VOL. 4

33-986

50 Sheets 11" x 8½" College Ruled

NATIONAL

NATIONAL BLANK BOOK COMPANY, INC. • HOLYOKE, MASS. 01040

MADE IN U. S. A.

B. Page

(312) 692-7520
Message board
Classroom 1

11/12/84

XA Differences

Don Wallin

MVS/SP 1.3 is the positioning level for XA.
XA can address 128 times as more storage.
VSAM has been modified to take advantage of XA. The other access methods have not.

True VSCR is only available by using the extended private area.

MVS/SP 2.1.1 - has lots of new address spaces:

SMF

CNRLST

MVS/SP 2.1.2 Primary feature is ASM improvement. The ASM improvements are so good, that they were applied to 2.1.1 + SP 1.3 via PTFs:

2.1.1 - OZ75721
UZ90290

1.3 OZ75720
UZ90289

Storage layout: the system area straddles the 16meg line.

Conversion can be fairly invisible to applications. However, this does not mean the applications will take advantage of XA features. XA may require about 2 meg more real storage.

[SHARE recently had a Nickle & Dime approach to VSCR under MVS/SP]

XA Instructions & 31 bit programming

New instructions:

TRACE	99	- Trace
SIE	B214	- Start Interpretive Execution
		- Not used by MVS. (VM)
BASSM	ØC	- Branch + save + set mode
BSM	ØB	- Branch + set mode
DXR	B22D	- Divide (extended precision)
IPM	B222	- Insert Program mask.

Plus several new I/O instructions. (The old ones are gone entirely).

The 370 branch and load instructions have changed. The instructions function differently depending on the execution mode of the machine. In 370-mode they function as before.

2 gig = 2,147,483,647 bytes
31 bit address layout:

a sssssssssss pppppppp dddddddddddd

a = Ø or 1 = 24 or 31 bit mode
s = segment. 2048 segments of 1 meg. size
p = page number. 256 pages/segment
d = displacement within 4K page.

The DAT process is the same except multiply the page number by 4 rather than 2.
All but 16 of the 2048 segments are above the line. Page tables occupy 8 meg!

XA requires a new format PSW
No BC mode psws under XA. Bit is always 1.
Difference:
Bit 32 is the amode bit (tells DAT whether virtual address is 24 or 31 bits in length).

BASSM is the primary branch command. RR instruction. This saves the execution mode of the brancher as well as the return address. Use BASSM instead of BAL and BALR.
When BASSMing from 31 to 24 bits, the high order byte of R2 is not used in loading the new PSW.

BSM - the way to get back without a return. The amode of the branching routine is placed in R1. The target address should reside in R2 before execution. If either reg. is ≠ zero, then that particular function is not performed. This may be used to check your own amode.

BAS and BASR replace BAL and BALR, but the latter two still work. These deal only with the right-most 31 bits. They leave the amode bit unchanged. Thus, execution mode will not change.

LA operates differently depending on mode. LA loads either 24 or 31 bits. It does not load the amode bit.

LRA - load real address always returns a 31 bit address. The execution mode governs the input to DAT (whether 24 or 31).

Page tables for the area below 16 meg are fixed in XA. Above, they are pageable.

IPM is used to recoup some of the information (cc, pgm mask, & ilc) lost when BAL & BALR are executed in 31-bit mode.

Assembler statements:

Amode - mode program expects to receive control in.
Rmode - where to load: below or anywhere.
RSECT - read only section. Specified for parts of Nuc.

These statements can be coded in source, in the linkage editor parm values, or even in the loader parm values.

Amode 24 -
Amode 31 -
Amode ANY - runs in either mode; preferable.
Rmode 24
Rmode ANY

If nothing is specified, Amode & Rmode are set to 24. Rmode 24 is the default unless Amode 31 is specified.
Assembler H V2 is required; as is the XA linkage editor. (Lked is part of DFP now)
At link edit, if one csect is Rmode 24, then the whole module goes below.
The SVC table now contains an amode bit. This is set at load time.
IOHALT (SVC33) requires reassembly.

GETMAIN now has some new parms.
VRC/VRC cause SVC 120 to be generated rather than 4+10.

A checkpoint taken on one system (SP/XA) will not restart on the other - a recovery consideration.

CCW Ø format only holds a 24 bit address. This is the IDAW which points to the IDAW.

XA has several new macros:

- CPOOL - for cell pool manipulation
- CPUTIMER -
- DATOFF - to create DAT-off modules.
- ESPIE - can trap your own page faults.
- NUCLKUP - for module look-up (since NUC is scatter-loaded)
- PGSER - address the paging services
- PTRACE - trace entry
- RACROUTE - RACF ...
- SMFIOCNT -
- SPLEVEL - can specify the old version of macros. + new assembler.
- SVCUPDTE - dynamic SVC changes.
- VSMLIST - VSM mapper
- VSMLOC - virtual verifier
- VSMREGN - maps TCB storage.

ASM is now of higher priority than the dispatcher.
Storage protection is now done on the page level, not segment. Read-only is enforced for the read-only nucleus
resident BLDL
PLPA/MLPA/FLPA
NUC map

CVTMVSE (+X'74') indicates whether processor was IPLed in XA mode.

XA has 157 user exits (not counting any JES exits).

IOCP

The IOCP macros are processed by the IOPIOCP to build the IOCDS. At IML time the IOCDS is used to build the Hardware System Area (HSA) in real storage.
UCWs and CUCWs are located in the HSA and define devices and control units, respectively.
HSA will hold 4096 UCWs and 256 CUCWs.

Chan path id (chpid)	8 bits
subchan #	16 bits
Device addr	8 bits
Dev number	16 bits

IOS only knows about the device number and subchannel number. DCS converts the device # into a CHPID & Dev. addr.

CHPIDs 00-07
10-17
20-27
40-47
50-57
60-67

The subchannel number is the index into the UCWs. subchannel is 3 hex digits. HSA has 4096 UCWs.

Channels and device addresses are no longer directly referenced.

XA handles Device Queuing via IOQ control blocks, Device types, interrupt processing, and sampling at the path level for SRM.
The channel subsystem handes:

Logical path queue
Path selection
Chan types
Control Unit type (data stream or OC interlock)
Device type
Rejected I/O redrive
Availability
Measurement

and the traditional funtions of Bus + Tag handling, UCW handling, + CCW fetch.
This split increases parallelism between the CPU and the channel processors.

IOCP deck:

Channel paths
Control units (also specifies data streaming or OC interlock)
I/O Devices
Preferred paths

IOCP comes in two versions, MVS and VM, which are interchangable.

CHPID macro:

```
CHPID  PATH=(xx, y, z),        xx = chan #, y = chan set  } for use when IPLed in 370 mode.
       TYPE = BY|BL
```

alternate paths:

```
CHPID  PATH=((05,1,0),(07,1,1),(06,2,0))
```

Control unit macro

```
CNTLUNIT  CUNUMBR=xxx,        ← arbitrary
          PATH=(xx,xx,xx,xx),
          PROTOCL=D|S         D = DC interlock; S = data streaming
          SHARED = Y          } no concurrent activity = tape
                   YB         } documentation only
                   N          } concurrent activity - DASD
          UNIT=               - unit type
          UNITADD=(xx,xx,xx,xx) - the wired address.
```

CNTLUNIT is mainly needed to know which paths pass through it.

I/O Device macro

```
IODEVICE  ADDRESS=xxy     - Device Number, wired by FE
          CUNUMBER=       -ties to control unit macro
          MODEL=
          PATH=xx         -preferred path
          TIMEOUT=Y|N     -8.3 sec
          UNIT=           -device type; not used
          UNITADD=xx      -device address; optional
                  If UNITADD is omitted, the right
                  two digits of ADDRESS is used.
```

Sysgen and IOCP can be combined in one deck, but be sure to code IGNORE=YES. Best to keep separate.

UZ90215 must be on to let IOCP work right for XA.

A Logical Control Unit is represented by a CUCW in DCS to queue requests.

The Device address & number can be different when a processor has more than 15 channels

The CUNUMBR is used by DCS in its rotation algorithm.

11-13-84

Channel Subsystem

Hierarchy:

1. Access method - VSAM
2. IOS Drivers - not much change
3. IOS - greatly changed
4. DCS - the same whether XA or 370
5. hardware

get → DCS publications:

IBM Journal of R&D, May 1983; "XA Channel Subsystem." Vol. 27, No. 3

SIO and SIOF are gone with XA. They are replaced in IOS by SSCH (Start Subchannel). Subchannel number is generated when the HSA is built at IML time. These numbers are plugged into UCBs at IPL.

