IBM VSE/ESA Turbo Dispatcher Performance

Wolfgang Kraemer

VSE Product Mgmnt
Dept 3221
71032-14 Boeblingen
WKRAEMER at DEVM
wkraemer at de.ibm.com

Update 2001-07-15

Contents

A. Overview

B. General MP Performance Aspects

C. VSE Implementation

D. Performance Considerations

Contents

E. Performance Results

F. Performance Modelling/Prediction

G. Turbo Dispatcher Performance Hints

Contents

H. VM/VSE Only TD Considerations

I. PR/SM LPAR Only Considerations

J. Appendix: Why now?

K. LSPR Results for Turbo Dispatcher

Turbo Dispatcher Evolvement

Turbo Dispatcher Evolvement

The VSE/ESA 2.1 Turbo Dispatcher is generally available since 07/95 and since then included in each VSE/ESA 2.1 shipment.

TD as of 2.1.0 GA, 07/95 (APAR DY43551):

Performance measurements in the Boeblingen lab environment with the 07/95 version of the VSE/ESA Turbo Dispatcher and different pure VTAM/CICS online workloads (no batch, no SQL/DS data base partition) have revealed that, in order to fully exploit 2-ways, an additional performance fix is required.

TD as of 2.1.1 GA, 10/95 (APAR DY43684):

Extensive additional measurements have shown a total sum of up to 190% on a 2-way with increased transaction throughput and much better response times (PTF UD49610/12/13).

This was achieved by implementation of several additional cases of intercommunication between the processors. This in turn resulted in higher CPU-times, and thus in lower MP-factors for some workloads.

In order to improve the MP-factors, more investigations were done to assess the workload specific individual costs and benefits of these actions.

TD as of 2.1.1+, 11/95 (APAR DY43757):

This performance PTF allows several 'read-only'-Fast SVCs to be run as parallel code and thus reduces the Non-Parallel share of workloads, especially with CICS monitoring (PTF UD49667/69).

Also some means have been taken to exploit-3-ways, where a total sum of 248% CPU utilization has been observed, at an MP-factor of 2.4.

TD as of 2.1.2+, 03/96 (APAR DY43919):

This PTF for the turbo dispatcher contains enhancements in functional areas, as well as performance (PTF UD49915).

TD as of 2.1.3, 07/96 (APAR DY43979):

This is plain VSE/ESA 2.1.3 and is still valid.
Any newer PTF level (starting with DY44052) requires newer vendor PTFs.

Turbo Dispatcher Evolvement ...

Turbo Dispatcher Evolvement (cont'd)

TD as of 2.1.3+, 07/96 (APAR DY44052):

This PTF for the Turbo Dispatcher contains enhancements for relative shares, described in APAR II09513. It has been superseded by DY44156 or DY44201

TD as of 2.2.0+, 12/96 (APAR DY44265):

This PTF for the Turbo Dispatcher contains functional enhancements to even better allow vendor products to run in parallel mode (TD level 7).

This enabling PTF UD50177 requires new levels of vendor code, using the new function, in order to bring performance benefits.

Also functional problems in connection with vendor code and with a singular PRTY SHARE problem have been fixed.

TD as of 2.3.0, 12/97:

This TD level 8 contains e.g. the QUIESCE enhancements.

On order to correct a (rare) QUERY TD overflow problem,
make sure you applied APAR DY44677 (PTF UD50680).

TD as of 04/99 (APAR DY44847, PTF UD50965):

Includes minor functional patches for Relative Share balancing.
Retrofitted from VSE/ESA 2.4.0 GA-level.
Not contained in VSE/ESA 2.3.2 refresh.

í Use always latest TD level

Starting with DY44052, additional vendor PTFs are required.

Refer also to the TD APAR/PTF list later in this document

Notes etc

Note

All information contained in this document has been collected and is presented based on the current status.

It is intended and required to update the performance information in this document.

It is the responsibility of any user of this VSE/ESA 2.1 document

- to use the latest update of this document
- to use this performance data appropriately

This document is unclassified and especially suited for VSE customers.

Trademarks

The following terms included in this paper are trademarks of IBM:

ES/9000 ESA/390 System/390 SQL/DS PR/SM
VM/ESA VSE/ESA ESCON ECKD RAMAC
Nways ...

The following trademarks are owned by their respective owners:

EXPLORE/VSE	by Computer Associates
TMON/VSE	by Landmark Corporation
R/2	by SAP AG, Walldorf, Germany
...	

Notes etc ...

Disclaimer

This document has not been subjected to any formal review or testing procedures and has not been checked in all details for technical accuracy. Results must be individually evaluated for applicability to a particular installation.

Any performance data contained in this publication was obtained in a controlled environment based on the use of specific data and is presented only to illustrate techniques and procedures to assist to understand IBM products better.

The results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicability of this data in their specific environment.

The above disclaimer is required since not all dependencies can be described in this type of document.

Acknowledgements

Thanks to all who contributed directly or indirectly, be it by measurements, suggestions or in other ways.

Special thanks is expressed to

- Ingolf Salm	the designer of the Turbo Dispatcher
- Hanns-J. Uhl	for numerous performance measurements

All mistakes and inaccuracies in this document are my own.

Please, as in the past, contact me if you have

- suggestions or questions regarding this document
- questions on VSE/ESA performance, not covered in any of the VSE/ESA performance documents

Note that some additional items are documented in IBM INTERNAL USE ONLY appendages, available to your IBM representative for discussion with you, if specific need exists.

Wolfgang Kraemer, IBM VSE Development, Boeblingen Lab, Germany

Notes etc ...

Base Document(s)

This document essentially deals with the performance aspects of the VSE/ESA V2 Turbo Dispatcher (N-way support).

For general VSE/ESA V1 and V2 performance, refer to the documents

'IBM VSE/ESA 1.1/1.2 Performance Considerations'
'IBM VSE/ESA 1.3/1.4 Performance Considerations'
'IBM VSE/ESA V2 Performance Considerations'
'IBM VSE/ESA I/O Subsystem Perf. Considerations'
'IBM VSE/ESA VM Guest Performance Considerations'
'IBM VSE/ESA Hints for Performance Activities'
'IBM VSE/ESA TCP/IP Performance Considerations'
'IBM DFSORT/VSE Performance Considerations'
'IBM VSE/ESA CICS Transaction Server Performance'
'IBM VSE/ESA V2.5 Performance Considerations'
'IBM VSE/ESA Performance on xSeries (NUMA-Q) Enabled for S/390'

The files are
VE13PERF.PDF, VE21PERF.PDF, VE21TDP.PDF, VEIOPERF.PDF, VEVMPERF.PDF, VEPERACT.PDF, VETCPPER.PDF, VESORTP.PDF, VECICSTS.PDF, VE25PERF.PDF, VEXEFSP.PDF

The VSE/ESA 2.1 base document is available since the 2.1 General Availability 04/95, it has been updated many times, and now contains also VSE/ESA 2.2 and 2.3.
VSE/ESA 2.4 performance info was appended in the CICS TS document.

All documents are also available from INTERNET via the VSE/ESA home page

http://www.ibm.com/servers/eserver/zseries/os/vse

(http://www.ibm.com/s390/vse/   former URL)

Starting with VSE/ESA 2.4 documentation, these documents are also available on the VSE/ESA CD-ROM kit SK2T-0060, in Adobe Reader format.

Subject documents contain references to further VSE/ESA performance documents.

Glossary

Glossary

DIM	Data in Memory A concept to store as much data as possible/reasonable in processor storage
DLAT	Directory Look Aside Table
ITR	Internal Throughput Rate A measure for processor and/or S/W effectivity: #transactions or batch jobs per CPU-second. On n-ways it is per n CPU-seconds, thus ITR higher.
ITRR	ITR ratio to a another (base) processor or S/W setup
LSPR	Large System Performance Reference IBMs method to characterize relative processor speed. Based on measurements
MIPS	Meaningless Indicator of Processor Speed (if you believe without reflection). Millions of Instructions Per Sec of a certain workload on a certain architecture and implementation. 'Effective MIPS' make some more sense, they are better suited to characterize absolute processor power. In any case only ITR-ratios to a base processor can be determined/measured/provided
MRO	CICS Multiple Region Option Provides the required communication of CICS partitions using Transaction Routing (TR) or Function Shipping (FS)
NP	Non-Parallel code that cannot run in parallel on more than 1 processor
PR/SM	Processor Resource Systems Manager An ES/9000 standard feature for logical partitioning
TD	VSE/ESA Turbo Dispatcher for support of multiple processors (MP)
MP, n-way	These terms are used here interchangeably. Any processor system with >1 processors ('CEC's), shared processor storage and I/O subsystem/channels
PB	Partition Balancing, a VSE function

References

Further References

The following are references for further performance information in the context of VSE/ESA V2 or support of multiple processors:

VSE/ESA Turbo Dispatcher Guide and Reference,
Version 2.1 SC33-6599-00, 07/95
Version 2.2 SC33-6599-01, 12/96

VSE/ESA 2.1/2.2 Performance Considerations,
Use W. Kraemer's latest update.
Part of VE21PERF PACKAGE on IBMVSE tools disk and on INTERNET

Modelling CICS Systems
(Performance impact of CICS/MVS MRO implementations)
Ellen M.Friedman, Enterprise Systems Journal March/April 88, p.28

Guidelines for Partitioning CICS/VS Systems,
GG24-1623, 12/87, 53 pages
(An introduction to CICS MRO)

CICS/VSE 2.1 MRO Function Shipping
ITSO Red Book, GG22-3883-00,   pages,

CICS/ESA 3.3.0 Shared Data Tables Guide, SC33-0887
(An outlook to CICS/ESA and N-way)

VM/ESA, Running Guest Operating Systems, SC24-5522-02, 12/92

VSE/ESA 2.1 'The Turbo Dispatcher',
ITSO Red Book, GG24-4674-00, 58 pages, 02/96

MVS Performance Capacity for 9672-Rxx Processors,
WSC flash 9505.1, 02/95
(Available to your IBM representative, IBM Internal Use Only)

Balanced Systems and Capacity Planning,
WSC Technical Bulletin, GG22-9299-04, 125 pages, 08/93
by P.T. Borchetta and R.J. Wicks
(Includes multiprocessor considerations for response times)

Are you Turbo Ready?, VM/VSE Tech Conf Orlando, 05/96 by Dan Janda

Sizing VSE/ESA Systems, VM/VSE WAVV Conf Green Bay, 10/96 by Dan Janda

VM/ESA Geater N-way Thoughts, VM/VSE Tech Conf Rome, 10/96 by Bill Bitner

References ...

Further References (cont'd)

Real World Turbo Dispatcher Considerations,
by Dan Janda,
VM and VSE Tech Conf Kansas City 05/97, session 33E
VM and VSE Tech Conf Mainz, Germany, 06/97, session 53E
VM and VSE Tech Conf Reno, Nevada, 05/98, session 32E

How Much Does a Hen Weigh? -Sizing VSE/ESA Systems-,
by Dan Janda,
VM and VSE Tech Conf Kansas City 05/97, session 33I
VM and VSE Tech Conf Mainz, Germany 06/97, session 53I

Turbo Dispatcher for the Real World,
by Dan Janda,
VM and VSE Tech Conf Orlando, 06/2000, session E77

Turbo Dispatcher information is also available from INTERNET via the VSE Turbo Dispatcher home page

http://www.ibm.com/products/vse/vsehtmls/turbod.htm

Overview

PART A.

Overview

General Note

```
Note that due to the high capacity of 9672 CMOS processors, workloads
must be carefully tuned in order not to encounter performance
bottlenecks, which also would have appeared on uni-processors, even
with the VSE standard dispatcher.
```

Why Multiple Processors

Ù **VSE/ESA with 'old' Standard Dispatcher**

Native, under VM or in PR/SM LPAR:

„ **Any VSE MACHINE can use only 1 processor's power**

Even if its workload needs more

Even if other processors are sitting idle

„ **Workload must be balanced among VSEs**

Ù **VSE/ESA with Turbo Dispatcher:**

„ **Any VSE PARTITION can use only 1 processor's power**

Even if its workload needs more

Even if other processors are sitting idle

„ **Workload must be balanced among PARTITIONS**

VSE/ESA Turbo Dispatcher

Approach

Ù **Assign processor on PARTITION basis**

```
rather than on SUB-TASK basis (MVS, if sub-tasks available)
```

General

Ù **Transparent support, keeps all 'external interfaces'**

Ù **Smooth transition, even with vendor products**

í **Allows to keep full transparency to subsystems and existing applications**

```
Only those programs or vendor products have an impact which

          - used dispatcher interfaces
          - did not use provided interfaces
          - managed job scheduling
          - updated the first VSE 4K page (!)
          - used POWER internal control blocks

Mostly, changes apply to

          - performance monitors
          - schedulers
          - accounting products

No change to VSAM, CICS/VSE or VTAM was required for functional
reasons
```

í **Basically same**

- **operating environment**
- **system structure**
- **administration**

```
By usage of several processors, naturally, it may be required to

          - re-adjust partition priorities
            (including partition balancing)
          - split up Online work into several CICS partitions
```

í **Provide cost-effective and seamless support, adequate to VSE customer expectations**

VSE/ESA Turbo Dispatcher ...

Basic Design

```
At any point in time ...
```

Ù **Each partition can be dispatched**

- **concurrently to any other partition**
- **on any (single) processor**
- **independent of its last dispatch**

Ù **System code or 'Non-Parallel work-units' can run only on 1 processor at 1 point-in-time**

í **No dispatch affinity or pre-assignment required/implemented**

```
of any task or partition to a specific processor
```

Warnings

Any exploitation of more processor power may need tuning effort (as on UNI-processors)

Any MP exploitation needs proper workload and also partition setup

VSE/ESA Turbo Dispatcher ...

Applicability

Ù 'Old' and Turbo Dispatcher (TD) available on any VSE/ESA V2 system

```
Selection via IPL LOADPARM:  IPL cuu ....T
```

Ù TD also runs on UNIs,

UNI-customer can ...

„ exploit new partition balancing function(s)

„ determine expected MP suitability of his individual workload and setup

```
Also suited for 'MP extensions' (adding processors):
        - install addt'l H/W
        - define/use >1 processor for 1 VSE
          (under VM or in LPAR, in already installed n-ways)
```

Ù Any number of processors function-wise supported

```
Most capacity benefits expected for up to 4 processors
```

Ù Runs on all IBM ESA/370 or ESA/390 n-ways or multiprocessors

```
'Attached processors' (APs, w/o I/O capability) NOT supported.
Parallel Sysplex (Coupled systems) NOT supported

CAUTION: 4381-92E processors may not correctly execute
         TS (Test and Set) instructions, potentially used
         in MP environments.

         VSE/ESA TD itself does not use this instruction,
         but potentially other components or vendor programs.
         More info is contained in the IBM APAR VM59052
```

ESA/390 Only for VSE/ESA 2.4

VSE/ESA Turbo Dispatcher ...

Additional Functions

Ù Customer requested VSE dispatch enhancements are (will be) part of the Turbo Dispatcher only

```
E.g.

   - equal balancing weights for static and    (ESA 2.1.0)
     dynamic partitions
     (-> PRTY command to be checked/changed
         if dynamic partitions in the partition balancing group)

   - more flexible partition priority settings (ESA 2.2.0)
     (relative SHAREs for balanced partitions)
```

VSE TD Startup

Ù IPL is done on 1 processor only

Ù Addt'l processors are started after IPL complete

- via startup-procedure or
- via operator command

```
(//) SYSDEF TD,START=cpuaddr|ALL

 Native:  ALL causes all physical processors to be started

 VM/VSE:  ALL causes all virtual processors to be started,
          which currently are defined for this guest
          or 'seen by VSE'
```

Uni and N-way Capacity Constraints

Traditional (Uni-) Constraints

Ù CPU speed

Ù Real storage

Ù I/O capacity

N-way Constraints

Ù All Uni-constraints

Ù Single engine power for single partition(s)

Ù Single engine power for non-parallel part of load

Ù Sufficient partitions to occupy all engines

Setting Correct Expectations

Areas where TD benefits are limited

by other reasons

Ù The 'biggest' VSE partition requires more CPU-power than is available on a single processor of the n-way

Ù The VSE system before was NOT at all CPU utilization bound

e.g. was limited by other system resources

This may have been

I/O bottleneck
- device bottleneck
- channel bottleneck
- subsystem bottleneck (incl. cache size)

Other system resources
- LTA
- Label processing
- Channel queue size
- Number of CCW translation buffers
- VSAM string numbers
- SVA-24 System GETVIS space
- ...

Ù TD increased thruput somehow, but a new bottleneck was created, which also would have appeared on a faster UNI-processor

(see examples above)

Ù Overall workload's Non-Parallel share is too high

compared to the number of processors

Ù Not enough partitions are active concurrently

General MP Performance Aspects

PART B.

General MP Performance Aspects

What you may know already

'Motherhood' Statements (hopefully)

Ù Multiprocessor 'MIPS' are not as easily exploitable as if the total equivalent processor power (capacity) is provided on a UNI

BUT,
- starting point (CMOS) is very cost effective
- some actions can be done, e.g. More partitions
More Data In Memory (DIM)
CICS MRO

Í The 'biggest' partition (mostly CICS production) can only consume at best as many 'MIPS' as provided by 1 processor of an MP

Refer to the Performance Considerations part, under which conditions even less than the power of a single processor can be exploited by a single partition.

On a UNI, for temporary peaks, a single CICS workload could exploit the total processor capacity and thus may block lower priority tasks from being processed

Ù MP support alone does NOT provide a higher S/W capacity to any operating system

> For a certain total VSE workload, setup and VSE release, the maximum achievable system throughput of a single VSE does NOT increase vs a UNI with same overall 'MIPS'

A VSE S/W bottleneck does not vanish by using several processors concurrently

Í Proper VSE System Planning and Setup required.

CPU-power is not always a means to solve performance/capacity problems (even on a UNI)

What you may know already ...

