Designing Interoperability into IA-64 Systems:

DIG64 Guidelines

Michael Demshki - Intel, DIG64 Chair

Melvin Benedict - Compaq, Hardware Architect

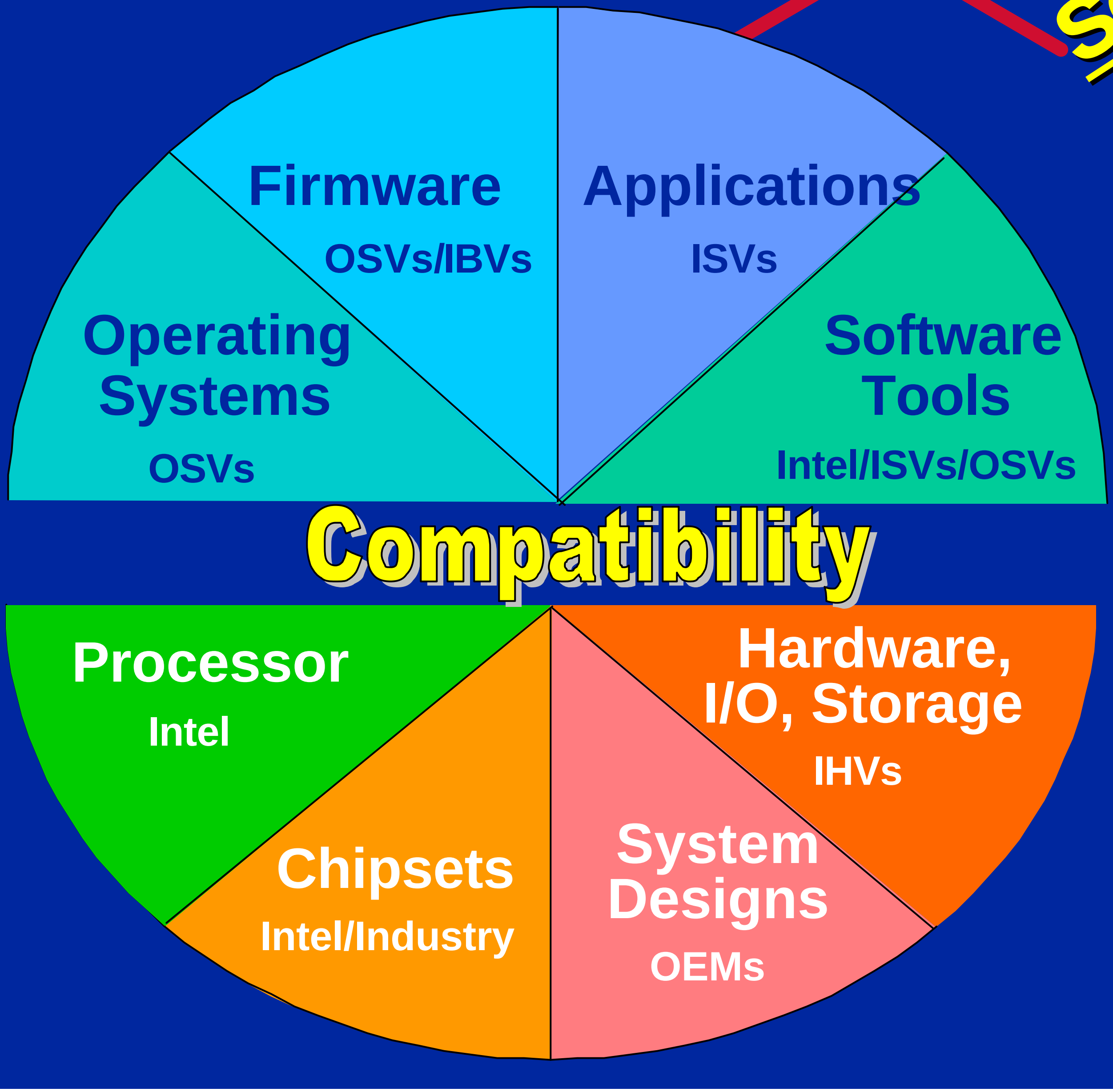
Dong Wei - Hewlett-Packard, Platform Architect

Tomm Aldridge - Intel, Architecture Manager

Agenda

- Introduction to DIG64
- Guidelines for Migration of Legacy Technology
- Guidelines for Firmware
- Guidelines for Core System
- Getting Involved with the DIG64

