Auditors won't like it :-)**

## **APARs of Interest**

### **T7M8512 - CAIRIM fails to load TMS modules**

**IDMS R14 RIM loads CAS9SEC**

**CA-1 (TMS) RIM loads CAS9SEC, CAILPAM and others**

**If CA-1 finds CAS9SEC loaded (IDMS initialized before it), it assumes CAILPAM and others are also loaded**

**CAIRIM completes with a RC 0 !!**

**Results in S806 Abends for all tape jobs**

**Circumvention is to insure CA-1 (S910INIT) occurs before IDMS R14**

**Fix changes S910INIT to handle CAS9SEC already being loaded**

**Future release of IDMS will not load CAS9SEC**

## **APARs of Interest**

### **OW34634 (F809) Copy Count Compatibility**

**Output NJE'd to JES2 or VSE from JES3 doesn't print**

**Output created with default copy count - copies not specified**

**Prior to JES3 2.4 the NJE header field NDHGDSCT (copy count) is set to 1; with JES3 2.4, NDHGDSCT is set to 0 indicating a default copy count**

**JES2 and VSE treat 0 as "no copies wanted"**

**APAR applies to all levels of JES2 (ESA 4.2 through OS/390 2.5)**

## **APARs of Interest**

### **PQ28126 (F907) TCP/IP Won't Restart**

**If TCP/IP is “forced”, it may not restart**

**Applicable to 3.4 & 3.5 (OS/390 2.5 and up)**

**Two circumvention's:**

- **Bring up TCP/IP with different PROC name**
- **IPL**

### **PQ29921 (Open) FTP to MVS Data Loss**

**Applicable to OS/390 2.7 only**

**FTP of FB binary to MVS uses double buffering**

**Only one buffer flushed at end of transfer**

**Circumvention's:**

- **Use APPEND instead of PUT to create a new dataset**
- **APPEND into an empty pre-allocated dataset**

## **CONFIGxx - An Oldie But Goodie**

**In SYS1.PARMLIB, CONFIGxx defines a standard configuration**

**The configuration can consist of: CP(s), Storage, Expanded Storage, CHPIDs, Devices, and Volumes**

**Commonly used to compare current vs. "standard"**

**Display differences via D M=CONFIG(xx) command**

**Can also be used to reconfigure**

# CONFIGxx - An Oldie But Goodie

## Sample CONFIGxx statements:

```
CPU      ( 0 , 1 , 2 , 3 , 4 , 5 , 6 , 7 ) , ONLINE
STOR     0M-2048M , ONLINE
ESTOR    0M-6144M , ONLINE
CHP      ( 00-03 , 0E-0F ) , ONLINE
DEV      2500-25FF , ( 4C , 4D , CC , CD ) , ONLINE
VOL      PAGE1A , SYSRES=25XX
```

## Handy when doing CPU upgrades/changes

- Avoids surprises after IPLs or maintenance
- Identifies missing items that might have been overlooked
- Provides early notification of potential problems

**Use automation to issue D M=CONFIG periodically**

**Low tech, but effective and at a reasonable price**

# **SYSMDUMP to SYSOUT**

**JES spool can be used as a temporary storage area for SYSMDUMPs**

**Avoids some of the shortcomings of SYSMDUMPs to DASD**

- **Dataset name ENQ**
- **Overlaying old dumps**
- **Allocated but unused dump data sets - wasted space**
- **Catlg'd but never Opened dataset - GDG**

