Enterprise Supercomputers

Ike Nassi
Executive VP and Chief Scientist
SAP Palo Alto

(Joint work with Rainer Brendle, Keith Klemba, and David Reed)

High Performance Transaction Systems Conference, 24 October, 2011

A Personal Reflection

Let me start with a story.

The setting. Early 1982:

Digital Equipment Corporation (DEC) had introduced the VAX (a super-minicomputer) three years earlier

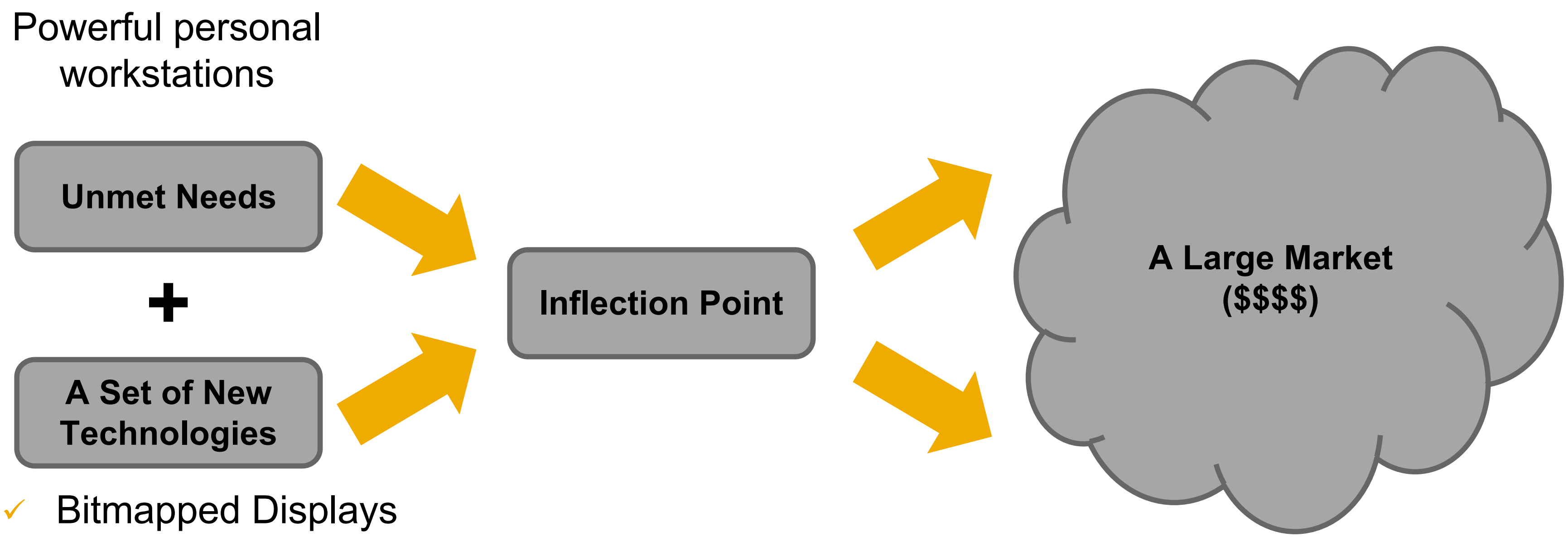
Sales were going well

The VAX was a popular product

And then one day…

The Moral of the Story

We see that…

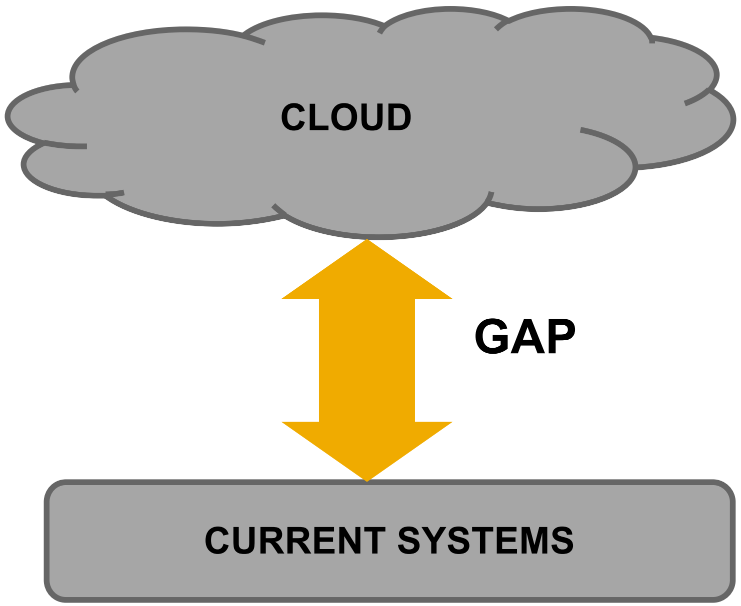


- ✓ Bitmapped Displays
- ✓ 10 MB Ethernet
- ✓ Powerful µprocessors
- ✓ Open source UNIX