'Motherhood' Statements (cont'd)

Ù Additional processors = added capacity

In general no improved Response or Elapsed Times

Except where CPU was/would have become an extreme bottleneck

Ù All processors in an MP system experience the SAME speed degradation

> There is no benefit if e.g. the first processor would be dedicated to the biggest VSE partition

Processor type	Capacity
9672-R1x 9672-R2x 9672-R3x	1 x 100% 2 x 85% 3 x 80%
4381-91E 4381-92E	1 x 100% 2 x 80%
Very rough values for illustration only. The relative capacities depend on - the workload - the operating system. They include both H/W and S/W overhead	

Ù Each workload has a certain share of code which may not run concurrently on more than 1 processor

Mostly system functions, share is also operating system dependent and varies with setup and workload

Í This is THE limiting factor for the number of processors fully exploitable by that type of workload,

provided that enough partitions/regions can be set up

MP vs UNI Performance

MP vs UNI Processor Performance

Ù More CPU-time is required
(sum from all processors used)
than on a single processor with identical technological characteristics

Reasons:

„ Reduced H/W speed (cache, DLAT and bus contention)

higher overall concurrency
As on UNIs, if concurrency increases
more task switches
more inter-processor communication

„ Increased S/W pathlength (extra instructions)

synchronizing and locking
dispatching
Number and individual cost of dispatching events
queuing (e.g. spin-loops)

For more details refer to the next chart

MP vs UNI Performance ...

Ù **UNI processor speed dependency**

With increasing UNI CPU utilization, the effective speed of any IBM and non-IBM processor ('effective MIPS') to perform a certain task decreases (i.e. the CPU-time increases),
e.g. by increased DLAT and cache misses.

This is also true for each individual processor of an MP system

Ù **Reasons for MP specific speed degradation**

Apart from additional S/W instructions in case of MP,
the additional degradation is caused by

High speed buffer (cache) consistency requirements (invalidation of updated cache entries in other processors via multiple copy bit)

Additional bus contention when communicating (propagating) with other processors
(via communication, via higher miss rates)

Higher cache/DLAT misses by tasks moving around between processors

(-> try to select a processor, which still may have data of the task in his local cache, but this costs S/W instructions)

Certain 'serializing instructions' causing processor idle times by waiting until all processors have finished their current S/390 instruction

Ù **Dependency of MP degradation**

At a given (!) total MP throughput, the MP degradation

is nearly independent of the number of processors

is very dependent of the total traffic on the bus

is to some extent processor type dependent

Í **MP effect is similar to UNI MIPS degradation, but more restricting and sensitive to workloads and processor implementation**

General MP Performance Targets

Besides good Response or Elapsed Times, there are ...

„ **Two principal targets for optimal MP performance:**

1. **Optimal exploitation of a given MP with a given customer workload**
(a given partition setup, even a single CICS partition ...)

2. **Maximum total VSE throughput on any MP ('bigger n')**

MP Performance Questions

1. MP Exploitation

„ **How many processors can I exploit effectively with my setup as of today ?**

Given partition setup and application mix

„ **What must/can I do in order to exploit more processors
and what alternatives do I have to change my VSE setup?**

„ **How many processors can I exploit effectively with a modified setup?**

When becomes the 'Non-Parallel state' the system bottleneck?
(The 'Non-Parallel processor' is that logical(!) processor executing Non-parallel work-units)

How many partitions do I need for that?

„ **Is enough power available on 1 processor to support my 'biggest' partition (production CICS)?**

MP Performance Questions ...

2. ES/9000 Processor Selection

„ **What MP processor fits to my needs/capabilites today/tomorrow?**

„ **What are the decision criteria to select an MP processor vs a UNI?**

How fast must the MP or UNI be?

3. Resulting Performance

„ **What are the performance benefits/impacts
throughput/capacity
response times, elapsed times
depending on workload, partition setup etc. ...?**

„ **What is the impact in VSE/ESA 2.1, in case I stay on a uni (and upgrade H/W later)?**

„ **What are the performance aspects to run VSE TD e.g. on an old dyadic 4381-92E ESA/370 processor (or equivalent, if supported)?**

„ **What has to be considered if I add additional processors in the same processor type?**

E.g. Going from 9672-R21 to a 9672-R31

May I see in certain cases a loss if I add a processor and not immediately need it capacity-wise?

VSE Implementation

PART C.

VSE Implementation

Code Classification

Non-Parallel Work Units

Ù **Traditional (UNI-processor) subdivision of code**

Key 0		Key >0
SUPVR-state	Non-SUPVR = PP-state = problem state	
Supervisor, POWER append.,	POWER, JCL, VTAM Transients, ...	Batch application, CICS code, VSAM ...

```
NOTE:
SUPVR state code runs in NP-status since it executes privileged
ESA/390 instructions.
This only indirectly has to do with Non-Parallel code, mostly
called NP-code here
```

Ù **An MP related performance target:**

Make as much code as possible/reasonable MP-capable

============>
'Parallelize' code

'UNI-code' 'NP-code' 'Non-Parallel'	'MP-code' 'parallel code or work units'
Any code requiring the Non-Parallel status:	All other code:
All key-0-code, except indicated otherwise	Non-key-0 code, except indicated otherwise

Concurrency Classification

MP Oriented View of Concurrency

Ù **'Non-Parallel code' (NP-code) cannot be executed concurrently to any other NP-code**

Ù **'Parallel code' may run concurrently with any other code**
Except if in direct functional dependency

Ù **The highest degree of concurrency is the TD, being able to run concurrently on all processors ('system-reentrant')**

Tasks/Code/Work Units can run concurrently to...				
	any other parallel task	any non-related NP-code	any related (called) NP-code	itself
'NP-code' 'Non-Parallel'	X	no	no	no
'Parallel code'	X	X	no	no
'system-re-entrant'	X	X	n/a	X

Storage Considerations

Real/Central Storage

„ **Real storage is shared between all processors:**

'Tightly Coupled'

Virtual Storage

„ **Read and Write to storage areas is controlled as today on UNIs:**

```
- key 0:   allows read/write from/to any area

- key >0:  Access (PSW) key must match storage key
            - to read data
            - to read from fetch protected areas
              (seldomly used in VSE)
            - to write data
```

„ **MP Aspects:**

- Shared areas (SUPVR, SVA-24, SVA-31)

Accessible by all processors concurrently
(in general, key 0 is required)

- Each processor has its own prefix-page (4K)

Not directly accessible by other processors

- Each processor has a private work area (SVA-31)

About 10K for work areas and control blocks,
includes a 'shared' copy of the 4K prefix-page.
S/390 architecture automatically mirrors updates

VSE Turbo Dispatching

VSE Turbo Dispatcher -a closer look-

Ù **VSE/ESA Turbo Dispatcher**

„ **can run at any time on any processor and even concurrently to itself**

since

- not the queue, only queue elements are locked, at the level of maintasks (not subtasks)

„ **recognizes key 0 and SUPVR state tasks and assigns them by default to 'Non-Parallel execution'**

Queuing occurs at transitions from parallel to NP-code

„ **requires JA=YES**
for more info on CPU-times
for optimal dispatch decisions and
partition balancing priority changes

Even with 'old' dispatcher the JA-tables were updated, as soon as a PB group is used or >1 partitions active in a dynamic partition class.
The overhead of JA=YES vs NO is only the call of the $JOBACCT dummy routine at end-of-jobstep

NP-code still can be interrupted as on a UNI, except code runs disabled already on a UNI

Ù **Design is open to enable MP capability on critical system paths (SUPVR) and subsystems**

VSE Turbo Dispatching ...

How VSE/ESA Turbo Dispatcher works

1. Work Units (UOWs)

VSE/ESA TD enables 'applications' to run on more than 1 CPU by dealing with 'work units'.
A UOW is a set of code or function or task that may be executed more or less independently.

In general, multiple UOWs exist in a VSE system, at least one per active partition. VSE/ESA TD does NOT allow any partition to have more than 1 UOW in the dispatch queue.

Non-Parallel UOWs are UOWs that cannot be processed in parallel to any other non-parallel UOW.

2. Dispatchability

A UOW is eligible for being dispatched, if all resources it is waiting for are available, e.g.

it is not waiting for a completion of an I/O operation, including page-I/O

it is not waiting for any other locked resource (e.g. LTA, locked record...)

3. Dispatching

VSE/ESA TD inspects each UOW and (if eligible for being dispatched) dispatches it on any idle processor.
If no processor is available and the priority of a newly dispatchable UOW is higher than the lowest priority of a currently processed UOW, that UOW is being interrupted and the processor continues with the newly dispatchable UOW.

A Non-Parallel UOW can only be dispatched, if the Non-Parallel state is not already active on any processor.

4. Dispatch history

Any partition may have run on any available processor of the n-way, but never on more than 1 processor at any point in time.

Any processor of an n-way may have processed instructions belonging to any VSE partition.

VSE Turbo Dispatching ...

Principal Dispatch Process (processor #x)

```
            -----------------------------------------------
           |  At any interrupt (SVC, I/O, SIGP ...)        |
           |  and before freeing a processor:              |
           |                                               |
           |  Processor #x enters Turbo Dispatcher,        |
           |               scans the Dispatch Queue        |
            -----------------------------------------------
                                 |
                                 V
            ----------------------------------------
 --------->|    Select task/work unit with          |
|          |    currently highest dispatch priority |
|           ----------------------------------------
|               |                           |
|               | Task found                | No task found
|               V                           |
|          -----------------                |
|         | NP-state        |               |
|         | required?       |               |
|          -----------------                |
|            |Yes         | No              |
|            V            |                 |
|     --------------      |                 |
|    | NP-state     |     |                 |
|    | already act.?|     |                 |
|     --------------      |                 |
|       | Yes     | No    |                 |
|       V         |       |                 |
 --------         |       |                 |
                  V       V                 V
         ----------------------      ---------------------
        | Dispatch this task   |    | Free this processor |
        | on this processor    |    | (no work to do)     |
         ----------------------      ---------------------
                     |                     |
                     V                     V
            ----------------------------------------------
           |             Exit Turbo Dispatcher            |
            ----------------------------------------------
```

Each processor dispatches independently from any other

NOTE: This is just the very basic principle,
details not shown

VSE Turbo Dispatching ...

Solutions for Non-Parallel State Contention

Ù **2 principal methods of solution**

1. Execution of a 'Spin Loop'
2. Return to dispatcher (new dispatch decision)

```
       -----------------------------------------------
      |  NP-state required, but not available         |
      |                                               |
      |  1.Spin loop            2.Dispatcher call     |
       -----------------------------------------------
              |                       |
              V                       V
       --------------           --------------------
      | Spin loop    |         | Set NP-state indic.|
      |    --->      |          --------------------
      |   |    |------>DIAG            |
      |   |    V     |          --------------------
      |   <----      |         | Turbo Dispatcher   |
      |              |         |                    |
       --------------           --------------------
              |                         |       |
              |                     |       ----------------
              |                         |
|
              V                         V
V
      -------------    ---------------------------------
---------
     | Continue    |  | Select highest dispatchable task|
|Free     |
     | (same task) |  | (higher, same, lower)           |
|processor|
      -------------    ---------------------------------
---------
```

Í **Both methods are being used by the VSE/ESA TD.**

The method selected is situation dependent

Spin Loop Considerations

Spin Loop Considerations

„ Purpose of 'active waits'

Spin loops are instructions executed by software instead of going into wait and being re-dispatched (hopefully) soon

Spin loops should be used in those cases where the cost of dispatch and re-dispatch is higher than the expected CPU-time for 'active wait'

As long as the processor cannot be used for other purposes, it is acceptable even if a spin loop formally costs more CPU-time than without.

> Spin loops should be designed even more carefully if

- under VM
- in PR/SM LPAR.

Holds also for native VSE if processor load is very high

„ Spin loops for Turbo Dispatcher

may be used for queuing for the NP-state e.g. in case of SVC or PC or External interrupts, but dependent e.g. on task and situation ...

are not used/required at all in case of a UNI-processor

do interrupt themselves after a certain time by issuing DIAG hex44 if under VM or in PR/SM LPAR.

This DIAGNOSE will invoke e.g. the VM dispatcher, which may select another VM task for being dispatched instead.

Depending on workload and also from vendor programs, about up to 3% spin time was observed:

RAMP-C	0.05%
DSW	0.15%
PACEX	0.40%

Up to 10% spin were observed for cases where vendors replace the SVC-new-PSW, which should not be done. They are aware. Please contact your vendor and inform us.

Provided TD CPU-times

CPU-times with VSE/ESA Turbo Dispatcher

Display of 'QUERY TD' command

```
                         Elapsed Time (ET)
  |--------------------------------------------------------------|

VSE JA:
        CPU-time                  OVHD-time
  |-------------------------------|-----------|

QUERY TD:
                TOT-time
  |--------------------------------------------|---|...............|
                          |----------|       SPIN-  ALLBND-time
                             NP-time         time    = idle time
                          ('Non-Parallel')
```

VSE JA results	CPU-time per active job step OVHD-time " " " " (sum of all processors only)
'QUERY TD' command (per processor and total, in current interval)	SPIN-time Spin loop time NP-time Non-Parallel time TOT-time Total time (w/o spin) NP/TOT Ratio ET Elapsed time
'More internal info'	+ALLBND-time processor idle time +#dispatcher entries +total SVC count

- JA=YES required for TD
- Current interval is since IPL, last SYSDEF TD,RESETCNT or SYSDEF TD, START|STOP command
- Internal SVC count: with FAST-SVCs, w/o re-SVCs
- SPIN-time: always 0 on a UNI, up to say 3% on a 2-way. Not contained in VSE JA and thus in IUI DSA screen
- Higher dispatch CPU-times via Turbo Dispatcher is fully counted in VSE JA OVHD time

Provided TD CPU-times ...

Sample QUERY TD console display

```
  CPU   SPIN_TIME     NP_TIME  TOTAL_TIME NP/TOT
   00         104      566303     1115630  0.507
   01           0      269776      565394  0.477
   02    INACTIVE
   03         161      319618      626749  0.509
       -----------------------------------------
 TOTAL        265     1155697     2307773  0.500

  ELAPSED TIME SINCE LAST RESET:       1901703
```

Resulting Performance Figures

```
                                          TOTj + SPINj
 Total/individ. processor utilization  = -------------
                                              ET

                             NP          = NP/TOT         on UNI
 Share of NP CPU-time = --------      |
                        TOT+SPIN         = about NP/TOT   on MP
```

'QUERY TD,INTERNAL'

Output as QUERY TD, but with addt'l information:

- Number of dispatcher entries
- Number of SVCs   in interval
  - all 'normal' SVCs
  - 'fast' SVC 107 (x'6B')
  - 'fast' SVC 117 (x'75')
  - 'fast' SVC 124 (x'7C')
  - 'OS/390 SVCs:
    - SVC 131 (x'83')
    - SVC 132 (x'84')
  - only SVCs intercepted by vendor pgms thru own vendor hooks are not included

Tools for Examining VSE TD Workloads

Determine individual partition's CPU consumptions

Ù VSE Job Accounting

„ CPU- and overhead time per active job step

No change vs UNI implementation

Ù Display System Activity (DSA) in IUI

- Total CPU utilization now may exceed 100% on an N-way.

  Consider this figure as a 'sum of utilizations of all processors'.

- The number of active processors is displayed, naturally

- For very CPU intensive test jobs, individual partition utilizations may exceed formally 100%, if other partitions run concurrently (Actual partition utilization may not exceed 100%)

  REASON: Partition utilization includes JA Overhead time, which is distributed across all partitions with the same relative amount.

Determine Non-Parallel shares

Ù QUERY TD command

Suited best.
QUERY TD described on previous charts

Ù VSE Work Desk CPU Activity Display

Configurable and flexible graphic display of QUERY TD results (Requires PTF UN83022 for APAR PN75762, on top of VSE/ESA 2.1.1)

- as a snapshot (bar charts)
- over time (history diagram)

Also available

Ù Vendor Performance Displays

TBD, TD usually provides the base info

VSE Turbo Dispatching ...

VSE Turbo Dispatcher -a closer look- (cont'd)

Ù **No Processor Affinity**
except for functional reasons

„ **No affinity of any partition to any processor**

Additional pathlength would have first to be compensated

Would be questionable anyhow if total utilization high, especially for small number of real processors

„ **No affinity of NP-code to any processor: 'floating Non-Parallel'**

Affinity would require additional dispatching overhead

„ **No affinity of I/O interrupts to any processor**

All processors are enabled

Interrupt 'storing' can be done in parallel state, but the proper interrupt handling requires the Non-Parallel state

One processor 'wins' (or 'loses')

Enabling only e.g. the that processor currently running Non-Parallel code would not be beneficial, since additional S/W overhead would be required and for other reasons

Naturally, the processor from which VSE IPL was done plays a specific role:
- cannot be STOPped
- is the only processor available for IUCV and VMCF interrupts if under VM/ESA

VSE Turbo Dispatching ... ...

VSE Turbo Dispatcher -a closer look- (cont'd)

Ù **No benefit of internal balancing of logical processors**

- TD roughly tries to balance processor usage
- TD always can select/find an idle processor and use it

Do not argue on how the TD spreads total VSE load across individual logical or physical processors

QUERY TD gives you processor individual data just for information, NOT for tuning or performance reasons.
Such type of balancing would not help to imjprove performance.

In spite of that, currently, CPU utilizations are well balanced.

NOTEs:

- Balancing of processors is dependent on the H/W. May change if H/W changes
- Under VM or in PR/SM LPAR, balancing of physical processors may be done by S/W or u-code

Ù **Reserving Processing Capacity**

It may be desirable to reserve certain processing capacities to specific partitions, without giving them higher VSE dispatching priority: e.g. for day batch

- Assigning or reserving a processor to a specific VSE partition may look as a solution, but ...
- The enhancement of VSE dispatching functions (e.g. VSE/ESA 2.2 TD) is a more flexible solution for that and works on any number of processors of any speed

Performance Effects of chosen TD approach

Í **VSE MP-dispatching is less granular, less sophisticated than in MVS**

„ **1 partition can only exploit the power of a single processor (at most and at best)**
(including all subtasks)

The effect that MVS/CICS uses some internal MVS subtasking to potentially use additional engines ...
'can be generally ignored for rough capacity estimates'

„ **More active and dispatchable partitions are needed to exploit a multi-processor system,**
e.g. the 9672-Rx1/Rx2 parallel CMOS servers

„ **Higher share of Non-Parallel code in VSE limits the maximum MP exploitation for a given workload**

„ **Partitions must queue more often/longer if Non-Parallel state is active on another processor**

„ **The MP-factor for a given number of exploited processors is potentially lower than seen for other MP supports**

Performance Considerations

PART D.