**Direct the dump to Spool**

```
//SYSMDUMP DD SYSOUT=(output_class)
```

**Spool space is only used WHEN a dump actually occurs**

**In the event of an abend, transfer the SYSMDUMP from spool to a permanent data set via a JES SYSOUT viewer.**

# **SYSMDUMP to SYSOUT**

**Using a JES SYSOUT viewer:**

**(1) Allocate a target data set with the following characteristics,**

**DSORG(PS) RECFM(FB) LRECL(4160) BLKSIZE(4160)**

**(2) Copy the SYSMDUMP sysout file to the target data set.**

**▲ In IOF, use SNAP**

**▲ In SDSF, use PRINT**

**Start IPCS, and point IPCS at the target data set.**

## **SYSMDUMP to SYSOUT**

**Usage considerations:**

**For output\_class, use a 'Held' output class**

**Potentially could use a output class that is quickly "archived"**

**For BLKSIZE > 4160, IPCS had problems finding certain data structures**

**When copying the SYSMDUMP sysout file from spool, avoid carriage control processing**

**Be sure you have available spool space**

**Probably not usable for everything, but solved a problem for us**

**[Http://www.s390.ibm.com/products/oe](http://www.s390.ibm.com/products/oe) has lots of free UNIX tools:**

- **Reminiscent of early MVS days**
- **Look for “Tools and Toys”**
- **Contains:**
  - ▲ **Code samples**
  - ▲ **Ported tools**
  - ▲ **OS/390 USS tools**

**Recent additions (verbatim from the website):**

- **Lynx is a freely available line mode browser allowing access to WWW documents from OS/390 Unix Services directly or via proxies; interactively via a curses-driven display or via scriptable commands**
- **GNU Emacs 19.34 -Perhaps the most popular programmable Unix editor. The non beta version is now available for downloading from MKS**

## UNIX Freebies

- **odb2** - A command line db2 interface. It uses CAF instead of CLI, so its generally quicker and easier to setup than CLI. Also included are several tools (cwrap and others) to make db2 USS application development easier.
- **Bruno's Bag of Tools** - Added the new newname utility, a tool that allows you to rename files using a sed substitute command (wildcard characters) allowing you to rename many files at once that require a similar change.
- **Cobol compile environment**: Tips, templates, and a makefile for building COBOL applications from the OS/390 shell.
- **Copytree**: a utility that can run under TSO or the shell used to make a copy of a file hierarchy preserving all file attributes.
- **Setstack**: A simple tool for OS/390 used to conditionally execute other programs dependent on the state of the specified TCP/IP stack.

# The Ghost of Bit Buckets Past

**D PROG,LPA works for PLPA as well as for DLPA**

```
D PROG,LPA,MODNAME=IEFACTRT
```

```
CSV550I 10.19.39 LPA DISPLAY 256
```

FLAGS	MODULE	ENTRY PT	LOAD PT	LENGTH	DIAG
P	IEFACTRT	84A0C7E0	04A0C7E0	00000910	0866A368

**II11230: OS/390 R6+ supports use of &SYSR1 in LOADxx for libraries in concatenated parmlib:**

- Supported in LOADxx
- \*\*\*\*\* and MCAT not supported
- Symbolics still not supported for Parmlib datasets

**D LLA,STATS - a newly discovered undocumented option for an undocumented command**

# The Ghost of Bit Buckets Past

D LLA,STATS

CSV620I 10.12.48 LLA STATS DISPLAY 152

TOTAL DASD FETCHES: 8743 TOTAL VLF RETRIEVES: 4852

76 LIBRARY ENTRIES FOLLOW

LIBRARY: SYS1.SBDTLIB

MEMBERS:	61		
MEMBERS FETCHED:	0	MEMBERS IN VLF:	0
DASD FETCHES:	0	VLF RETRIEVES:	0

LIBRARY: EWX.SEWXAUTH

MEMBERS:	3		
MEMBERS FETCHED:	0	MEMBERS IN VLF:	0
DASD FETCHES:	0	VLF RETRIEVES:	0

LIBRARY: EOY.SEOYLOAD

MEMBERS:	11		
MEMBERS FETCHED:	0	MEMBERS IN VLF:	0
DASD FETCHES:	0	VLF RETRIEVES:	0

LIBRARY: SYS1.CMDLIB

MEMBERS:	293		
MEMBERS FETCHED:	43	MEMBERS IN VLF:	48
DASD FETCHES:	20	VLF RETRIEVES:	740

<SNIP>

## Vertical microcode:

- Introduced on G4 processor
- Common to “Alliance” processors (G4-G6)
- Processor enters millicode mode instead of ESA/390 mode to execute complex instructions that are not “hard wired”