Subchannel numbers are assigned one-for-one with devices. There is no relationship implied to channels or path. They are simply sequentially assigned.

The new I/O instructions do not make reference to devices. They handle channel paths and subchannels.

With XA IOS, once IOS is disabled for interrupts can remain disabled and then query DCS for additional interrupts (through Test Pending Interruption instruction). This saves significant overhead involved in interrupt processing.

The CAW is replaced by an Operation Request Block (which may be placed anywhere in memory). The CSW is replaced by the Subchannel Status word. The SSW contains 12 bytes of status.

Due to DPR, IBM says that channel busy can now be 70% if four paths are available.

I/O Instructions

Privileged and S-format. Op code: B23x
R1 points to UCB (DCS number, subchan #).

Start Subchan - points to ORB (which points to UCB, & the CCWs via real address.
RSCH - Resume subchan
HSCH - Halt Subchan
CSCH - Clear Subchan - if Halt fails, clear is issued.
STSCH - Store Subchannel - issued by XIH and points to Subchan info block (SCHIB).

MSCH - Modify Subchannel - permits IOS to change flags. Used by DCTEP, for instance, to alter path for diagnostic reasons. Executes synchronously.

RCHP - Reset Channel Path - used by channel check handler. Executes async & causes a machine check when DCS informs CPU that reset has been accomplished. MCIH will figure out that this return is a pseudo-machine check.

STCRW - Store Channel Report Word

STCPS - Store Chan Path Status - SRM and RMF use this to measure busyness.

SCHM - Set Channel Monitor - R1 points to ORB, R2 points to MBO (Measurement Block Origin). Used by SRM. Directs DCS to the MBO as a repository for the measurement info that it gathers. This enables for measurement. It does not actually initiate measurement by DCS. Each MBE is 32 bytes long and contains:

- Dev. connect time
- Function pending time
- Dev. disc time
- SSCH Count

TPI - Test Pending Interrupt - issued by disabled IOS. DCS replies with a pointer to the UCB for which it has a pending interrupt. This saves a PSW swap and status saving involved in interrupt processing.

TSCH - Test Subchannel - would be used following an interrupt or after a TPI. This brings in the IRB (Interruption Response Block), which contains the status of the I/O.

When the IRB is collected, DCS frees the path for re-use.

Subchannel Information Block - (SCHIB)

The SCHIB is a copy of the UCW when a store subchan is performed. A SCHIB is moved into the UCW when Modify subchan is performed. The SCHIB points to the UCB. It has an Interrupt Subclass Code which permits paging I/O interrupts to be distinguished from all other I/O interrupts. If SRM is requesting measurement then the MM bit will be set. Another field will point to a Measurement Block Entry. The SCHIB contains a PMCW (Path Management Control Word), SCSW (Subchannel Status Word), and a block for model-dependent information. Some UCW fields are not present in the SCHIB.

CCWs come in two formats, Ø and 1. The fields are the same but rearranged. These CCWs do not identify themselves as Ø or 1, this information must be picked up elsewhere.

Only one CPU can be busy with a particular subclass of interrupts at a time. The other(s) mask for that class in CR 6. IOS takes care of the masking. In practice, one processor will tend to handle all paging interrupts and another will handle all other (subclasses 3 and 5, respectively).

Channel Measuring:

Chan measurement block lives in real memory. The MBE holds counts of the number of instances when the device is in various conditions.

Chan Report Word - this is the response to Reset Chan Path. Completion is indicated by a machine check.

XA IOS

Little change in STARTIO and EXCP. Lots of changes to UCBs. The new IOS modules begin with IOSV. Old modules were IEC.

LCH is gone. If UCB is busy, I/O is queued off of UCB.

Any processor can start I/O to any device. No shoulder tapping. IOS makes no path management decisions. This is handled by DCS.

Interrupt processing is similar. Status is returned in the IRB, not a CSW. Any processor may be interrupted. Interrupts may be fielded while IOS is disabled (TPI). Two stage process to recognized interrupt & get status.

The IRB of every interrupt is examined.

At NIP time only the sysgened UCWs are initialized. Done by STSCH and MSCH. UCWs are retrieved one at a time. If a UCB exists for it, the UCW is returned via MSCH. If no UCB exists, the next UCW is retrieved & the process conditions.

UCWs are all linked together and are in device class queues. Queue names: Tape, COMM, DASD, DISP, UREC, CHAR, & CTC.

Diagnostics

One IOWA per processor (LCCA + 2E0 is ↑)

IOWA holds:

- IRB
- ORB
- workareas

No CAW, see ORB

No CSW, use IRB (IRB may be incorrect)

I/O queued to UCB+ indicates type of I/O.

Missing Interrupt Handler

Times have been reduced significantly.

MIH needs to know what type of I/O has been initiated to the subchannel.

Dynamic Path Reconnection (DPR)

Requires XA and 3380 type device.

CPU

path { Channel
storage Director
Head of string

string { Actuator
Actuator
:

String switching enhances availability, not performance.

The head of string knows which system sent the reserve and can use this to pick another path for return.

A CPU identifies itself by sending its CPUid and IPL timestamp down every path. Every head of string which receives this knows it has a path to that CPU. This is a path group id.

Because of DPR, effected DASD can be 20 to 35% busy and be efficient.

Initialization

11-14-84

ACR is no longer a sysgen option. It will be there.

IEAIPL00 - clears storage, builds segment tables (8K) for each address space, initializes work areas, and the loads and deletes IRIMs.

IPL uses nucleus SVCs even though there is no SVC interrupt handler or SVC table. Some of these twelve SVCs are only used at this time. (SVC 0-11) Control is passed via a R+D table of twelve entries containing branch addresses.

MSSFCALL permits IEAIPL00 to query MSSF for some hardware specific information - NUC ID, location of HSA, etc.

Then memory is cleared all the way up to the HSA. This generates the usable Frame Queue.

IEAIPL02 brings in the nucleus (+ locating according to Rmode + Rsect). The modules in sysl. nucleus are scatter loaded.

IEAIPL05 builds a NUCMAP.

IEAIPL03 inits UCB for sysres volume

IEAIPL04 lays out virtual storage. It does not necessarily initialize any of the areas which it creates.

IEAIPL06 - the RSM IRIM; builds PFT for RSM. Also creates RIT + RAB but no initing.
IEAIPL07 - fills in entries in the SVCTABLE for the nucleus SVCs.
IEAIPL99 - IPL cleanup. Builds available frame queue. Passes control to NIP0
NIP0 inits lots of control blocks. Control then goes to RIMs.

An alt nuc must be specified at the system support console before IPL. Cannot be done a "specify System Parameters."
For a failure during IPL, check the SVC stack, PSA, + IPL Diagnostic Area.

Parmlib changes

New members:
ADYSETxx - dump analysis & elimination: permits dump symptoms to be recorded and match against specs, and then either kept or discarded.
CPACSTxx - defines lpa list.
IEACMD00 - another command lib. Don't use.
Deleted:
IEABLDxx
IEALOD00

Supervisor Services

New XA locks:

RSMGL	RSMAD
VSMFIX	RSM
ASMGL	VSMPAG
RSMST	TRACE
RSMCM	CPU
RSMXM	

New kind: shared/exclusive in addition to spin and suspend. (RSM & TRACE are of this type)
The lockword holds CPU identity.

CPU lock does not serialize anything.
It records requests to run disabled on a particular processer.

IOSYNCH has five new sub-uses. IOSCAT and IOSLCH are gone.

Storage locks have proliferated. (9)
RSM lock is lowest in priority & is shared/exclusive.
Must be held to get other RSM locks.

RSMST
RSMCM
RSMXM
RSMAD
} Hierarchical order; Each Address Space has these.
RSMST - high — Page steel
RSMXM — Cross memory
RSMAD - low — address space

RSMGL - the highest RSM lock. Global.

Rules:

1. Only one of each type lock per processor.
2. Only one lock of each type per address space.
   [Hierarchy is meaningless since only one can be held].

PSA HLHI is gone & replaced by PSA CLHS (Current locks held string) - no hierarchy implied.