Performance Considerations

MP (N-way) Processor Environments

VSE Native

„ Important focus for MP performance and exploitation

Single VSE Guest under VM/ESA

„ All considerations for VSE native apply to the VM task 'VSE'

„ VM CP may exploit additional processor(s) and thus increase total host MP exploitation

„ VM CMS tasks likewise exploit additional processors

Multiple VSE Guests under VM/ESA

„ Single VSE's MP exploitation capability less critical

„ All considerations done here for VSE native apply to the individual VM tasks 'VSEx'

Multiple VSE LPARs

„ Single VSE's MP exploitation capability less critical

„ All considerations done here for VSE native apply to the individual LPARs

í VSE native is considered here primarily

Maximum Number of Exploitable Processors

Even with optimal partition setup...

The 'Non-Parallel processor' is fully saturated at 100%, for CPU queueing time reasons we assume here only 90%:

Max. number of fully exploitable processors

$$nMP = 0.9 / NPS \qquad (A1)$$

NPS = share of Non-Parallel CPU-time

(any mix of batch and/or CICS partitions)
(estimated or extrapolated or directly measured)
(may vary across a day, depending on load mix)

The resulting number nMP of processors is INDEPENDENT from the speed of a single processor in the MP environment. But the faster each individual processor, the more total load is required for exploitation.

TABLE A (nMP)

nMP = 0.9 / NPS (A1)

Fraction of NP-code NPS	.20	.25	.30	.35	.40	.45	.50	.55
Max # of processors nMP	4.5	3.6	3.0	2.6	2.2	2.0	1.8	1.6

Since under VM, the Non-Parallel code is 'enlarged'...

As VM guest, the effective NPS must be taken:

$$NPS\_effective = NPS \times TV\_ratio \qquad (A2)$$

Refer to the VM/VSE Only part

Number of Batch Partitions for Saturation

Number of (equal) batch partitions for saturation

$$nsat = 0.9 / (NPS \times \%CPU) = nMP / \%CPU \qquad (B)$$

%CPU = resulting CPU utilization if 1 batch partition would run alone on 1 single processor of the n-way (refer to TABLE B)

```
               KItot/MIPS               KItot
 %CPU   =  -----------------  =  ----------------      (C)
           KItot/MIPS + IOT      KItot + IOTxMIPS
```

TABLE C (%CPU)

KItot Relative I/O-intensiveness	5 Heavy	10 Heavier	15 ...	20 Avg	30 Lower	50 Low
IOTxMIPS 50	.09	.17	.23	.28	.37	.50
100	.05	.09	.13	.17	.23	.33
150	.03	.06	.09	.12	.17	.25
200	.02	.05	.07	.09	.13	.20

MIPS = equivalent number of millions of instructions executed in the average per processor second on a single processor of the n-way (it is reasonable to use the total n-way capacity/n). Naturally, (C) also can be applied to a UNI

KItot = average number of thousands of instructions between 2 successive I/O operations

IOT = average duration of a physical I/O operation in msec, e.g. 6..14 for cached, 15 to 20 for uncached I/Os

(In general only very few batch applications overlap I/Os. POWER CPU-time is considered as part of this consideration, though POWER I/Os are overlapped to partition I/Os)

Number of Batch Partitions for Saturation ...

TABLE B (nsat)

nsat = 0.9 / (NPS x %CPU) = nMP / %CPU (B)

Share of NP-code NPS	.15	.20	.25	.30	.35	.40	.45	.50	.55
%CPU= .1	60	45	36	30	26	22	20	17	15
.2	30	25	18	15	12	11	10	9	8
.3	15	11	9	8	7	6	5	4	3

í High #partitions for high MIPS (even at fast I/O)

Examples

NP share NPS	KItot per IO	MIPS for 1 proc.	IOT (msec)	%CPU on 1 proc.	#Batch partitions nsat	#expl. proc. nMP
				(C)	(B)	(A)
.25	20	8	15	.14	25.7	3.6
			8	.24	15.0	"
		12	15	.10	36.0	"
			8	.17	21.2	"
.35	20	8	15	.14	18.4	2.6
			8	.24	10.7	"
		12	15	.10	25.7	"
			8	.17	15.1	"
.45	20	8	15	.14	14.3	2.0
			8	.24	8.3	"
	10	8	15	.077	26.0	"
			8	.135	14.8	"

í Many batch partitions can/must be run before the Non-Parallel processor becomes the bottleneck

Reason is the high MIPS for the individual processors

Single CICS Consideration

Single CICS Consideration

„ **A single CICS partition can consume just the processing power of a single processor, but only if queuing of this CICS partition for getting the Non-Parallel state is negligible**

```
Refer to the chart 'Maximum Utilization by a Single Partition'
```

„ **Definitions and Assumptions:**

```
  - CPU-time share of a tx-workload consumed in the VTAM
    partition:

    Since this share is in most cases very small (.03 to .05),
    it is not considered separately in the following.
    It is simpler to include that here in the total CICS
    partition.
    For the same reason, it is not separately considered that
    all VTAM code is NP-code.

    Also, the number of CICS partitions needed in order to
    exploit 1 full processor with VTAM only is very high


  - CPU-time share of a tx-workload consumed in the POWER
    partition by the CICS Report Controller:

    This share is very small, even if RCF is used, thus neglected

  - Each CICS is assumed here with same characteristics and load
```

MP Capacity with Multiple CICSs

MP Capacity with (Independent) Multiple CICSs

„ **Max. number of processors exploitable**

```
The maximum number of exploitable processors (nMP)
as a function of the number of CICS partitions (nCICS) is:

     nMP =   nCICS / %max

    %max = max((nCICS x NPS / .9), 1)
                      |            |
                    'NP'         'home' processor is bottleneck

                                    nCICS = #CICS partitions
                                     NPS  = share of NP-code

From nCICS = 1 up to .9/NPS  CICS partitions the 'NP processor'
is not yet the bottleneck, but instead just the number of CICS
partitions (each running on its own logical processor, here
simply called 'home processor').
If more CICSs are active, the 'NP-state' becomes the bottleneck
```

nMP = nCICS (if nCICS < .9/NPS)
(#CICSs is bottleneck)
(D)
nMP = .9 / NPS (if nCICS > .9/NPS)
('NP proc.' is bottleneck)

„ **Example for NPS=0.30 Non-Parallel share**

```
    --> nMP = .9/NPS = 3.0 CICSs

  Maximum number of exploitable processors and bottleneck:

     nMP(1CICS) = 1.0     home  processor
     nMP(2CICS) = 2.0     home      "
     nMP(3CICS) = 3.0     NP/home   "
     nMP(4CICS) = 3.0     NP        "
     nMP(5CICS) = 3.0     NP        "
```

CICS MRO TR/FS

CICS MRO Transaction Routing/Function Shipping (TR/FS)

„ **On a UNI, TR and FS are used performance-wise**

- **to split required virtual storage across several specialized CICS partitions (some kind of VSCR)**
- **at cost of increased CPU-time (reduces CPU effectiveness of a UNI)**

„ **On an MP, TR and FS**

- **likewise give VSCR**
- **but MP-exploitation (throughput) will increase if additional processors are available (in spite of increased total CPU-time per tx)**

„ **The maximum achievable MP throughput is determined by the utilization of (whatever comes first)**

- **the NP-state**
- **any processor running an affected CICS partition (TOR, AOR, FOR, or any mix)**

```
TOR/AOR/FOR = Terminal/Application/File Owning Region
```

CICS MRO TR/FS ...

Multiple CICS Workload Setup for MP

Ù **The following principal alternatives exist**

„ **Independent CICS partitions**

```
- TOR,AOR,FOR-combination in independent partitions
- Brings high MP benefit if loadwise doable
  and function-wise possible
```

„ **MRO TR to several target CICSs**

```
- Separate (AOR,FOR) into independent combinations
- Brings high MP relief if doable
- Not possible if some files are required by all txns
```

„ **MRO FS to a target FOR**

```
- Move FOR processing into separate CICS(s).
  Leave TOR and AOR
- If all file requests to be function shipped:

  Small MP relief, due to high FS overhead in base partition
- If only few file requests to be function shipped:

  Small MP relief, since only few requests offloaded
```

„ **Mixtures of TR and FS**

```
- Not considered here
```

„ **Distributed tx-processing**

```
- Not considered here
```

Ù **If MP processing power is available for attractive costs, MRO overhead is less critical**

CICS MRO TR/FS ...

CICS Transaction Overview (MRO)

Relative CPU-times/pathlengths per tx (DSW workload)

- Values taken also from MVS results (but relative figures are applicable both to VSE and MVS)
- UNI processor ratios, no MP-effects
- 2 CICS partitions maximum for a transaction
- All values include Key0 and SUPVR parts
- VTAM partition not included

```
   A  Rel. CPU-time                      1.45 *
   |                                     ---
1.4-                                    |   |      1.23 **
   |                          1.17      |   |.49   ---
1.2-                          --- .19   |34%|FOR  |   |.35
   |      1.0                |16%|TOR   |   |     |28%|FOR
1.0-      ---       .95      |...|      |...|     |   |
   |     |///|      ---      |///|      |///|     |...|
 .8-     |///|     |///|     |///|      |///|     |///|
   |     |///|     |///|     |///|.98   |///|.96  |///|.88
 .6-     |///|     |///|     |84%|FOR+  |66%|TOR+ |72%|TOR+
   |     |///|     |///|     |///|AOR   |///|AOR  |///|AOR
 .4-     |///|     |///|     |///|      |///|     |///|
   |     |///|     |///|     |///|      |///|     |///|
 .2-     |///|     |///|     |///|      |///|     |///|
   |     |///|     |///|     |///|      |///|     |///|
   |_____|///|_____|///|_____|///|______|///|_____|///|__________

         VSAM      CICS   |  VSAM   |   VSAM      CICS
          LSR       DT    |   TR    |    FS        SDT
                                                (MVS only)
```

- Splitting CICS partitions via TR or FS brings only MP benefit if several target FOR/AORs or source TOR/AORs are used (refer to next chart)

* FS overhead (percentage-wise) depends on the relative intensity of function-shipped logical file requests to the FOR

** Shared Data Tables (SDT) use cross-memory services for all(!) READs (i.e. very minimal overhead), for all WRITEs FS overhead is included

Here the FS overhead for SDT is bigger since DSW R/W ratio low (1/3.7, i.e. many WRITEs as compared to READs)

- DSW is an IBM internal workload used to assess CICS tx performance

CICS MRO TR/FS ...

CICS Partition Split via MRO TR or FS

- Values extrapolated from previous chart

2 target cases shown (both are worst case regarding overhead):

- All transactions transaction routed, 50% of load to each of the 2 target CICSs
- 2 TOR/AOR CICSs (both loaded equal) function ship all requests to a 3rd CICS which owns all files

```
   A   Rel. CPU-times                               1.45
   |                                                ---
1.4-                                               |   |
   |                         1.17                  |   |.49
1.2-                         --- .19               |34%|FOR
   |     1.0                |16%|TOR               |   |
1.0-     ---                |...|                  |...|
   |    |///|               |///|.50               |///|.48
 .8-    |///|               |42%|FORb+             |33%|TORb+
   |    |///|      TR to    |///|AORb       FS to  |///|AORb
 .6-    |///|  ==> 2 target |///|       ==> 1 FOR  |///|
   |    |///|      CICSs    |...|           from 2 |...|
 .4-    |///|               |///|.50        CICSs  |///|.48
   |    |///|               |42%|FORa+             |33%|TORa+
 .2-    |///|               |///|AORa              |///|AORa
   |    |///|               |///|                  |///|
   |____|///|_______________|///|__________________|///|__________

                                   ---         ---
         ----           ---  ---> | a |    -> | a | ---> ---
     -> |    |      -> |TOR| TR    ---         ---  FS  |FOR|
         ----           ---  ---> | b |    -> | b | ---> ---
                                   ---         ---
```

- Pre-req is that total workload can be split up function-wise

í **Highest CICS partition requirement reduced to about half**

```
but at cost of
          - about 17% to 45% more total CPU-time
          - an (estimated) increase of the NPS value to
            (0.17+NPS)/1.17   for TR    e.g. 0.4 -> 0.49
            (0.45+NPS)/1.45   for FS         0.4 -> 0.58
```

These figures do not include NPS improvements by APAR PQ13099 (PTF UQ19908) as of 07/98

Maximum Utilization by a Single Partition

Maximum Utilization by a Single Partition

Ù **A single partition can exploit even less than the power of a single processor, if it must wait for the NP status, caused by other partitions running also NP-code**

The following formula can be used as a very rough estimate for this effect:

$$\%CPUpart\_max = \frac{1}{1 + \frac{\%NPrest}{1 - \%NPrest} \times NPS} \qquad (E)$$

%CPUpart_max = maximum CPU utilization of a single partition

%NPrest = utilization of the 'Non-Parallel processor' by all other partitions. It is the part 'seen' by the considered partition (i.e. the part which cannot be interrupted). It may be small, if the considered partition has higher priority than the other partitions.

The formula assumes that this partition has a high dispatching priority

If the non-parallel share NPS approaches 0 ... or if the utilization of the other partition approaches 0 ...

%CPUpart_max approaches 1.0

If, for example, other partitions utilize the Non-Parallel state by 20%, and the Non-Parallel share is 0.3, %CPUpart_max is 0.93

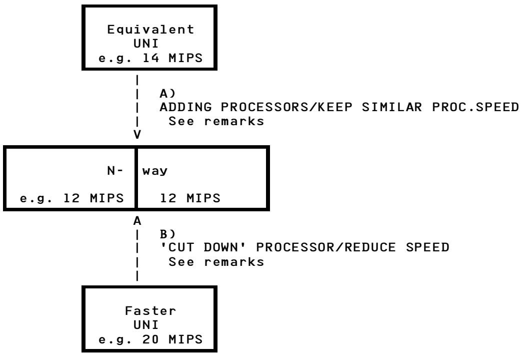
This means in practice ...

í **The power of a single engine of an n-way must exceed the actual processing requirements of the biggest VSE partition**

Migration from Uni to N-way

2 principal ways to migrate to N-ways

A) Coming from an equivalent Uni



Equivalent
UNI
e.g. 14 MIPS

A) ADDING PROCESSORS/KEEP SIMILAR PROC.SPEED See remarks

N-way
e.g. 12 MIPS 12 MIPS

B) 'CUT DOWN' PROCESSOR/REDUCE SPEED See remarks

Faster
UNI
e.g. 20 MIPS

B) Coming from a faster Uni

	System/VSE Capacity	Partition Capacity	Response Time
A) Add processors	INcreases	Similar, INcreases if CPU offloaded	Similar, Improves if CPU offloaded
B) Cut down proc.	Depends	Reduces	Higher
This summary contains very rough classifications only			

The example above simply assumes a 2-way and VSE native

Migration from Uni to N-way ...

A) Coming from an equivalent Uni

= Adding processors

„ In general, no TD specific problems

Except, when 1 VSE partition uses >70% of total VSE power

Watch out for problems which
are caused by higher throughput, and which
would also have appeared on a UNI

Any emerging VSE or setup bottleneck

B) Coming from a faster Uni

= Having 'smaller per-engine-ITR'

„ Problems if speed/capacity of 1 engine not sufficient for biggest VSE partition

„ CPU intensive night single-batch jobs may run slower

Likewise applies e.g. to (long running) single thread update transaction

Í Restructure night batch work to achieve more parallelism

Refer to 'Night Batch Window' in VSE/ESA 1.3 document

Caution for both cases: Be aware of 'Latent Demand' (source processor >90% full at peak hour)

N-way Related Properties of Workloads

N-way Related Properties of Workloads

1. Share of Non-Parallel code

   NPS is a rough overall number

2. Relative frequency of transitions into NP-state

   This number is the frequency of potential conflicts when NP-state is required.
   It is one indication for N-way overhead, be it via
   - more dispatcher calls/cycles or
   - more spin time

3. Relative dispatch intensiveness

   This relative frequency is determined by SVC, I/O and timer interrupts and by the design dependent SIGP frequency.
   Roughly spoken, it is THE major impact factor for N-way overhead (TD overhead on UNI + MP-factor)

These 3 main characteristics result mainly from ...

„ Relative I/O intensiveness

This value is SVC0 related, the real number of I/Os is setup-dependent. It also determines the I/O interrupt frequency

„ Distribution and type of supervisor calls (overall = normal + fast)

The type of individual SVC (plus the Function Code FC for Fast-SVCs) determines whether a call could be made Non-Parallel.
Also it is a measure of the pathlength spent in NP-state per SVC

„ Relative frequency of timer interrupts

This frequency depends on
- MSECS
- the number of active partition balanced partitions
- the usage of other timers, by CICS, monitors ... etc

Performance Results

PART E.

Performance Results

Overview

Ù General Remarks
Ù Overall Performance
Ù Measurement Results (mostly 9672-Rx1)

RAMP-C Online
DSW Online
DSW+EXPLORE/VSE
DSW+EXPLORE/VSE on 9121-320/480
with Variations under VM/ESA
PACEX Batch
Mixed Online/Batch

General Remarks to TD Results

General Remarks

Ù Worst case workloads were kept deliberately for development reasons

- partly high I/O or file intensiveness (RAMP-C, PACEX)

  High file intensiveness

  -> high supervisor and dispatch intensity

  -> high Non-Parallel share

  Refer to workload descriptions e.g. in the VSE/ESA 2.1/2.2 base document

Ù The following processors were used so far

```
9221-170  (UNI) and 9221-200 (DYAD)
9672-Rx1  (UNI to  6-way)
9221-211  (UNI) and 9221-421 (DYAD)
9121-320  (UNI) and 9121-480 (DYAD)
```

„ Performance does not differ between CMOS and 'non-CMOS'-processors,

at same basic ESA/390 MIPS

„ 9672-Rx2 TD MP-factors do not differ from Rx1

The same applies to 9672-Rx4 and to 2003 processors (refer to LSPR results at the end of this document).

