2054 OpenVMS Performance Update

Gregory Jordan

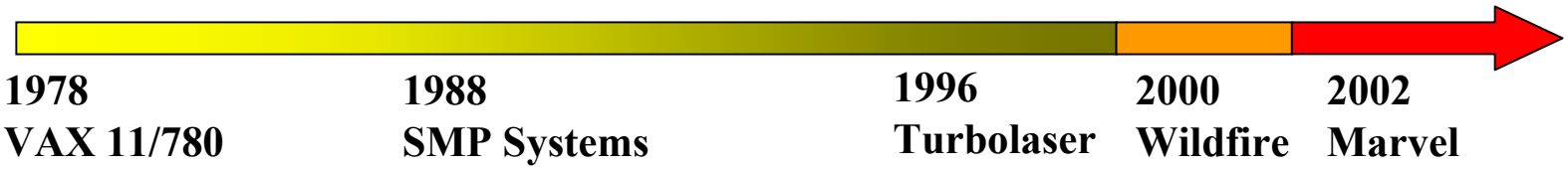
Sr. Member Technical Staff
OpenVMS Engineering

HP WORLD 2003
Solutions and Technology Conference & Expo

Agenda

- Some History
- Recent OpenVMS Performance Work
- Current OpenVMS Performance Work
- Some Performance Data
- Comparing GS160 and GS1280 Systems

Evolution of System Architecture

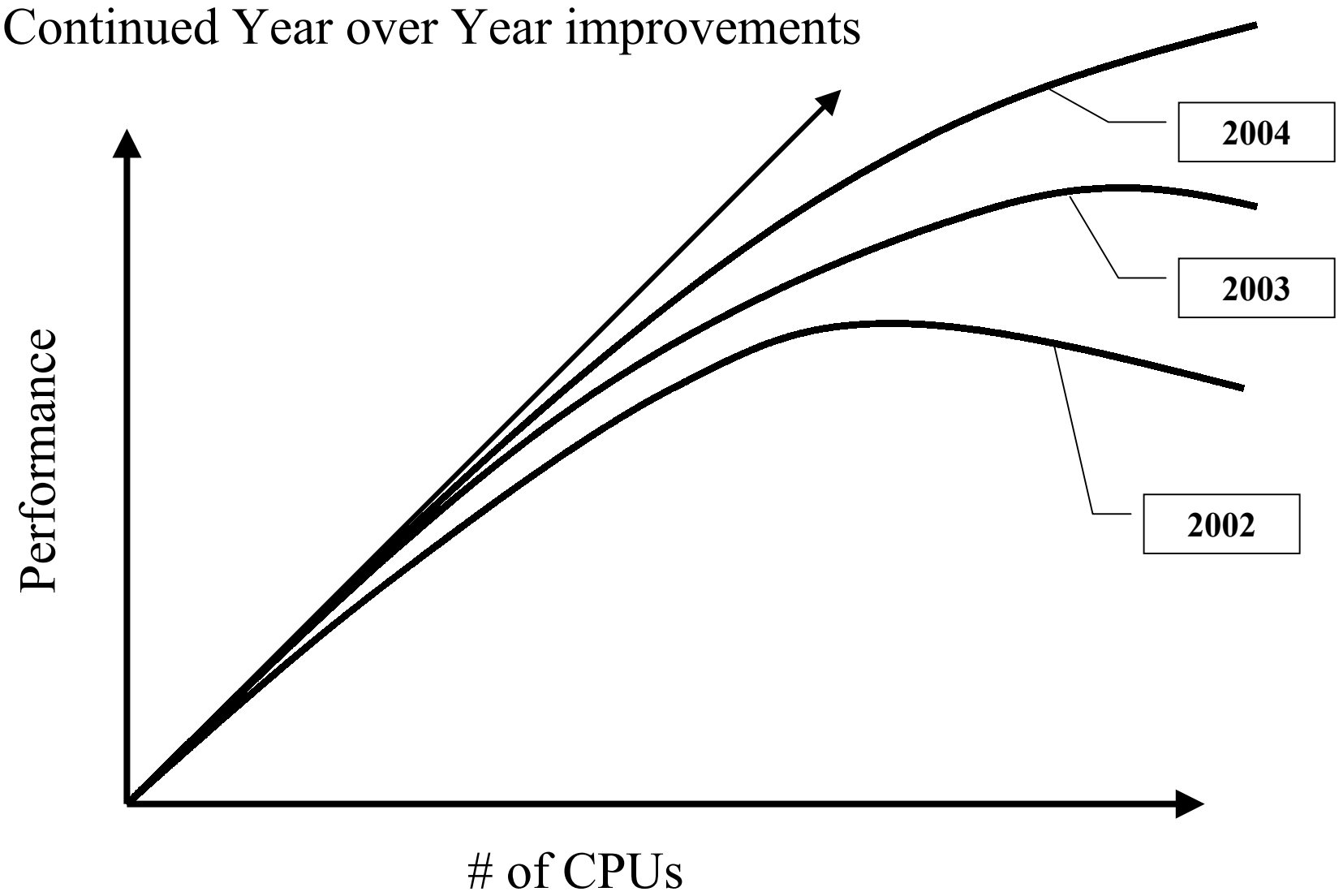


- History
  - Single shared system bus
- Past
  - EV6, Crossbar technology
  - NUMA
- Current
  - EV7, Mesh architecture CPU interconnect on-chip

OpenVMS Performance Focus

- SMP Performance
  - Find and reduce existing spinlock bottlenecks
  - The above work must be driven based on the bottlenecks customers see
- Scaling
  - Find and alleviate issues in the OS that limit scaling
- Single Stream performance
  - We are looking at general performance of the CRTL and various e-commerce applications