SVC Changes

Amode is set in SVC table.
New SVCUPDTE macro permits SVCs to be dynamically added or changed. May only be issued by Authorized programs.

VSM

VSM is pretty much rewritten.

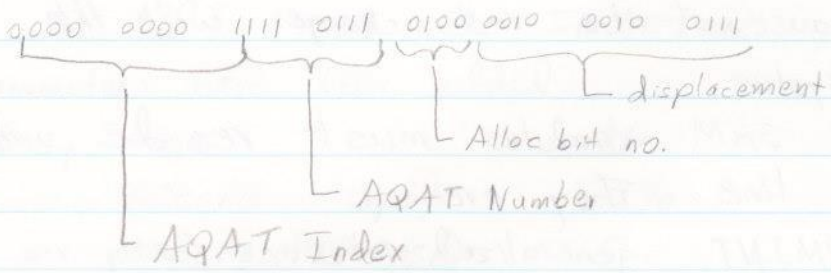
Extended Private	Ext LSQA/SWA/229-230 Ext user
Extended Common	ECSA EMLPA EFLPA EPLPA ESQA ENUC
Common	NUC SQA PLPA FLPA MLPA CSA
Private	LSQA/SWA/229-230 user area system region
Common	PSA

Control Blocks

For Getmain/Freemain problems, check VSWR fixed + pageable control blocks.

Virtual Address processing of FREEMAIN:

ØØ F7 42 27



AQATs and SQATs add more queues but the searches are faster.

VSM is documented in INIT + TUNING and Logic Library.

RSM

Greatly rewritten.

Page tables are pageable. The old paging macros still work but only with 24 bit support. One new macro replaces all functions (PGSER).

VARY STOR is replaced by CONFIG STOR.

3 flavors of page tables:

Lower - fixed in storage below 16 meg.

Higher - pageable table to describe the pageable page table which describes storage above 16 meg.

Top Page Table - fixed page table describing the higher page table.

SRM

11-15-84
George Fraley

SRM was totally re-written (in PLS 3).
SRM has no new actions but many algorithms have been reformulated.

The channel measurement procedure is totally different because of the rast changes with the channel subsystem.

Only 4 SRM modules must reside under the 16meg line. They are:

IRARMINT - generalized interface (entry via SVC 95).
IRARMFIP - SRB entry
IEAVNP1Ø - resource init routine for SRM.
IEAVNP1F - NUC resource init module (new w/ ZA).

The rest of SRM is Amode = 31 and Rmode = Any.

The new SRM has 3 new modules:

IEAVNP1F -
IRARMEV2 - expands SYSEVENT processing
IRARMCHM - channel measurement

All but two SRM modules begin with the characters IRAR.

New SYSEVENTS:

UCBCHG - updates UCBs in response to VARY or CONFIG. (sysevent 7Ø).
DDR (sysevent 71) - issued during device swap operations.
CHANNEL (sysevent 72) - channel check processing.
Turns off chan measurement facilities for permanent errors.

Parmlib:

No changes to the ICS.

IPS has added IOSRVC - I/O service rate which permits I/O counts or service time for devices which may be used for reporting.

IOSRVC = count | Time

In OPT, ICC and INIT are ignored. Several parameters have been added.

Busy times:	under	over utilized
ICCLPB (Tape)	50%	80%
ICCLPB (NDPSDASD)	15	30
ICCLPB (DPSDASD)	20	35
CPENABLE	10	30

- percent of time CPU enabled for interrupts. Another CPU will not be enabled for interrupts until the first is spending 30% of its time servicing interrupts. This deserves investigation for a particular environments.

SRM now sleeps for a little longer before interrupting.

Event Notification Facility (ENF) - SK 95 4/ a syserent code queues for handling by SRM.

IRARMEVD$_{01}^{DD}$ listens for sysevents. Then specific CSECTs handle each particular sysevent.

SRM only measures tape and DASD I/O devices.

SRM collects tape and DASD channel path info with modules IRARMCHM and IRARMFIP.

CHM is invoked every 3 SRM second, to collect chan info.

FIP - device ...

Evaluation is done by IRARMIOM to identify a significant users of a resource, device allocation, and logical path busy.

RMCT (Resource Management Control Table) is the main SRM control blocks. It points to many other control blocks. Start here for diagnostics. RMCT also holds Constants and Variables (identified by "C" and "V," respectively). The variables are input via parmlib.

IRARMFIP is entered every 200-250ms, and calls IRARMCPS which issues STCPS to store Chan Path status. Then, IRARMOBS is called to evaluate.

Control Blocks for Device + Path measurement:

CPMT - Chan Path Measurement Table:
- Sample count (as updated by DCS)
- # of times found busy

LPBT - Logical Path Block Table:
- Logical Path utilization
- Percent connect time
- Calculated status from last evaluation.

DMB - Dev. Measurement Block
- Pointer to UCB
- Request pending time
- Dev connect time
- Dev active time
- Sample count base
- UCB queue length

CMB - Chan Mea Block
- # of SSCH & RSCH instruction issued
- Dev connect
- Req. pending times
- subchan disc time

See Debugging Books and Vol. 12 of System logic for the better documentation.

RMF

a Program Product. RMF has been largely rewritten using PLS. It uses MSSF for info gathering. RMF also uses the SRM measurement blocks. It runs in Amode 31. The reports have been reformatted.

RMF collects data on a timed interval from

ICT + ASCB → I/O service

MCT + ASXB → storage management (AFQ, # pages stolen...)

CCT + OUCB → Processor info (MTW), (APG high, PG rate ...)

TCT + OUXB → for type 30 SMF records.

RMF issues 3 syseverts of its own (45, 46, 47).

RMF uses SVC 122 to read the IOCDS from MSSF and then construct control structures for all devices, paths, and CCUs. Most Chan data is taken from SRM and also uses the STSCH for subchan gaoving data. See p. RMF 0060 for a picture of I/O and the RMF labels. Copy this.

All RMF records have changed. Each has an SMF header, and an address pointer, length of section, and count of bytes in a floating section.

Monitor III - a realtime contention monitor. See GG22-9350.

SMF

No functional changes. With XA 2.1.1 SMF becomes a separate restartable address space. All SMF buffers are in the SMF private area. Gives some VSCR.

One new macro: SMFIOCNT now replaces IEFSMFEX (which counted EXCPs). Many SMF records have changed (particularly with Devices). Also, the accuracy has improved.

SMF is started through CMD00.
VSAM datasets may be used and are pre-formatted.
User entry is through SVC 83. In SRB or disabled mode, entry is through IEFU84.
Device address has been changed to device number.

Contents Supervision

Formerly called Program Manager.
Responsible for searching libraries for load modules, scheduling the loading of modules, maintaining the use count, and fetching modules. (Virtual fetch is available but only used by IMS).

Macros:

Link	Delete
Attach	Identify
Load	Synch
XCTL	Program Fetch

Module and messages are not prefixed by "CSV."
All can accept 31-bit addresses, but some live down under. All LPA modules must truely be reentrant.

Reason codes have been added to the messages.

With XA 2.1.1, the Linklist Lookaside address space was added. This holds PDS directories for linklisted libraries. Not all concatenations to linklib must be APF authorized. SVCLIB may be concatenated to linklist.

A new abend S023 has been added in support of the linklist lookaside address space.

Search Sequence

LLE Queue (searched for LOAD macro only).
JPA Queue
Job, step, task private libraries.
Private libraries are searched first if specified.
LPA Queue
SVCLIB (when OCB specified on macro)
LNKLST (in concatenation order)
S806 - if not found.

Libraries may be concatenated to LPALIB.

LNKLST LOOKASIDE - optional. In-stor hashed directory. XMS is utilized to find module and the Pgm transfer back. Can be stopped and refreshed - necessary if a library is compressed.
Four new modules support this.
commands: start - S LLA
stop - P LLA
refresh - F LLA, Refresh

Fetches occurring while stopped will cause physical I/Os to read the PDS directories.

No resident BLDL table. No tuning of LNKLST is necessary.

Once SMF, LLA, + Trace are shut down, their ASIDs are retired until the next IPL.

Linkage Editor

Now supports Amode, Rmode, and Rsect specifications.

Rsect only applies to Nuc modules. Rsect is specified in source code instead of Csect.

The linkage editor now requires 96K and is no longer in overlay. It should run faster.

MODE is a new control word.

In PDS directory record at +31 (PDS2FTB2) are bit settings to indicate Rmode and Amode.