Ù Exploitation problems for 2- and 3-ways resolved and response times improved,
at cost of CPU-time

(TD overhead on Uni and MP-factor)

Overall Performance

1. Maximum Number of Fully Exploitable Processors

Ù **Up to about 3 processors can be fully exploited**

(NP-share varies from about 0.25 to 0.5)

Workload	Approx. Non-Parallel share NPS	Max # processors (native) nMP ***
Customer workloads	TBD	TBD
SAP R/2 production	.20	**
DSW Online - " - +EXPLORE/VSE	.27 .30	3.3 3.0
RAMP-C (DIM setup)	.31	2.9
RAMP-C (I/O intens.)	.41e	2.2e
PACEY (ESA expl.)	tbd	tbd
PACEX Batch *	.47	1.9

```
 -    nMP = 0.9 / NPS
 e    estimated
 *    Very file and thus supervisor intensive load.
      NP-share varies from 0.30 and 0.55 for
      individual jobs
**    SAP R/2 loads can hardly be split across
      multiple CICS partitions
 -    NP-share only slightly increases when going
      from a UNI to an n-way
 -    NP-share may vary across a day, depending
      on load mix
***   For VM/VSE  the number of fully exploitable
      processors is reduced by the T/V-ratio
```

Naturally, a lot of dispatchable batch partitions are required, especially on high-capacity 9672 CMOS n-way processors

Overall Performance ...

2. CPU-Time Costs

	VSE/ESA 2.1		
Dispatcher Processor	UNI UNI	TD UNI	TD MP
CPU-time cost	5-10% ------>	MP-factor -------->	
Overall thruput ratio	------------------>		

Ù **Turbo vs old dispatcher on a UNI: about 5-10% cost**

Measured values (latest status):

```
+15%   for PACEX (I/O and supervisor intensive)
       (real worst case)

+4%    for DSW-CICS (CICS function intensive) and DY43919

+7%    for RAMP-C DIM (very file intensive)
```

3. MP-Factors

Definition

Throughput ratio of an n-way to corresponding UNI, at SAME total processor utilization:

$$\text{MPfactor} = \frac{\text{CPUT\_uni}}{\text{CPUT\_nway} / n} = \frac{n}{\text{CPUT\_nway} / \text{CPUT\_uni}}$$

Pre-req is that the selected total CPU utilization can be achieved for the specific type of workload!

Overall Performance ...

MP-factors (cont'd)

MP-factors on 9221-200 vs 9221-170 (UNI) at 70%/90%
TD 2.1.1 results from 9672-R21)
2-way

Workload	VSE TD	VM/VSE (2xSD)	MVS/SP 4.2.0
TSO IMS LSPR CICS	- - -	- - -	1.73 1.62 1.8x
RAMP-C (DIM setup) TD 2.1.2+	1.65 @80%	1.70e	-
RAMP-C (IO-intens.)	tbd	1.63	-
DSW TD 2.1.2+	1.72 .. 1.75	1.82e	-
PACEX TD 2.1.2+	1.4	-	-

VM/VSE: 2x VSE/ESA 1.3 under VM/ESA 1.2.1
MVS/SP: Source is WSC Flash 9418
- Be aware of manifold dependencies of MP-factors

Ù **VSE MP-factors for 9672-R CMOS processors**

Processor	VSE RAMP-C DIM	VSE DSW&LSPR	MVS IMS/TSO	MVS LSPR
9672-R21 2-way	1.65	1.75	1.8	1.88e
9672-R31 3-way	2.2	2.4	2.5	2.69e
9672-R41 4-way	2.7 e *	2.8e *	3.1	3.41e
9672-R51 5-way	*	*	3.7	-

```
e   estimated/expected
-   Base is the 9672-R11 UNI processor
-   MP-factors are very workload dependent
*   No claim to exploit 4-/5-ways fully
```

Í **Entry MP performance for VSE TD**

Overall Performance ...

NPS and MP-Factor Relationship

„ **NPS simply gives the relative amount of CPU-time running in NP-state**

It does NOT tell directly anything on

how often transitions from Parallel to Non-Parallel state are done.
This is one contributor influencing N-way performance

how often the dispatcher is called (relative dispatch intensiveness).
This is another parameter influencing N-way performance

Í **It does NOT directly give an indication of how effectively a given n-way can be exploited**

Only that it can be fully utilized, if at all

„ **MP-factor simply tells how effectively a selected n-way can be exploited:**

Throughput ratio at same overall CPU util., mostly 90%, sometimes 70%, PROVIDED the NPS allows you at all to exploit the n-way to that level

„ **If NPS does NOT allow to exploit a selected n-way, NO MP-factor at all exists for this n-way**

MP-factors at <70% make no sense

Í **Having the same NPS for 2 loads does NOT mean that their MP-factors are also same.**

BUT: A very rough first guess for an MP-factor is what has been measured for another workload with a similar NPS

There is only a statistical relationship, no load specific one

('Your mileage may vary')

Overall Performance ...

What Factors Determine N-way Performance?

Ù Processor type

- Only minor MP-factor deltas between newer 9221s, 9672s, 9121s and 9021s

Ù Workload type

„ Frequency and type of system services called
„ Non-Parallel Share NPS
„ Relative Intensiveness of

transitions into Non-Parallel state

dispatcher calls (includes I/Os, normal-SVCs, timer interrupts, ...)

Fast-SVCs which must run Non-Parallel

Ù Workload setup

„ Number and type (Batch/CICS) of active partitions
„ Required CPU-power for 'biggest' partition

Ù All factors determining performance on a UNI, at same throughput

Avoid system bottlenecks with higher loads

Ù TD PTF level

Overall Performance ...

Results on 9672-Rx1 (RAMP-C and DSW Online)

„ Multiple CICS partitions (partition balanced)

In general, on n-ways more CICS partitions are used. So any performance deltas due to more CICS partitions are contained in the measured MP-factors.

„ Each CICS with 300 or 400 terminals and 6 user volumes

„ Cached 9345 devices,
6 channels, 2x 64M cached CUs

Refer to next pages

TD Results for RAMP-C

RAMP-C Results on 9672-Rx1

„ TD status as of 03/96 (2.1.2+, DY43919)

#proc.	#CICSs	tx/sec	RT (sec)	CPU% sum	IO/sec	Rel. CPUT/tx	NPS
1 SD	2	52.2	0.32	83.5%	365	0.93	-
1 TD	2	51.8	0.43	89.4%	363	1.00	0.291
2	2 3	52.4 72.7	0.26 0.37	108.5% 165.4%	366 550	1.20 1.22	0.322 0.316
3	4 4	100.4 105.5	0.79 1.80	244.5% 250.9%	713 717	1.40 1.38	0.313 0.293
4	4	101.1	0.70	271.7%	724	1.55	0.317

- SD = standard dispatcher
- Runs with 4 CICSs started to be bound by the Non-Parallel utilization (was 76.5%/73.5%/86.2%)
- Any runs with fewer CICS partitions than processors would have been partition number bound

í TD overhead on Uni:

about 7%

í MP-factors (RAMP-C, DIM setup):

2/1.22 =1.65 3/1.38 =2.17

Both cases at about 82% total CPU utilization

í Overall throughput ratio (n-way vs UNI with SD):

1.65/1.07 =1.54 2.17/1.07 =2.03

About 53%/103% more RAMP-C throughput on 2-/3-way

TD Results for DSW+EXPLORE/VSE

DSW+EXPLORE/VSE Results on 9672-Rx1

„ VSE/ESA 2.1.1+ TD status: DY43697

#proc.	#CICSs	tx/sec	RT (sec)	CPU% sum	IO/sec	CPUT/tx (rel.)	NP util.
1 SD	2	37.1	0.22	74.3%	210	0.96	-
1 TD	2	37.2	0.25	77.3%	210	1.00	22%
2	2 3	55.8 75.8	0.30 0.66	132.0% 180.0%	321 421	1.13 1.14	41% 53%
3	4	91.1	1.74	239.4%	505	1.25	72%

- SD = standard dispatcher
- Different tx-rates from different terminal think times (varied from 7 to 12 sec).
  Each CICS partition had 300 active terminals
- NP-share was .29/.30/.30 on 1/2/3-way

í TD overhead on Uni: about 4.0%

í MP-factors (DSW+EXPLORE/VSE):

2/1.16e = about 1.72 3/1.25 = about 2.4

2-way at about 90%, 3-way at 80% total CPU utilization

Similar n-way related figures here as w/o EXPLORE/VSE

í EXPLORE/VSE overhead here:

CPU-time:	about 4% to 6%
I/Os :	very minor
RT :	minor
NPS :	.30 vs .27 (2- and 3-way)

(Base were corresponding runs without EXPLORE/VSE)

Note that EXPLORE/VSE overhead depends on the monitoring options. SVC monitoring is CPU-time expensive and was not used here.

TD Results for DSW Online

DSW Results on 9672-Rx1

„ **VSE/ESA 2.1.2+ TD status: DY43919**

#proc.	#CICSs	tx/sec	RT (sec)	CPU% sum	IO/sec	CPUT/tx (rel.)	NPS
1 SD	2	32.4	0.20	62.1%	188	0.99	-
		48.9	0.64	91.2%	276	0.96	-
1 TD	2	32.2	0.29	65.0%	185	1.04	0.259
	2	48.1	0.51	93.0%	265	1.00	0.253
	2 CIT	38.4	0.57	90.6%	220	1.22	0.212
2	2	82.1	2.68	180.5%	457	1.135	0.273
	3	85.0	1.43	185.9%	463	1.13	0.274
3	4	101.3	3.86	252.3%	561	1.29	0.285
4	4	103.2	3.72	279.3%	567	1.40	0.296
	4 CIT	95.6	4.45	299.8%	537	1.62	0.255

```
- SD = standard dispatcher
- Different tx-rates from different terminal think times
  (varied from 3 to 15 sec).
  Each CICS partition had 300 or 400 active terminals
- Runs with higher tx-rates started to be Non-Parallel State
  bound (up to 83% utilization)
- NP-share was .25/.27/.27 on 1/2/3-way
* 2 CICSs on 2-way discussed separately
CIT is with CICS Internal Trace  on
```

í **TD overhead on Uni: about 4%**

í **MP-factors (DSW):**

2/1.135 = about 1.75   3/1.29 = about 2.35

2-way at 93%, 3-way at 84% total CPU utilization

TD Results for DSW Online ...

DSW Results on 9672-Rx1 (2.1.2+) (cont'd)

í **Overall throughput ratio (n-way vs UNI with SD):**

1.75/1.04 = 1.69 (2-way)   **2.33/1.04 = 2.24** (3-way)

Higher DSW tx-throughput on n-ways

„ **2 CICSs on 2-way consideration**

With Online CICS transaction workload alone (no Batch on top), the following observations hold:

Full 2-way exploitation still possible
Response times somewhat higher at high utilization

„ **CICS Internal Trace impact**

Costs about 17% to 22% CPU-time

Gives higher response times at same tx-rate

Reduces Non-Parallel Share

```
from .25 to about .21 (2-way)
from .27 to about .23 (3-way, est.)
from .29 to about .25 (4-way)
```

TD Results for DSW Online (VM/VSE)

DSW+EXPLORE/VSE Results on 9121-320/480

„ **VSE/ESA 2.1.1+ TD status: DY43697**

#proc. etc.	#CICSs	tx/sec	RT (sec)	CPU%	ITR	CPUT/tx (rel.)	NPS
Native, with EXPLORE/VSE							
1 SD	4	71.2	0.23	91.2%	78.1	0.770	-
1 SD L	4	74.1	0.18	90.4%	82.0	0.683	-
1 TD	2	65.7	0.26	90.3%	72.8	0.826	0.375
	4	65.6	0.25	92.7%	70.8	0.849	0.389
1 TD L	4	68.9	0.18	90.7%	76.0	0.633	0.351
2	2	95.8	0.70	73.8%	129.8	0.926	0.373
	3	85.3	0.21	70.2%	121.5	0.989	0.385
		108.2	1.06	87.3%	124.0	0.969	0.372
	4	85.3	0.21	71.9%	118.6	1.013	0.391
		108.7	0.96	90.4%	120.2	1.000	0.370
Native, no EXPLORE/VSE							
2	4	113.1	0.44	88.4%	127.9	0.940	0.355
VM/ESA Guest (incl. EXPLORE/VSE)							
1 SD V=R	4	65.8	0.23	90.0%	73.1	0.822	-
1 TD V=R	4	60.2	0.23	90.3%	66.7	0.901	0.396
1 TD V=V	4	54.6	0.27	92.7%	58.9	1.020	0.405
2 V=R	4	98.1	0.99	91.0%	107.8	1.115	0.386
2 V=R, D	4	101.1	0.59	87.1%	116.1	1.035	0.387
2 V=V	4	84.9	0.63	89.8%	94.55	1.271	0.391
2 V=V, D	4	86.0	0.46	85.1%	101.1	1.189	0.390
VM/ESA V=R Guest, 4 VSE log. processors defined							
4 on 1	4	45.2	0.27	90.3%	50.1	1.201	0.450
4 on 2	4	84.0	0.48	91.7%	91.7	1.311	0.418

```
- 3380-K's and 3390-2's attached via non-cached 3990's,
  using up to 38 volumes at up to 10 parallel channels
- All terminals simulated with a VSE internal driver,
  running all key-0 (i.e. Non-Parallel) and highest VSE priority
- L means 'VSE internal driver with Low priority'
- Different terminal think times (9 to 15 sec)
  and total terminal numbers (960 to 1680)
- ITR is the number of tx's per n CPU-seconds on n-way
- NPS is the Non-Parallel share of CPU-time: small variations.
  Highest NP util. (at 108.7 tx/sec): 0.370x2x90.4%=66.9%
- All VSE DASDs defined to VM/ESA 2.1.0 as DEDicated DASDs
- D means 'DEDicated 2nd processor'
```

TD Results for DSW Online (VM/VSE) ...

DSW+EXPLORE/VSE Results on 9121-320/480 (cont'd)

í **TD results would not differ on same speed CMOS**

a) Native Conclusions

í **TD overhead on Uni:**

```
- higher than w/o the internal VSE driver
- depends on driver setup (rel.# disp.calls per tx)
```

	Terminal Driver	Disp.calls/tx
about 9%	Internal, Hi prior.	1.5
about 8%	Internal, Lo prior.	1.3
about 4%	External	1.0 Base

í **MP-factors (incl. EXPLORE/VSE):**

2/1.178 = 1.70

(4 vs 4 CICSs at 90%)

2/1.142 = 1.75

(3 vs 4 CICSs at 90%)

2/1.173= 1.705

(3 vs 2 CICSs at 90%)

í **EXPLORE/VSE overhead here:**

```
CPU-time:  about 6%

RT      :  measurable, since utilization high

NPS     :  0.38 vs 0.35
```

Note that EXPLORE/VSE overhead depends on the monitoring options.
SVC monitoring is CPU-time expensive and was not used here.

TD Results for DSW Online (VM/VSE) ...

DSW+EXPLORE/VSE Results on 9121-320/480 (cont'd)

b) VM/VSE Guest Conclusions

í MP-factors (incl. EXPLORE/VSE):

2/1.237 = 1.62

(4 vs 4 CICSs at 90% V=R)

2/1.250 = 1.60

(4 vs 4 CICSs at 90% V=V)

This is about 5% smaller than the native MP-factor

í VM/VSE guest/native ratios:

		g/n ratio	Remark
V=R	Uni SD Uni TD 2-way	0.936 0.916 0.897	 2% lower than SD 4% lower than SD
V=V	Uni SD Uni TD 2-way	0.840 0.832 0.786	 very similar to SD 6% lower than SD

í More VSE logical than physical processors:

Higher CPU-times required on any processor

```
1.201/0.901 = 1.33   on 1-way (9121-320)
1.311/1.115 = 1.18   on 2-way (9121-480)
```

... as expected. Just a functional test

í DEDicated 2nd processor:

About 7% better CPU-time, thus better response times, but...
2nd processor unavailable for other VM tasks.

Dedication recommended where feasible

VM/VSE Guest/Native ITR-Ratios

VM/VSE Guest/Native ITR-Ratios for TD

Ù General aspects:

„ TD on uni does not use more privileged instructions than the SD

They require CP interception under VM

„ TD on n-way uses DIAGNOSE and SIGP on top

„ Dispatching is only a smaller part of total load

Ù DSW Measurement Results for VM/ESA 2.1.0:

„ On uni:

VM/VSE guest/native (+ T/V) ratio
same as for SD

„ On 2-way:

VM/VSE guest/native ratio
(and VM/VSE MP-factor)
... about 4% lower vs uni

Refer to the 9121-320/480 VM/VSE DSW results

TD results for DSW (Overview)

TD results for DSW (Overview)

```
   UNI SD                      UNI TD                    2-WAY
               TD Ovhd                      MP-Factor
               on Uni
 -----------               -----------               ----------
|           |  0.91       |           |  1.70       |          |   NATIVE
| ITR=78.1  |---------->|    70.8    |---------->|  120.2   |
|           |           |NPS=0.389  |           |  0.380   |
|RT=0.23 sec|           |   0.25 sec|           |  0.96 sec|
 -----------               -----------               ----------
      |                         |                        |         Guest/
      | 0.94                    | 0.94                   | 0.90    Native
      |                         |                        |         Ratio
      V                         V                        V
 -----------               -----------               ----------
|           |  0.91       |           |  1.62       |          |
|   73.1    |---------->|   66.7     |---------->|  107.8   |   V=R
|           |           |   0.390    |           |  0.386   |   GUEST
|  0.23 sec |           |  0.23 sec  |           |  0.99 sec|
 -----------               -----------               ----------

      |                         |                        |         Guest/
      | 0.84                    | 0.83                   | 0.78    Native
      V                         V                        V         Ratio
 -----------               -----------               ----------
|           |  0.90       |           |  1.60       |          |
|   65.6    |---------->|   58.9     |---------->|   94.5   |   V=V
|           |           |   0.405    |           |   0.391  |   GUEST
|  0.24 sec |           |  0.27 sec  |           |* 0.63 sec|
 -----------               -----------               ----------
```

```
All figures apply to DSW Online workload and the specific setup used
     - 9121-320/480, CMOS values are very similar
     - with EXPLORE, VSE internal driver at highest priority,
       and 4 CICS DSW partitions
     - VM/ESA 2.1.0, for V=R guest: DEDicated devices
                     for V=V guest: no MDC was used
     - 90% overall utilization
     - ITR = #tx per n CPU-sec
     - tx/sec = 0.9 x ITR (here)
The NPS values only slightly vary, just for illustration.
Consider that response times hold for different transaction rates
and are given for illustration only (* RT is at much lower tx rate)

All values are workload dependent: 'Your mileage may vary'
```

Acknowledgement

All the 9121-320/480 runs were done by Greg Kudamik,
VM Development, Endicott (NY).