OpenVMS Performance Goals



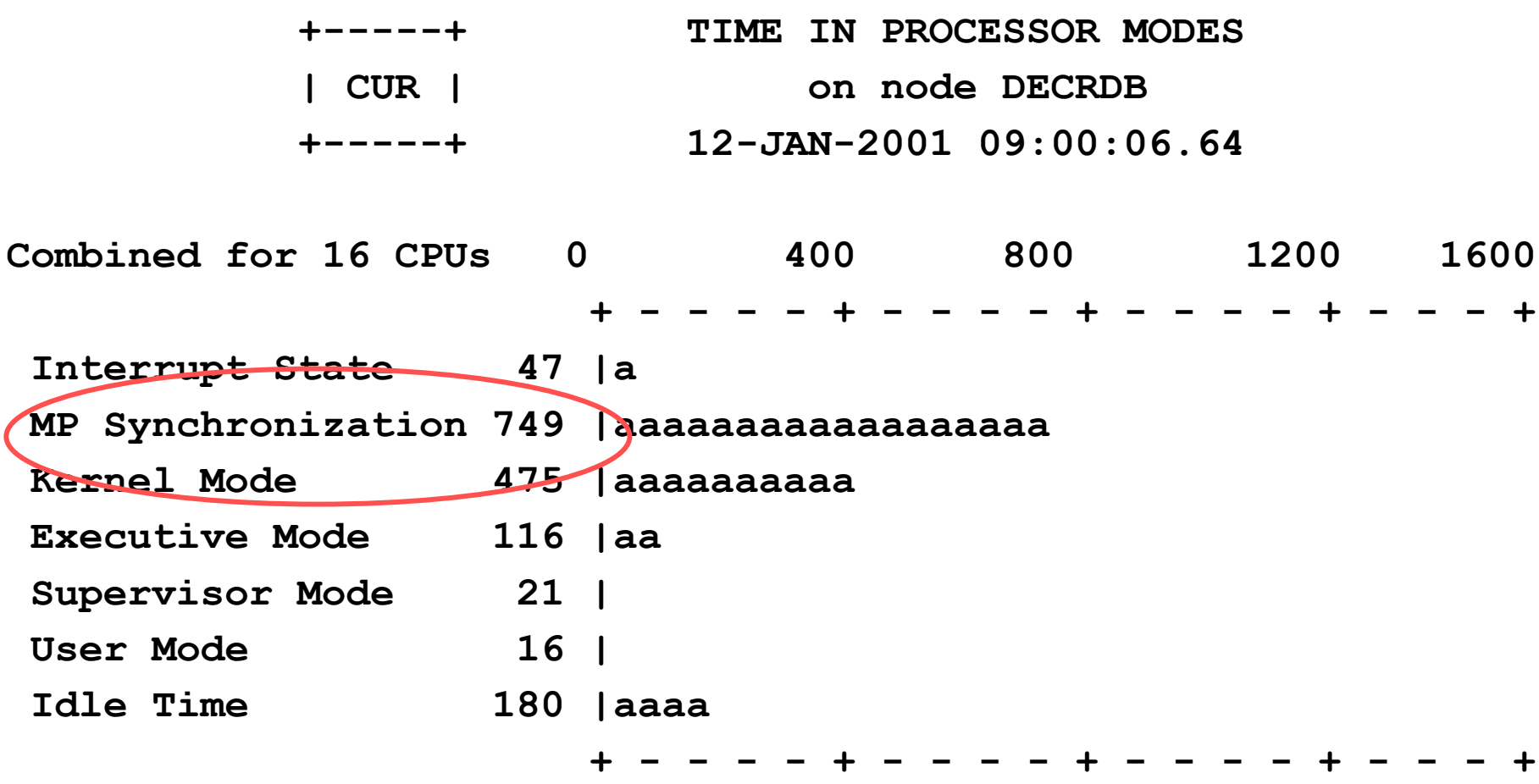
SMP Performance Drivers

- Contention
  - Spinlocks
    - Used for inter-processor synchronization
    - Only 1 CPU can hold - other CPUs “spin” waiting for these locks
  - Lock Manager
    - Application synchronization, various VMS components
- Memory Bandwidth and Latency

The Turning Point

- The GS160/GS320 series systems was a turning point for performance work
  - Most Large systems typically had 8-10 CPUs maybe 12 - SMP scaling was reasonable
  - Now customers were trying to run with 16 or more CPUs
    - The NUMA aspects of these systems exacerbated the SMP scaling issues
- The new GS1280 systems are also a turning point due to a combination of improved processor speed and low overall memory latency

MONITOR MODE



Spinlock Tracing

- Our primary tool when looking at SMP performance is called the spinlock trace tool
- This tool has shipped on OpenVMS (since at least V7.1H1)
- It is run from a privileged account by:
  - $ @sys$examples:spl.com
- The tool provides detailed data of spinlock usage and hold times

Performance Improvements in V7.2-2 and V7.3

- V7.2-2 and V7.3
  - Dedicated-CPU lock manager
  - Process scheduling, idle loop
  - MUTEX without SCHED spinlock
  - SYS$RESCHED (used by DECthreads and Oracle)
  - SYS$GETJPI
  - MailBox driver
- V7.3
  - Fibre fastpath
  - SCSI fastpath

Performance Improvements in V7.3-1

- AST Delivery
- Mailboxes Specific Spinlocks
- RMS Global Buffer Locking
- Reduce IOLOCK8 usage by Fibre/SCSI
- Improved IO Completion for RAMdisk, Mailbox and Shadowing IO
- Reduced Balance Slot size
- Improved Timer Queue Processing
- Distributed Interrupts for Fast Path Drivers
- Various NUMA Changes

Performance Improvements in V7.3-2

- LAN
  - Fastpath LAN drivers
  - Fastpath PEdriver
  - TCPIP
- Scaling changes
  - Remove WSMAX and BALSETCNT restrictions
- XFC
  - Alleviate SMP bottlenecks with very high cache rates
- Miscellaneous Updates

LAN and PE Fastpath

- LAN Drivers
  - Move off of IOLOCK8 to LAN device specific spinlocks
  - Allow device interrupts to CPUs other than the primary
- PEdriver
  - Move off of IOLOCK8 to PE specific spinlocks
  - Allow a specific CPU to be chosen for PEdriver processing

TCPIP Performance
Future Synchronization Mechanisms

- Multiple dynamic spinlocks
  - No more IOLOCK8
- Queue KRP (kernel request packet)
  - Handled by fork thread on non-primary CPU
  - Similar to dedicated lock manager
- Improve concurrency
  - Multiple concurrent network I/O
  - Multiple processes queue TCPIP requests concurrently

Remove WSMAX and BALSETCNT restrictions

- Currently balance slots live in S0S1 space
  - S0S1 is a shared 32 bit space of 2GB in size
  - Balance Slot size is heavily based on WSMAX
  - Some customers today must trade-off large number of resident processes (BALSETCNT) vs. large working set (WSMAX)
- We are breaking balance slots into a balance slot and working set slot
  - The Working Set List is the major part of a Balance Slot
  - The working set slot will exist in S2 space