## Millicode runs in its own environment:

- Code resembles programs
- Uses some instructions unique to millicode
- Has its own registers
- Can be stored on Read Only Storage (ROS) or in HSA
- Transparent to end user

# CFMON

## Free Coupling Facility Monitor

Collector runs as Started Task; ISPF viewer

Publication *Parallel Sysplex Coupling Facility Online Monitor: Installation and User's Guide, SG24-5153-00* available at [www://ibm.redbooks.com](http://www.ibm.redbooks.com)

Available at

<ftp://www.redbooks.ibm.com/redbooks/sg245153>

- Self extracting zip file creates two xmitted files
- Easy installation

If RMF for OS/390 R6 is installed, CFMON spits out ugly messages indicating RMF should be used

CFMON is a great price-performer!

## CF Monitor - Option Menu

Version 1.1.0

Option ==>

Select one of the following options:

- 1 - Coupling Facilities
- 2 - Defined Structures
- 3 - Allocated Structures
- 4 - Connections

---

Current time	99.227	07:48	Elapsed secs	409981
Monitor start time	99.222	13:51	CPU time secs	2493
Monitor cycle time	9	secs	Monitor status	SLAVE
			Monitored Systems	1

SYSPLEX ID	P390PLEX	HRDW name	
SYSID	P390	LPAR name	
SYSTEM	SP6.0.6	CPU model	7490
CFS	2	Structures	4

CF Monitor 1.1.0 - Compiled at 06/11/98

# CFMON - Option Menu

```

                                CF Monitor - Option Menu                                Version 1.1.0
Option ==>
    CF Monitor - Coupling Facilities Activity Summary                                Row 1 to 2 of 2
                                                                                               num-sys 1
Command ==>
    CFRM policy                                POLICY1                                System P390__
                                                                                               Activated at: 99.222 19:12:20
                                                                                               Updated at: 99.222 19:10:54

Enter T on command line to display connected Systems
Select A for Allocated Structures                                C for Connections
       F for CF activity                                        S for Systems activity

<--- last cycle 9 sec ----- interval 114:00:38 secs 410438 ---->

      <----- sync -----> <----- async ----->
      bsy max stg total req interval dly chg total req interval que nsc
      CFname % bsy % avg % % avg % %
      - - - - - b--m--k--- -r/t- -rate -- -- b--m--k--- -r/t- -rate -- --
CF01 0 1 20 82044 114 0 0 0 368498 6039 1 0 0
CF02 0 3 19 195745 271 0 0 0 929899 5350 2 5 5
  
```

```

                                Systems Activity Summary                                Row 1 to 1 of 1
                                                                                               num-sys 1
Command ==>
    Enter S to select a System for Connections                                System P390__
CFname CF02 Busy % 0 ( 3 ) CP 1 Used space K: 48640 19 % CF lvl. 6
<----- last cycle 9 sec ----- interval 115:37:57 secs 416277
      <----- sync -----> <----- async ----->
      total req total req interval dly chg total req interval que ns
system 1151864 203404 avg 0 % % 948460 avg 2 % 5
      - - - - - b--m--k--- b--m--k--- -r/t- -rate -- -- b--m--k--- -r/t- -rate -- --
_ P390 1151864 203404 271 0 0 0 948460 5342 2 5 5
  
```

# CFMON - System Activity Summary

CF Monitor - Option Menu

Version 1.1.0

## Systems Activity Summary

Row 1 to 1 of 1

num-sys 1  
System P390\_\_

Command ==>

Enter S to select a System for Connections

CFname CF02 Busy % 0 ( 3 ) CP 1 Used space K: 48640 19 % CF lvl. 6  
 <----- last cycle 9 sec ----- interval 115:37:57 secs 416277

	total req	total req	interval	dly chg	total req	interval	que ns
system	1151864	203404	avg 0	% %	948460	avg	2 % 5
- - - - -	b--m--k---	b--m--k---	-r/t-	-rate --	b--m--k---	-r/t-	-rate --
_ P390	1151864	203404	271	0 0 0	948460	5342	2 5 5