The Market for Workstations

Is “the Cloud” all there is?

Emergence of Cloud indicates that we may have been missing something



Current systems represent the work of the last two decades

Emerging Needs

MOBILITY

BIG DATA

NEW UI

ELASTIC

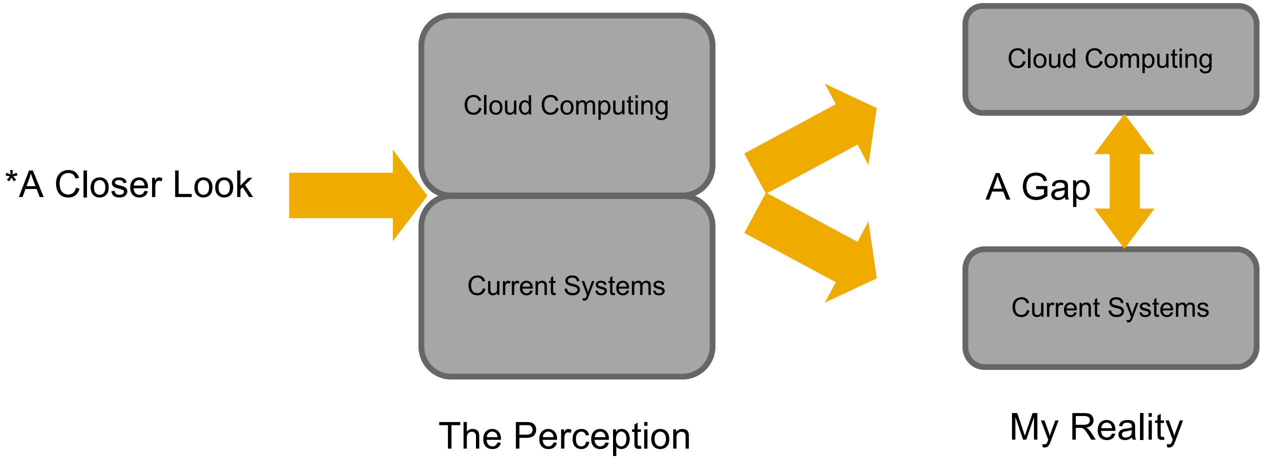
SOCIAL

NEW BUSINESS

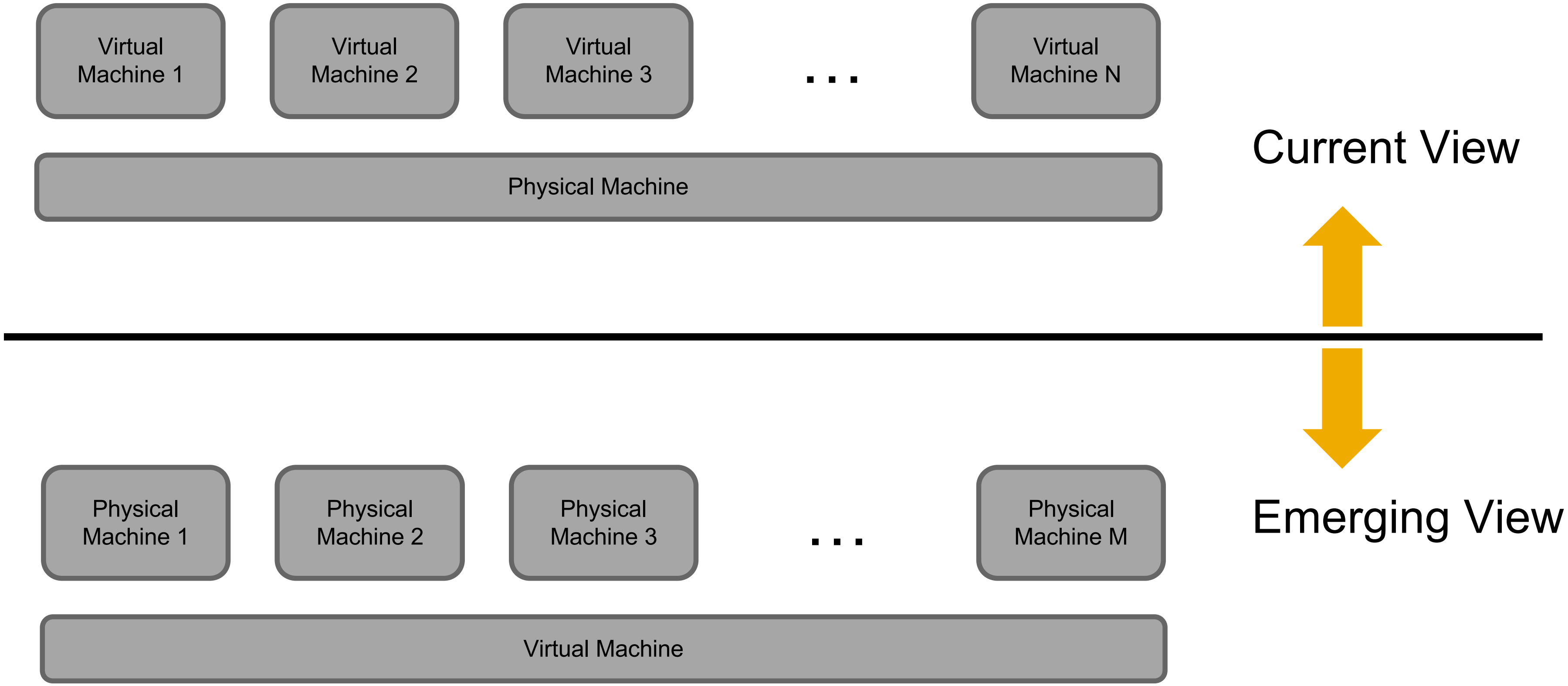
Drivers

Today I'll talk a bit about

- Current commodity server design
- How this creates a computing gap for the enterprise *
- How businesses of tomorrow will function
- And how enterprise supercomputers will support them
- A different layer of cloud



The short story…



And this creates a perceived technology gap

Problems with today's servers:

- Cores and memory scale in lock step
- Access latencies to large memories from modern cores
- Addressing those memories

Current and emerging business needs:

- Low latency decision support and analytics
- "What if?" simulation
- Market/Sentiment analysis
- Pricing and inventory analysis
- In other words, we need "real" real time

This creates a technology conflict:

Cores and memory need to scale more independently rather than in lock step

How can we resolve this conflict?

Reincarnating the “mainframe”

- In-memory computing provides the real time performance we’re looking for
- In-memory computing requires more memory ***per core*** in a symmetric way
- I remember what Dave Cutler said about the VAX:
  - *“There is nothing that helps virtual memory like real memory”*
  - Ike’s corollary: *“There is nothing that helps an in-memory database like real memory”*
- The speed of business should drive the need for more flexible server architectures
- Current server architectures are limiting, and we must change the architecture and do so inexpensively
- And how do we do this?

The Chip

How do we leverage advances in:

- Processor Architecture
- Multiprocessing (Cores and threads)
- NUMA (e.g. QPI, Hypertransport)
- Connecting to the board (sockets)
- Core to memory ratio
  - We have 64 bits of address, but can we reach those addresses?