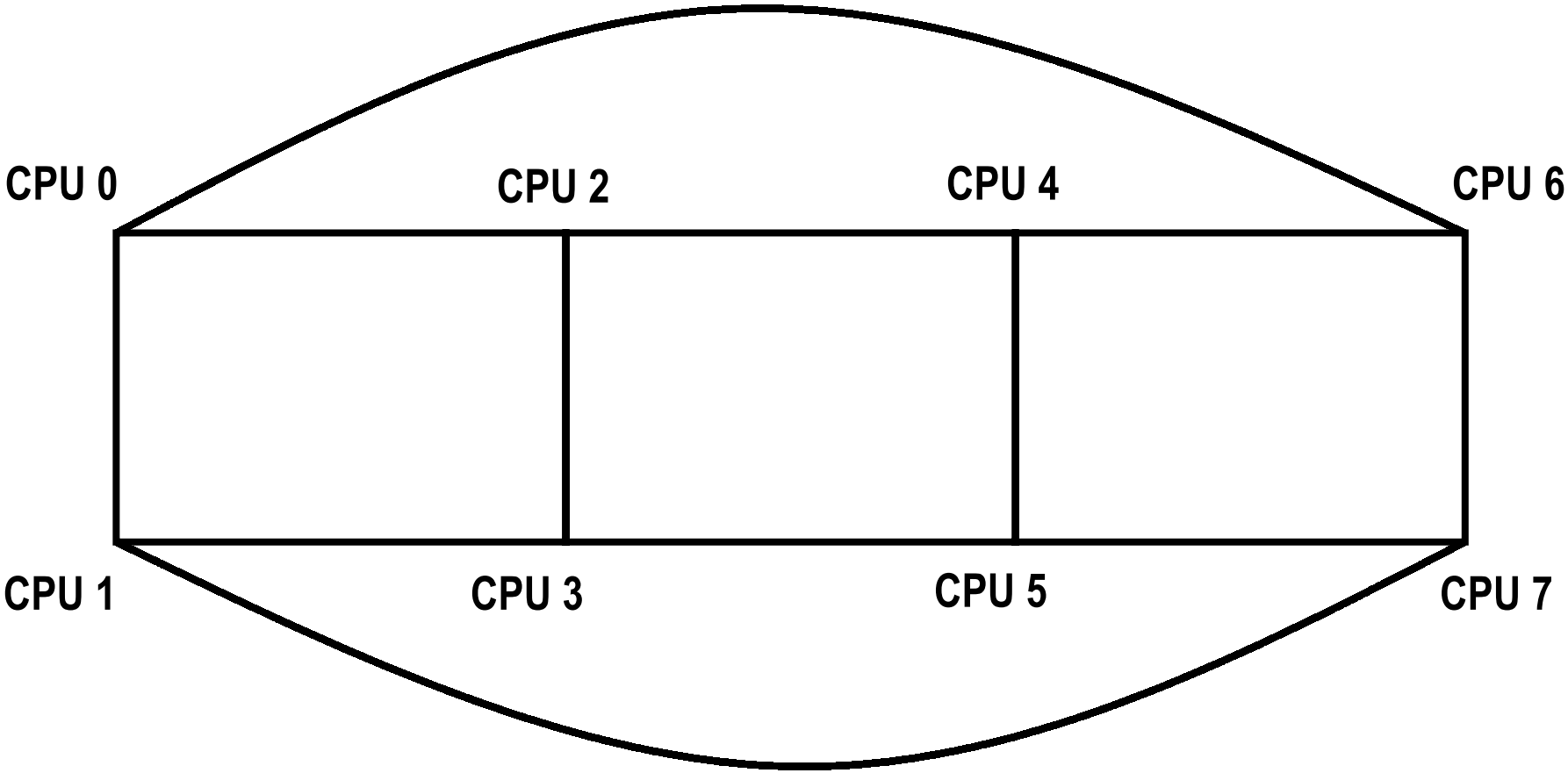
Miscellaneous Updates

- COPY and SEARCH Improvements
  - The IO Buffer size has been increased from from 64k to 127k
  - Results in fewer IOs necessary for copying or searching large files
  - Can have a significant improvement in the elapsed time for SEARCH and COPY operations on large files

The new AlphaServer Systems - GS1280

- System changes from a Hierarchical Switch Architecture to a Mesh Architecture
- The system is still a NUMA system
- EV7 chips have a smaller (1.75MB) but faster cache
- We can support NUMA and RADs, but at a granularity of each CPU and associated memory and IO is a RAD
- All commercial workload testing has shown the system performs similarly with and without RADs
  - We are currently defaulting to turning RADs off

Mesh for an 8p GS1280



16 CPU GS1280 Memory Latency

208	172	136	172
172	136	70	136
208	172	136	172
244	208	172	208

Average 170 ns

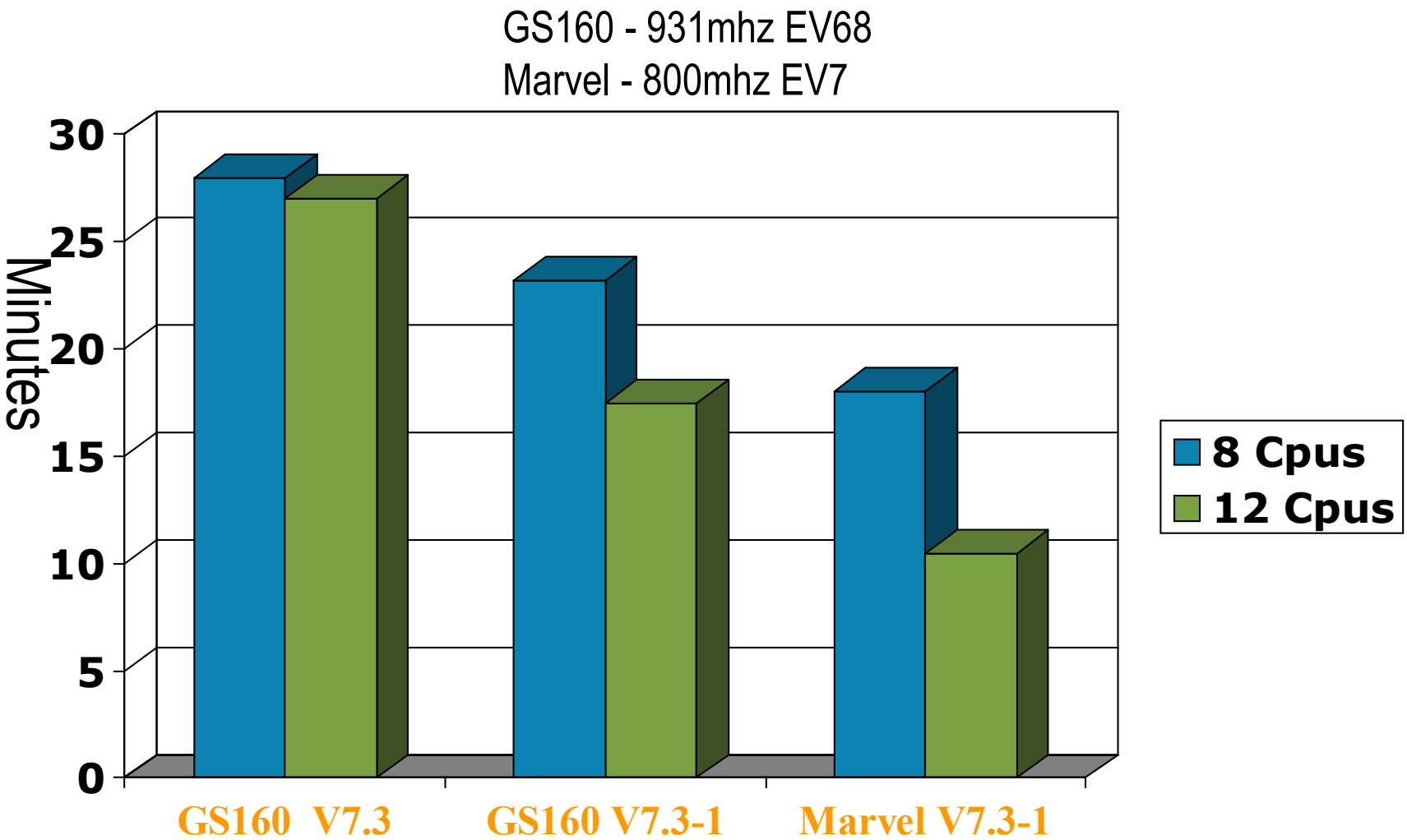
5 CPUs <= 136 ns **6 CPUs <= 172 ns** **5 CPUs <= 244 ns**