## Connections Display

Row 1 to 2 of 2

num-sys 1  
System P390\_\_

Command ==>

Interval start 99.222 13:51 end 99.227 09:43:02  
 Interval length 115:47:59 secs 416879 <- last cycle 9 secs ->  
 Enter V structure-name on command line to view a structure, R to reset  
 Enter C on command line to see connection name and stats for last cycle

CF busy%	0	type	stat	total req	<- sync	interval	async ->	cf
structure name	] cf-name	sysid	] 1154591	0	avg	3	avg lv	
- - - - -	- - - - -	- - - - -	- b--m--k---	-rate	-s/t-	-rate	-r/t---	
IXCLIST2	L	CF02	P390	A 720046	0	0	2 5807 1	
JES2CKPT1	S	CF02	P390	A 434545	0	271	1 3883 0	



# CFMON - Allocated Structures

CF Monitor - Allocated Structures

Row 1 to 4 of 4

Command ==>

SYSID P390

CFRM policy

POLICY1

Activated at: 99.222 19:12:20

Updated at: 99.222 19:10:54

Enter L Structure-name on command line to locate a Structure

Enter S to select a Structure for Connections

```

-----
S Structure name  CF name      Status      Total req  TOD alloc.  Type  Size-K  RB%
-----
_ ISGLOCK         CF01        ALLOCATED   83745     99.222 18:40          LOCK   10240  1
_ IXCLIST1        CF01        ALLOCATED  379812    99.222 18:39          LIST   40192
_ IXCLIST2        CF02        ALLOCATED  730261    99.222 18:39          LIST   40192
_ JES2CKPT1       CF02        ALLOCATED  451188    99.222 19:16          LIST   6400
  
```

Structure Display -

SYSID

P390

Row 1 to 2 of 2

Command ==>

Structure JES2CKPT1

CF CF02

Allocated at: 99.222 19:16:02

Size,K 6400

Entries 1515

In use 158 Highest 158

CF % 2

Elements 1495

In use 158 Highest 158

Type Serialized LIST

Lock-ent 2

In use 0 Highest 1

Preference list CF02

Exclusion list

Disposition KEEP

Max connections

32

Access time

0

```

-----
Sel Connection Name  -Cycle rate- Interval
                   Sync  Async  Tot reqs  Sysname  Jobname  ASID  Status  CFL
                   b--m--k---
_ JES2_P390          1      1    451698  P390    JES2    0017  ACTIVE  0
_ JES2_P392          0      0         0  P392    JES2    0017  ACTIVE  0
  
```

# CFMON - Structure Activity

```

Structure Activity from System      P390
Command ==>
Cycle time          9              secs      Interval start 99.222 13:51          end 99.227 11:32:35
-----
Structure           JES2CKPT1      CF          CF02        Allocated at: 99.222 19:16:02
Size,K   6400        Entries     1515        In use     158        Highest    158
CF %    2           Elements    1495        In use     158        Highest    158
Type Serialized LIST      Lock-ent    2           In use     0          Highest    1
Disposition KEEP      Max connections 32        Access time          0
=====
Sysid   P390        Asid       0017        Jobname    JES2        Status     ACTIVE
      <----- last cycle -----> <--- Interval 117:37:25          secs 423445
      Total req 452837
SYNC s-t(MIC) 272      Rate 0     SYNC req 212816          % 46      Avg s-t    271
ASync s-t(MIC) 3200    Rate 1     ASync req 240021        % 53      Avg s-t    763
-----
External req 452762
Async  queued  53180      amortized q-t 117
Sync   changed  14          % 22 Avg q-t 531
Delayed no-sch 53182      % 0 <-incl in ASync
Req. contention 82          % 22
                                % 0.0