TD Results for PACEX Batch

Worst Case Results for PACEX Batch (03/96)

„ PACEX as a very heavy I/O intensive workload

„ 9672-Rx1, VSE/ESA TD status 2.1.2+ (DY43919)

„ 5 user volumes per 4 active batch partitions

„ Cached 9345 devices, 6 channels,
2 x 64M cached CUs

#proc.	#batch part.	ET (sec)	jobs /min	CPU% sum	IO/sec	CPUT/ part. (sec)	NPS
1 SD	8 stat 8 dyn 16	245 255 431	13.7 13.2 15.6	68.8% 69.8% 82.6%	695 696 806	21.09 22.27 22.29	- - -
1 TD	8 stat 8 dyn 16	252 259 469	13.3 13.0 14.3	77.8% 80.3% 87.8%	676 685 741	24.46 26.01 25.79	.452 .484 .471
2	16	370	18.8	159.4%	937	36.87	.484
3	16	393	17.1	198.2%	885	48.66	.454
- PACEX16 consists of 8 static + 8 dynamic partitions							

> Dynamic vs static partition overhead:
about 6% CPU-time, 4% I/Os here for PACEX

í TD overhead on UNI:
about 15% here for PACEX

í MP-factor for 2-way (PACEX):

2/(CPUT-ratio) = about 1.4

Varies with utilizations

TD Results for PACEX Batch ...

Results for PACEX (cont'd)

í **Overall throughput ratio (2-way vs UNI with SD):**

1.4/1.15 = 1.22 for PACEX

About 22% more PACEX throughput (worst case, 2-way)

Comment to 3-way trial:

- Since 2-way processor is 'nearly maxed out' (high Non-Parallel utilization), adding a 3rd processor even reduces throughput.

  Normal case is increased throughput, also at increased CPU-time

More Details for Individual Workloads

More Details for Individual Workloads

- System Resource View -

„ **9672-Rx1, VSE/ESA 2.1.1+ TD status: DY43697**

Run-ID	RAMP-C alone (PB) no EXPLORE R21 09289501	DSW alone (PB) no EXPLORE R31 09279510	PACEX16 alone (PB) no EXPLORE R21 10239503	PACEX8 alone (PB) EXPLORE R21 10049503
tx/sec	78.4	103.9	-	-
RT	0.31 sec	3.53 sec		
CPU% sum	162.1%	248.7%	165.8%	127.6%
IO/sec	545	559	959	674
msec/IO	-	-	-	8
Max CHANQ used (255)	109	72	25	15
Max Copyblks (BUFSZ=3000)	571	513	2997	ca 2435
Max GETV.used				
SVA-24	1116K (89%)	1088 (87%)	508K (40%)	ca 388K
SVA-31	3816K	3028	512K	
Locks fail				
Ext.	0.6%	0.9%	8.9%	ca 8.6%
Int.	7.5%	1.7%	5.2%	ca 1.0%
NP/TOT	0.307	0.271	0.453	0.530
NP util.	50%	67%	75%	67%
Batch ET	-	-	362 sec	264 sec

- PB = Partition Balancing
- All 'external' locks here internal since w/o lockfile

Some potential VSE system resource bottlenecks are shown

TD Results for Mixed Workloads

'Mixing Online and Batch' on a 2-way

„ **9672-Rx1, VSE/ESA 2.1.1+ TD status: DY43697**

„ **DSW Online + PACEX8 Batch (incl. EXPLORE/VSE)**

- 2 CICS partitions with 300 terminals each (9 sec thinktime)
- 8 PACEX partitions, 7 batch jobs each, 1 dynamic partition/class

Run-ID	DSW alone 10059501	DSW + PACEX8 mix 10099501	PACEX8 alone (no PB) 10049502	PACEX8 alone (PB) 10049503
tx/sec (rel)	45.82 (1.00)	37.96 (0.83)	-	-
RT	0.18 sec	0.30 sec		
CPU% sum	108.6%	170.1%	121.7%	127.6%
IO/sec	266	539	649	674
msec/IO	9	10	8	8
Max CHANQ used (255)	62	52	14	15
Max Copyblks (BUFSZ=3000)	1246	2057	2435	na
Max SVA-24 GETVIS used	1136K = 90%	1176K = 93%	388K = 31%	na
Locks fail				
Ext.	3.0%	6.7%	8.6%	na
Int.	1.7%	1.2%	1.0%	na
NP/TOT	0.308	0.383	0.532	0.530
NP util.	33%	65%	64%	67%
Batch ET	-	642 sec	274 sec	264 sec
(rel.thruput)		(0.43)	(1.00)	(1.04)

- All runs above on 2-way 9672-R21
- Shared I/O with channel/CU/device contention
- No PRTYIO set
- PB = Partition Balancing

Also some potential VSE system resource bottlenecks are shown

TD Results for Mixed Workloads ...

'Mixing Online and Batch' on a 2-way (cont'd)

í **This mixed load exploits 2-way to 85% overall,**

- with 83% of 'Online alone' throughput
- with 43% of 'Batch alone' throughput

í **Costs are**

- increased RTs (here by 0.12 sec)

- about 19% more CPU-time

vs running loads sequentially

í **Batch partition balancing**

- costs 1% CPU-time

- brings 4% more 'Batch alone' throughput

TD Results for Mixed Workloads ...

3-Way-Study with Mixed Online/Batch

„ **9672-R31, VSE/ESA 2.1.1+ TD status: DY43697**

„ **DSW Online + PACEX8 Batch (incl. EXPLORE/VSE)**

- 3 CICS partitions with 300 terminals each (5 sec thinktime)
- 8 PACEX partitions, 7 batch jobs each, 1 dynamic partition/class

Run-ID	DSW alone	DSW + PACEX8 mix 10129501	PACEX8 alone (no PB)
tx/sec (rel) RT	-	94.05 - 2.48 sec	-
CPU% sum IO/sec msec/IO Max CHANQ used (255) Max Copyblks (BUFSZ=3000) Max GETV.used SVA-24 SVA-31 Locks fail Ext. Int. NP/TOT NP util.	206%e	221.3% 691 11 1176K (93%) 2792K 0.323 71%	130%e
Batch ET (rel.thruput)	-	1628 sec -	-
- 3-way 9672-R31 - Shared I/O with channel/CU/device contention - No performance relevant console messages			

„ **This example is a still ongoing study**

TD Results for Mixed Workloads ...

3-Way-Study with Mixed Online/Batch (cont'd)

í **Mixed load here exploits 3-way to 74% overall**

For higher exploitation with this mix, tuning is required.
Potential candidates:

- I/O contention
- CHANQ
- BUFSIZE (copy blocks)
- SVA-24 GETVIS space
- Wait-on-string
- LTA usage/misusage
- others, TBD

POWER Spooling in VSE/ESA 2.2

Results for 'More Parallel POWER'

„ **9672-R11 Uni processor**

Suffices for the NPS demonstration here

„ **Average spool intensive PACEX workload**

(1 and 16 partitions here)

and Heavy spool intensive LIBR LIST job

Workload	Case	Elapsedtime ET	Rel.CPU-time CPUT	NP-share NPS
PACEX1	2.1.1 SD 2.1.1 TD 2.2 TD	189 sec 180 sec 180 sec	0.86 1.00 1.005	- 0.456 0.414
PACEX16	2.1.1 TD 2.2 TD	367 sec 371 sec	1.00 1.003	0.488 0.446
SPOOLINT	2.1.1 SD 2.1.1 TD 2.2 TD	18.3 sec 19.3 sec 19.0 sec	0.84 1.00 1.03	- 0.597 0.325
- SPOOLINT is LIBR LIST of a big member - VSE/ESA 2.2 POWER with autostart statement SET WORKUNIT=PA				

„ **At only very small CPU-time increase ...**

NPS reduced by 10% and over 40% (relative)

í **Non-Parallel state is offloaded for any other system activities (non-POWER related)**

Direct benefits are experienced only
- if base load spool intensive
- other load(s) can profit from Non-Parallel state offload

- APAR DY44442 (UD50251/50252) also helps to reduce the increase of SVC7s for parallel POWER with the associated overhead
  - on all processors, when POWER (using the NPC specification) has lower priority ...
  - on 2-ways when POWER (as usually used and recommended) has higher priority ... ... than the spooled partition

Performance Modelling/Prediction

PART F.

Performance Modelling/Prediction

Overview on Prediction

1. **Maximum #Processors Fully Exploitable**
2. **CPU Requirements (per partition/total)**
3. **Check Capacities of Selected N-way**

MIPS

'MIPS'

„ **Technical aspects of real and effective 'MIPS'**

They have been discusssed in 'To MIPS or Not to MIPS ?'
in 'VSE/ESA Hints for Performance Activities'

„ **Any (even 'effective') MIPS value only makes sense if the MIPS value of another processor for the same load is cited:**

If customer or anybody else rates processor A as A 'MIPS'
(whatever 'MIPS' is and whatever the dependencies are),
processor B has A x ITRR 'MIPS'

ITRR is the ITR-ratio (e.g. reciprocal CPU-time-per-txn ratio)

„ **IBM only claims ITRRs**

This is the only technical feasable way

LSPR ITRRs are based on actual runs
Any discussion on precise MIPS rating is fruitless
One only can see and measure ITRRs
Naturally ITRRs also vary with the workload
So even ITRRs are subject to some variation

í **WHATEVER 'MIPS'-rating you prefer to apply, the target processor has ITRR times the 'MIPS' of your base**

You better rely on the applicable ITR-ratios from LSPR (which are based on actual measurements) rather than for any anonymously paper-derived IBM or non-IBM figures.

Evaluate Max. #Processors

1. Maximum #Processors Fully Exploitable

(at optimal partition setup)

„ **1a) Determine/Estimate the Non-Parallel share of the workload (NPS)**

VSE Release	Tool
Pre VSE/ESA 2.1	NPS must be estimated, refer to sample loads (*1)
VSE/ESA 2.1	QUERY TD Performance Monitors
*1 A rough estimate would be a tool to determine the key 0 CPU-time share (tool not known/planned). Would have higher CPU-time overhead, thus only for temporary use	

Note, that NPS
only slightly varies with CPU-utilization
only slightly increases from 1 to 2-ways
may vary across a day, depending on load mix

> Even smaller 2.1 test loads on equivalent UNI may suffice for a first estimate

Refer to 'NP-Share Determination' in part 'Performance Hints'

„ **1b) Use formula (A) to determine max #processors fully exploitable**

```
nMP  = 0.9 / (NPS x TV_ratio)   (Overall load,       (A)
                                  CICS and/or Batch)
```

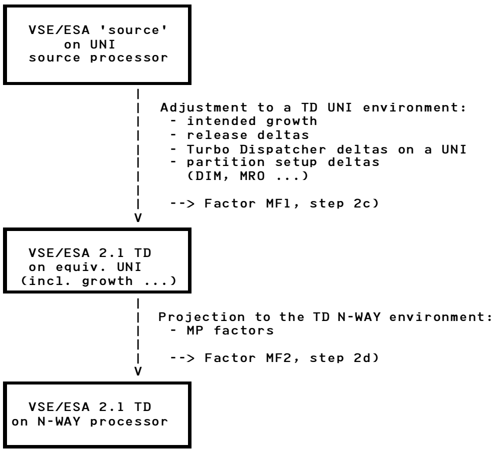
For TV_ratio use 1.0 if native, use T/V-ratio if under VM

Predict CPU Requirements (Summary)

2. CPU Requirements for VSE Partitions (and Total)

„ **Rationale to Calculate Ratios in CPU Requirements**

Individual partition and total CPU/processing requirements
(Refer to next pages for details)



```
 VSE/ESA 'source'
     on UNI
 source processor
        |
        |  Adjustment to a TD UNI environment:
        |   - intended growth
        |   - release deltas
        |   - Turbo Dispatcher deltas on a UNI
        |   - partition setup deltas
        |     (DIM, MRO ...)
        |
        |   --> Factor MF1, step 2c)
        V
 VSE/ESA 2.1 TD
 on equiv. UNI
(incl. growth ...)
        |
        |  Projection to the TD N-WAY environment:
        |   - MP factors
        |
        |   --> Factor MF2, step 2d)
        V
 VSE/ESA 2.1 TD
on N-WAY processor
```

„ **Note**

For the two S/W related factors MF1 and MF2 above, the actual speed/power of the processors does not play any role.

The only link to REAL H/W here is that the MP-factor (MF2) should correspond to the real MP-factor on the target processsor, to be selected later.

Predict CPU Requirements

2. CPU Requirements for VSE Partitions (cont'd)

(e.g. 'MIPS consumed') for a representative peak hour

„ **2a) Determine CPU utilization of each partition**

Use CPU-time/Elapsed Time ratio, if not directly given

VSE Release	Tool
Pre VSE/ESA 2.1 or VSE/ESA 2.1	VSE Job Accounting (JA) Display System Activity (DSA) Performance Monitors
Note that QUERY TD only gives CPU-times for all active partitions together. Make sure that you really use data from a peak hour, e.g. values from several 5 min intervals	

„ **2b) Multiply CPU utilizations with the total processing power of the source system**
(e.g. 'MIPS', to get 'consumed MIPS')

„ **2c) Adjust CPU requirements to TD S/W environment (still as UNI)**

- Select a factor corresponding to your intended growth (caused by more or bigger transactions, or by consolidating from other VSEs)
- Select a CPU-time factor for release deltas, e.g. 1.04 for VSE/ESA 1.3 to VSE/ESA 2.1
- Select e.g. 1.05 to reflect TD overhead on a UNI

Note: If partition setup differs for the TD environment (e.g. more DIM, or use of MRO to split partition), this has to be taken into account in addition

These factors result in Multiplication Factor MF1:

```
MF1 = (growth x release x TD on uni x setup) CPU-time factor
```

• Cont'd on next page

Predict CPU Requirements ...

2. CPU Requirements for VSE Partitions (cont'd)

„ **2d) Adjust CPU req'ments to N-way environment**

í **The resulting maximum partition CPU requirement (biggest partition) determines the type/class of N-way to be selected**

Select an N-way processor which may satisfy partition and total CPU req'ments, ...

(and does not exceed the maximum number of fully exploitable processors, based on the Non-Parallel share).

If done here, it allows to consider processor dependent MP-factors (in case they should differ measurably)

Consider CPU-time increase in the N-way environment

```
- Select/Estimate the expected VSE MP-factor MPf
  (refer to VSE TD measurement results)

--> Increase in CPU-time by going with the TD from 1 to n-way
    is another Multiplication Factor
```

```
MF2 =  n/MPf   (>1)
```

Predict CPU Requirements ...

2. CPU Requirements for VSE Partitions (cont'd)

„ **2e) Consider target processor utilization(s)**

To switch over from 'consumed MIPS' to 'processor speed-MIPS'...

multiply CPU requirements e.g. with 1/0.8,

```
MF3 =  1/target_utilization(s)
```

e.g. MF3 = 1/0.8, if average target processor utilization (single processor and overall of N-way) should not exceed 80%

í **Required speed-MIPS for biggest partition and total**

```
Required speed-MIPS values

 = (source_utils)x(source-MIPS) x MF1 x MF2 x MF3   (B)
```

3. Check Capacities of Selected N-way

„ **Conditions an N-way must fulfill (Overview)**

```
a) Processsing requirement of biggest partition
   must fit on 1 processor

b) Total processing requirement must fit on n-way

c) The Non-Parallel share NPS must allow n-way exploitation

d) Naturally, the following values must be selected correctly

  - ITR ratios  (Uni-ratio, not MVS, not VM, from VSE LSPR)

  - MP factors  (workload specific, from this document)
    (DON'T use MP factors e.g. from MVS)
```

Predict CPU Requirements ...

3. Check Capacities of Selected N-way (cont'd)

„ **a) On a single processor of the N-way**

```
Target-MIPS_on_1_proc

      = Source-MIPS x ITR-Ratio_equivUNI x (MPf / N)
```

This ITR ratio is the ITR ratio from the source processor to the technology-wise equivalent UNI. So you must know e.g. what processor is the 'UNI-version' of your target n-way:

3-way	2-way	Equiv. UNI
- - -	9221-200 9221-421 9221-221	9221-170 9221-211 none (0.70 x 211)
- - 9121-732	9121-480 9121-521 9121-621	9121-320 9121-411 9121-511
9672-R31 9672-R32 9672-R34	9672-R21 9672-R22 9672-R24	9672-R11 9672-R12 9672-R14
2003-135 2003-136	2003-125 2003-126	2003-115 2003-116
<--------- 3-way --------- MPf <--- 2-way ---- MPf		

„ **b) On the total N-way**

On an n-way, in total n times the processing power is available as on 1 engine of the N-way:

```
Target-MIPS_on_total_N-way

       = Target-MIPS_on_1_proc x N

       = Source-MIPS x ITR-Ratio_equivUNI x MPf
```

Predict CPU Requirements ...

Notes

„ **'MIPS'**

The use of effective 'MIPS' is the simplest way to make the calculations as easy/understandable as possible.
Refer to the chart 'MIPS'

Whatever 'MIPS' classification or 'philosophy' you may apply, the following condition must be fulfilled:

```
MIPS of (UNI) target processor
----------------------------  = ITR ratio from (VSE) LSPR
MIPS of (UNI) source processor
```

Here, as long as VSE LSPR does not include N-ways, the (equivalent) UNI-processors are being considered

„ **Conceptual step via 'Equivalent UNI'**

This is a pure conceptual step for better understanding and planning of the various impacts.