EV67 GS320: local latency ~330 ns; remote ~960 ns

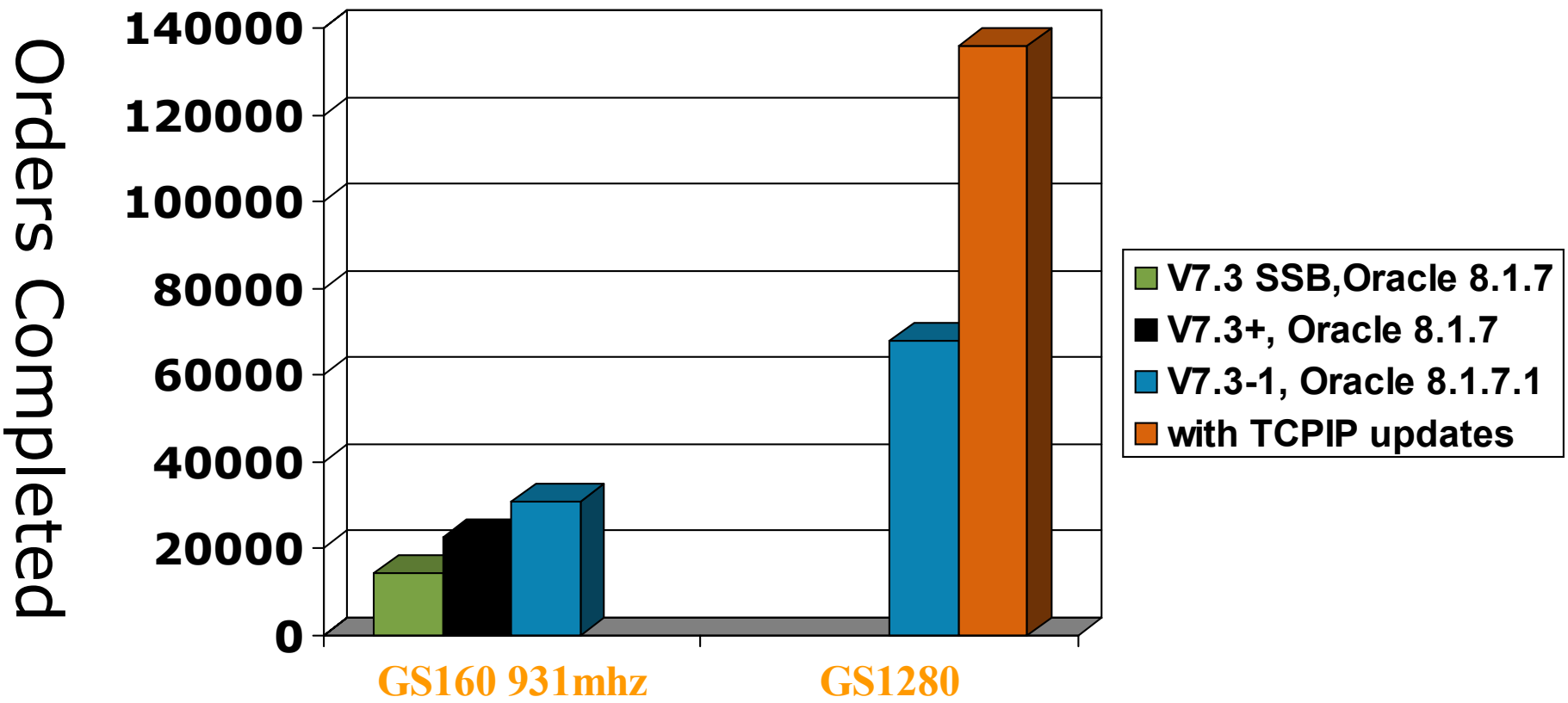
Workload Examples

- Various Workload Examples:
  - Financial End of Day Processing - RMS, RAMdisks
  - Oracle Application/DataBase Sever – TCP/IP requests
  - Real Customer Applications
    - Bank Austria
    - others