What is an IA-64 ~~system~~? Solution



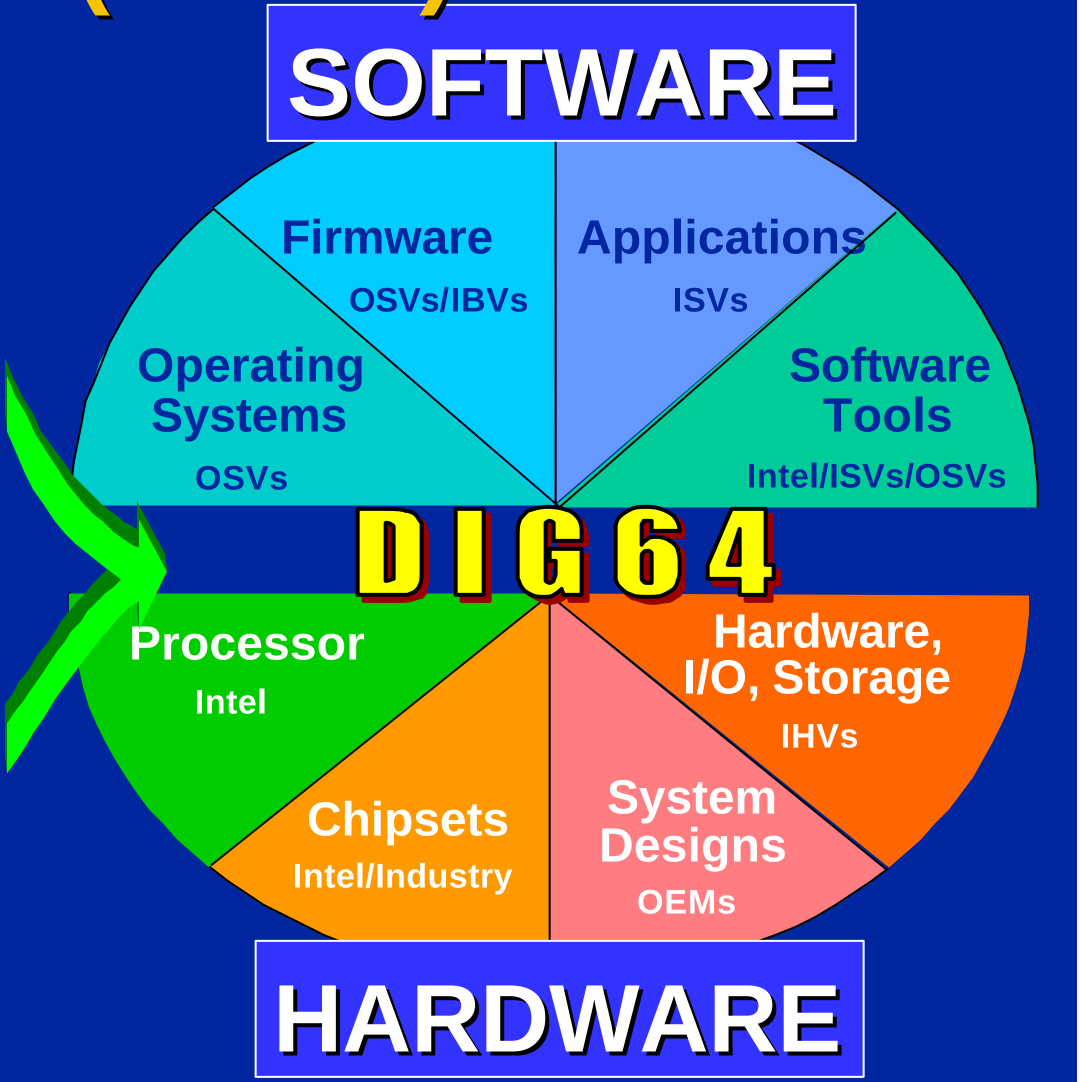
Developer's Interface Guide for IA-64 Servers (DIG64)

Goals:

Define HW/SW compatibility for IA-64 system solutions:

- System building blocks
- SW interfaces
- Legacy Transitions

Enable technology innovation



What will DIG64 do?

Developer Benefits:

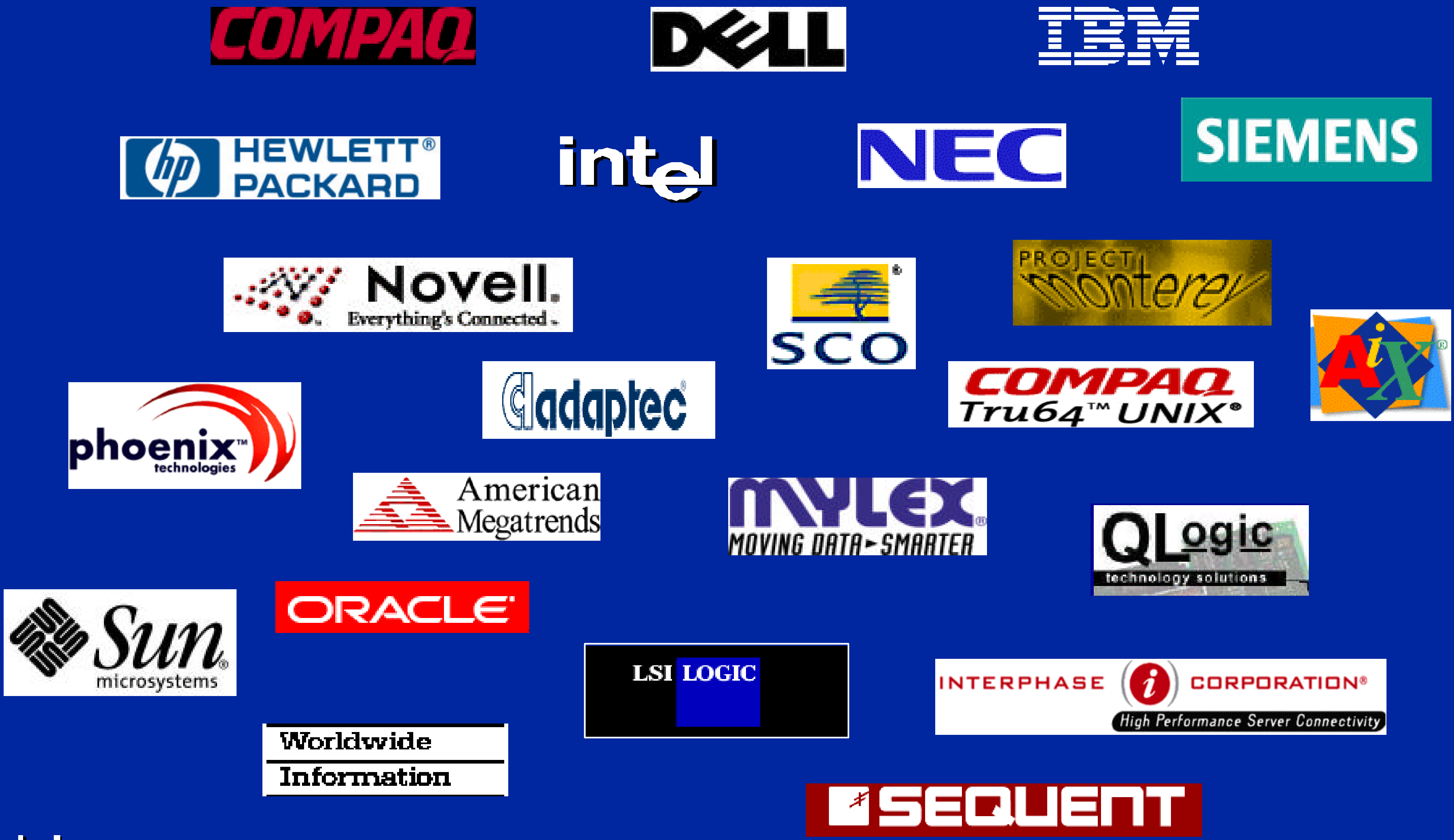
- ✓ Accelerate time-to-market for IA-64 solutions through concurrent development of HW & SW
- ✓ Increase developer efficiency
- ✓ Enable focus on innovation and differentiation
- ✓ Provide a technology migration roadmap

IT Benefits:

- ✓ Increase the number of interoperable solutions available to IT
- ✓ Increase reliability of solutions
- ✓ Lower qualification costs
- ✓ Lower support costs of obsolete technology

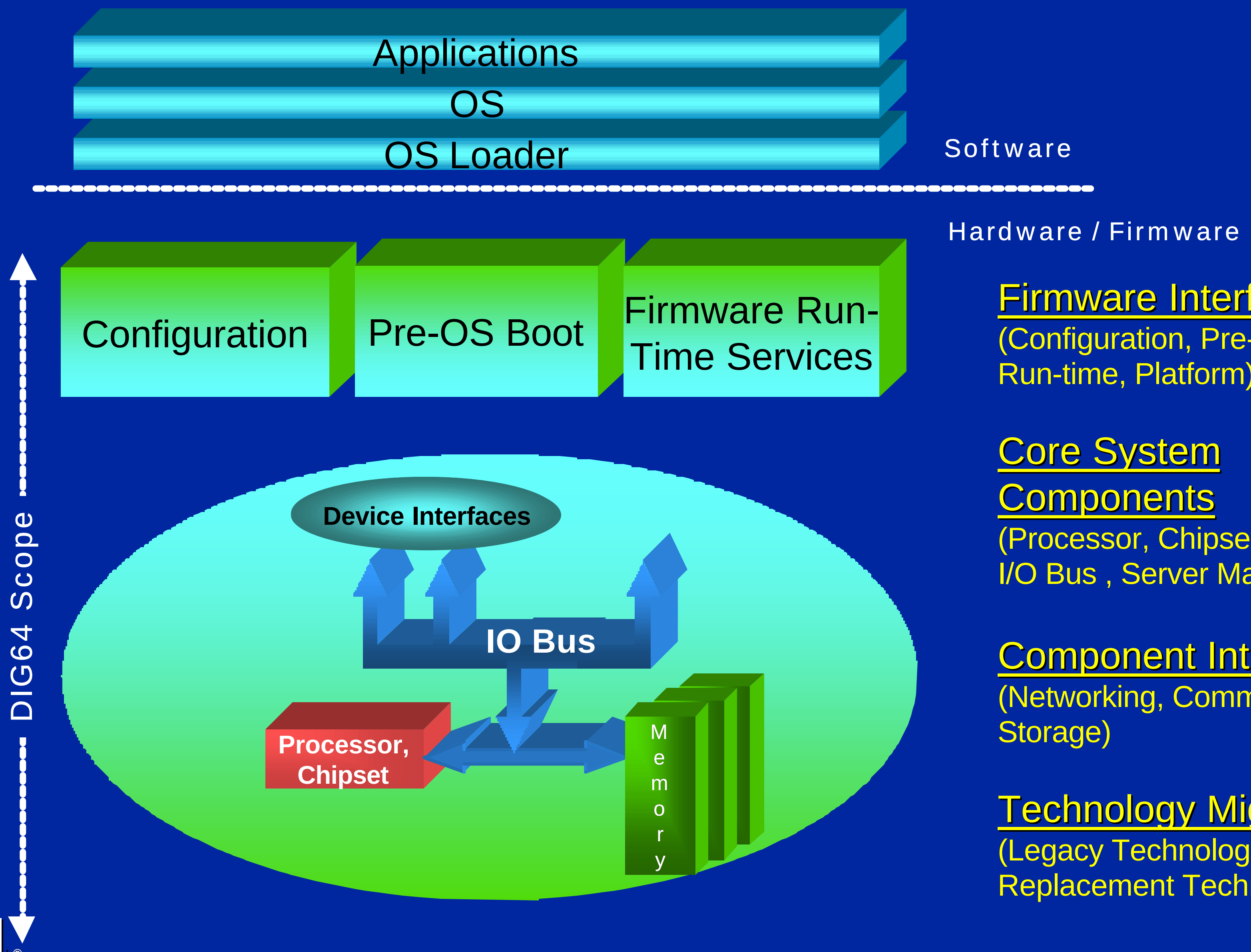
Who Developed DIG64?

Promoters and Contributors*



*third party brands and names are the property of their respective owners
Sun, Sun Microsystems, and Sun logo are trademarks or registered trademarks of Sun Microsystems, Inc. in the US and other countries

DIG64 Scope



<u>Firmware Interfaces</u>
(Configuration, Pre-OS, OS Run-time, Platform)

<u>Core System Components</u>
(Processor, Chipset, Memory, I/O Bus , Server Management)