Would not be explicitly needed, if

```
    - MP-factors would be implicitly included
      in VSE n-way capacity figures (e.g. LSPR)
    - MP-factors would vary less
```

Again, this step helps for better understanding

4. Check any Required Refinement for VM/VSE

If applicable and not already taken into consideration, (re-)check the following VM/VSE refinements:

```
  - slightly lower MP factor vs native

  - changed guest native ratio
    (via changed setup of VM guest)
```

Predict CPU Requirements (Example)

Requirements for Individual Partitions (Example)

On a 9221-191 with VSE/ESA 1.3 native (about 10 VSE/ESA MIPS), the CPU utilizations shown in Table 1 have been observed during a representative peak hour interval:

TABLE 1 Partition	Utilizations ut on Source processor 2a)	Consumed CPU Power 'MIPS' on Source 2b)	TD Uni 2c) (x MF1)	TD N-way 2d) (x MF2)
VTAM	ut_VTAM =0.05	MIPS_VTAM = 0.5	0.7	0.8
CICS1	ut_CIC1 =0.60	MIPS_CIC1 = 6.0	7.8	8.9
CICS2	ut_CIC2 =0.05	MIPS_CIC2 = 0.5	0.6	0.7
POWER	ut_POWR =0.01	MIPS_POWR = 0.1	0.1	0.1
BATCH1	ut_BAT1 =0.07	MIPS_BAT1 = 0.7	0.8	0.9
BATCH2	ut_BAT2 =0.02	MIPS_BAT2 = 0.2	0.2	0.2
Total	ut_tot =0.80	MIPS_tot = 8.0	10.2	11.6

- Note that batch throughput ratio is hard to project
  - if going from Uni to N-way or changing processor speed
  - and if any online utilization approaches 100%
- 'MIPS' here is ANY reasonable figure for the effective speed/capacity of a processor (Refer to separate chart)

Assumptions for this Example:

- The intended growth for the production CICS is 20%.

- The total workload is not heavy I/O intensive, thus roughly
  - the release delta figure may be assumed as 3%,
  - the TD vs UNI overhead is assumed as 5% more pathlength

- No change in partition setup is planned, VSCR provided by VSE/ESA 2.1 is used for growth, without exploiting more DIM.

- Non-Parallel Share is estimated to be about 0.35

Predict CPU Requirements (Example) ...

Example (cont'd)

The maximum number of fully exploitable processors here is

```
nMP =  0.9 / 0.35 = 2.6   (A)
```

Thus only a 2-way can be fully exploited, though, a 3-way may be filled in certain periods, too.

So, if hypothetically VSE/ESA TD would be used on a UNI-processor, the source CPU requirements would have to be multiplied here by

```
          1.20 x 1.03 x  1.05 = 1.30
MF1 =
          1.00 x 1.03 x  1.05 = 1.08
```

depending on where the anticipated growth will take place.

Assuming a 9672-R21 2-way target processor and an estimated VSE MP-factor of 1.75, ... results in a factor

```
MF2 =  2/1.75 = 1.14
```

Table 1 shows that an n-way will be required which allows to consume

```
       -  8.9 MIPS  consumed on a single processor alone
                    ('biggest partition')
       - 11.6 MIPS  consumed on the total processor.
```

If the adequate N-way utilizations are e.g. about 80%, the nominal capacity/speed of the n-way must be multiplied with

```
MF3 =  1/0.8 = 1.25
```

The following Requirements result here:

```
Required speed-MIPS values

 = (source_utils)x(source-MIPS) x MF1 x MF2 x MF3   (B)

->  8.9/.8 = 11.1 MIPS  for a single processor alone
-> 11.6/.8 = 14.5 MIPS  for the total processor
```

Predict CPU Requirements (Example) ...

Example (cont'd)

Check feasability of selected n-way:

From LSPR, refer e.g. to the LSPR/PC figures in this document, the (UNI-)processor ITR-ratio can be calculated:

```
             ITR_9672-R11      0.72
ITR-Ratio = -------------- =  ----- =  1.41 (UNI-ratio)
             ITR_9221-191      0.51
```

(actually, LSPR shows ITR-ratios to a common source processor)

So, since we assumed here about 10 MIPS for the 9221-191, the equivalent UNI 9672-R11 has about 14 MIPS.

Using the same conditions as described in Step 3 'Check Capacities of Selected N-way', it results:

a) 14/MF2 = 14x(MPf/n) = 14x1.75/2 = 12.25 is larger than 11.1 MIPS (1 single engine has enough power)

b) 12.25 x n = 24.5 is larger than 14.5 MIPS (Total processor power is big enough)

c) NPS allows full 2-way exploitation

d) These conditions have been observed

> The selected 9672-R21 is OK

Simple H/W Migration Case

'Simple' H/W Migration Case (no S/W or setup change)

VSE/ESA 2.1 with TD already used on Source-UNI

```
   SOURCE                   EQUIVALENT                   TARGET
    UNI                     TARGET UNI                    N-WAY
              ITR-ratio                MP-factor
  -----------              -----------              -----------
 |           | ITRR_UNI |             |   MPf    |             |
 | ITR-S-UNI |--------->| ITR-T-UNI   |--------->|             |
 | from LSPR |  calc.   | from LSPR   |  from    |ITRR-T-NWAY  |
 |           |from LSPR |             |this doc  |             |
  -----------              -----------              -----------
```

```
              ITR-T-UNI  |                                  MPf
 ITRR_UNI =  ----------  | ITRR-T-NWAY = N x (ITRR_UNI   x ----)
              ITR-S-UNI  |                                   N
```

```
   9121-411                 9672-R12                9672-R22
  -----------              -----------              -----------
 |           | ITRR_UNI |             |   MPf    |             |   Native
 | ITR-S-UNI |--------->| ITR-T-UNI   |--------->|             |   example
 | =  5.44   |  calc.   | = 4.77      |   1.7    |ITRR-T-NWAY  |
 |  in LSPR  |from LSPR |  in LSPR    |          |             |
 |  (1.0)    |          | (->0.87)    |          |(= 2x0.74)   |
  -----------              -----------              -----------
```

```
   0.87 = 4.77/5.44      |     2x0.74 = 2 x (0.87 x 1.7/2)
```

Without any S/W change, the 9672-R22 2-way provides 2 times 74% of effective processing power i.e.

- 74% for any VSE partition
- 148% for the total VSE/ESA

This result must be checked whether it fits in a specific case (see 'Predict CPU Requirements')

Here is the corresponding example for a V=R guest:

```
   9121-411                 9672-R12                9672-R22
  -----------              -----------              -----------
 |           | ITRR_UNI |             |   MPf    |             |   VM/VSE
 | ITR-S-UNI |--------->| ITR-T-UNI   |--------->|             |   V=R
 | =  6.20   |  from    | = 5.12      |   1.6    |ITRR-T-NWAY  |   example
 |           |  LSPR    |             |          |             |
 |  (1.0)    |          | (->0.825)   |          |(= 2x0.66)   |
  -----------              -----------              -----------
```

Benefits/Costs of Additional Engines

Benefits/Costs of Additional Engines

Here it is assumed that the additional engine can be exploited for the assumed type of workload and setup:

```
        i.e. from the - TD NPS value
                      - biggest VSE TD partition
                      - #VSE guests
                      - biggest VSE uni guest
```

„ **MP-factors for various workloads**

Without workload specifics, which may have a lot of impact, you usually may see the following average MP-factors (and power per engine) for the different environments shown:

Total ITRs and (ITR per engine)				
Environment	1-way	2-way	3-way	4-way*
VSE TD native	1.0 (1.0)	+0.7 ---> 1.7 (0.85)	+0.6 ---> 2.3 (0.77)	+0.5 ---> 2.85 (0.71)
VM/VSE, 1 TD guest	1.0 (1.0)	+0.65 ---> 1.65 (0.82)	+0.55 ---> 2.2 (0.73)	+0.45 ---> 2.65 (0.66)
VM/VSE Uni-guests	1.0 (1.0)	+0.85 ---> 1.85 (0.92)	+0.7 ---> 2.55 (0.85)	+0.65 ---> 3.2 (0.80)
MVS/ESA	1.0	+0.9 ---> 1.9 (0.95)	+0.8 ---> 2.7 (0.90)	+0.7 ---> 3.4 (0.85)

- All values may vary with processor type and workloads
* 4-way sizing for single VSEs needs careful checks

By adding an engine and by full exploitation

í **total capacity increases**

í **capacity per engine decreases**

Turbo Dispatcher Performance Hints

PART G.

Turbo Dispatcher Performance Hints

Where to use TD?

Where to use TD?

Ù **Use Turbo Dispatcher ONLY if**
a single VSE needs more than a single processors power

(naturally, not if 1 partition alone eats up say >70% of all)

or

you need the enhanced partition balancing or VSE/ESA 2.2 relative SHAREs

This is of benefit for specific cases

or

you need info on the Non-Parallel share

or want to test whether your programs would run

This may be done only temporarily

There will be more reasons to use the TD, when some dispatching/balancing enhancements are implemented.

Refer to VM/VSE regarding consolidation of VSE guests

Where NOT to use TD

Ù **On N-ways, without check of applicability**
Ù **Under VM, to define an N-way on a uni**

Note
Ù **If possible, start with 2 processors**
More than 2 processors require careful evaluation

Tuning VSE for the TD (Summary)

Tuning VSE for the TD (Summary)

Apart from
- having enough concurrently dispatchable partitions (see 'Partition Setup', later)
- having installed the latest TD level
- having installed the latest TD PTFs from vendors ...

1. Tune VSE as for the Standard Dispatcher

„ **Reduce total CPU-time**
- More or More intelligent setup of Data In Memory (CICS Data Tables, Mult. VSAM LSR with shorter subpools)
- Better usage of VSE system resources, e.g. GETVIS/FREEVIS (GETVIS subpools, clustering of GETVISs, includes LE enclave creation)
- Careful specification of performance relevant parameters (CICS SIT, VSE standard options, Trace options, POWER DBLK, 3800 spooling ...)
- More effective application design

2. Tune VSE in order to reduce Non-Parallel CPU-time
All aspects apply as for Standard Dispatcher above, with specific care for non-parallel CPU-time

„ **Reduce inefficient use of system services**
SVC statistics from SIR MON may help
„ **Reduce, if possible, usage of key 0 programs**
Measure NPS for varying environments and/or activities
„ **Check for TD related PTFs**
„ **Reduce, if possible, number of task switches, timer interrupts**
„ **Run POWER in parallel mode**

3. Tune for lowest CP overhead (T/V ratio)

Performance Hints for Customers

Achieve Throughput Economically (lowest CPU-time)

„ **Use/Define only as many processors as required**

Additional processors

- if not exploitable due to limited #partitions

- if not required since others are used below say 70%-80%

increase the total CPU-time per job or tx.

This is caused e.g. by 2 effects:

- more frequent ALLBOUND processing (ALLBOUND costs more than on a UNI-processor)

- more communication to idle processors via SIGP

Defining more processors may hurt even if 1 processor is already fully utilized with a single partition (CICS):

Single processor speed is reduced and hence the processing capacity of the biggest CICS partition

„ **Use as many partitions (especially CICS) as required on the selected n-way**

The impact of more CICS partitions than required is much smaller than the impact of additional (non-required) processors (as long as that does not mean a higher frequency of MRO)

Performance Hints for Customers ...

Partition Setup

„ **Set up more batch and/or (independent) CICS partitions**

Exploitation of fast single processors and of multiple processors

„ **If req'd and possible ...**

Split up huge CICS partitions into multiple partitions

with CICS MRO

- Transaction Routing

- Function Shipping

- Shared Data Tables (not with CICS/VSE 2.x)

Refer to the CICS MRO section in this document

„ **'Go relational'**

SQL/DS on 1 processor can run concurrently with CICS on another processor

This split of required partition CPU-power is an extra bonus when 'going relational'

Consider CPU-time increase with increased relational functionality

SQL/DS 3.5 even allows to do data base switching

Í **Multiple SQL/DS partitions (or application servers) for 1 CICS allow more concurrency**

Non-Parallel Share

NP-Share Hints for Maximum MP Exploitation

„ **Reduce share of NP-code by exploiting DIM**

Saves I/Os and thus also supervisor code

Gives also better response times and/or allows higher processor utilization

Naturally requires sufficient real storage

„ **Be aware that Virtual Disk is mostly running Non-Parallel**

Contention may occur if NP-utilization is already very high and VD use extensive

Maybe in such a case using a VM VD is an alternative

„ **Do NOT use the NPA parameter option in // EXEC, except where really required**

This parameter is an 'emergency exit only', is 'unsocial' if used w/o urgent need

„ **DUMP is also running Non-Parallel**

Not expected to be a production system problem

Non-Parallel Share ...

„ **Usage of DEBUG option for problem analysis: slightly higher Elapsed and Response time overhead**

compared to Standard Dispatcher

3 DEBUG areas (SVA-31) as for SD,
- each used wraparound, switched to next at cancel condition
- size specifiable with DEBUG, default= 64K each

Extra DEBUG area (64K) in SVA-31 used wraparound for TD specific entries

Short critical path is locally locked to ensure proper trace entry sequence

DEBUG code runs in parallel or non-parallel state

NP-Share Determination

„ **Determine and monitor your share of NP-code**

Use e.g. QUERY TD to check

Non-Parallel Share ...

„ **Be aware of specific key 0 programs, increasing NP-share overproportionally**

„ **Determine NP-share in varying situations**

night batch / batch alone
varying day batch
with/without a key 0 program
(if possible, see above)

Will give you hints on the sensitivity of your load mix, which may change after moving to TD and n-ways

Very small dependency of NP-share from
- CPU utilization (typical job/job mix sufficient, not full load required)
- number of processors (1-/2-way)

„ **NP-share of composed workloads**

Mixing or adding workloads will change the overall NP-Share (NPS)

Just extending the tx-rate or number of concurrent batch jobs means adding load with the same NPS

Resulting NPS when 2 loads a) and b) are mixed:

$$NPS = \frac{(\%CPUa \times NPSa) + (\%CPUb \times NPSb)}{\%CPUa + \%CPUb}$$

%CPUa and %CPUb is the resulting individual CPU utilization (as indications for throughput) in the mixed environment.

These values are not easily to be determined. This also applies to the sum %CPUa+%CPUb which is the total resulting CPU utilization

More Turbo Dispatcher Hints

More Turbo Dispatcher Hints

Ù **Usage of SDAID trace will require UNI-processor mode**

- Additional processors must be stopped before via SYSDEF TD
- Use SDAID not during peak hours

Ù **Do NOT use TPBAL**

For TD, costs are even higher than benefits

Refer to TPBAL charts in VSE/ESA 2.1 base document

Ù **Install TD PTF for APAR DY43919 (or higher)**

This PTF is included in VSE/ESA 2.1.3

TD enhancements to exploit 3-ways more efficiently

- Improved processor communication via SIGPs
- More parallel running SVCs

Ù **Install VSAM PTF for APAR DY43952**

This PTF (included in VSE/ESA 2.1.3) is available since 04/96

Apart from general VSAM improvements for 2.1, some benefits especially hold for the TD:

„ **Savings of non-parallel SECTVAL SVCs**

„ **Savings of TD calls when data compression is used and ICCF is up**

This VSAM PTF pre-reqs the TD PTF for APAR DY43919

VSE System Load Balancing

Workload/Partition Balancing

„ **Multiple processors require new load balancing**

After having spread a VSE system across several processors it may be required to balance the VSE system anew:

PRTY

With multiple processors, the need for careful setup of PRTY is still important

PRTYIO

Should be re-checked

Partition Balancing (PB) and MSECS

PB internal priority rearrangements occur at 'MSECS times'.

Refer to the following charts for more info.

„ **Aspects for Workload Balancing on n-ways**

Less 'discrimination' between partitions of different priority
i.e. partitions not in same partition balancing group

í **CPU allocation is 'more social'**

More dispatching of lower priority jobs
- More Non-Parallel code
or - More noninterruptile (disabled) NP-code

If NP-code of a lower priority job runs disabled...

í **Higher priority partition must queue for NP-state**

VSE System Load Balancing ...

Aspects for Partition Balancing for TD

(also valid on UNIs)

„ **Each active partition (dynamic or static) within the PB group has the same weight**

> With 'Old Dispatcher',

- a total dynamic partition class has same priority (time slice) as any other dynamic class or static partition in the PB group
- the priority of a dynamic partition depends on the number of active dynamic partitions in that class in the PB group

> With TD,

- any balanced partition has SAME priority within the PB group (not considering different PRTY SHAREs), thus...
- DYNAMIC partitions are weighted same as STATIC ones

„ **Example for a PB group**

PRTY ....,C=D=F4,.............

Assume, currently 5 dynamic partitions of class C are active, 10 of class D.

The priority of each partition within the partition balancing group (size of time slice) is

```
'Old Dispatcher':
        C      D      F4
      1/3,  1/3,  1/3    for each class/static partition
      1/15, 1/30, 1/3    per individual partition

Turbo Dispatcher:
        C      D      F4
      1/16, 1/16, 1/16   for each partition
```

VSE System Load Balancing ...

Aspects for Partition Balancing for TD (cont'd)

„ **With the TD, PRTY setup has to be changed,**

```
 except all of the following conditions are fulfilled

     - you stay on a UNI

     - no partition balancing group defined

     - no dynamic partitions in PB group

     - only 1 partition per dynamic class used

  In the exceptional case above TD cannot show benefits,
  TD clearly was NOT done for this case.
```

„ **Do not 'overconsolidate' VSE systems or VSE partitions, even on MP processors**

```
  ... as long as only 1 PB group is provided

Refer to chapter 'VSE/ESA Workload Balancing' in

          'IBM VSE/ESA 1.3/1.4 Performance Considerations'
```

Í **VSE TD Relative SHAREs reduce (or even avoid) the need for >1 PB groups**

```
Refer to the separate foil
```

VSE/ESA 2.2 Turbo Dispatcher

Load Balancing Enhancements in VSE/ESA 2.2 TD

Ù **Situation**

„ **Any terminal/user driven Online load can monopolize CPU consumption**

„ **If processor not powerful enough, no chance to get even a small 'day batch' throughput,**

even if customer is willing to limit CICS performance slightly

```
(increased response times and lower Online throughput)

It is impossible to have a Batch and a CICS partition in the
same PB group, before VSE/ESA 2.2
```

Ù **Problem solved with VSE/ESA 2.2 Turbo Dispatcher via 'Relative CPU Shares'**

Í **Better (more flexible) control of VSE/ESA partitions in case of high overall CPU utilization**

VSE/ESA 2.2 Turbo Dispatcher ...