At +32 (PDS2RLDS) is the number of RLD records.

Linkage Editor is part of DFP. The new 370 DFP preserves RLD counts + amode/rmode. The old Linkage will not.

RLD counts will be added just by re-linking a module with the new linkage editor.

Program Fetch

Now is Nucleus resident.
Uses format 1 CCWs. A single SSCH will read the text record and up to 12 RLDs. The CRLD contains the RLD count so PF can set up the CCW.

Linkage editor can put out 18K blocks. PF has a 64K buffer and can thus hold three text blocks. IEBCOPY can now reblock PDSs. This has performance implications.

See Linkage Editor Logic (LY26-3902) and Sys Logic Library (LY28-1228).

Unless programs are reblocked with IEBCOPY the old program fetch is faster. The new PF also attributes I/O to the user (to SMF), if the old modules are not provided with RLD counts.

Utilities

IBCDASDR, IBCDMPRS, and IEHDASDR are gone. IEBCOPY is modified. IEBIMAGE is new. The standalone utilities are the deleted utilities. IEBIMAGE support 3800-3.

IEBCOPY - Alter-in-Place will provide CESD with RLD counts (making the CESD into a CRLD). Use this on all old module libraries. Can also reblock load modules. Text blocks and RLDs are treated separately. Thus, the number of RLDs must be considered when deriving optimum block sizes. Requires magic.

Generic device names can now be specified since all utilities use the EDT for allocation instead of an internal (untouchable) table.

Ref: MVS/XA Utilities GC26-4018

DFP

DFP/370 is the positioning product for XA.
DFP 1.1 is the XA version and is different from DFP/370.

VSAM buffers may be above 16 meg.
To do, specify MACRF=AMODE31 on the ACB before the OPEN.

Numerous VSAM routines will accept 31-bit callers.
Addition VSAM return codes are provided.

See XA DFP Planning GC26-4040.

System Trace

11-15-84

Trace in 5370 had some problems. The new trace is designed to circumvent some of these problems.
With XA 2.1.1, trace got its own address space.

370 limitations:

32 bytes/entry was too small
The microcode had to be changed every time MVS control blocks were changed.
Serialization problems occurred with MPs.
Only 4 digits of the TOD clock were stored.

XA changes:

Implicite tracing of these instructions:

BACR	PC
BASSM	SSAR
BASR	PT

Recording is maskable.
Each processor has its own trace buffers.
Events are tied to a processor.
New formats: 32 bytes and 64 bytes.

CR12 contains the address of the next available trace buffer. It also contains bits which can mask the three kinds of trace:

1. Branch Tracing
2. ASN Tracing
3. Explicit Tracing

} if bit is Ø tracing of this type can occur.

PIC 16 is trace fault and occurs when the trace buffer reaches the end of a page.

XA has a new TRACE instruction.

GTF and trace can run concurrently.

A single trace address space has all of the trace buffers for all processors. Trace will validate and correct all trace control blocks - the ultimate paranoac.

Portions of the trace buffers relating to a particular address space can be selectively extracted.

Trace most always executes as a secondary address space. Users call trace.

The trace AS controls both system and master trace. Both traces default to being on. Status of trace may be adjusted dynamically.

The PTRACE macro may be used by pgms in supervisor state or in keys Ø to 7.

SNAPSHOT will provide the trace entries for an address space that is currently dumping. (May also be explicitely invoked without dumping).

JES2 +3 still have their own trace facilities.

23 bytes of PSA are given to trace for pointers + stuff:

PSATRCEL (+2D4) addr of trace lock

PSACLHS (2F8) Current locks held Ø4 = trace

PSATRACE (7EC) - trace active? ≠ '80' = No
PSATBVTR (7F0) - real & virt addr of TBVT
PSATBVTV (7F4)
PSATRVT - (7F8) addr of TRVT
PSATOT (7FC) addr of trace operand table.
PSATRSAV (+828) PTrace save area

CBs:
TBVT - trace buffer vector table
TRVT - trace vector table (↑ to trace routines)
TOT - trace operand table
TOB - trace options block (trace environment info).
TTCH - Trace table copy header - for data being copied out.

The Trace Operand is placed in the trace entry and determines what the entry describes. This is not as simple as scanning the trace entries a la s/370.

User routines may be added to trace. Entries may be printed in either hex or in both hex & character.

Two new abend codes: S09D = bad PTRACE parameter. S09E = 80 reason codes(!) showing something wrong with trace control block validation.

Trace does not depend on control block dependent microcode.

ZA Diagnostics

New wait state codes.
Machine checks have changed.
GTF now incorporates CCW trace as a standard feature.

Look up wait state codes when they occurred to check for new information.
The logrec I/O error wait state has a new format. Also, logrec does not have to be on sysres, it merely has to be cataloged.

Diagnostic information for specific system components has been taken out of the Diag Tech manual and placed in the appropriate Sys Logic manual.

Mach Chk
Interval Timer + External Damage MCICs deleted.
MCICs added:
CRW pending
Service proc. damage
Chan Subsys

ENQ/DEQ nodes that are new:
SYSZTRC - sys trace
SYSZVARY - path

DAE - check IEACMD00 to see which dumps are automatically supressed.
DAE functions by automatically setting SLIP traps.
For each dump to be surpressed, five characteristics must be specified.

The dump criteria are called keys.

Criteria:

- failing load module - required
- failing csect - required
- Any three of:
  - abend code
  - RC
  - recovery routine csect
  - component id
  - subfunction name
  - PSW/reg difference
  - failing instruction area

Dump specifications:

Alter IEAABD00 (sysabend), IEADMP00 (sysadump), and IEADMR00 (sysmdump) to specify what system and program areas are desired in each dump.

MVS/XA permits up to 100 sys.dump datasets. These datasets must be added and deleted specifically. The ADD should be placed in CMD00.

```
DUMPDS  ADD
DUMPDS  DEL
DUMPDS  CLEAR
```

SLIP - enhanced.

The RC may now be trapped.

A Nucleus module may be dumped in the case of a trap.

Display Dump command will list:

- Title
- Error id
- Abend code
- Reason code
- Module + csect name
- TOD clock
- Procedure name
- ASID
- PSW + Regs

The dump options currently in effect may be displayed.

Force and cancel have been enhanced. This permits killing address spaces that sometimes are reluctant to come out (like during initialization). Non-cancellable jobs may also be killed with the ARM operand of the FORCE command. (Possibly used against SMF, trace, + address spaces that are restartable).

Diagnostic Tools

IPCS has been enhanced with new verbs and better displays.

CCW trace for GTF is no longer an option.

Different CSECT in the same load module may have different Amodes. AMBCIST notes this.

AMDPRDMP -

Now has an INDEX (!) DD statement!

Other new verbs:

- IOSDATA - Validates UCBs, IOQs, IOSB, SRB, IOS ERP.
- LPA MAP
- NUCMAP
- RSMDATA - use VERIFY, not PRINT. (Lists internal RSM trace table)
- TRACE - formats trace data.
- VSMDATA

For IOSDATA, specify the operand EXCEPTION. This validates all CBs but only prints errors.

Standalone Dump

Each possible console to be used must be generated. No more pressing enter on any console. SADUMP must enable the specified subchans to look for interrupt.

New optrons:

ASID
SUBPOOL
RANGE
LOADPT
MSG
PROMPT

SADMP has a one step generation process.

Get yellow card - 5 version.

MVS Measurement & Tuning

4-22-85

Joan Arnold-Rolando
Rich Polisch

Aimed at SP1 but XA differences will be noted.

Definition + Methodology - Rich

Have established tuning goals. The users requirements determine the goals. Performance is what is perceived by the end user, not by the numbers. Capacity is allocated among resources. Capacity is constant for a given configuration. Tuning is getting the best use of that capacity. Problems can be avoided by increasing the intrinsic capacity.

Three options for improving performance:

- add hardware
- rearrange configuration (physical or logical)
- reduce overhead. (ever re-write code!)

Methodology

1. Measure the system
   (What you measure determines how - SMF, RMF, hardware monitor).
2. Alter a variable (one change at a time).
3. Re-measure
4. Compare result to objectives

Tuning to "as good as you can" is not an established goal. Get numbers.

Top-down: inspect whole system and spin-off detail projects.

Bottom-up: when the user calls with a problem, investigate what is inteferring with his job.