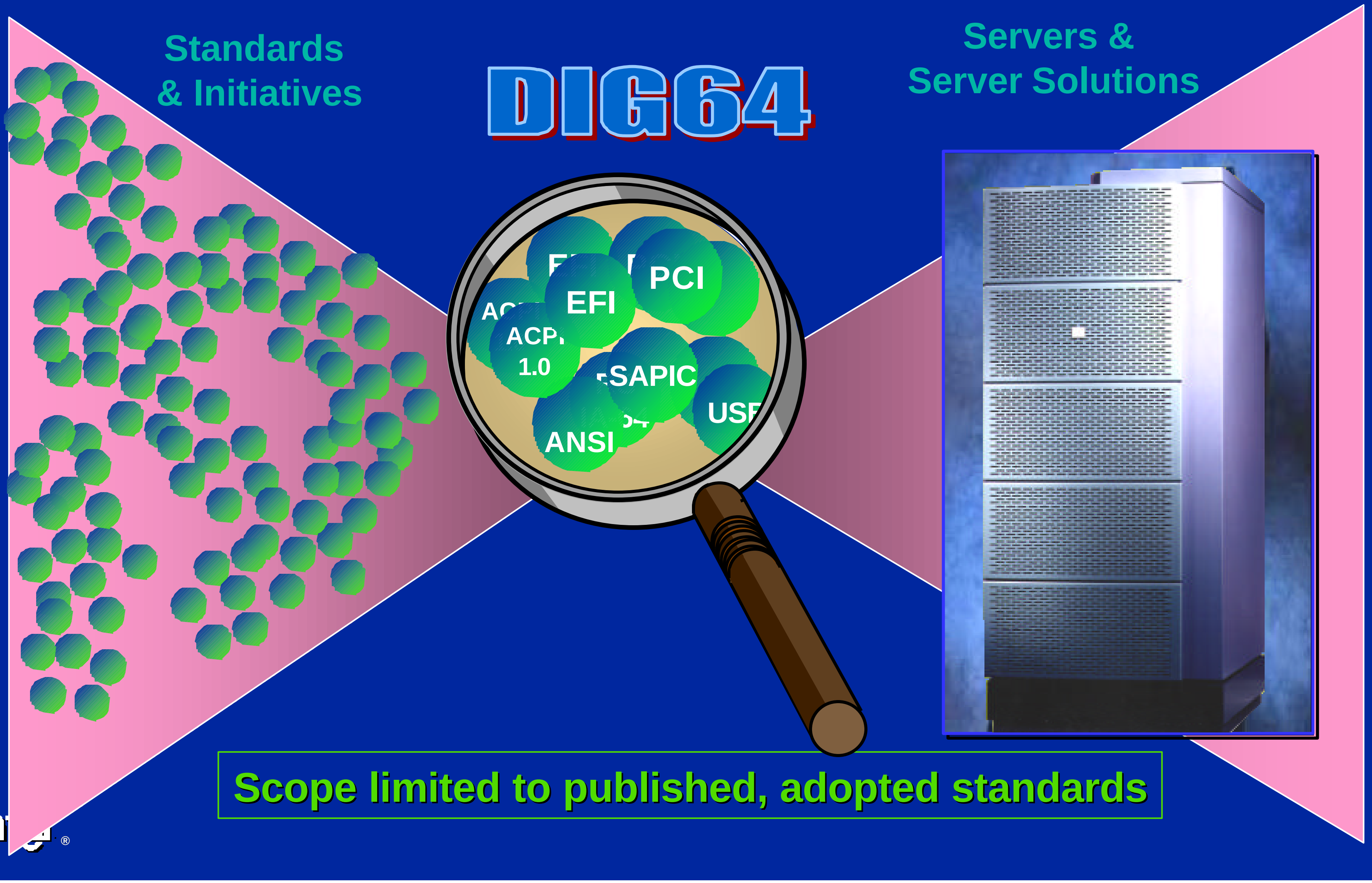
<u>Component Interfaces</u>
(Networking, Communications, Storage)

<u>Technology Migration</u>
(Legacy Technologies, Replacement Technologies)

DIG64 Scope - continued

- **DIG64 Release 1.0 is intended for Itanium™-based products**
  - ◆ There will be subsequent DIG64 releases targeting McKinley and future IA-64 processor generations
- **DIG64 is operating system-independent**
  - ◆ Oriented toward cross-platform interoperability
- **DIG64 is not bounded by system size, scalability, architecture, or application**
- **DIG64 does not address: packaging, form factor, environmental design, or proprietary solutions**

DIG64 Content: Key Standards & Implementation Requirements



Standards & Initiatives

DIG64

Servers & Server Solutions

Scope limited to published, adopted standards

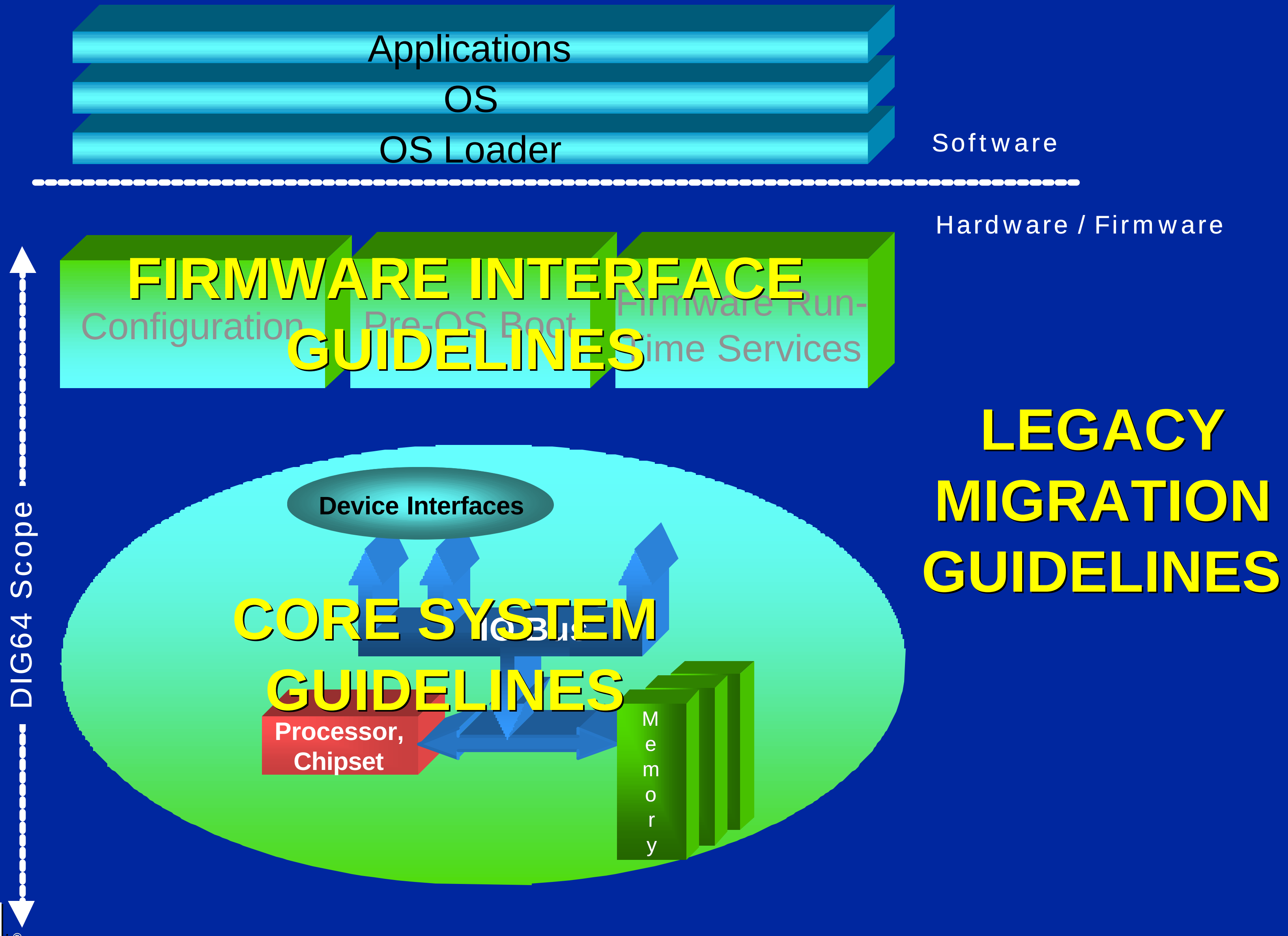
DIG64 Product Compliance

- Compliant products will satisfy all "required" guidelines
- Compliance is determined through self testing
- Test suites and plugfests will be used to facilitate compliance testing
- A DIG64 web site and marketing programs will be established to promote compliant products