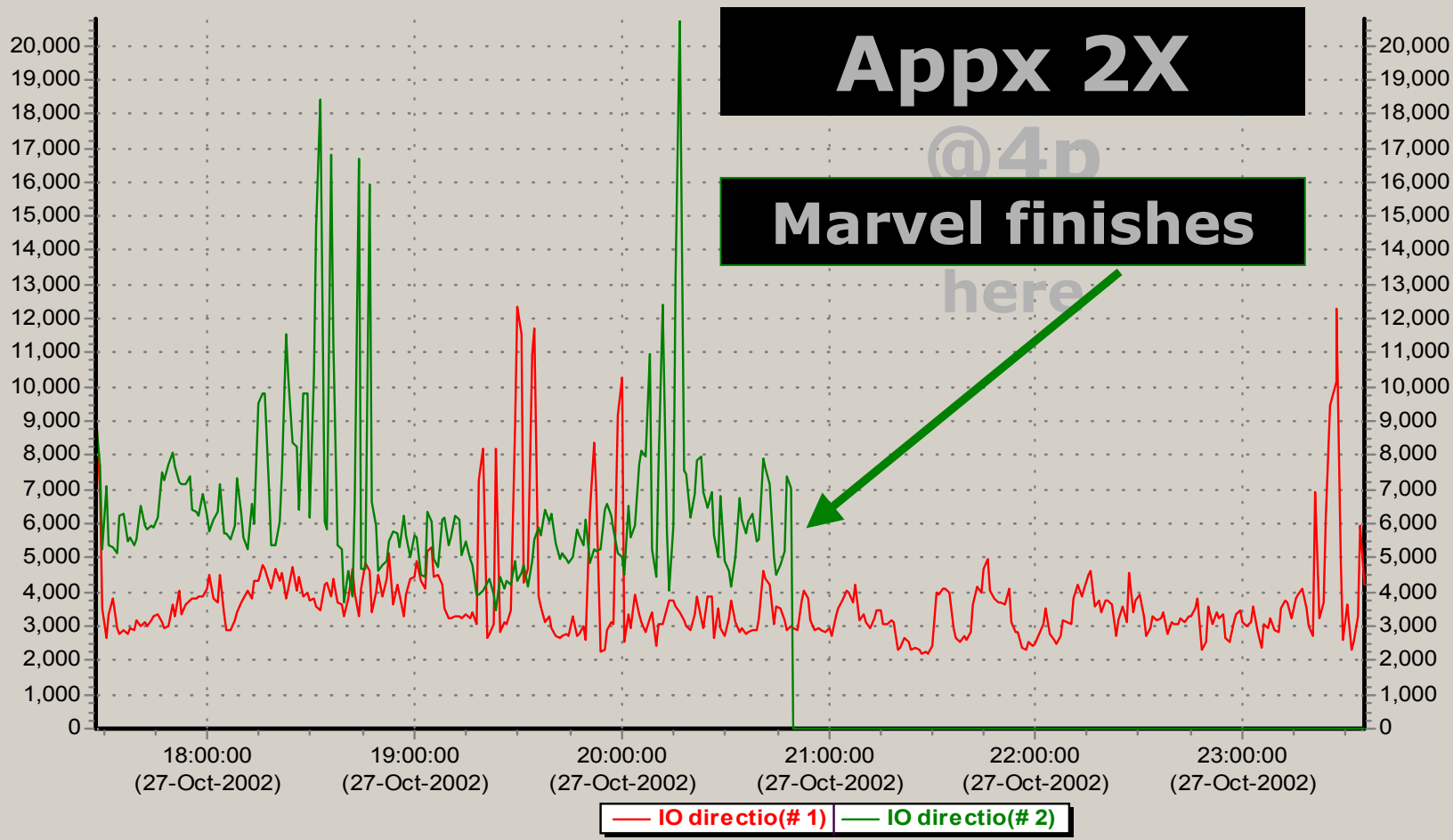
RMS Application with DECram



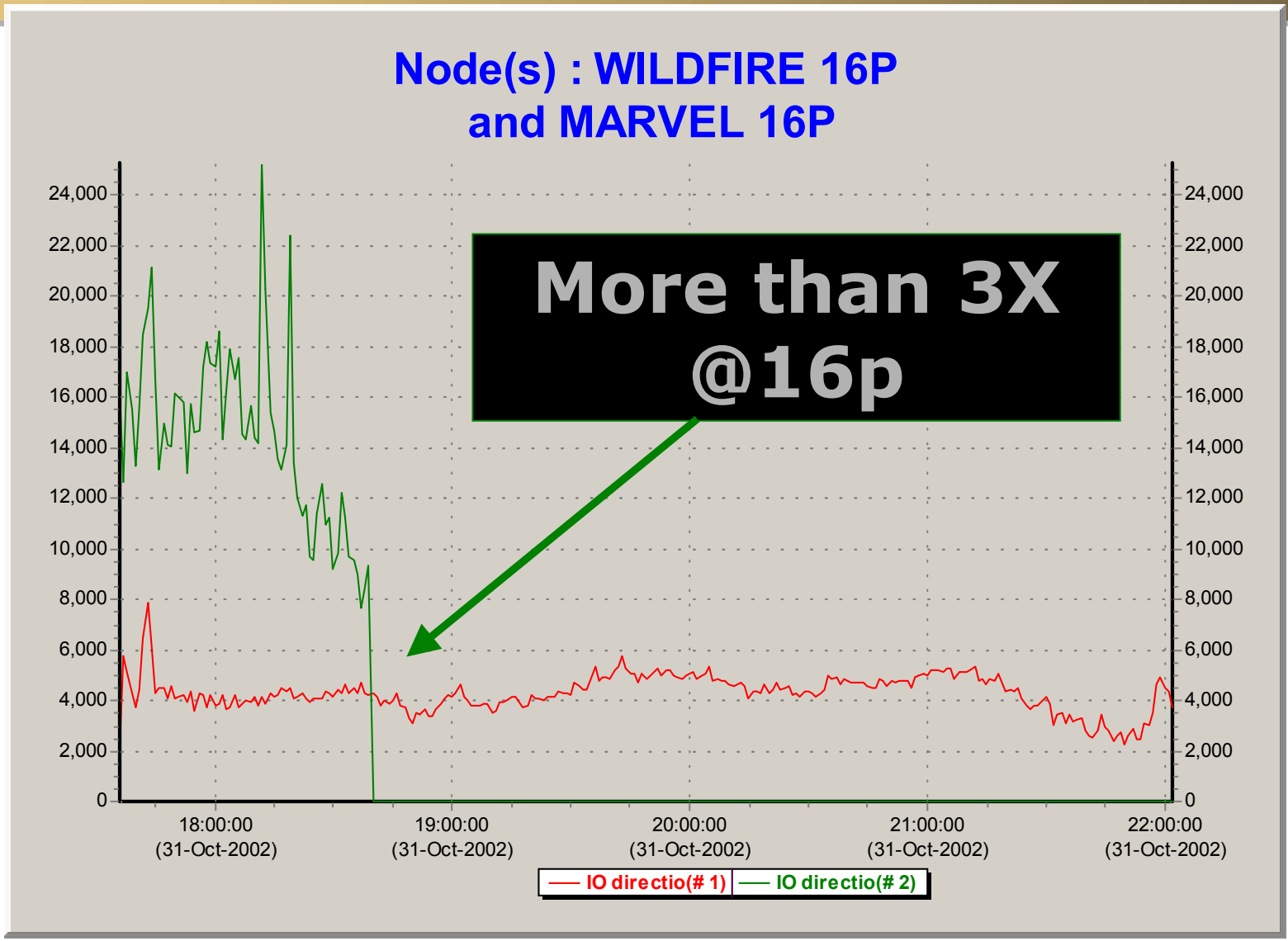
Oracle Application/Database Server



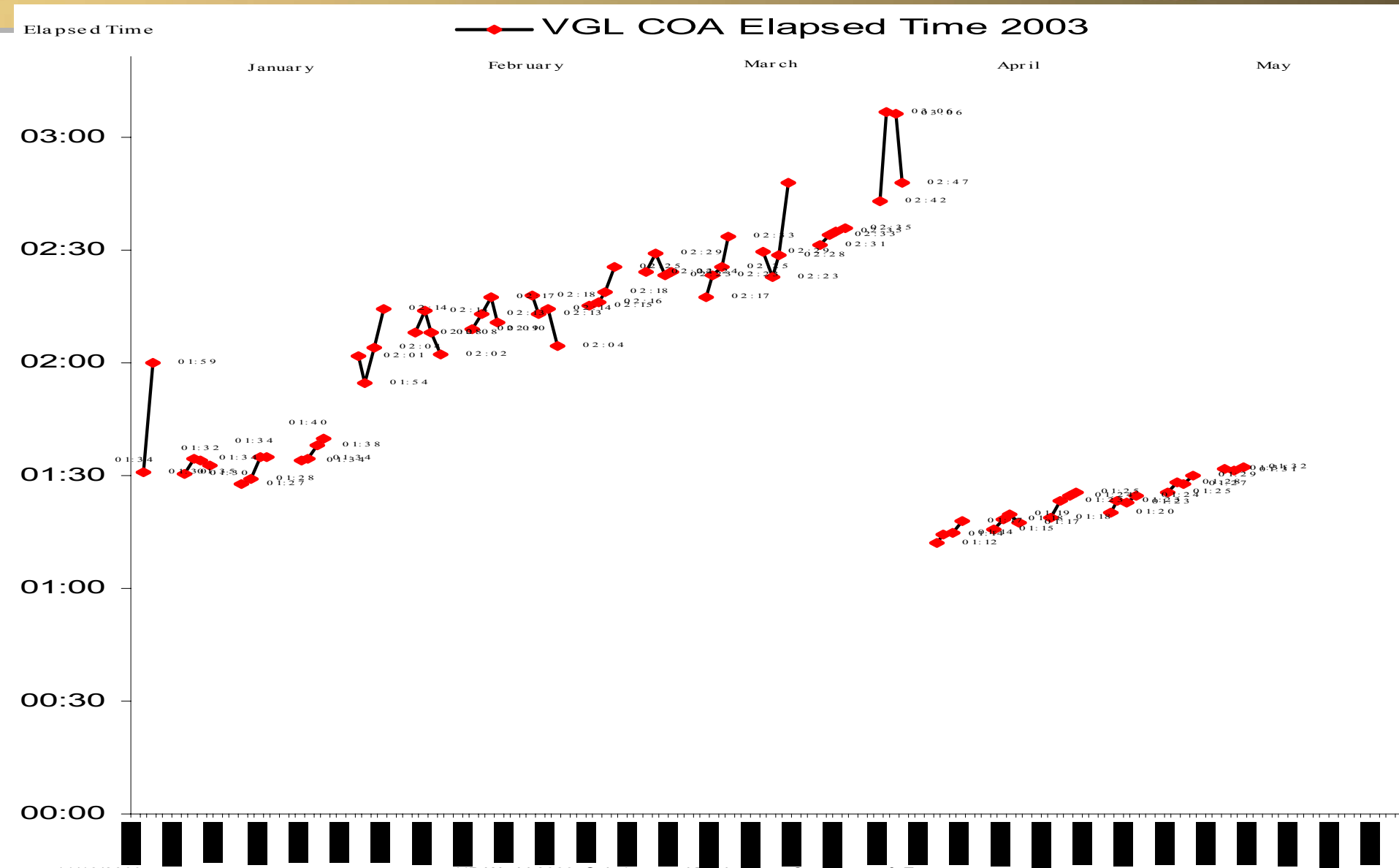
Bank Austria 4P head-to-head test



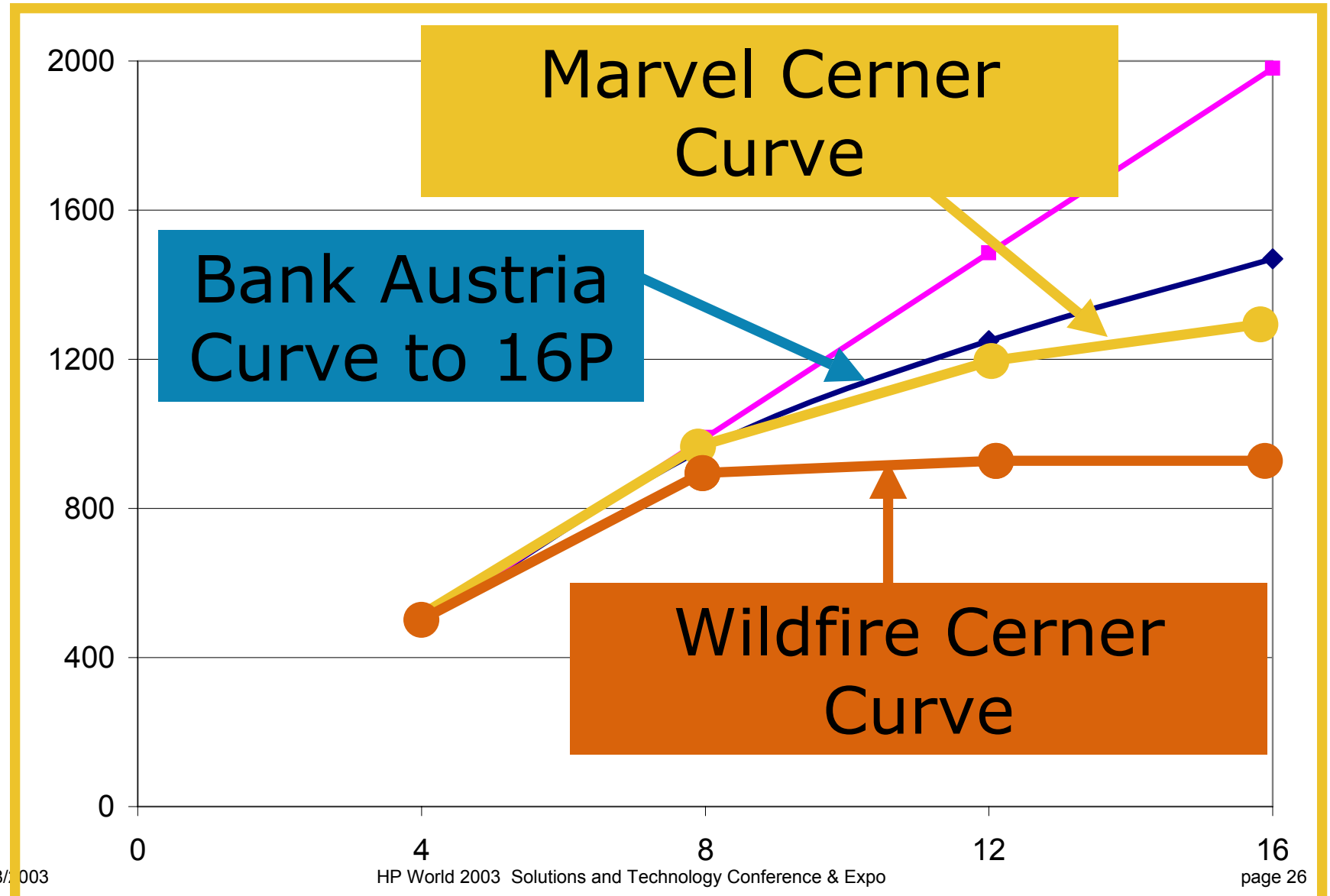
Bank Austria 16P head-to-head test



Bank of Austria
Close of Area

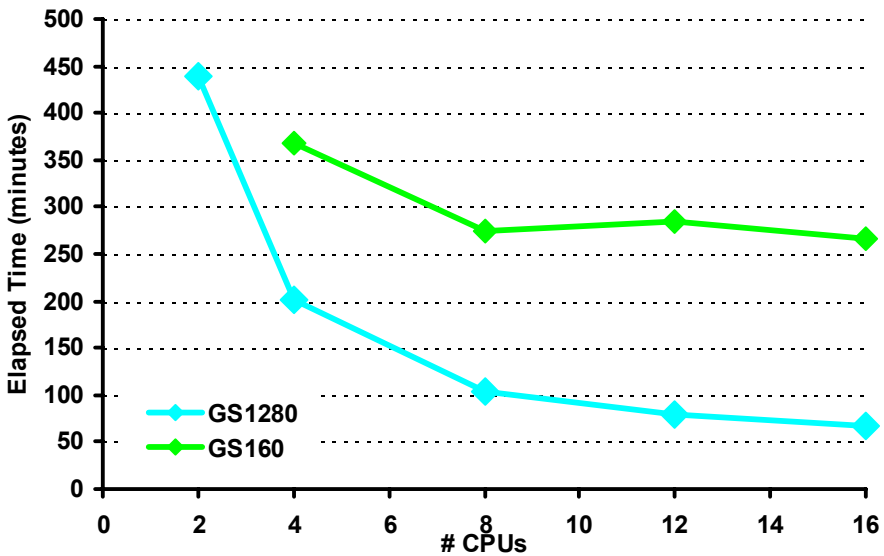


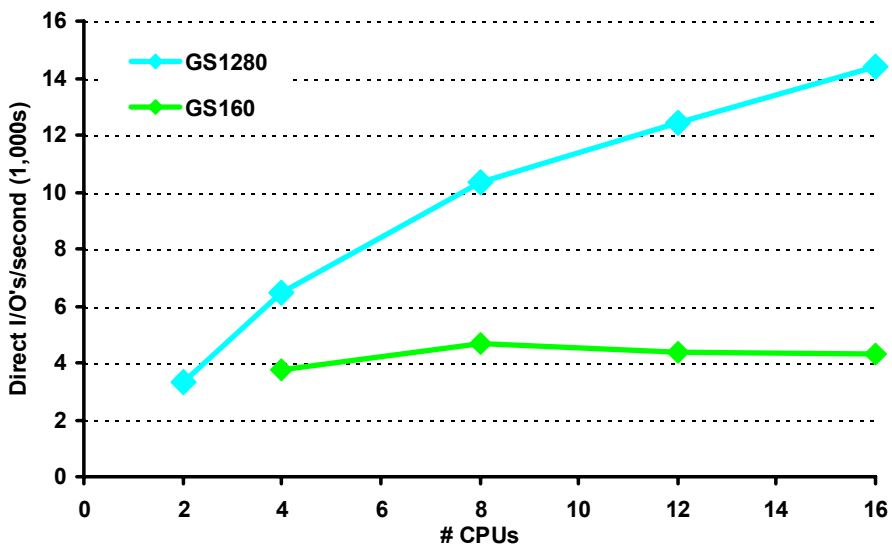