Top-down is the better approach. Bottom-up is real life. When you allocate scarce resources, somebody *is* going to end up on the bottom. Management must decide

the priority order.

Service Level Agreements (SLA) - the negotiated agreement with the users regarding what they want and how much they are willing to pay for it.

Tuning Objectives -

- Respone (online)
- Turnaround (batch)
- Thruput (best utilization of equipment; low multiprogramming...)

Trivial transactions are most important for the TSO user because the user expects them to be fast. In a response oriented system, capacity must equal peak demand.

Performance Indicators

CPU utilization
  Problem state
  Supervisor state
Channel utilization
Control Unit utilization
Device utilization
Paging rates
Swapping rates

} the biggest payback

If capacity is less than requirements, tuning will fail. Beware of PTF updates to SRM. Test performance during maintenace testing.

Preliminary Tuning Considerations - Joan

Sysgen - (Stage 1) - especially w/ IOCDS
Sysl. Parmlib - (other than IPS) IEASYS00
Jes - spool
Catalogs -
Data Set Placement -

MVS Module search:

LLE - created at each issue of load
JPAQ - Link, Attach, XCTL
Tasklib
Steplib
Joblib
Active CDE Queue
LPDE use count ≥ 1
IEALODxx → forces CDE to be built
IEAFIXxx (loaded in nucleus area)
Some CICS or IMS modules
IEALPAxx - MLPA modules
(used to test new modules)
LPDEs (all others] → Link Pack Directory Entry
Some things must go in LPALIB.
BLDL/FBLDL (Directory entries)
Linklib + concatenation
TSO Daly - Sysproc dd for Clists
else - S806.

Sysgen Parameters

CTRLPROG operands:

ACRCODE = No - eliminates extra code needed to support ACR in MP/AP environment.

OPTIONS= CRH - for MP/AP only.

OPTIONS= RER - reduced error recovery on tape read errors (99 to 5). Requires DCB=OPTCD=Z to be effective.

Options= BLDL - makes BLDL fixed. (8K or so)

Storage= - omit & allow IPL to determine size of real storage. (Unless running an MP system that gets reconfigured).

Review default blocksizes (a 3380 has an interblock gap of about 480 bytes).

Extra UCBs waste real storage.

System trace is 2-3% overhead. Trace should be on unless the system is very stable.

IEABLDxx - each entry 60 bytes; must be in alphabetical order. Can be pageable but will stay in.

IEAFIX. NO page faults for these modules.

IEACPAxx - name of module & all aliases.

IEAOPTxx - beware of default values.

IEAPAK00 - do not use.

Amdahl CPAMOD program can gather info to help build one that is efficient

CPALIST - for XA; libraries concatenated to IPALIB.

IEASYSpp - SQA/CSA ; SQA get 256K added to whatever is specified. Usually overallocated.

RA - SQA = (below, above)
below gets 3 extra segments
above gets 4 extra segments

Put CVIO in to clear all VIO datasets at IPL.

Maxusers - # of entries in ASVT.

Coglmt - increase, to reduce OPENS + CLOSES to log dataset. ROT = 888888

REAL - make as small as possible. OLTEP requires 76K. (+ then only sometimes).

VRREGN - make this Ø.

WTOBFRS - increase to speed up IPL.

Duplex - ; if used, do only for PLPA.
Used to protect against paging I/O errors on PLPA.

LNKLSTxx - don't put in linklib because it is automatically searched.

IEADMPxx - avoid NUC, SQA, CSQA, + SWA because users don't need to see this.

Channel load - 35 to 50%. Selector chans on the high end. Shared DASD should have less than 35%.

3350 fixed head - HASPCKPT, OS CVOLs, VTOC, or Linklist Directory (something that fits in 2 cyl).

If VS in no problem, put more things in PLPA.
(Must be reentrant).

JES2 -

remove job journaling (only used for checkpoint restart). Require 36% of JES2 overhead. Default is on!

Minimize PROCLIB concatenations.

&NUMTGV:

3350 - 2775

3380 - 4425 or 8850
large small - datasets

Buffersize should be Blksize = 4008

&Numcmbs - console buffers for JES. Raise this.

Sys1.Haspchpt get the whole volume resered during use. Don't place on the spool volume.

Don't mix device types for spool space. HASP doesn't know speed differences.

&DEBUG = NO ; only put 1 copy of checkpoint in storage.

Catalogs

Use CVOCs for non-VSAM datasets.

Increase VSAM buffers to permit multiple searches of VSAM catalogs.

Avoid GDGs in VSAM.

Convert VSAM to ICF catalogs.

VSAM + ICF catalogs have extensive CSA overhead.

Keep catalogs on unshared device.

Recommended:
Summary (INT)
Reports (ALL)
DINTV (0050)
RTOD (0800, 1700)
STOD (0800, 1700)

VSAM Share Options - GG22-9043
VSAM Performance Options - GG24-1584
Keeping Current Tape - "VSAM Performance"

RMF - Runs as a Started Task with high DPRTY. Rich
RMF records are in SMF format - can be on SMF dataset.

type Rec.
70 - CPU activity
71 - Paging (rates)
72 - Workload (Generated by SRM)
73 - Channel Activity (Phy/+ logical)
74 - Device
75 - Page dataset (performance)
76 - Trace (Can trace fields in control blocks that RMF samples. Analysis must be home-grown).
77 - ENQ Activity - not too useful for history.
78 - Monitor I Activity (XA - Virt Storage, I/O Queuing)
79 - Monitor II

RMF is controlled by ERBRMFxx parameters in Sys1.Parmlib.

CPU -
Busy
Wait

Channels -
Busy
Queueing
Service Time

Devices
Busy
Waiting
Queueing - important Types

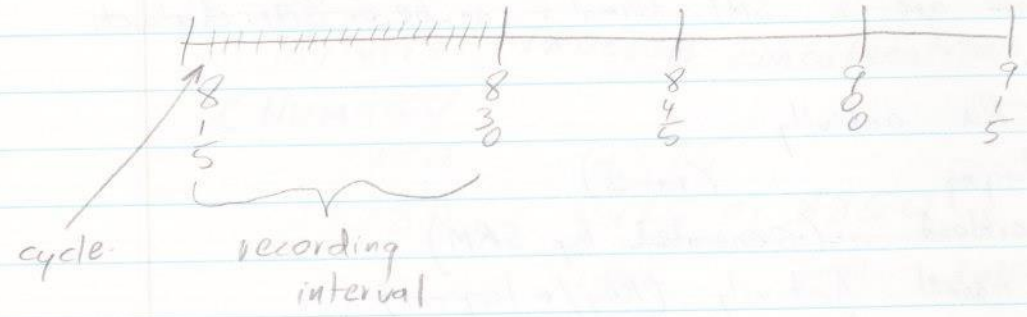
RMF has three kinds of times

cycle – sampling time; 300-500 ms; never > 1 sec.

interval - writes records based on accumulated cycles.

15 min to ½ hour. is max. (1 min).

Reporting interval - coded in Post Processor. "DINTV" 60 min



RMF cannot report on any resolution smaller than an interval.

Number of samples will be number of cycles per second * number of seconds in interval.

DASD Activity

Don't just balance channels on the numbers - segregate batch/interactive + tune appropriately.

A logical channel id is assigned to each set of physical channels assigned to devices.

$$Q\ time = \frac{Ave\ Q\ length}{Ave\ Rate}$$

Response Time = Service Time + Q time

Paging I/Os are queued in ASM + do not show in the DASD Activity report.

DASD SIO Rate = $\frac{\#\ SIOs}{\text{Interval in seconds}}$

Compare this with total SIO rate. DASD should be 2/3 of all SIOs.

Paging Activity

Demand Page Rate Vs. Total Page rate (Page in + Page out)

non-swap
non-VIO

Logical swap avoids paging I/O & is good.
A logical swap is better than physical swap.
Better than 90% of swapped out users should be logical swaps. Less than 90% may indicate a real storage shortage.

Page/Swap Dataset Activity

8405 has major enhancement in allocation of page slots.

ASM

Joan

3 components:

I/O Contiol (IILRIODRV) - handles paging & talks to RSM

VIO Control

VIO Group Operators

ASM is critical to response time.
Paging instruction path length in SP1 is apx 2000 instructions.

The PART points to the queues for cache, Fixed heads, & movable head. Each dataset will have a PARTE. Each PARTE of a type points to the next of the same type. The last one points to the first - Circular queue.