```

# CFMON - Connections Display

## CF Monitor - Connections Display

Row 1 to 4 of 4

Command ==>

```

Interval start 99.222 13:51 end 99.227 11:39:40 SYSID P390
Interval length 117:44:28 secs 423868 <- last cycle 9 secs ->
Enter V structure-name on command line to view a structure, R to reset
Enter I on command line to see CF-name sysid and stats for interval
structure name type stat total req <- sync .cycle. async -> cf
] connection name ] 1650666 0 3 lv
-----
IXCLIST2 L SIGPATH_010000A9 A 731802 0 0 2 6305 1
JES2CKPT1 S JES2_P390 A 454027 0 278 1 3262 0
IXCLIST1 L SIGPATH_010000A9 A 380950 0 0 0 0 1
ISGLOCK K ISGLOCK#P390 A 83887 0 112 0 0 0

```

## Connection Activity

Command ==>

```

Cycle time 9 secs Interval start 99.222 13:51 end 99.227 11:45:12
-----
Structure JES2CKPT1 Type S-LIST CF CF02
-----
Sysid P390 Asid 0017 Jobname JES2 Status ACTIVE
<----- last cycle -----> <--- Interval 117:49:56 secs 424196 ---->
Total req 454949
SYNC s-t(MIC) 281 Rate 1 SYNC req 213812 % 46 Avg s-t 271
ASYNC s-t(MIC) 2802 Rate 1 ASYNC req 241137 % 53 Avg s-t 3764
-----
Async queued 53425 % 22 amortized q-t 117
Sync changed 14 % 0 Avg q-t 531
Delayed no-sch 53427 % 22 <-incl in ASYNC
Req. Contention 82 % 0.0
External req 454874

```

## **9672 G5 and G6 processors support Capacity Upgrade on Demand (CUoD):**

- Add one or more CPs without IPL or POR
- Driver 22 required (June 99)

**After Dynamic CPU Upgrade traditional sources of CPU model identification will not be updated until an IPL is performed**

## **New facilities available to provide information:**

- STSI instruction
- CSRSI service
- ENF

## **ENF support:**

- Request Code 20 qualifier X'80000004'
- Signal initiated by Machine Check Handler

## STIDP:

- Privileged instruction
- Returns CPU type, model code, and serial number in a doubleword
- Example:

```
STIDP    SAVE_CPUYPE
```

```
SAVE_CPUYPE = FF01234596720000
```

```
Model code    =    X'FF'  
Serial number =    X'012345'  
CPU type      =    X'9672'
```

## Information returned by STSI *does not* exactly match STIDP information

- Last half word from STIDP is not documented
- STSI returns much more information

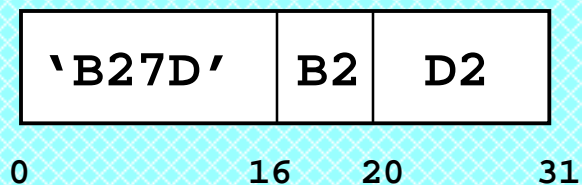
## STSI:

- Privileged instruction
- Documented in “dash six” version of Principles of Operations
- Returns environmental information for basic mode, LPAR mode, and VM
- S0C1 diagnostic aid
- Information returned to a System Information Block (SYSIB)
- Target (SYSIB) address must begin on a page boundary

## Condition Codes:

- 0 - Information returned or CPCV value returned in GPR 0
- 3 - Requested information is not available

## STSI D2(B2)



## Specify CPCV, Selector 1, and Selector 2

**Central-processing Complex Variety (CPCV) is identified in GPR 0 bits 0-3:**

- 0 = Request to return highest CPC in GPR0
- 1 = Basic Mode (V1 CPC)
- 2 = LPAR Mode (V2 CPC)
- 3 = VM (V3 CPC)

**Selector 1 specifies major CPC component in GPR 0 bits 24-31:**

- 1 = CPC
- 2 = CPU

**Selector 2 indicates the level of information requested in GPR 1 bits 16-31:**

- 1 indicates:
  - ▲ When Selector 1 = 1: General CPC information
  - ▲ When Selector 1 = 2: Single-CPU general information
- 2 multiple CPU general information; only valid when Selector 1 = 2