Comparing SMP Scaling Curves



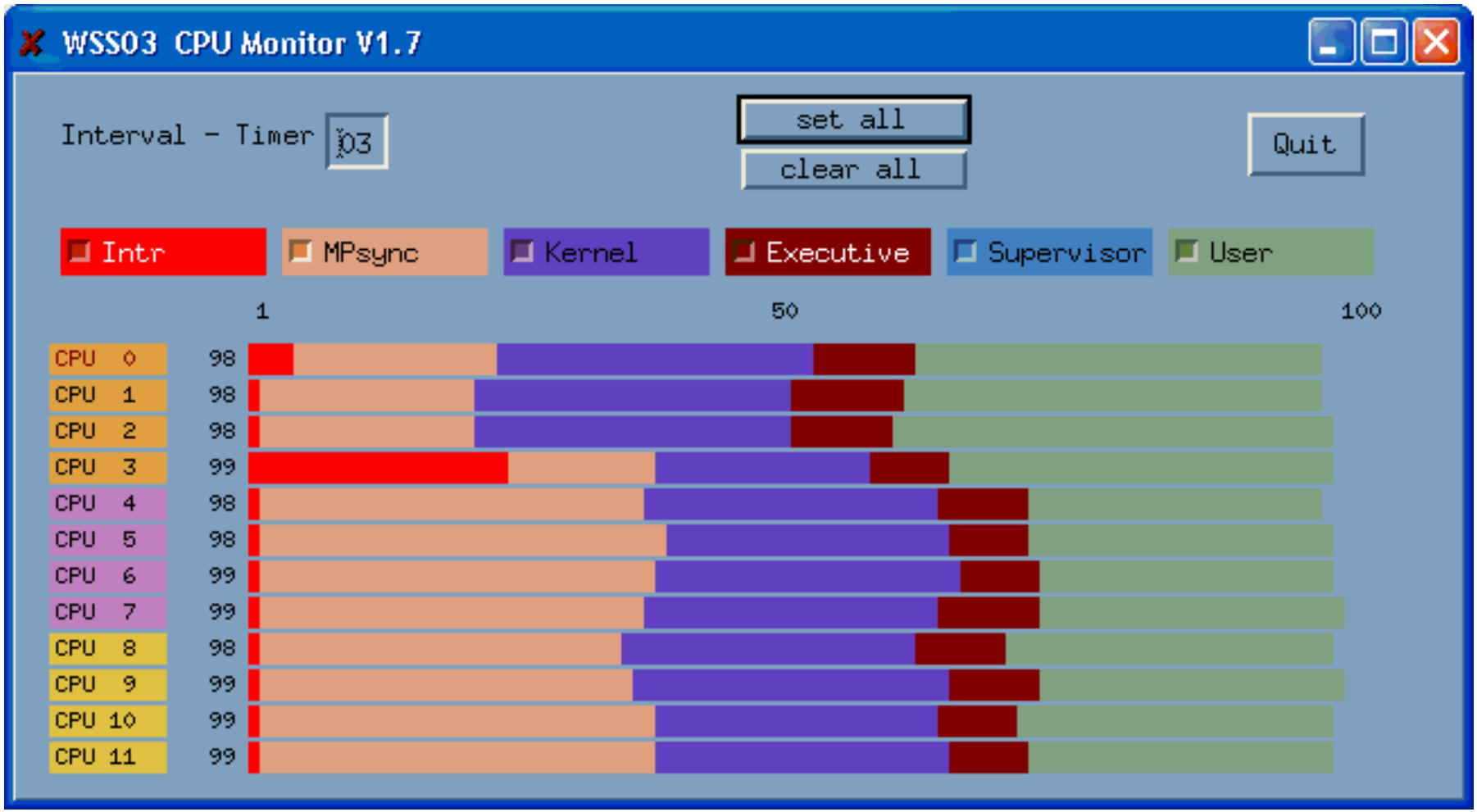
IO Rates

Significant improvements in application performance, application scalability and sustainable I/O rates with new AlphaServer GS1280 systems

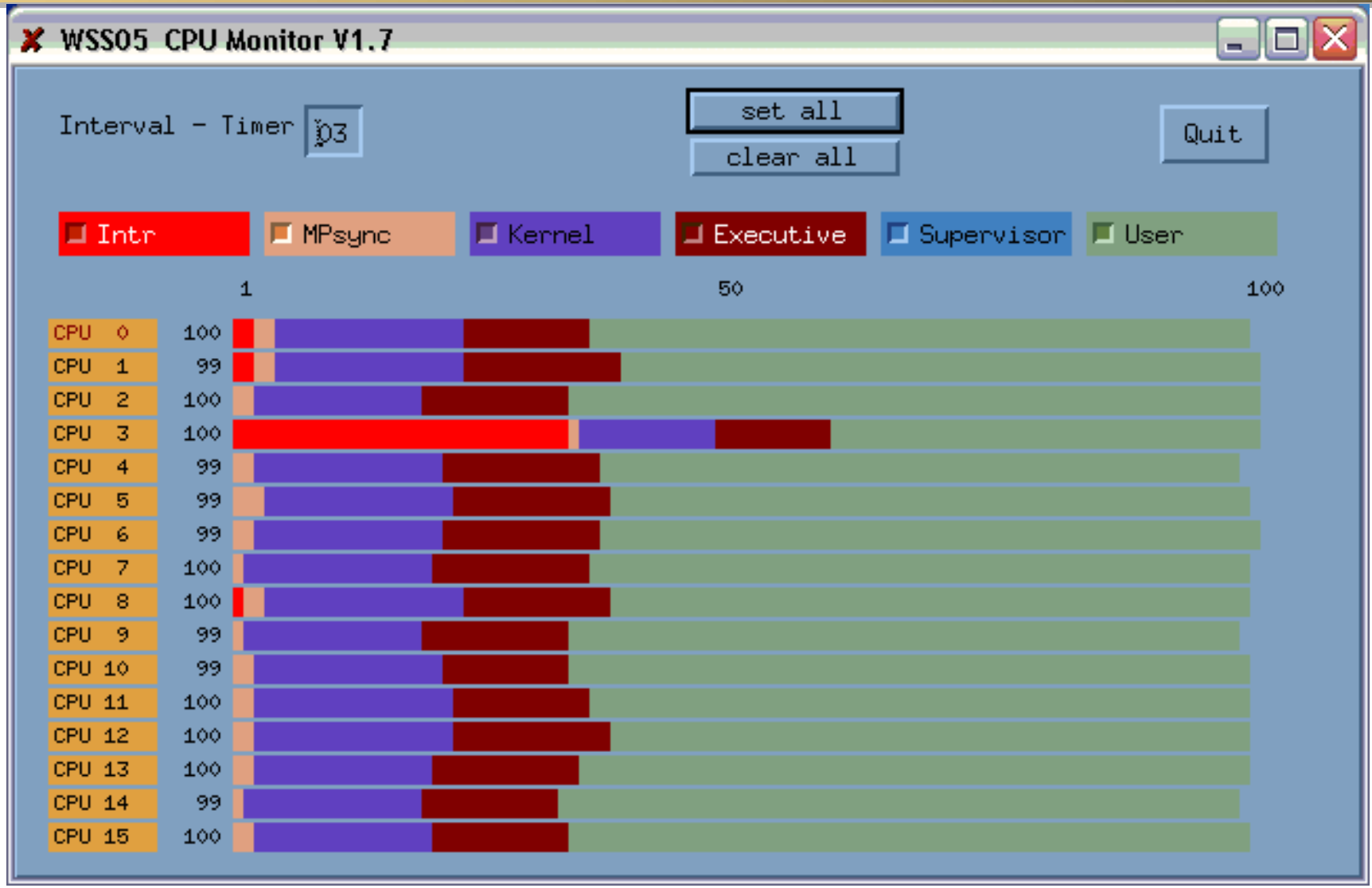




Bank Austria GS160 CPU Modes



Bank Austria
GS1280 CPU Modes



Looking Ahead

- Various Project Under consideration (post V7.3-2) include:
  - SCHED Contention with heavy use of $SETPRI
    - Deferred Priority
  - Additional improvements to COPY/SEARCH
    - Perform multiple IOs in parallel
  - Continued Work in XFC
- We are continually looking at customer workloads for bottlenecks and contention
  - This often leads to a variety of small projects that can have a major impact on a set

Summary

- OpenVMS continues to make performance and scaling enhancements to the operating system
  - Don’t forget that even larger gains are often possible with application changes
- OS Improvements coupled with the latest generation hardware are resulting in substantial performance gains for customer applications
- With a common code base, all performance work applies to both Alpha and Itanium
- Our goals are to scale most commercial workloads to 16 to 32 processors

Interex, Encompass and HP bring you a powerful new HP World.