The Chip

Typical Boards

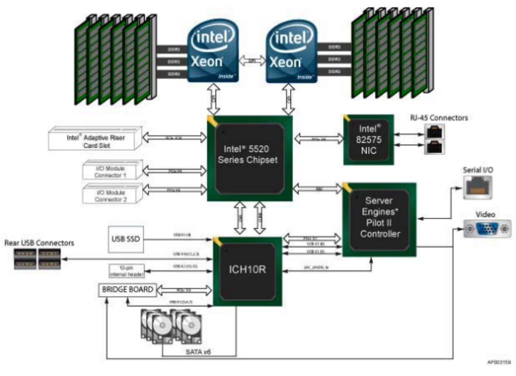
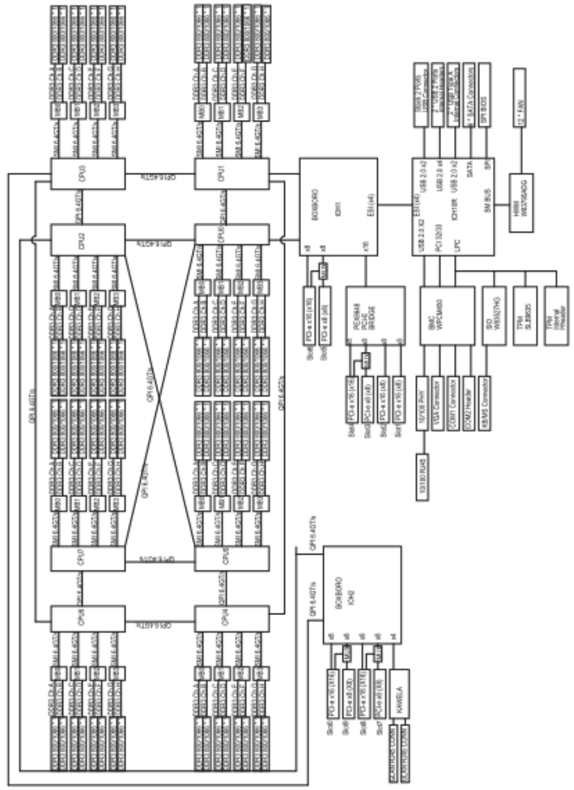


Figure 10. Intel® Server Board S5520UR, S5520URT Functional Block Diagram



An 8-socket board layout

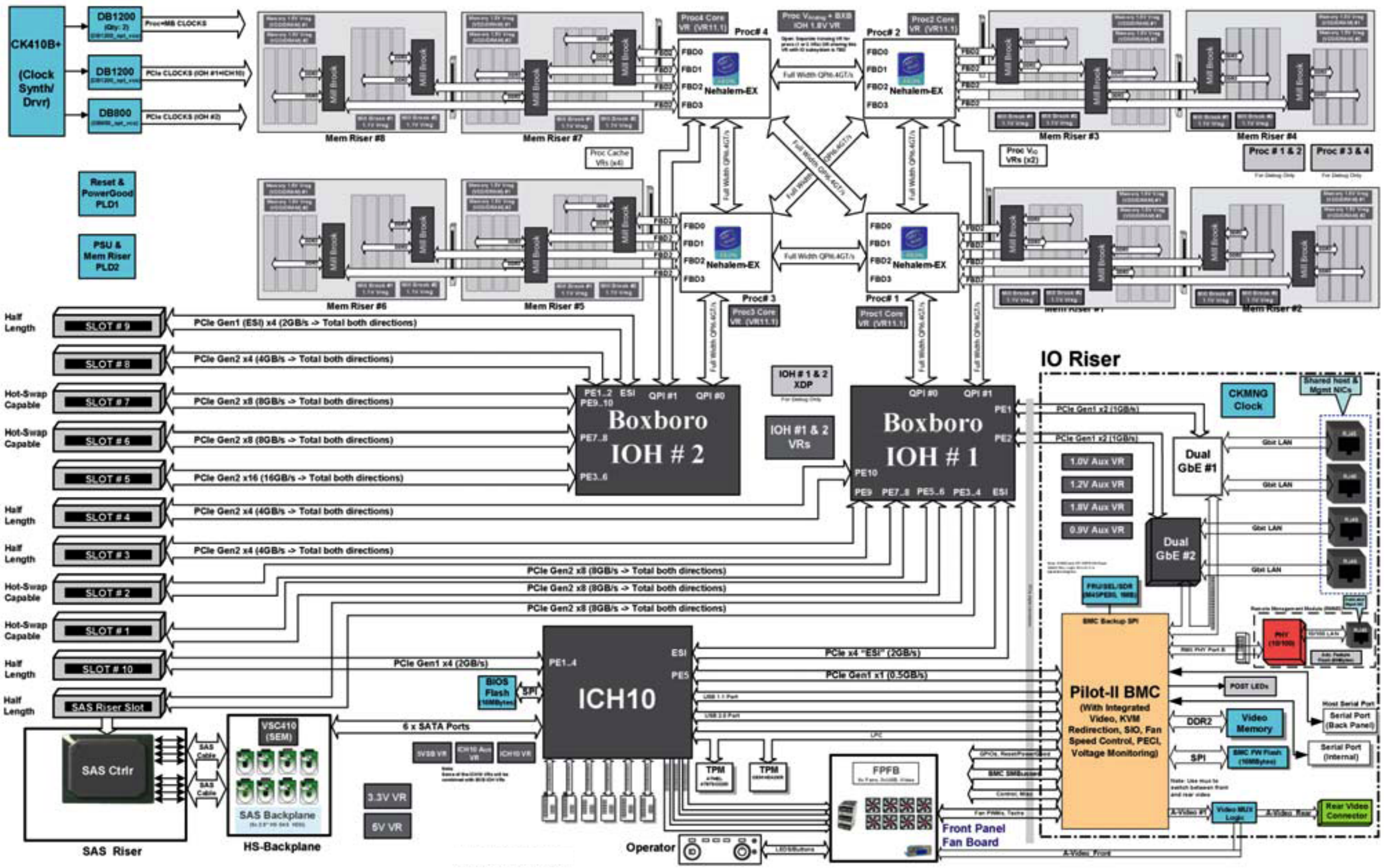
Highlights:

- Sockets (2, 4, 8)
- Speed versus Access

How do we make these choices for the enterprise?

Putting it together – The “Big Iron” Initiative

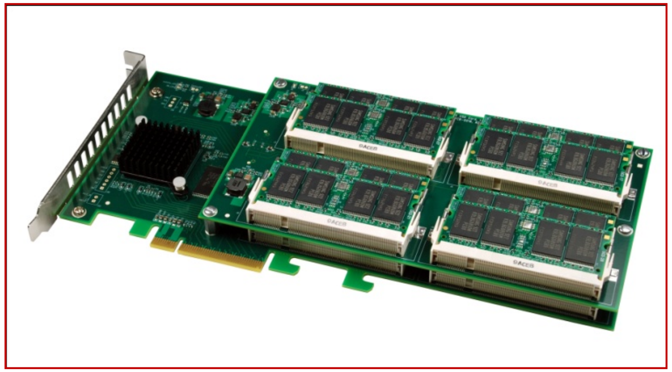
S4R System Block Diagram



Storage

Considerations for storage

- Cost per Terabyte (for the data explosion)
- Speed of access (for real time access)
- Storage network (for global business)
- Disk, Flash, DRAM
- How can applications be optimized for storage?

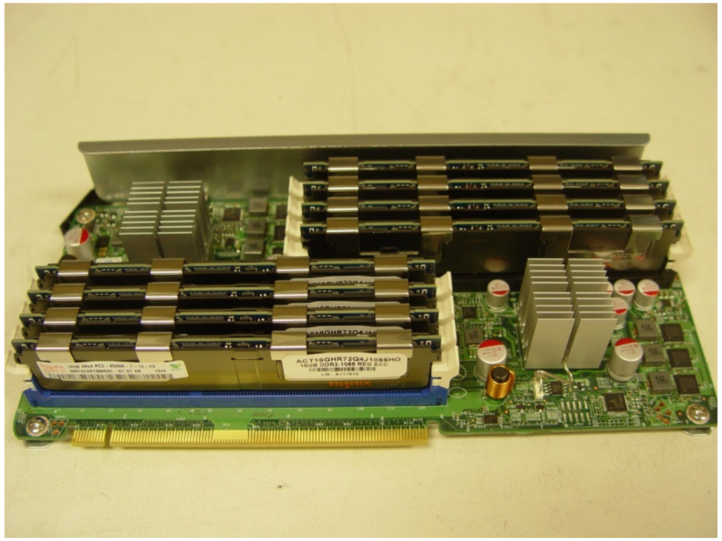


Example: A 2TB OCZ Card

Memory Considerations

Highlights

- Today, cores and memory currently scale in lock step, ***but core density is increasing faster than DRAM density!***
- Large memory computing (to handle modern business needs)
- Collapse the multi-tiered architecture levels
- NUMA support is built in to Linux
- Physical limits (board design, connections, etc.)
- ***Connections between boards are evolving, lowering latencies between interconnects!***