ASM changed at 8405. UZ90289 SP XA = UZ90290. Described as "ASM enhancements." A total re-work of ASM algorithms. New ASM attempts to schedule work for the least utilized page datasets.

Burst calculation - ASM calculates the milliseconds/page

pre 8405 {
using 1/2 historical time (PARTE)
1/4 expected performance (PCT)
1/4 last previous I/O
}

Now, the # of pages/burst is
6 - 3330
10 - 3350
33 - 3380

ASM in SP1.3 attempts to group related slots together. Then, at 8405, ASM forgets the groups and just tries to limit the number of slots used until 25% of the total are occupied & then opens up the rest of the data set slots. Most sites over-allocate page datasets (to keep management from placing other datasets on the pack).

The RIO has a significantly shorter path length than SIO. However, if another access to a different dataset on the same volume occurs, the SIO must be issued again.

In resume I/O even the seek & set sector is omitted (after 8405) if the slot is contiguous. Very fast!

Swapping

4-23-85

Jean

Swapping moves a user in or out of memory.
Swapping is SRMs weapon.
With extended swapping:
the CSQA + working set go to the swap dataset.
the non-working set pages go to local datasets.
the unchanged non-working set pages are already on local.

If common pages too highly, watch the wss definition in IPS.

In OPT, CSCTMTE the think time is set + heavily influences swaping. 15 seconds is added to this value (for TSO users only).
All address spaces are candidates for logical swap.
Logical swap mean SRM will not consider them in the multiprogramming target level; and physical I/O has been avoided.
Data set Selection - ASM will select the ds with the shortest I/O wait time. Swapping to locals may occur if only one swap ds exists.
One solution <u>might</u> be to not use swap datasets and add a few more locals. (after 8405).
Page + Swap datasets are defined with IDCAMS.
Pageadded dataset will be preserved across IPLs until a CVIO or CLPA is done.

VIO

Simulates dataset I/O by paging. The tracks of the device are simulated by "windows." The window even holds the interblock gap. If a VIO dataset is to be passed from step to step, a Sys1.stgindex dataset is required. VIO can be significantly faster than sysda allocations.

Implementation:

Unitname VIO=Yes

VIO work can be excluded from specific datasets.

IEAOPTxx DVIO=YES
IEASYSxx NONVIO=(dsn, dsn, ...)

If real storage is abundant, VIO is very good.

Try VIO for:

2314 - simulation uses little space
2305 - " " " "
Short jobs
Small dataset
Compiler work data sets
TSO edit work files

Do <u>not</u> use VIO for:

sysut1 of IEHMove
sysut3 & 4 of IEBCOPY

Monitor performance when implementing VIO. Erratic TSO response may be a function of VIO problems.

Recommendations:

1 page dataset / device
several small locals vs few large
dedicated paths & devices
spread out over channels
swap datasets on separate devices
use fastest devices for
PLPA & common if activity is high
swap ds w/ extended swap
Keep devices similar.

I/O Considerations

Rich

Types of chans:

Selector - no disconnection until the channel is done with the chan pgm. Selector channels are not available with XA.

Byte Mux - printers & readers; chan can disconnect after each byte. Chan interleaves bytes to different devices.

Block Mux - DASD; Blocks of data are transferred & disconnects are possible between blocks.

Burst mode - standard (1.5mb)

Data Streaming - changed protocol.

3880 Controllers -

A 3880 has two controllers which are called directors. So, a 3880 is really to control units in one box.

Head on string controls 2 to 8 devices and determines path options (string switch).

3350 A2 / C2 - C2 is alternate

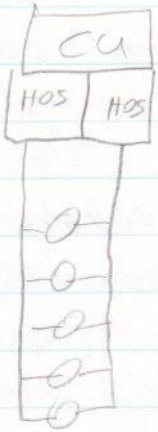
3380 A4 / AA4 -

A logical channel is a data contrivance. A logical channel describes a set of paths. Requests for service are queued to a logical channel.

CA - Logical control unit - is a set of physical control units with a common group of devices. In CA, requests are queued on the CCU.

DPS: Dynamic Path Selection

Dual porting: two heads of string



STC 8380

Amdahl 6280
6380

IBM 3380DD

IBM's regular 3380

AA4

One of the 4 internal paths may be used. 3 devices are unavailable whenever 1 is accessed.

Shared DASD - contention is now down on the CU and HOS level.
Reserve/Release is implemented on the device level and reserves a volume. Omegamon permits examination of cross-system reserves.
Reserve problems won't show up in RMF PP data. Look at the system while the problem is happening.

GG22-9346 XA I/O Performance

Path Selection - Sys1.Parmlib (IECIOS)
CCH=(x,y), select=
Seq - try primary first
Rseq - Reverse seq., try secondary first
Rotate - alternate back & forth
LCU - last chan used
If it worked last time, try again.
Balance -

For XA, this all changes. XA Default is Rotate. Seq & RSEQ can be forced on IODEVICE macro in IOCP. PATH must be coded for each device.
See Phy Chan Act reports to see how things are performing.

I/O Nitty Gritty -

Queue Delay	CU Delay	Device Delay	SIO Delay (service Time)

XA RMF has better data.

RPS - the disk is turning at 3600 rpm

Latency 8.3ms

- Seek - CCHH (Disconnect during seek)
- Set Sector - find the wedge. (disconnect)
- Search ID Equal - Reconnect (RPS miss if path is busy)
  - TIC *-8
  - Read/Write

Reconnects

With XA, chan busy can be 50% because reconnect can go back on any path.

Path Busy = Sum of Path Utilization of all other Devices on path

With channel switching: (or string switch)

Path Busy = Controller busy + (Controller Free & Channel Busy)

The busiest devices suffer least on reconnect.

String & channel switching buy availability but do little for performance. To improve performance, use shorter strings.

Dual porting

Summary

RPS Reconnect Delays are critical.
String & Channel Switching improve reliability
and availability but degrade performance.
Low activity devices suffer most.
Shorter strings are best.
TO improve Dev Busy, reduce service time
by watching ds placement.
TO improve CU busy - isolated shared DASD
Put only shared data on shared DASD.
Device service times can be improved by using
efficient block sizes. Also, VTOC placement.
Reorganize PDSs to put highly used members
near the front (use FASTDASD).

It may not help much to place highly used datasets closely together. 15ms is required just to move the arm one cylinder. Thereafter, overhead is not too bad (30ms average). Best bet is to move the concurrently accessed dataset to another volume.

Dual Density Devices - just makes matter worse. Has 1 physical volume + 2 logical packs. Very difficult to tune.

Blocking

As blocksize goes up, number of EXCPs + SIOs goes down. Half track blocking is probably preferable to full track blocking in TSO.
Eliminate secondary extents. (Write a PSS tool?)
Use cylinder allocations

4-24-85

Planning for lots of TSO

Rich

	ENV #	CPU Busy	A Chan Busy %	Total SIO	Disk SIO	Total Physical Paging	TSO Users	Resp Time
untuned	1	51%	55%	191	182	323/sec	90-100	erratic
Coral on Disk PLPA/Com on Dram	2	48.9%	37.9	193	153	295	"	smooth
12MB; on 7 locals	3	50.2	16.79	167	123	137	133	very good
Swap on 2 drums	4	50.01	21.01	151	119	169	135	" "
16meg	5	71.21	9.79	166	134	95.2	140	Great

Plot pages/sec/terminal against terminals. Watch swapping.