Setting Relative CPU Shares

Ù **PRTY SHARE command allows to set and retrieve the SHAREs for the balanced group**

```
which holds static partitions/dynamic classes

Balanced Group defined e.g. via PRTY BG,C=F5=F6=F8,F2,F3,F1
```

„ **Each member of the balanced group has a default SHARE**

```
Default SHARE value is 100
```

„ **All dynamic partitions have the SHARE of the corresponding dynamic class**

Ù **PRTY SHARE,<x>=n to set a SHARE value**

```
where <x> = static partition or dynamic class
       n  = any value out of 1 .. 9999 (low .. high priority)
            (0 means lowest priority in PB grp, but unbalanced)
```

e.g. PRTY SHARE,C=50

Ù **PRTY also displays the SHAREs**

Ù **Current time slice of balanced group member calculated via MSECS and SHARE of member**

```
(individual SHARE / sum of all SHAREs of active PB partitions)
```

Í **RELative SHAREs**

```
Complex customer load situations can be handled w/o the need for >1
partition balancing groups
```

Some Hints for PRTY SHARE Settings

Background

„ **Dispatching of partitions outside the PB group is not affected**

```
Still absolute priority of a higher priority partition (or the PB
group)
```

„ **SHAREs have higher effect at times when processor full**

```
Partitions below the PB group are not affected by SHAREs, except
that now they can be put into the PB group for the first time
```

„ **SHAREs only have effect to partitions when they are dispatchable**

```
A very I/O intensive partition may benefit less from higher SHAREs
```

„ **Within the PB group, the SHAREs result in 'soft capping'**

```
A PB partition can get more than its share in a PB group, IF others
can not use their shares
```

„ **Classification of partitions regarding traditional PB suitability**

Type of partition	PB suitability (w/o VSE SHAREs)
CICS Online	Balancing several production CICSs was usual (Concurrent I/O per partition)
Data Base Server	Traditionally had lower, same, or higher priority as CICS (Concurrent I/O per partition)
Batch	Was not balanceable in practice with any CICS (I/Os mostly single thread)

```
A batch partition was so far not balanceable with a production
CICS, since a CPU intensive batch could dominate the processor
(provided I/O was completed, more CPU could always be consumed)
```

Some Hints for PRTY SHARE Settings ...

SHARE Hints

„ **Assign the SHAREs in the PB group in an evolutionary manner**

An 'uplifted partition' (moved from below the PB group into it) should start with a lower SHARE value than the average value. Even the lowest SHARE suffices to give a partition absolute priority over a partition below the PB group.

A 'downgraded partition' (moved from above the PB group into it) should start with a high SHARE value (maybe higher than the sum of all the rest in the PB group).

„ **Select SHARE values noticeably different, to cover the entire priority spectrum:**

Use values between say 50 to 2000

„ **Observe the balancing results**

Consider the conditions above on impact of SHAREs, at different loads across a day

Be aware that by more concurrent activities of partitions in the PB group e.g. file contention may show up.

This effect may be lowered for server partitions, which may differentiate between requests originated from Batch vs Online

„ **Correct/Refine values**

Since customer workloads vary a lot, also across a day, finding optimal values is an iterative way.

More Hints

For more info refer e.g. to

```
 II09513   Information APAR, describing these balancing enhancements
 DY44052   with PTFs UN49992(94,95) providing this function
```

TD and 1 Big Rel.-Share-Balanced Group

TD and 1 Big Rel.-Share-Balanced Group

Ù **Background**

„ **TD PRTY SHARE settings are very effective, but only apply to the partition balanced (PB) group in PRTY**

„ **Only 1 PB group is available**

„ **Sometimes a partition priority is beneficial,**

```
which
    - is high enough to avoid e.g. overruns (TCP/IP)
    - but still allows others to continue in case of
      high temporary CPU demand
      (or even a loop, especially on a UNI processor)
```

What to do if PB group is already used, e.g. for batch?

Ù **Potential Solution**

„ **Try to get better overall VSE partition dispatch by setting up 1 big PB group**

Exploit TD Relative CPU Shares to the max

Assign Rel. Shares such that

- relation is roughly like desired CPU utilizations
- increase those values which are RT critical

Ù **Customer Experience**

Very good (running since about 01/99, posted in VSE-L 03/99)

Make sure the PTF for TD APAR DY44847 (as of 04/99) is installed.

MSECS Setting for Balancing

MSECS Setting for Balancing

Ù **Background**

- PB internal priority rearrangements occur at 'MSECS times' in a VSE system.

  Not only the potential rearrangements of temporary dispatch priority within the PB group is done (also using the PRTY SHARE values), but also a scan of all other active partitions.

- A PB internal partition priority is changed, essentially, when a partition has consumed 'enough' CPU-time since the last change.

Ù **Some Measurement Results**

- PACEX I/O Intensive Workload

  - 9672-R11 CMOS processor (roughly)
  - 7 I/O-intensive batch jobs per partition
  - 8 (dynamic) partitions, all in 1 PB group
  - Every partition had the same total work to do (to get a mix, sequence of jobs was 'rotated')

	MSECS 976 (Default)	MSECS 100	MSECS 9760
Elapsed Time Ending Window *1	250 sec 23 sec	245 sec 13 sec	270 sec 89 sec
CPU Time	208.4 sec (1.000)	209.7 sec (1.005)	206.9 sec (0.995)
NPS # Disp. Entries # SVCs	0.502 1021.3K 1068.4K	0.503 1027.5K 1068.5K	0.503 1024.9K 1068.9K

*1 'Ending Window' is the time from 'first partition available' until 'all partitions completed'

> Fairest balancing is obtained for MSECS 100, at only 0.5% CPU-time cost

> High MSECS saves only 0.5% of CPU-time, but balancing is not granular enough (biggest ending window)

MSECS Setting for Balancing ...

Ù **Recommendations**

„ **The MSECS default of about 1 sec (976) is reasonable in most cases**

„ **In general, MSECS should be**

- small enough, to provide enough granularity for control by PB
- big enough, to avoid unnecessary CPU-time overhead

In any case, you may try for your environment, but no major other results are expected than sketched above.

„ **MSECS may be set lower on faster processors**

„ **The MSECS value for an n-way usually can stay the same as on the same single engine UNI**

VSE/ESA 2.3 TD Enhancement

VSE/ESA 2.3 TD Enhancement

„ **QUERY TD Enhancements**

The QUERY TD command provides additional information concerning the workload:

Spin Share:

```
     (SPIN_TIME) / (SPIN_TIME + TOTAL_TIME)
```

This is the share of time spent by processors in so-called spin-loops.

Overall utilization sum:

```
     (TOTAL_TIME + SPIN_TIME) / ELAPSED_TIME
```

This value corresponds to the sum of all individual processor utilizations, which can add up to n x 100% (native)

NP Utilization:

```
     (NONPARALLEL_TIME / ELAPSED_TIME)
```

This value is additional info to the well known 'Non-Parallel Share' NPS (or NP/TOT). It is the utilization of the non-parallel status and can reach at most 100% (native).

It is a good indicator of the remaining potential for achieving more total throughput, especially with more processors

Performance Hints for Vendors

Performance Hints for Vendors

„ **Enable and/or favor actions by customers**

Refer to 'Performance Hints for Customers'

„ **Avoid key-0 code where appropriate, check whether running disabled is required**

„ **Use those ESA features (if possible) which do not require SUPVR state**

Avoids that by default the processing state of the code is running Non-Parallel

- Use AR-mode (can run completely in PP-state)

„ **Do Not replace SVC-new-PSW**

This action leads to performance degradation, since non-parallel status is enforced. The effect is a major increase of the spin-time, showing up to about 10%, vs about 2%.

Use the provided vendor interfaces, as described in SC33-6331.

TD Exploitation by IBM/Vendor Programs

TD Exploitation by IBM/Vendor Programs

There are different grades/steps of exploitation possible:

Ù **Level 1: TD 'Toleration'**

- This step is a real pre-requisite for any further step.

  It simply means that the program runs with the TD functionwise. For very most programs, this is already fulfilled, if the program runs at all in a VSE/ESA V2 environment. This really is the lowest level of 'support' imaginable

Ù **Level 2: TD 'Toleration+'**

- This step is fulfilled if several copies of the same program can be run CONCURRENTLY in several partitions of VSE/ESA V2 with the TD.

  It especially applies to those programs (e.g. SORTs) for which so far it was not reasonable/beneficial to allow several copies to be run, since already 1 copy was very CPU intensive. The TD with support of several processor makes this option mandatory

Ù **Level 3: 'TD Exploitation'**

- Allow to split a partition load (so far in 1 partition only) dynamically across several partitions

Ù **Level 4: 'TD Exploitation+'**

- Allow that the load of a single task is split up into several tasks and later on be combined

This is a theoretical option only, SYSPLEX not supported by VSE

Ù **In general, to get a low Non-Parallel share:**

- Avoid Key 0 where possible
- Allow code to run Non-Parallel (even key 0 code may run Non-Parallel, if synchronization is done by the program)

Software AG's Exploitation of TD

Software AG's Exploitation of TD

Ù **ADABAS C data base product exploits n-ways**

Was implemented in 2 phases:

1. **Non-Parallel share was reduced**
   - via improved 'Hand-Shaking' on TD and Vendor side (needs TD level 7 or above)
2. **Can run concurrently in >1 VSE partitions ('SMP')**
   - 1 for Updates (should get higher dispatch priority)
   - n for Reads

   - Data buffers in separate address spaces (only small duplication)
   - Efficient use of 'invalidation bits' for data consistency
   - High benefits from a common ESA data space used as S/W cache (further reducing the Non-Parallel Share)

Environment

- ADABAS V6.1.3 (or 6.1.2 +PTFs) or 6.2.1 (SMP)
- SAG phases in SVA: ADASTUB, ADANCHOR, ADASVC61

Ù **Customer Production Results**

Obtained 01/97, on 9672 2-way, 1 Update, 2 Read partitions. ADACSH was used to apply Data In Memory (reduce SSCHs) in Phase 1

	Original	Phase 1 6.1.3	Phase 2 6.2.1
Non-Parallel Share	0.36	0.29	0.19

Ù **More info**

- 'Software AG's Enablement of VSE/ESA Turbo Dispatcher' by Peter Harris, SAG, VM/VSE Tech Conf, Kansas City. May 13-16, 1997. Session #36F. sagph@sagus.com June 16-18, 1997. Session #56D.

Í **Contact vendor to get latest vendor level for TD**

CA Products and the TD

CA Products and the TD

Ù CA System Adapter History

„ Originally used SVC NEW PSW swap

causing increased SPIN-time

„ Now uses new TD functions

- Specific 'Hand-Shaking' functions for system related vendors (12/96, DY44265)

  SWITCHPU, SWITCHNP, RESETPU

- specific FLIH intercept for TD

Ù Required CA Software levels

- LO16881-9705 CA90s GenLevel
  meanwhile replaced by LO22343

  Exploits new TD functions, as cited above

- LO16874-9705 CA90s GenLevel

Ù Customer Non-Parallel Share Results

	Original	With System Adapter PTFs (9705)
Non-Parallel Share	0.35 - 0.45	0.20 - 0.35

Ù Newest level for System Adapter is 9907

Cont'd

CA System Adapter in General

Ù Avoid that CA-products run in BG

SYSCOM and BG COMREG reside in the first page and thus cannot be updated in NP-mode.

This applies to all vendor products, IBM products not affected

CA System Adapter in General

Most of the System Adapter is VLA-31-bit capable

„ Should be loaded into the VLA-31

Just to save virtual and thus real storage,
since concurrently used from several partitions.
Ensures availability of SVA-31 space

„ Should NOT be loaded into the VLA-24

Would be a waste of shared space below the line

„ If not loaded into any VLA, it would be loaded automatically into 'SVA-ANY', when required

'Load deferred'

„ It may be of performance benefit, to start >1 engines only AFTER the System Adapter

More info

- 'CA Products and the VSE/ESA Turbo Dispatcher' by Greg Lee,
  CA, VM/VSE Tech Conf, Kansas City, 05/97. Session 36F

í Contact vendor, to get latest vendor level for TD

Note
IBM cannot confirm the accuracy of performance, compatibility, or any other claims related to non-IBM products.

Questions regarding the capabilities of non-IBM products should be addressed to the suppliers of those products.

Addt'l VSE/ESA TD Performance APARs

Addt'l VSE/ESA TD Performance APARs/PTFs

The following are hints to additional TD specific performance related performance PTFs.

Please refer first to

- the VSE V2 performance PTFs in the Base document
- the APAR/PTFs referred to in this TD document

```
* DY43952    UD49914     VSAM performance PTF
```

This PTF reduces VSAM CPU-time by avoiding SECTVAL SVCs for SD and TD (provided the partition crosses the 16M line) and improves VSAM data compression with ICCF started.
It requires a TD PTF (for APAR DY43919) and is contained in in VSE/ESA 2.1.3.
Please make sure that the performance benefit of this PTF is not reduced by a newer VSAM PTF, i.e. make sure that also VSAM PTF UD50015 is applied.

```
* DY44055  UD50003     Parallel POWER for VSE/ESA 2.2
           UD50004
```

This PTF upgrades to POWER 6.1.1 which allows to run POWER tasks in parallel mode.

Still the default is POWER running non-parallel, since some preconditions have to be met functionwise with vendor programs.
Refer to this APAR for more information.
Make sure DY44112 with PTF UD50016 is applied, too.

```
* DY44172  UD50112     TD System Enhancements when ICCF started
           UD50115
           UD50118
```

This PTF avoids that whenever ICCF is started in a VSE/ESA system, monitor class MC(4,1) is 'hot' across all VSE partitions (causing additional dispatcher entries).
With this PTF (exclusive to the TD), MC(4,1)s are only hot when an ICCF interactive partition is started and only in this interactive partition, not for the whole VSE.
It is also required for the CA System Adapter enhancements.

This PTF belongs to VSE/ESA 2.2.

VM/VSE Only TD Considerations

PART H.

VM/VSE Only TD Considerations

VM/ESA Multiprocessing and VSE TD

VM/ESA MP Features

Ù VM/ESA can provide

„ Real Multiprocessing

An individual guest logical processor gets exclusive use of a physical processor (selected by VM)

(V=R|F guests only)

„ Virtual Multiprocessing

Guest 'can see' more processors than actually available

Even on a Uni-processor

Í Defining more virtual than real processors results in poor guest performance
(just recommended for testing purposes)

Í No performance reason to define >1 logical VSE processors under VM/ESA on a UNI

Ù Some VM specific definitions

„ Master and Alternate (Real) Processors

Master processor is one of the real processors, where certain VM/CP work must run (Mostly the IPLed processor, this is one method, VM uses to serialize work).
The Master Processor cannot be dedicated to any guest

Alternate processor is any other real processor

„ Base (Virtual) Processors

Base processor is that virtual processor of a guest, to which VM/CP associates total guest resources in the virtual machine definition block.
This is used only by CP internally

Guest Definitions

Guest Definitions

„ Directory for Guest:

```
    MACHINE ESA <max_no_of_virt._proc.>
```

- <max_no_of_virt._proc.>
  If omitted, number is given by the number of the CPU directory control statements

„ Directory control statement or CP DEFINE cmd:

```
    CPU <cpuaddr>  NODEDICATE|DEDICATE
          ...
```

- cpuaddr = virtual address, e.g. 00..05, at most <max_no_of_virt._proc.> CPUs

- NODEDICATE|DEDICATE specifies whether this virtual processor is to be dedicated to a physical processor (selected by VM).
  Default depends on guest type and OPTION statement

- If V=R and VM/ESA on real MP (and automatic dedication enabled) VM/CP dedicates 1 processor by default

- NODEDICATE is used in general, DEDICATE gives performance benefits (details below)

The CPU statements in the directory are 'static', the CP DEFINE commands are 'dynamic' (i.e. can be issued when the guest is up).

„ Attach/Detach of virtual processors

```
    DEFINE CPU <cpuaddr>
    DETACH CPU <cpuaddr>
```

- Attaches/detaches a virtual processor from your virtual guest configuration

All definitions (except DEFINE) cannot be reset without a new guest IPL

Refer to 'VM/ESA CP Command Reference'

Specific conditions must apply to keep/have VM IOASSIST active

Importance of Low VM Overhead

Why is a low VM overhead important for TD?

Technical Background

VSE non-parallel code is key-0 code (often supervisor code) which is often intercepted by CP

Í Non-parallel code is 'enlarged' by the CP overhead (T/V ratio)
This is also true for parallel code, but ...
this can be compensated by adding more processors

„ Native VSE:

The non-parallel utilization is

```
    NPU_native  = NPS_native x (total sum of CPU utilizations)
```

with NPS_native as the (native) Non-Parallel Share.

„ Under VM:

The effective NPS is

```
        NPS_eff  =  NPS_guest x TV_ratio
```

Actually, the NPS shown by QUERY TD in case of VM guest (NPS_guest) is only minimally bigger than in case of native (NPS_native)

Í The lower/better the T/V ratio is, the higher is the maximum TD throughput under VM:

Example

	Non-Parallel Shares NPS	Max# fully expl. proc. nMP = 0.9 / NPS
Native	NPS_native = 0.35	nMP = 2.6 (native)

Under VM	NPS_guest = 0.36
	NPS_eff = NPS x TV_ratio
		= 1.9 (T/V=1.3)
		nMP = 2.1 (T/V=1.2)
		= 2.4 (T/V=1.1)

Reduce VM Overhead

Reduce VM CP overhead as far as possible

„ Refer to VM/VSE performance documentation, e.g.