DIG64 Terminology

- **Required:**
  - Features marked as *required* must be implemented to comply with the DIG64
- **Recommended:**
  - Features marked as *recommended* need not be implemented to comply with the DIG64, however, implementation is encouraged
- **Optional:**
  - Features marked as *optional* need not be implemented to comply with the DIG64. There should be no dependencies on *optional* features by other architectural levels.

Key DIG64 Guidelines

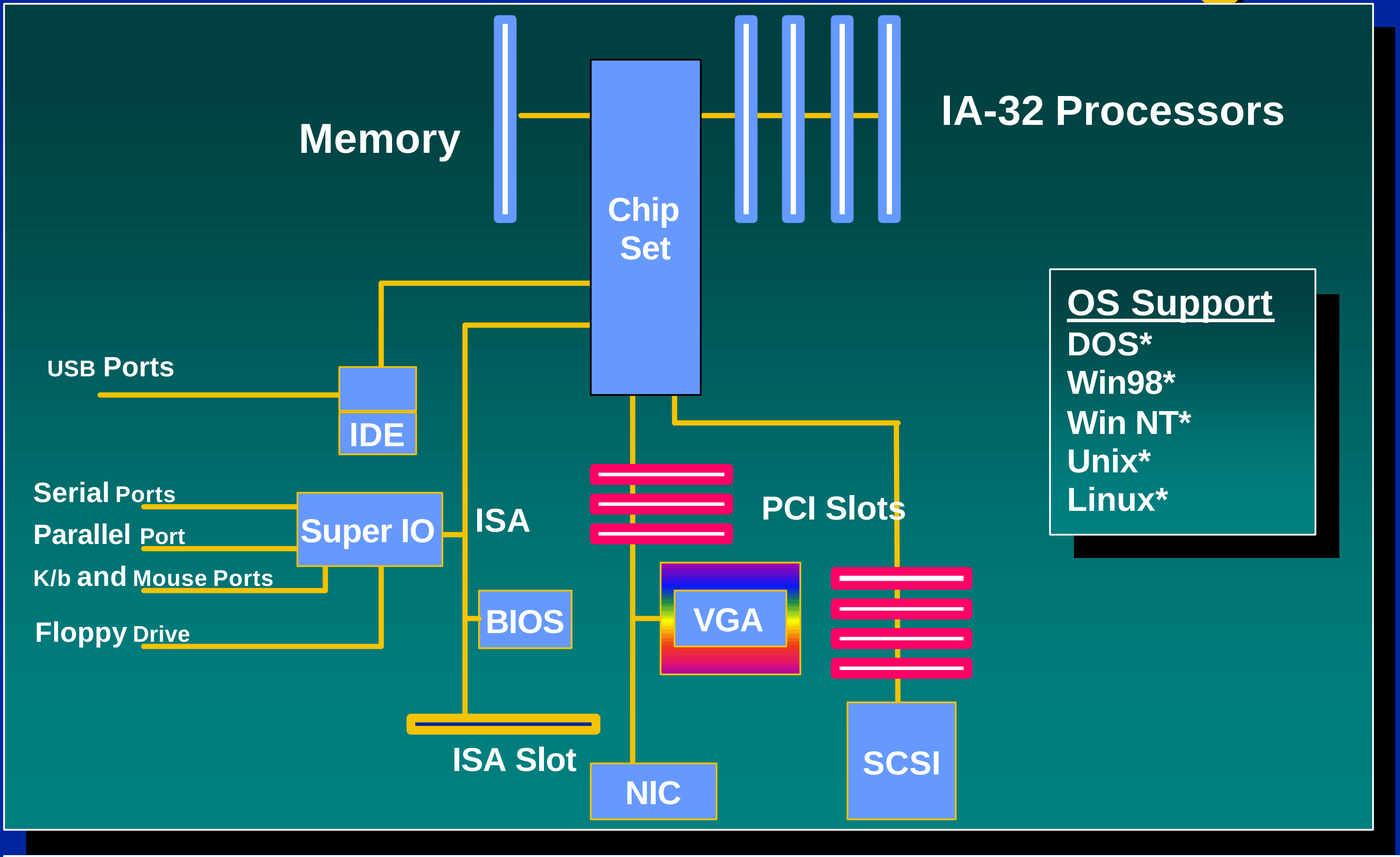


Guidelines for Migration of Legacy Technology

Melvin Benedict - Compaq,

Hardware Architect

Classic IA-32 Server Design



OS Support
DOS*
Win98*
Win NT*
Unix*
Linux*

Definition: Legacy is the retention of functions used solely to support compatibility with older technology.

*other brands and names are the property of their respective owners

Problems with Classic Design

- Legacy I/O does not scale
  - Slower clock rate and narrow data paths
  - No support for dynamic resource allocation
- ISA Memory regions
  - Non contiguous memory maps
  - Fixed addresses
- ISA slot support
  - Uses precious real estate
- VGA support
  - High end servers moving to “headless” operation
- Large OS validation costs

But, Legacy Removal Does Occur !