TSO Tuning

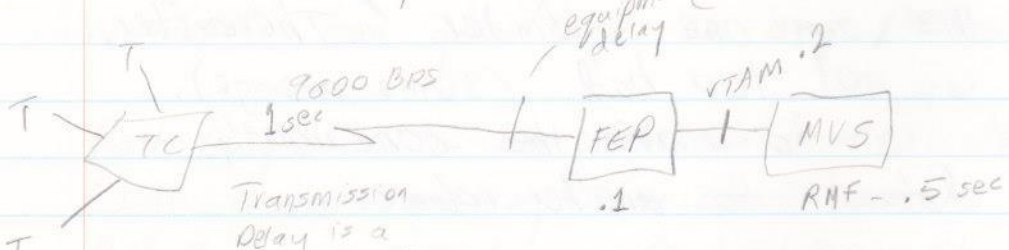
Joan

Objectives:

Good response time (no complaints)

Save CPU

Save memory



Typical Case

1	sec
.001	
.1	
.2	
.5	
1.8	- one way
*2	
3.6	- rand trip

TSO recommendations:

IKJEFT02
Sys1.Cmdlib re-entrant modules } Place in LPA
some AMS modules

Reformat UADS with large blocksize (get entire entry).
Place TSO above APG for first 2 periods, then move to APG to compete w/ batch.
Reorder CVOLS with share COPYCAT.
Force of inactive users.
Do not share Sys1.Brodcast. (Reserve gets issued).
Logon JCL Msglevel=(0,0) prevents messages from being created & then thrown out.
Write commands to replace frequently used Clists (Procman [by Tone] will convert clists into pgms).
Sys1.CMDLIB high on Inklstx order.
Don't use steplib in logon Proc. Else command search is slowed.
Commands not in LPA should be in Fixed BLDL.
Use % to speed Clist initiation.
Eliminate type 40 SMF records w/ share zap.

Buffer Tuning (for VTAM).

Buffers are specified in Sys1.Vtamlst(ATCSTR00).

Rules of Thumb

CPU Utilization > 85%

Storage:

AFQ > 15 (Below 10, Page stealing occurs).

Real storage:
- UIC - overcommitted < 2
- underutilized > 64
- Ø means fixed

Channels

Selector < 50%

Blk mux < 35% (SPI)

Chan Busy & CPU Wait < 10% of chan busy.

If it's greater, the chan may be holding up the CPU.

DASD

Dev util < 35%

Ave Q len. < .05

seek distance 90% within 10 cyl (FASTDASD)

Paging Rates

3350 - 30 pages/sec per device

3380 -

1 page movement requires about 2000 instructions.

so figure # of pages/sec * 2000 and divide MIP value for processor by this to figure overhead.

Keep # of Domains < 10 (Else SRM does too much)

RMF reporting interval ≥ 15 minutes.

# of swap ds = Ave pages per swap-in / 12

Virtual Storage Constraint Relief

Joan

With CICS & IMS, the CSA can get to 5 megabytes very easily. The private area in some cases is down to 4 megabytes. (VSCR became a problem for IBM when SMP would no longer run!)

VSCR techniques differ greatly depending on the type of work. A TSO machine will look different from an IMS layout.

Get rid of old modules (out-of-date SRCs, old products in CPALIB, etc).

ROT:

PCPA 3.4 - 4.5 mb
Nuc .87 - 1.2 mb (Top side is segment boundary)
CSA 2 - 6 mb (Bottom side is segment boundary).
SQA .5 - 1 mb (SQA will overflow into CSA. So it is okay to allocate minimum and watch for an overflow). 256K is added to what ever is specified.

Eliminate unnecessary UCBs.

UCB size:	CPI	ZA
DASD	104	144
Non-DASD	80	120
Consoles	234	234

For abend 822 (Fragmentation of CSQA) drain (JPI) and restart the initiator.

CSA

Don't over-allocate VTAM buffers.

If JES2 trace is not being used, don't allocate the buffers.

Investigate key & CSA areas and make sure they are being freed.
Watch IPL time for IEA913I msg - CSA expanding into private area. Look at CSAMON product from PSS Tools. Search Infosys on CSA & Subpool for map of CBs and keys. (try CSA Frag*)

See Virtual Storage Map with XA RMF.
See:

VS Assessment Methodology	ZZ2Ø-4151
VS Tuning Cookbook	ZZ20-4185

4-25-85
Joan

Capacity Planning

1983	1988	
26%	7%	Batch (backup the database).
17	20	On-line
1	4	End user (mailbox, etc)
25	39	Idle (due to online emphasis)
20	22	OS
11	8	Per (more outside packages)
100%	100%	

Procedures:
- Workload characterization
- Performance requirements
- Measurement capability
- Workload forecasts

user objectives - good performance
business objectives - maximum use of resources

Data collection:
- SMF/RMF
- JARS
- TSO/MON
- CMF
- Control/IMS/CICS
- MICS

Capacity planning must be aimed at a time period:
- peak-peak (users love this; management dies).
- average-peak (occasionally inadequate capacity).
- average (insufficient capacity at peak times).

This is a management decision.

Pay attention to workload cycles. Is every day the same? week? Does the utilization vary across the month? How about seasonal or end of year? How does management want to meet the requirements of these peaks.

Ref:
Queueing Theory for Computer Applications, Klinerock

Mean
Standard Deviation
Confidence Limits

Tools

Linear Regression/Trend Analysis - project from data.
Modeling - queueing theory
Simulation - too empirical
Benchmarks - expensive

Linear Regression -

Look at the past to judge the future. Fairly simple + straightforward and sometimes even works. SAS is good at this.

For inconsistent growth or new applications, the linear method is of little help. New devices and different speeds compound the problem.

Modeling -

Best 1 - System activity is defined in queueing theory equations. Once the system is characterized changes in coefficients may be introduced. Fairly good accuracy once the system is described.

Can be very CPU intensive.

MAP - amdahl product.

Simulation -

GPSS, Snap/shot, Sceit - packages.

CPU intensive.

Benchmark

The best... but... the process can be very difficult. A new application cannot be tested before it is written.

SRM

Rich

Tune everything else before starting on SRM.

SRMs objectives: Distribute system resources based on user-assigned priorities and optimize system thruput. Also, avoid resources shortages (SQA, Real Page Frames, ASM slots).

SRM weapons:

- Swapping - the primary mechanism.
- Frame stealing
- altering ASN priorities (APG)
- device allocation balancing
- inhibit address space creation

Resource monitor 1 collects info every second & places it in the RCT. Resource Monitor 2 wakes up every 20 seconds (30 in XA) and makes averages of what is in the RCT. The result is deposited in the RCT & DMDT (Domain Descriptor Table).

The CPU, storage, & I/O managers can be excluded from voting in the recommendation value by specifying in the RTB-D. Then, the Workload Manager makes the entire recommendation. The recommendation values go into the OUCB. The workload manager gets his priorities from the IPS.

The whole SRM process is SYSEVENT driven. The various managers are invoked on SRM second coefficients. (IRARMCPU)

The APG supplies the rules for ordering tasks on the dispatcher queue. This ranking

only applies to folks who reach the dispatcher queue (i.e. swapped in).

The RMTC (SRM control Table) infamation is what appears in the CPU Activity Report. OUCBs are placed on the SRM queues.

Domains

SRM's way of catagorizing wrk. The characteristics are short batch, long batch, TSO, CICS + IMS, System Tasks, + JES. TSO + long batch may have 4 performance groups. Short batch will have two. The rest have one. Keep the number of domains low (<10) or else SRM stays busier trying to run all the comparisons.

Non-swappable and STCs are not considered part of MPL + do not count toward min + max unless CNTNSW is specified in OPT.

For each domain, SRM keeps the current MPL, min MPL, max MPL, and target MPL. The current chases the target.

Resources:

Real storage utilization is tracked by the system high UIC for each addr space (OUXB). RCCUICT=(low, high) in OPT is compared to system high. Resource Monitor 2 looks here every 20 seconds.

CPU

If the lowest priority task on the dispatcher queue is not getting dispatched, then SRM considers the CPU at > 100%. OPT field in RCCCPUT=(low, high). If real load is < low, the swap in + try to get the CPU busier. If real useage is

greater than high, swap out.

The page rate PAGERT1,2 is model dependent + values should be supplied by vendor.

The system is overutilized if any high threshold value is exceeded. SRM is very sensitive to over-utilization.

The system is underutilized only if all the low thresholds are exceeded. If nothing is high, and everything is not low, then SRM is happy.

Contention for a domain exists if the Ready User Average $\geq$ (Target MPL $-$ 1).

Then contention index =

(RUA * Weight) / Max (TMPL or 1 [whichever is higher])

If RUA $<$ (TMPL $-$ 1) then reduce the Target MPL.

TMPL = Max (MinMPL, TMPL $-$ 1)

SRM will continue to reduce the TMPL until contention occurs!

Swap Control

Once Current MPL and Target MPL are decided SRM scans the Domain Descriptor Table and:

1. if CMPL > TMPL then swap out until CMPL = TMPL
2. Swap in oldest ENQ holder.
3. if CMPL < TMPL then swap in until CMPL = TMPL.
4. if CMPL = TMPL, and if

(max CMRV(out)) > (min CMRV(in) + 1)
then exchange swap.
Swap control wakes up every SRM second.
But the TMPL get updated every 20 seconds.