- VM/ESA 2.1.0 Performance SC24-5782
- 'IBM VSE/ESA Guest Performance Considerations'

„ Most preferrably use V=R/F guests with DEDicated DASDs

Benefits from I/O passthru/assist and VM CCW translation bypass.
Even w/o dedicated DASDs, still SIGP Interpretation Assist helps

„ Especially check that IOASSIST is really active

Refer to 'IOASSIST' below

„ Dedicate CPUs if possible

a) All started VSE CPUs can be dedicated

This is the best case, all processors have same speed (seen by VSE)

b) Not all started VSE CPUs can be dedicated

May be your workload already benefits even if not all guest processors can run on a dedicated engine.
No general statement possible so far. Individual trial required.

Refer to 'Dedication of Processors' below

Reduce VM Overhead ...

ASSIST Aspects

„ **VM IOASSISTs are only active, if ALL logical processors currently defined via CPU <cpuaddr> of a guest are started via SYSDEF TD,START=..**

```
Check IOASSIST status via QUERY IOASSIST in VM.

SIE assist include IOASSIST, beneficial for all V=R/F guests with
DEDicated DASD devices

-> A new TD function in VSE/ESA 2.3 to QUIESCE a processor 'STOPQ'
```

„ **SIGP Interpretation Assist**

Important for performance of VM preferred guests or LPARs for n-ways

```
Part of SIE assist, avoids interception of SIGPs.
Standard on all 9672 Enterprise Servers and newer ES/9000s
```

í **Avoid that a CPU defined for VSE under VM is not started**

```
Add via CP DEFINE CPU only those virtual processors, you
immediately start via SYSDEF TD

Unfortunately a CP DETACH CPU is not possible w/o a VSE re-IPL to
stay in IOASSIST.
```

Relative/absolute VM SHAREs

```
If a VSE TD system runs in competition with other VM tasks
(e.g. CMS, or VSE-test) ...
```

„ **Increase VM SHAREs for the VSE TD guest when defining addt'l logical processors**

```
VM SHAREs of a guest are divided amongst all currently defined
guest virtual processors, independent of their state.
```

Reduce VM Overhead ...

VSE logical processors in stopped state

„ **Defining virtual processors w/o using them**

Causes more CPU-time overhead

Lowers the effective SHARE value of a guest

Causes loss of VM IOASSIST

Dedication of Processors (more details)

„ **Dedication of a physical processor**

```
- means exclusive use by 1 (specific) VSE (logical) processor

- excludes other VM tasks (e.g. CMS, VSEs) from using this
  physical processor

- likewise applies to standard VSE dispatcher

- applies to any type of guest (V=R/F/V)

- may not be reasonable on a dyadic processor, if other
  VM tasks exist (see below)

- is another VM means to reserve processing power

- reduces total CP overhead for VSE guests
```

„ **Utilization of a DEDicated processor**

```
Both VMPRF and the IND command show 100% utilization for any
DEDicated processor
```

í **Use VSE to determine actual utilization of a DEDicated processor**

Dedication of Processors

Dedication of processors (cont'd)

„ **Imbalance of processor speeds imposed by CP or other VM tasks**

```
Since VM/CP runs on the VM Master Processor, this processor 0
seems to have lower speed for a VM MP guest. But, VM tries to put
lesss load onto the VM Master Processor.

MVS MP experiences revealed that it is/was of benefit for MVS to
have 'equal speed processors' (spin aspects).
Nevertheless also in such cases dedication of processors may be
beneficial.
```

„ **When/How to use DEDicated processors?**

```
Since VSE TD has no processor affinities, there would be no means
to preferrably select the DEDicated processor for VSE work
(in order to hurt other lower priority VM tasks less).
```

í **You may UNDEDicate the 2nd processor (processor 1) on 2-ways**

```
via  UNDED VSEmach CPU  ALL|cpuaddr

Dedication only reasonable if those processors not needed for
other VM tasks
```

í **A mix of DEDicated and non-DEDicated processors for a single guest has to be evaluated on an individual customer base**

```
You may try it for your environment,
no general rules can be given
```

í **DEDicate all VSE processors if you have enough real processors**

VSE/ESA 2.3 TD Enhancements

VSE/ESA 2.3 TD Enhancements

„ **New Supervisor Services for Vendors**

```
With the new TD level, additional performance optimized services
for vendors have been provided. They help in order to save
non-parallel CPU-time and thus to reduce the Non-Parallel Share
NPS.
```

„ **Quiesce CPU**

Problem

```
Dependent on the workload it may be necessary or beneficial to
temporarily stop an engine (CPU) in order to avoid the overhead
of an additional CPU that can't be exploited or is not required
(this may apply e.g. during off-shift).

However, VM/ESA V=R guest environments with any 'not started
(stopped)' CPU will have no I/O assist for dedicated devices. So
the VM overhead may increase and not allow to benefit from a
stopped processor.
```

Solution

```
A CPU can be quiesced via a new command 'STOPQ'.

Such a CPU will no longer participate in processing the workload.
The overhead of the CPU, that is not required, can be avoided, and
the VM/ESA guest continues to run with I/O assist.
```

Performance Results

```
Refer to next foil
```

VSE/ESA 2.3 TD Enhancements ...

QUIESCEing a Processor under VM

Ù **Background**

- A STOPped (INACTIVE) processor of a preferred VM guest (V=R/F) causes total loss of VM IO assists

  -> potentially significant increase of total CPU-time in case of DEDicated devices

Ù **VSE/ESA 2.3 TD enhancement: QUIESCE = STOPQ**

TD allows to STOP a processor w/o losing IO assist
(required native CPU-utilization on QUIESCEd processor: <<1%)

```
                                  PACEX16 I/O intensive batch
   A  Rel. Ext. Thruput ETRR      VSE/ESA 2.3 under VM/ESA 2.1.0
   |                              V=R and DEDicated devices
   |  +(1/rel.CPU-time)(ITRR)
   |
   |      1.0
1.0-      ----       0.95                   0.88
   |     |(1.0)      ----                   ----
 .8-     |    |     |(.94)      0.68       |    |
   |     |    |     |    |      ----       |    |
 .6-     |    |     |    |     |(.68)      |(.66)
   |     |    |     |    |     |    |      |    |
 .4-     |VSE |     |2+  |     |2+  |      |VSE |
   |     |2-way     |1Quie <---|1Stop      |3-way
 .2-     |    |     |    |     |    |      |    |
   |     |    |     |    |     |    |      |    |
   |_____|....|_____|....|_____|....|______|....|__________

A+Q+S=   2+0+0      2+1+0      2+0+1       3+0+0   VSE logical
                                                   processors

Tot.CPU   164%       162%       165%        220%
```

Í **To QUIESCE (STOPQ) a processor instead of STOP gives higher throughput AND lower CPU-time**

VSE TD guest 'stays in IO assist'

Delta and benefit is smaller
- if workload less I/O intensive
- if not all DASDs DEDicated

VM/VSE TD on an MP

Multiple VSE under VM vs Single TD Guest

Ù **Reminder**

Running TD instead of 'old' dispatcher

- costs some CPU-time
- allows

  better balancing of static and dynamic partitions

  a single VSE to exploit more than 1 physical processor

  to get info via QUERY TD (Non-Parallel share)

  to exploit forthcoming balancing enhancements

Ù **Consolidate VSE guests if**

possible function-wise
e.g. test should remain separate from production
workload balancing can be done by VSE/ESA
e.g. day batch throughput not jeopardized by high Online load

TD in VSE/ESA 2.2 allows better balancing
extensive data sharing and/or
frequent communication
is required between guests (e.g. via IUCV)

Í **Consolidation of guests**
may only have marginal benefits regarding total CPU-time
may provide a performance improvement by less data sharing
is more often possible by TD

VM/VSE TD on an MP ...

Multiple VSE under VM vs Single TD Guest (cont'd)

Ù **Use a single VSE with Turbo Dispatcher if**

a single VSE needs more than a single processor's power

or

you need the enhanced partition balancing, maybe with the Relative Shares (VSE/ESA 2.2) to balance batch with CICS partitions

or

you need info on the Non-Parallel share or want to test whether your programs would run

Í **These are the same reasons which apply natively**

but, under VM, the question of consolidation is on top

Ù **If none of all applies, use the 'old' dispatcher**

Run TD only temporarily, to get info on 'your NP/TOT ratio'

VM/VSE TD Example on a Dyadic

VM/ESA and a single VSE TD Guest on a Dyadic

```
         -----------         -----------
        |           |       |           |
        |   CPU0    |       |   CPU1    |
        |           |       |           |
         -----------         -----------

      VM Master processor    VM Alternate processor
      - req'd for certain    - may be dedicated to
        CP work                a guest processor
                               (not reasonable here)
```

Total load consists of:
- 1 VSE guest (TD) for production requiring >1 processors for its total load
- CMS work (optional)
- other VSE guests (optional)

Definition Steps:

1. Define, if possible, the TD guest as preferred guest

- gives the (V=R/F) preferred guest benefits for best ITR: I/O passtru and VM CCW translation bypass (DEDicated devices)
- same as for standard dispatcher environments

2. Define 2 processors (only) for the TD guest

- 1 is not sufficient for this VSE guest
- 2 is beneficial if
  - more than 1 partition contribute to total VSE load
  - the total VSE load would be more than say 70% of a single processor (as peak hour average)
  - this production guest gets enough preference (share) by VM dispatching
- >2 results in poorer guest performance

3. Decide on dedication of an alternate VM processor to a TD guest logical processor

Usually makes only sense on >2-ways, if
- processors not required for CMSs or other VSE guests
- all processors of the VSE TD guest can be dedicated
(you may try for your environment)

PR/SM LPAR Only Considerations

PART I.

PR/SM LPAR Only Considerations

PR/SM LPAR Multiprocessing and VSE TD

PR/SM LPAR features for n-way operating systems

Ù PR/SM Logical Partitions (LPAR) can provide a

„ **Dedicated LPAR(s)**

A logical partition that has exclusive use of its processors
(its logical processors are mapped to a separate subset of the physical processors)

„ **Shared LPAR(s)**

A logical partition that shares all physical processors (not assigned to any dedicated LPAR) with all other shared LPARs

Naturally, only the maximum number of processors defined for an LPAR can be 'seen' (concurrently used)

Processing weights for shared LPARs are defined and always hold for the total LPAR guest,
independent of the number of defined or currently active processors

LPAR guests stay in SIE passthru, even if not all defined processors are started

Í Any LPAR may consist of multiple processors

Í No processor is shared between a dedicated and a shared LPAR

PR/SM LPAR Multiprocessing and VSE TD ...

PR/SM LPAR Performance for N-ways

Ù LPAR in general gives lower ITR than basic mode

„ Shared LPAR overhead is very similar to overhead of preferred VM guests

„ Dedicated LPAR overhead is smaller than for Shared LPARs

Defining more logical processors than required gives higher overhead (like under VM)

When a large single-image processing is NOT required ...

Ù LPAR overhead is reduced or even compensated when each partition has fewer logical processors assigned than there are physical processors
(especially for dedicated LPARs)

Ù MVS examples (IMS on ES/3090-600E)

# and type of LPAR partitions	ITR ratio vs 3090-600E (basic mode)
2 DEDICATED 3-ways 2 SHARED 3-ways	110% * 102%
* Both LPARs had 84% of a 3090-300E	

Í Major factor for LPAR performance:

```
 # logical processors
-----------------------
 # physical processors
```

This ratio should be as low as possible

For more info, refer to 'PR/SM Planning Guide', GA22-7123-13, GA22-7236-00 (for G3 and Multiprise 200)

Appendix: Why now?

PART J.

Appendix: Why now?

The following is an article (courtesy of Jerry Johnston, IBM) for more background information

Why VSE MP support now?

Why VSE MP support now?

Multi-processing and VSE. Those are words many of us never thought we'd see together in a sentence. For years, one of most fundamental and enduring 'principles' of S/370/390 was that MVS and VM covered MPs, while VSE limited itself to small and intermediate uni-processors. The reason was perfectly logical - right up until technology changed the ground rules and it wasn't logical anymore.

MPs first came into common use in the early 70s. System/370 158 and 168 offered MP models. They were super, high end systems of little interest to most VSE customers. In the late 80s, 3090 systems pushed MP models to even higher levels of performance while most VSE users were content with the performance of IBM 9370 or 4381 uni-processor models. Some larger VSE customers took advantage of MP support in VM, but there was no broad-based requirement for native VSE support of MP models. For most VSE customers, there was always a bigger uni-processor available. The world was simple.

Things began to change in the early 90s. The entry ES/9000 was the ES/9221, a small rack-mounted system. The ES/9221 was (and is) an ideal choice for many VSE customers. Until recently, the top ES/9221 was the M 200, a 2-way system. Since there was no VSE support for MP, a VSE customer outgrowing the largest ES/9221 uni-processor was encouraged to go to an ES/9121 uni-processor and skip the more obvious M 200. It was a small anomaly, but it showed a weakness in the VSE 'uni-processor only' strategy. If MP models became common across a broad spectrum of performance, not just the high end, could VSE avoid MP support and still meet customer needs for choice and growth?

Now the change in ground rules is even starker. IBM introduced the System/390 Parallel Enterprise Server (IBM 9672 R) in September 1994. The 9672 R comes in six models - one uni-processor and five MPs. Because of its lower total cost of computing (acquisition cost, power, cooling, space requirements, reliability, growth potential, etc), the IBM 9672 R is an ideal enterprise server and a sizable portion of the VSE population found it appealing.

Equally significant, CMOS and the new parallel technology was clearly the future for S/390. Instead of several unique processor designs, System/390 systems of the future will be based on a common S/390 CMOS microprocessor technology (the same basic technology used to make low cost, commodity microprocessors and memory chips). Today the 9672 R and '211' models of the ES/9221 share a common S/390 CMOS microprocessor. To address a range of performance requirements, S/390 servers will simply add more CMOS microprocessors in parallel.

Why VSE MP support now? ...

The challenge to VSE's uni-processor strategy was clear. Unless we convinced ourselves that each and every VSE customer could be content with one single uni-processor model (the ultimate 'one size fits all' approach) MP support would soon become critical for VSE.

Fortunately VSE was ready for the change. VSE/ESA Version 1 had just completed a massive upgrade. In a span of only 3-4 years, VSE transformed itself from VSE/SP, with all its restrictions and limitations, to VSE/ESA V1.3 with support for 2 GB of real storage, up to about 200 partitions, 31 bit virtual addressing and virtual storage constraint relief, ESA data spaces, and much, much more. The capacity and extendibility of VSE had grown enormously but it still lacked MP support.

In September 1994, along with the System/390 Parallel Enterprise Server, IBM announced VSE/ESA Version 2. Building on VSE/ESA V1, Version 2 added client/server to traditional VSE strengths in cost/effective batch and transaction processing. Version 2 also included something called the 'Turbo Dispatcher'. MP support is coming to VSE and will be generally available in July, 1995.

The Turbo Dispatcher is a unique VSE design. One obvious objective was to support IBM's new n-way systems, exploiting multiple processors in a cost/effective way to improve throughput. Another, more important, objective was a design that minimizes the effect on staff and programs. Early experience says the Turbo Dispatcher meets both objectives.

A dispatcher distributes the jobs and various work units that make up each job to the available hardware resources. Work units are pieces of a job that begin at the instant of dispatching and continue to run up to the point when an interrupt request is posted. The Turbo Dispatcher assigns work units to the next available processing unit (PU). All PUs have 'equal rights'. That is, every PU has access to the shared virtual areas of VSE/ESA (including the supervisor) and every PU may receive interrupt requests from any I/O or other external sources. While a PU is processing a work unit, no other PU can process any work units of the same job. The Turbo Dispatcher does not dedicate a specific PU to a job. Instead, it distributes jobs evenly to all PUs. Thus, while a job cannot run on more than one PU at a time, during the life of any job it will run on all the PUs in the system.

As complicated as it may sound, the design is really quite simple. It works 'natively' or under VM/ESA. The Turbo Dispatcher does not change the system structure (for example, layout of VSE address spaces). That means no changes are required for most user or vendor written programs. In addition, there is no impact on systems administration or operating environment. Again, the most important objective of the Turbo Dispatcher design was to minimize staff and people cost.

Why VSE MP support now? ...

The VSE/ESA V2.1 Turbo Dispatcher runs on any ESA-capable processor. An 'ESA-capable processor' may be a uniprocessor or an 'n-way' model. 'N-way' means any system with two or more processing units (PUs) with shared main memory and channels. The current 9672 R goes up to 6-way (6 processing units), the 9221 goes up to 2-way, and the 9121 goes up to 4-way. All ES/9000 models will be supported by the Turbo Dispatcher. ESA capable processors also include the 4381-9XE models and most later 3090 models.

The Turbo Dispatcher does not support 'Parallel Sysplex' (known as 'coupling'). The latest MVS/ESA supports 'coupled systems'. That is, multiple systems, each of which may be a 'n-way' (where 'n' may be different for each system). MVS/ESA manages the entire complex as a 'single system image'. You can think of MVS as supporting 'm x n-way' (or maybe more accurately: a*n1 + b*n2 + c*n3 +...). VSE doesn't plan to go that far. Thus, although VSE has added MP support, the relative positioning of VSE and MVS remains unchanged. VSE is still positioned for small and intermediate systems. It's just that today's 'small' systems are often more powerful than the 'jumbo' systems of just a few years ago.

Turbo Dispatcher demonstrates that IBM has the good sense to change even the most fundamental 'principles' when technology and customer requirements indicate that what was once logical is no longer so. With the Turbo Dispatcher and the System/390 Parallel Enterprise Server, IBM gives VSE customers the sort of cost/effective capacity and growth opportunities that are needed in the emerging client/server world.

Jerry Johnston

LSPR Results for Turbo Dispatcher

PART K.

LSPR Results for Turbo Dispatcher

This section was updated and moved into the new document 'IBM VSE/ESA Hints for Performance Activities'

For official LSPR results and more info, refer to

- LSPR in Internet URL=http://www.s390.ibm.com/lspr/

EOD

END OF DOCUMENT   Have a nice day