Memory Riser

Flattening the layers

- We need to innovate without disrupting customers. (Like your multicore laptops.)
  - Seamless platform transitions are difficult, but doable (Digital, Apple, Intel, Microsoft)
- Large memory technology is poised to take off, but it needs appropriate hardware
- High Performance Computing (HPC)* has yielded results (Vanguard 1991) that can now be applied to Enterprise Software; today it's different:
  - We have a lot of cores now and better interconnects
  - We can "flatten" the layers and simplify
  - We can build systems that improve our software now without making *any major modifications*
- *Nail Soup argument: But, if we are willing to modify our software, we can win bigger. But we do this on our own schedule. ("Stone Soup" on Wikipedia)*

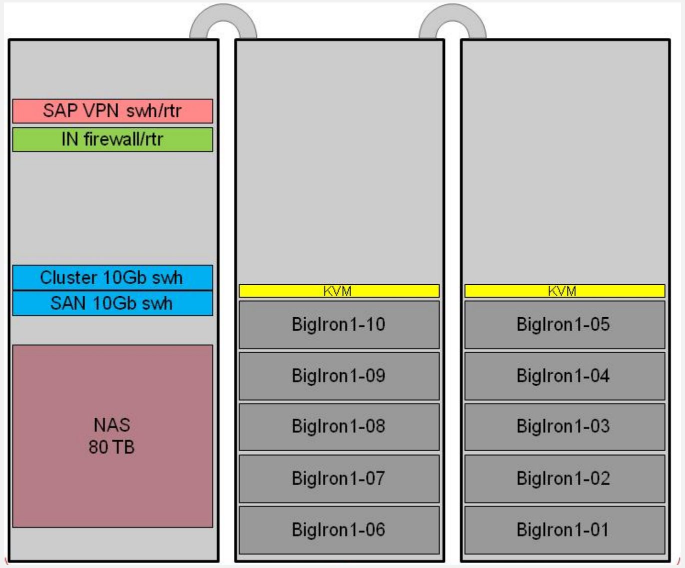
*Scientific/Technical Computing

What is "Big Iron" Technology Today?

Big Iron-1

Extreme Performance, Low Cost

Traditional Cluster



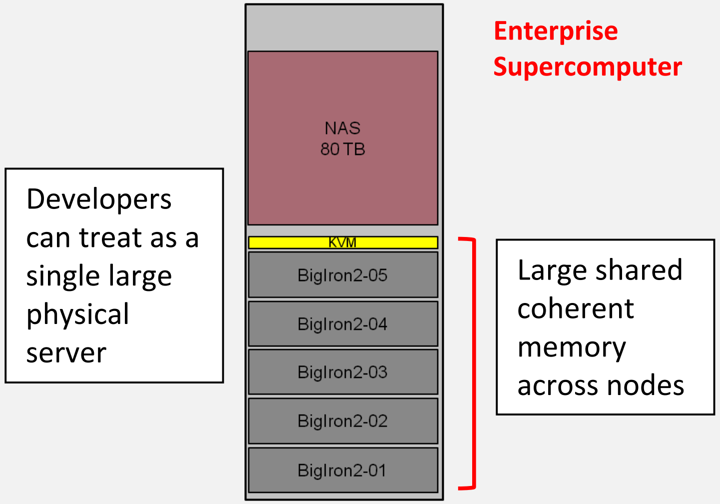
10 x 4U Nodes (Intel XEON x7560 2.26Ghz)

- 320 cores (640 Hyper-threads), 32 cores per node
- 5TB memory (10TB max), 20TB solid state disk

System architecture: SAP

Big Iron-2

Extreme Performance, Scalability, and much simpler system model



5x 4U Nodes (Intel XEON x7560 2.26Ghz)

160 cores (320 Hyper-threads), 32 cores per node

5 TB memory total, 30TB solid state disk

160 GB InfiniBand interconnect per node

1 NAS (72TB Expandable to 180)

Scalable coherent shared memory (via ScaleMP)

System architecture: SAP

The Next Generation Servers a.k.a. “Big Iron” are a reality

<u>Big Iron -1</u>
Fully Operational
September 30, 2010 2:30pm

All 10 Big Iron-1 nodes up and running

<u>Big Iron -2</u>
Fully Operational
February 4, 2011

All 5 Big Iron-2 nodes up and running

The “Magic” Cable (move cursor on box to play video)

But what about…

Distributed Transactions?

- We don't need no stinkin' distributed transactions!

What about traditional relational databases?

- In the future, databases become *in-memory* data structures!

(Do I really believe this? Well, not really. Just wanted to make the point.)