**Selector 3 is unused but is defined in POO**

**A callable service that returns SYSIB information:**

- **OW38489 (7/99)**
- **PTFs back to 6J1 (???)**
- **(Obviously) issues STSI under the covers**

**CSRIIDC maps SYSIB (for requests supported by CSRSI)**

**Why use CSRSI?**

- **Available to problem state programs**
- **Available to HLL programs (C declares in Samplib)**
- **Can return information for multiple environments from a single call**

**Unlike STSI, target address does not have to be on a page boundary (implies data is copied into caller's storage)**

# STSI (Store System Information)

CPC	Mode	Sel 1	Sel 2	GPR 0	GPR 1	Dsect
V1	Basic	CPC		X'10000001'	X'00000001'	SI11V1
V1	Basic	CPU	Curr	X'10000002'	X'00000001'	RYO
V1	Basic	CPU	Mult	X'10000002'	X'00000002'	SI22V1
V2	LPAR	CPU	Curr	X'20000002'	X'00000001'	RYO
V2	LPAR	CPU	Mult	X'20000002'	X'00000002'	SI22V2
V3	VM	CPU	Curr	X'30000002'	X'00000002'	SI22V3

CSRIIDC does not provide a Dsect for SYSIB 21

# STSI Coding Example

```
*      +-----+
*      +                Check STSI Availability                +
*      +-----+
L      R1,16(0,0)                load cvt addr
USING CVT,R1                    map cvt
L      R1,CVTSCPIN              load sccb addr
DROP  R1                        drop cvt mapping

USING SCCB,R1                   map sccb
TM     SCCBCON5,SCCBSTSI        stsi available?
BNO    NO_STSI                  bif no
DROP  R1                        drop sccb mapping

*      +-----+
*      +                Obtain Storage for SYSIB                +
*      +-----+
LH     R0,=Y(4096)              load length for obtain
STORAGE OBTAIN,LENGTH=(0),ADDR=(8),LOC=(ANY,ANY),COND=NO,      x
      BNDRY=PAGE
LTR    R15,R15                  work ok?
BNZ    OBTAIN_ERROR             bif no

LA     R0,4095(,R8)             calc end addr
ST     R0,END_ADDR              save it

MODESET MODE=SUP                switch to supervisor state
```

# STSI Coding Example

```
SR      R0,R0                query environment
STSI    0(R8)                doit
BNZ     STSI_ERROR          bif problemo
ST      R0,HIGHEST_CPV      save cpv value

LA      R1,SI00CPCVARIETY_V1CPC_MACHINE rqst basic mode info
SLL     R1,28                shift into position
LA      R0,STSI_CPC(,R1)     rqst cpc info
LA      R1,STSI_CPC_GENINFO  rqst info from single cpu
SR      R15,R15             allow breakpoint
STSI    0(R8)                doit
BNZ     STSI_ERROR          bif problemo

USING   SI11V1,R8           map sysib
MVC     CPU_MAKER,SI11V1CPCMANUFACTURER  manufacturer
MVC     CPU_MODEL,SI11V1CPCTYPE          cpu type (ex: 9672)
MVC     CPU_TYPE,SI11V1CPCMODEL          cpu model (ex: RD6)
MVC     CPU_SEQ,SI11V1CPCSEQUENCECODE    sequence code
MVC     CPU_PLANT,SI11V1CPCPLANTOFMANUFACTURE  plant code

L       R2,END_ADDR          load end address
PGSER   R,RELEASE,A=(8),EA=(2)  clear sysib
```