- **Lack of usage**
  - ◆ ISA slots
- **Slow, takes several generations**
  - ◆ Usually requires a catalyst
  - ◆ Check off purchasing
  - ◆ The other OEM may still support (fear)
- **IA-64 is a Catalyst!**
  - ◆ Architectural break in system growth
  - ◆ DIG64 accelerates removal process
- **Other Initiatives**
  - ◆ EASY PC
  - ◆ Hardware Design Guide for Microsoft Windows NT Server vers. 2.0 +

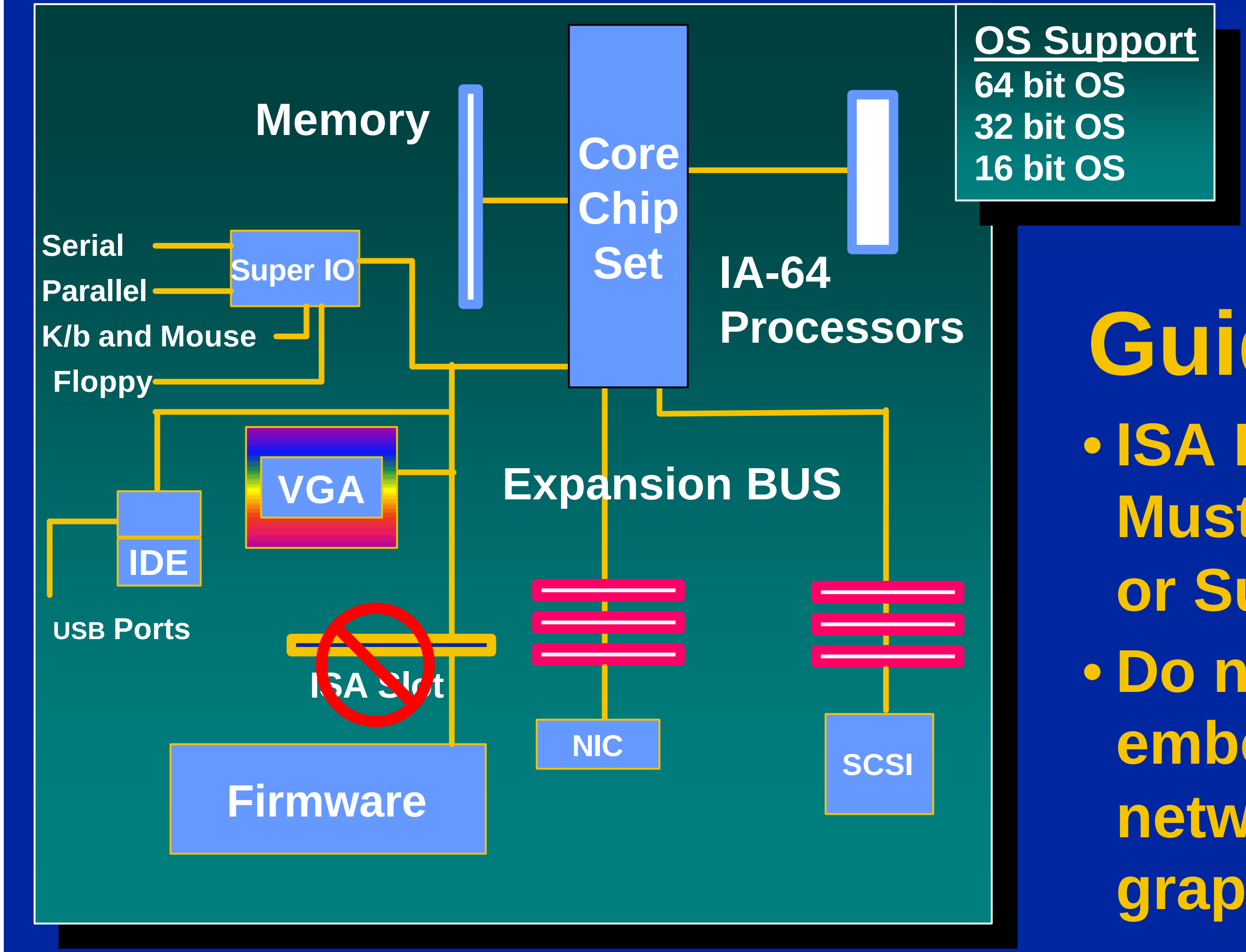
DIG64 Platform and Legacy

- **The retention of legacy in architecture will:**
  - Add complexity to new technologies
  - Stifle innovation
  - Add costs
- **DIG64 is the opportunity to make a break with the past**
- **Keys to migration**
  - Abstraction layers in firmware
  - Extensible Firmware Interface (EFI) boot loader
  - Optional guidelines

DIG64 Provides Evolution Roadmap

- **Phase out the legacy in servers over time by creating Optional guidelines**
  - ◆ Feature support is OEM responsibility
  - ◆ OEMs dictate own pace of legacy migration
  - ◆ Does not “forbid” features if OEM desires to support
  - ◆ Provides a bridge for manufacturing and utilities
- **Abstraction of hardware functions is key enabler**
  - ◆ EFI boot services at “power on” of IA-64 systems
  - ◆ Abstraction services added over time