Composite Swap Recommendation Value (CMRV) contains the opinions of the CPU, I/O, and storage managers, but their combined opinion only counts 20% of the CMRV. the workload manager's opinion counts most.

CMRV = WRV + (CPU * CRV) + (IOC * IRV) +
(MSO * MRV)

The CPU, IOC, & MSO values come from OPT & if zero do not count at all. These values judge the importance of the resource, not of any job's use of resource. Each manager decides whether his resource is over or underutilized. As the resource utilization value declines, the recommendation value goes up.

Workload manager is different. He deals in service units for I/O, memory, & CPU (SRB & RB). A service unit is amount of consumption and time. Then a swap recommendation value is calculated.

Service Units:

CPU - TCB execution time divided by a constant to reflect 10000 instructions. This makes the IPS machine independent.

I/O - sum of all EXCP counts for a user (including work done by JES)

MSO - Real storage frames * CPU service units / 50.

SRB - SRB execution time (local & global).

Weighting factors for these service units are contained in the IPS. Defaults = CPU=10, IOC=5, MSO=3, SRB=10. These probably don't need to be changed.

As a user accumulates service units, the recommendation value goes down. (probably).

Performance Groups

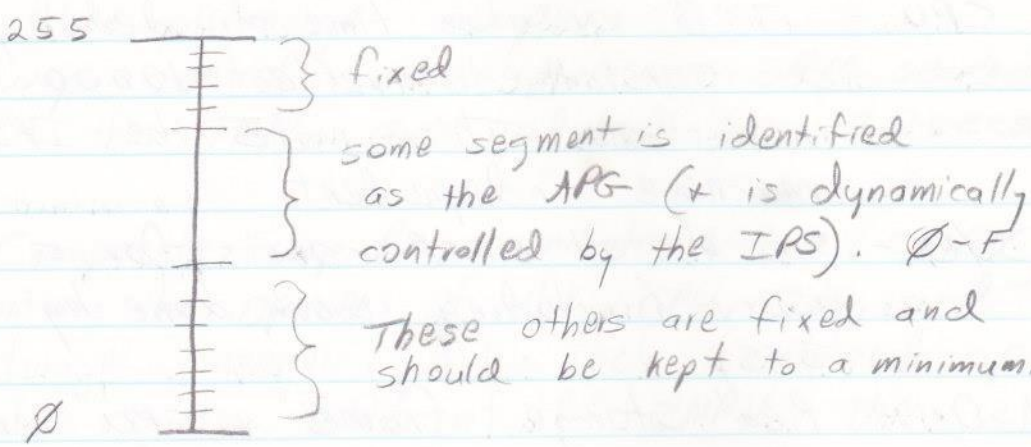
Assignments can be made in all sorts of places, but IPS over-rides everything. Each performance group period determines the eligibility characteristics of a transaction. The performance group assigns the domain and the performance objectives.

4-26-85
Rich

Automatic Priority Group (APG)

Controls how tasks are dispatched once they are swapped-in.

Priority ladder:



3 ways to be ranked in APG:

Mean-time-to-wait - CPU intensive jobs get low ratings. I/O intensive jobs should get all the CPU they want because they don't ask for much.

Rotate - timeslicing. This provides for the CPU hogs.

Fixed - position is fixed absolutely within the APG.

The APG range is Ø - F.

IEASYSØØ contains an APG parameter that is the first digit of where the APG appears on the priority ladder. The mean-time-to-wait, rotate, & fixed assignment supply the second digit.

Don't use the APGRNG parameter in IPS. APGRNG provides for multiple APGs. APGRNG and APG are mutually exclusive.

APGRNG = (6-15) positions the big APG on the ladder and then the MTW; Rotate, & Fixed positions are set with PP = Mx
Rx
Fxy

Dispatching Priority Management

Time slice control - possibly of use with multiple IMS or CICS address spaces.

TSDP - time slice dispatching priority

TSGRP=n - this assigns a performance group.

TSPTRN=(n1, n2, n3...) - the relative positions of time slicing.

Ex: Tsptrn=(1,2,1,3,2) - group 1 gets 40%, group 2 gets 60%.

Logical Swap - the swapped out user remains in storage until SRM's think time expires (LSCTMTE). If this is too low, too much swapping may occur. SRM adds 15 secs to this value for TSO users.

ENQ Delay

An address space holding a resource is made non-swappable for the ERV value i.e. OPT. This value is in service units and is set at 500 which may be low. Watch out for hot users who grab a resource to keep themselves swapped in.

Storage Isolation

Common + private work the same way:

Pvss = (min, max)

max	steal here first
	steal sometimes
min	never steal here

The purpose is to try to eliminate page stealing. Frames above the max are stolen first when stealing happens.

Address spaces holding frames above the max will

have their frames stolen first. Once all those are gone, everybody looses frames equally; but SRM will never steal to reduce an address space below the minimum.

PPGRT - Page rate coded in CPU seconds also controls storage. Raise the Target Working Set Size (TWSS) by 3% if the page-in rate is greater than the max. Lower TWSS if page-in rate is lower than the minimum. This limits paging if the task really needs to page but prevents the task from getting the whole program in real storage. SRM will keep an address space at the minimum even if there is plenty of storage available. Beware of forcing paging.

Common is controlled with CWSS and CPGRT parameters. Check RMFMON menu
SRCS - CPA Fixed + CSA Fixed = CWSS
SPAG - CPA in + CSA in = CPGRT
Do not over-fence unless plenty of real storage is available.

The CPGRT is coded in wall-clock seconds. Beware of this difference between PPGRT + CPGRT.

Response Throughput Objective (RTO) - Valid for 1st period TSO and can smooth out TSO response. This is what knocks down response time when the new CPU comes in. If the average time to end of a first period TSO transaction is < the RTO, then the difference is added to all incoming

transactions. This amount gets added to all incoming transactions since SRM can't know which will end in the first period. RTO can be specificed in fractions of seconds.

ICS

Assigns performance group to transaction type, users, jobs, etc. Reporting performance groups may be established for tracking by dept or application without actually changing performance group.

In CICS + IMS, individual transaction codes may actually be controlled by SRM.

- CICS -

SIT CMP=YES
MCT EVENT=YES
Command CSTT MON, ON=EXC

A PGN for CICS must be specified in ICS. This seems to add a lot to CICS overhead.

Clists

CNTCLIST, in OPT control whether a CLIST is considered as one transaction or whether each statement in the CLIST should be a separate transaction.

The invocation of SRM can be set by RMPTTOM in OPT. This is at 1000ms + Should not be changed.

IPS

6 sections:

1. Service Definition Coefficients - alter the rate that service units are accumulated.
2. Keywords - globals like APGRNG, CWSS, CPGRT, IOQ,
3. Workload level - one per IPS. Become workload recommendation values
4. Perf Obj - resource useage by service level.
5. Domains - catagorization of work
6. Performance groups - connects transactions (TCS) to domain.

PROCESSORS/MIPS/SRM SECONDS/SERVICE UNITS

Processor Model	MIPS	SRM seconds per second*	Service Units per second**
IBM 158	1.0	1.220	51.2
IBM 4341-12	1.6	2.083	87.6
NAS 6630	2.0		
AMDAHL 470/V7C	2.6	3.683	154.7
AMDAHL 470/V7B	3.5	4.511	189.5
IBM 3083E	4.1	4.878	204.8
AMDAHL 470/V7A	4.5	5.689	239.0
IBM 3033U	5.2	6.250	262.5
AMDAHL 470/V7	5.5	7.299	306.5
NAS 9040	6.0		
AMDAHL 470/V8	6.8	8.192	344.2
IBM 3083J	7.25	9.259	388.8
NAS 9050	8.0		
IBM 3081D	10.0	6.579	276.3@
AMDAHL 5860	13.4	14.840	623.6
IBM 3081K	14.0	8.772	368.4@

3084 = 2-81Ks

NOTES:

All numbers are based on software estimates coded for use by SRM. All service unit weight factors are equal to 1.0 for CPU, IOC, MSO, and SRB.

* Value calculated by dividing 1024 by SRM speed constant in IRARMCPU

** Value calculated by multiplying SRM second by 42.01686

@ Value to be multiplied by 1.85 for both sides of the processor