# STSI Coding Example

```
C      R1,HIGHEST_CPV      check for vm
BH     NO_VM              bif not available
LA     R1,SI00CPCVARIETY_V3CPC_VM  rqst vm info
SLL   R1,28              shift into bit 0
LA     R0,STSI_CPU(,R1)   rqst cpu info
LA     R1,STSI_CPU_MULT  rqst info from all cpus
SR     R15,R15           allow breakpoint
STSI   0(R8)             doit
BNZ   STSI_ERROR        bif problemo
USING  SI21V2,R8        map sysib
```

```
SAVEAREA    DS      18F      savearea
END_ADDR    DS      F        sysib high addr
HIGHEST_CPV DS      F        cpv returned by 'request'
CPU_MAKER   DS      CL16     manufacturer
CPU_TYPE    DS      CL4      cpu type
CPU_MODEL   DS      CL16     cpu model
CPU_SEQ     DS      CL16     cpu sequence code
CPU_PLANT   DS      CL4      plant
WALEN       EQU     *-WORKAREA  workarea length
```

```
CSRSIIDF
CVT DSECT=YES,PREFIX=NO
IHASCCB
```

# CSRSI Coding Example (Untested)

```

L      R1,16(0,0)          load cvt addr
USING CVT,R1              map cvt
TM     CVTOSLV4,CVTCRSR1  csrsi available?
BNO    NO_CRSR1          bif no

LOAD  EP=CSRSI,ERRET=NO_LOAD  load csrsi
SLL   R0,1
SRL   R0,1                clear hob
ST    R0,CSRSI@          save csrsi addr

LH    R0,=AL2(X'4040')    obtain info area
STORAGE OBTAIN,LENGTH=(0),ADDR=(8),LOC=(ANY,ANY),COND=NO,      x
      BNDRY=PAGE

ST    R8,CSRSI_INFO@     save info area addr
MVC   CSRSI_TYPE,=AL4(Csrsi_Request_V1CPC_Machine)
MVC   CSRSI_INFOLEN,=X'00002040'  set return area length
XC    CSRSI_RC,CSRSI_RC   set rc to zero
L     R15,CSRSI@         load csrsi addr
CALL  (15),(CSRSI_TYPE,CSRSI_INFOLEN,CSRSI_INFO@,CSRSI_RC)
LTR   R15,R15           work ok?
BNZ   BAD_CRSR1        bif no
USING SIV1,R8          map sysib

```

```

CSRSI_TYPE    DS    F    csrsi type indicator
CSRSI_INFOLEN DS    F    length of info area
CSRSI_INFO@   DS    F    info area addr
CSRSI_RC      DS    F    return code
CSRSI@        DS    F    csrsi routine addr

```

# STSI V1 11

+32	DS	CL16	Manufacturer
+44	DS	CL4	CPU Type
+64	DS	CL16	CPU Model
+80	DS	CL16	Sequence Code
+96	DS	CL4	Plant Code

07135000.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
07135020.	C9C2D440	40404040	40404040	40404040	*IBM*
	F9F6F7F2	00000000	00000000	00000000	*9672.....*
07135040.	D9C4F640	40404040	40404040	40404040	*RD6*
	F0F0F0F0	F0F0F0F0	F0F0F0F1	F2F3F4F5	*0000000000012345*
07135060.	F0F24040	00000000	00000000	00000000	*02.....*
	00000000	00000000	00000000	00000000	*.....*
07135080.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
071350A0.	00000000	00000000	00000000	00000000	*.....*

# STSI V1 21

+80	DS	CL16	CPU Sequence Code
+96	DS	CL4	Plant of Manufacture
+102	DS	H	CPU Address

```
07102000. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000000 *.....*
07102020. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000000 *.....*
07102040. 00000000 00000000 00000000 00000000 *.....*
           F0F0F0F0 F0F0F0F0 F0F0F0F0 F0F2F000 *0000000000000020.*
07102060. F1F04040 00006DD4 00000000 00000000 *10  .._M.....*
           00000000 00000000 00000000 00000000 *.....*
07102080. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000000 *.....*
071020A0. 00000000 00000000 00000000 00000000 *.....*
```

# STSI V1 22

+32	DS	F	CPU Capability
+36	DS	H	Total CPU Count
+38	DS	H	Configured CPU Count
+40	DS	H	Standby CPU Count
+42	DS	H	Reserved CPU Count
+44	EQU	*	Multiprocessing CPU Capability Adjustment Factors

07102000.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
07102020.	00001E10	00050004	00000001	005C005A	*.....*!*
	00580054	00000000	00000000	00000000	*.....*
07102040.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
07102060.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
07102080.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
071020A0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
071020C0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
071020E0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*