Legacy Removal in the DIG64

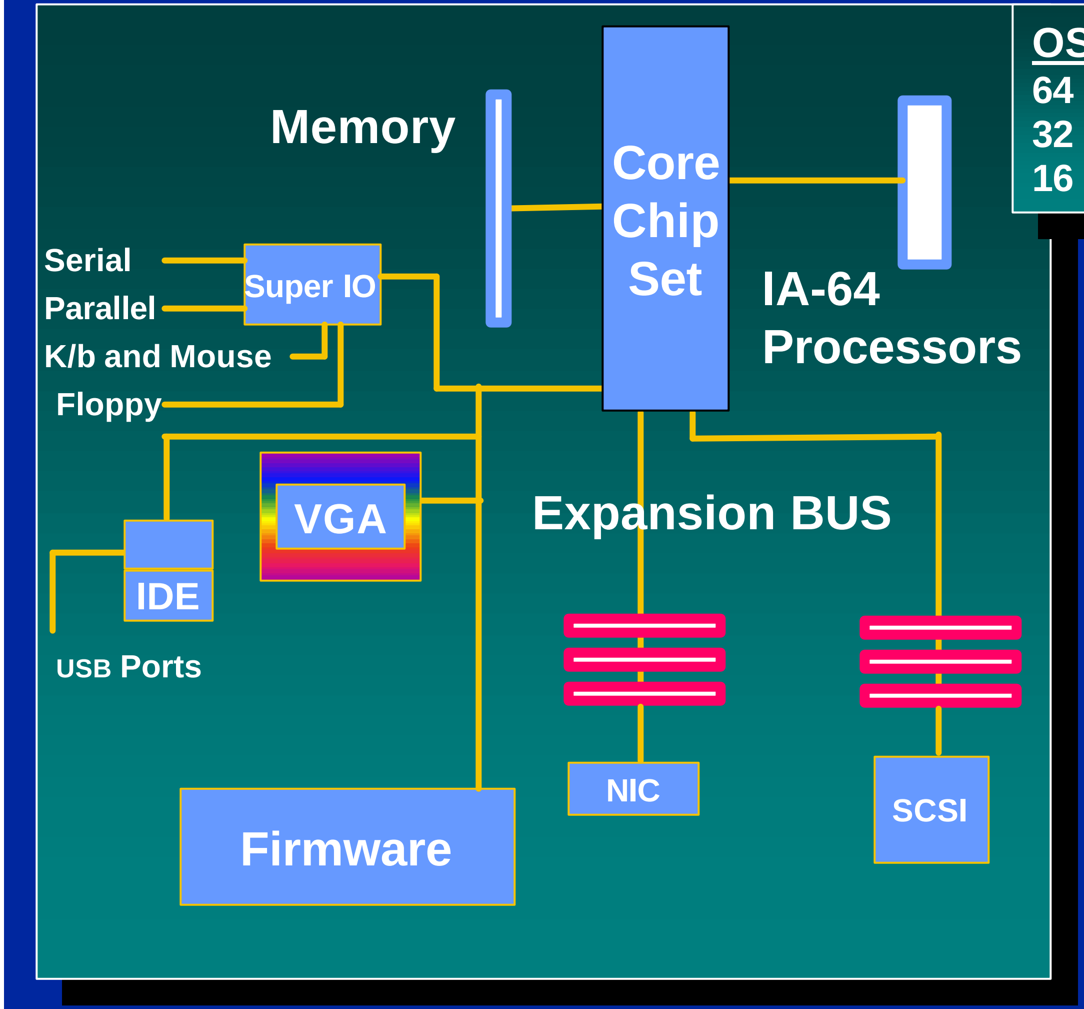


OS Support
64 bit OS
32 bit OS
16 bit OS

Guidelines:

- ISA Expansion Slots Must Not Be Included or Supported
- Do not include embedded ISA network, storage or graphic adapters

IA-64 OS support



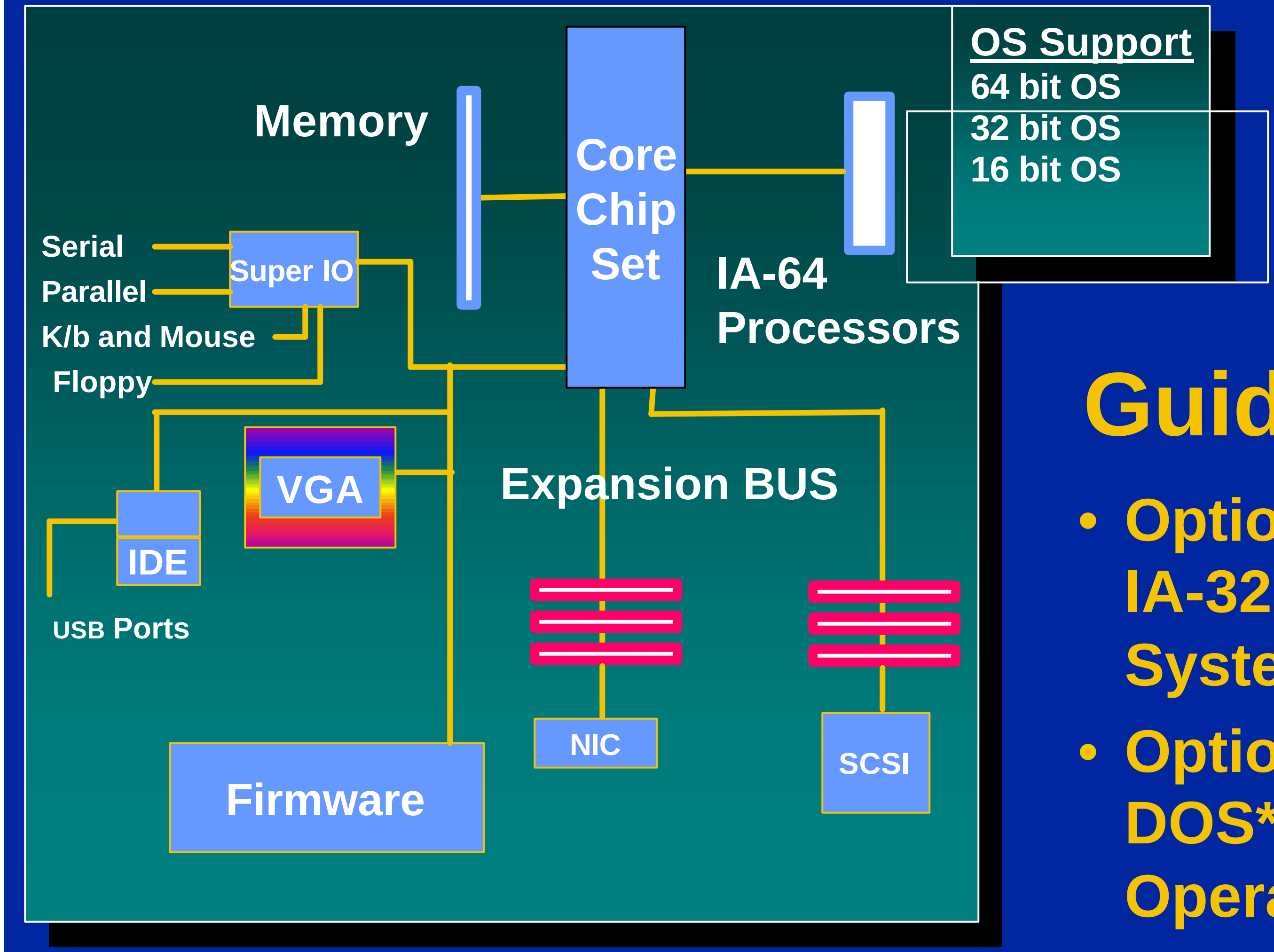
OS Support

64 bit OS
32 bit OS
16 bit OS

Guidelines:

- EFI Boot loader for 64 bit OS. Required
- EFI pre-boot environment Recommended
- EFI option ROM support Recommended

Non 64bit OS Support



OS Support
64 bit OS
32 bit OS
16 bit OS

Guidelines:

- Optional support for IA-32 Operating Systems
- Optional support for DOS* and Win98* Operating Systems