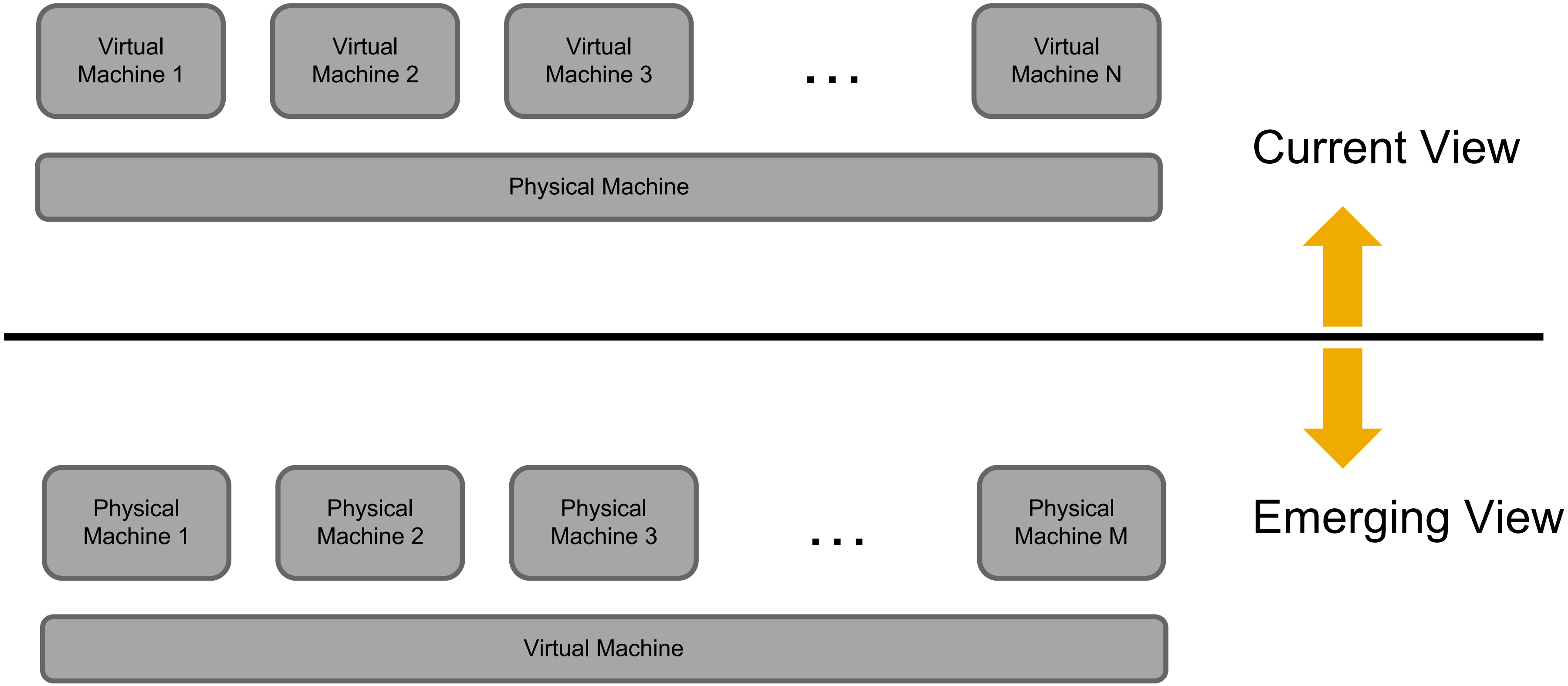
Why not build a mainframe?

- From a SW perspective, it **is** a mainframe (with superior price/performance for in-memory computing)

Is it Virtualization?

- In traditional virtualization, you take multiple virtual machines and multiplex them onto the same physical hardware. We're taking physical hardware instances and running them on a single virtual instance.

A simplified programming model and landscape



(Single OS instance)

To sum up

- Cloud computing has emerged
- The businesses of tomorrow will function in real time. Really!
- There is a growing computing technology gap
- There is an unprecedented wave of advances in HW
- The “mainframe” has been reincarnated with it’s attendant benefits, but can be made from commodity parts

Enterprise supercomputers leveraging coherent shared memory principles and the wave of SW+HW advances will enable this new reality

Questions?

Contact information:

Ike Nassi
New Email: nassi@nassi.com