# STSI V2 21

+80	DS	CL16	Logical CPU Sequence Code
+96	DS	CL4	Plant of Manufacture
+100	DS	H	Logical CPU ID
+102	DS	H	Logical CPU Address

1075A000.	00000000	00000000	00000000	00000000	* .....
	00000000	00000000	00000000	00000000	* .....
1075A020.	00000000	00000000	00000000	00000000	* .....
	00000000	00000000	00000000	00000000	* .....
1075A040.	00000000	00000000	00000000	00000000	* .....
	F0F0F0F0	F0F0F0F0	F0F0F0F0	F0F5F600	*000000000000056.*
1075A060.	F1F04040	00010000	00000000	00000000	*10 .....
	00000000	00000000	00000000	00000000	* .....
1075A080.	00000000	00000000	00000000	00000000	* .....
	00000000	00000000	00000000	00000000	* .....
1075A0A0.	00000000	00000000	00000000	00000000	* .....
	00000000	00000000	00000000	00000000	* .....
1075A0C0.	00000000	00000000	00000000	00000000	* .....
	00000000	00000000	00000000	00000000	* .....
1075A0E0.	00000000	00000000	00000000	00000000	* .....

# STSI V2 22

+32	DS	H	CPU Number
+35	DS	XL1	Logical CPU Characteristics
+36	DS	H	Total Logical CPU Count
+38	DS	H	Configured Logical CPU Count
+40	DS	H	Standby Logical CPU Count
+42	DS	H	Reserved Logical CPU Count
+44	DS	CL8	CPC Name
+52	DS	F	Capacity Adjustment Factor
+72	DS	H	Dedicated Logical CPU Count
+76	DS	H	Shared Logical CPU Count

1075A000.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A020.	00010040	00020002	00000000	E2E8E2C1	*... ..SYSA*
	40404040	000000FA	00000000	00000000	*.....*
1075A040.	00000000	00000000	00000002	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A060.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A080.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A0A0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A0C0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*
1075A0E0.	00000000	00000000	00000000	00000000	*.....*
	00000000	00000000	00000000	00000000	*.....*

# STSI V3 22

```
+31 DS X11 Description Block Count
+32 EQU * Description Block
```

## Description Block

```
+4 DS H Total Logical CPU Count
+6 DS H Configured Logical CPU Count
+8 DS H Standby Logical CPU Count
+10 DS H Reserved Logical CPU Count
+12 DS CL8 CPC Name
+20 DS F Capacity Adjustment Factor
+24 DS CL16 VM/HP Identifier
```

```
07102000. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000001 *.....*
07102020. 00000000 00040004 00000000 C5E3D7D6 *.....ETPO*
           E2C2D940 000000FA E5D461C5 E2C14040 *SBR ....VM/ESA *
07102040. F24BF34B F0404040 00000000 00000000 *2.3.0 .....*
           00000000 00000000 00000000 00000000 *.....*
07102060. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000000 *.....*
07102080. 00000000 00000000 00000000 00000000 *.....*
           00000000 00000000 00000000 00000000 *.....*
071020A0. 00000000 00000000 00000000 00000000 *.....*
```

## **Bibliography**

**“ESA/390 Principles of Operation”, SA22-7201-6, IBM Corporation**

**“OS/390 Authorized Assembler Services Reference Volume 2”, GC28-1765, IBM Corporation**

**“Parallel Sysplex Coupling Facility Online Monitor Installation and User’s Guide”, SG24-5153, IBM Corporation**

**Contributors, willingly or unwillingly, knowingly or unknowingly:**

**Bruce Black, Innovation Data Processing**

**Jerry Dearing IBM Corp**

**Harv Emery, IBM Corporation**

**Terry Fong, State of California HWDC**

**Ed Jaffe, Phoenix Software International**

**Doug Nadel, IBM Corp**

**Brian Peterson, St. Paul Fire & Marine Ins Co**

**Deb Carnes, The Boeing Company**