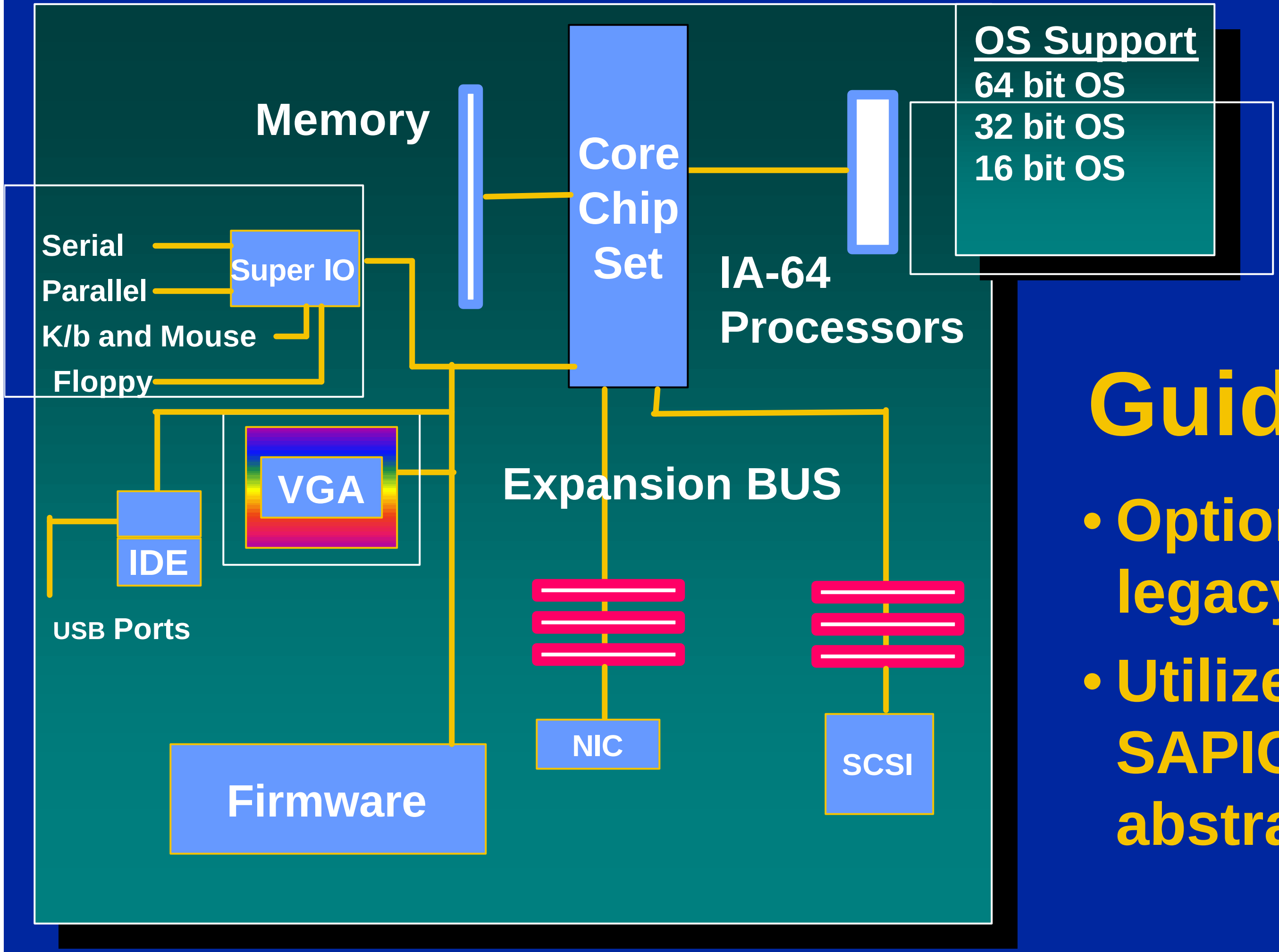
*other brands and names are the property of their respective owners

What does this mean?

- **The 64bit platform is NOT required to support non 64 bit OS's for compliance**
- **The legacy hardware hooks on the platform can now be removed**
  - New 64 bit OS should NOT depend upon legacy hooks!
- **Transition period to allow critical processes time to adjust**
  - DOS* is used to support manufacturing tests and other adapter utilities
  - Replace DOS with EFI pre-boot environment

*other brands and names are the property of their respective owners

Legacy I/O Support...



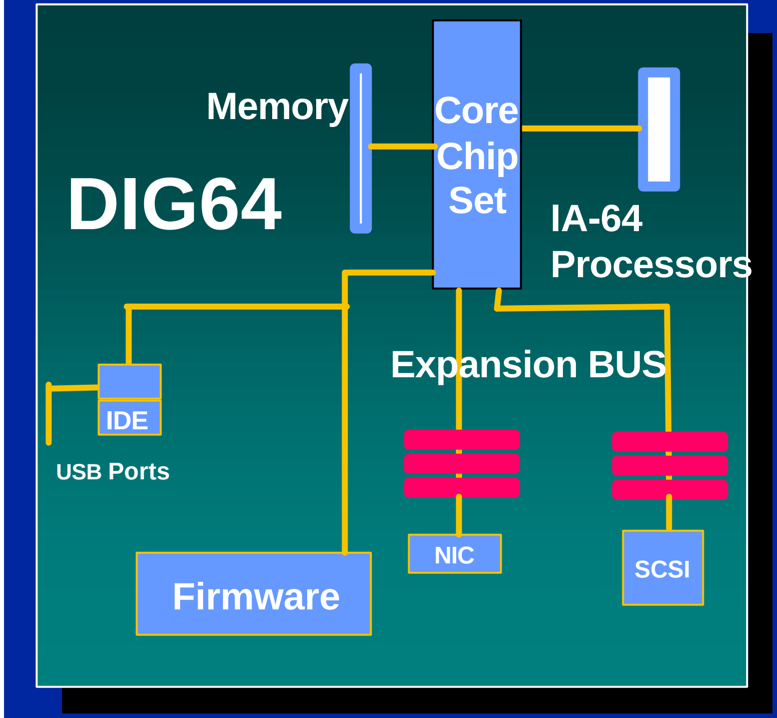
Guideline:

- Optional Support for legacy I/O, VGA
- Utilize IDE, USB, SAPIC, and EFI abstractions

What does this mean?

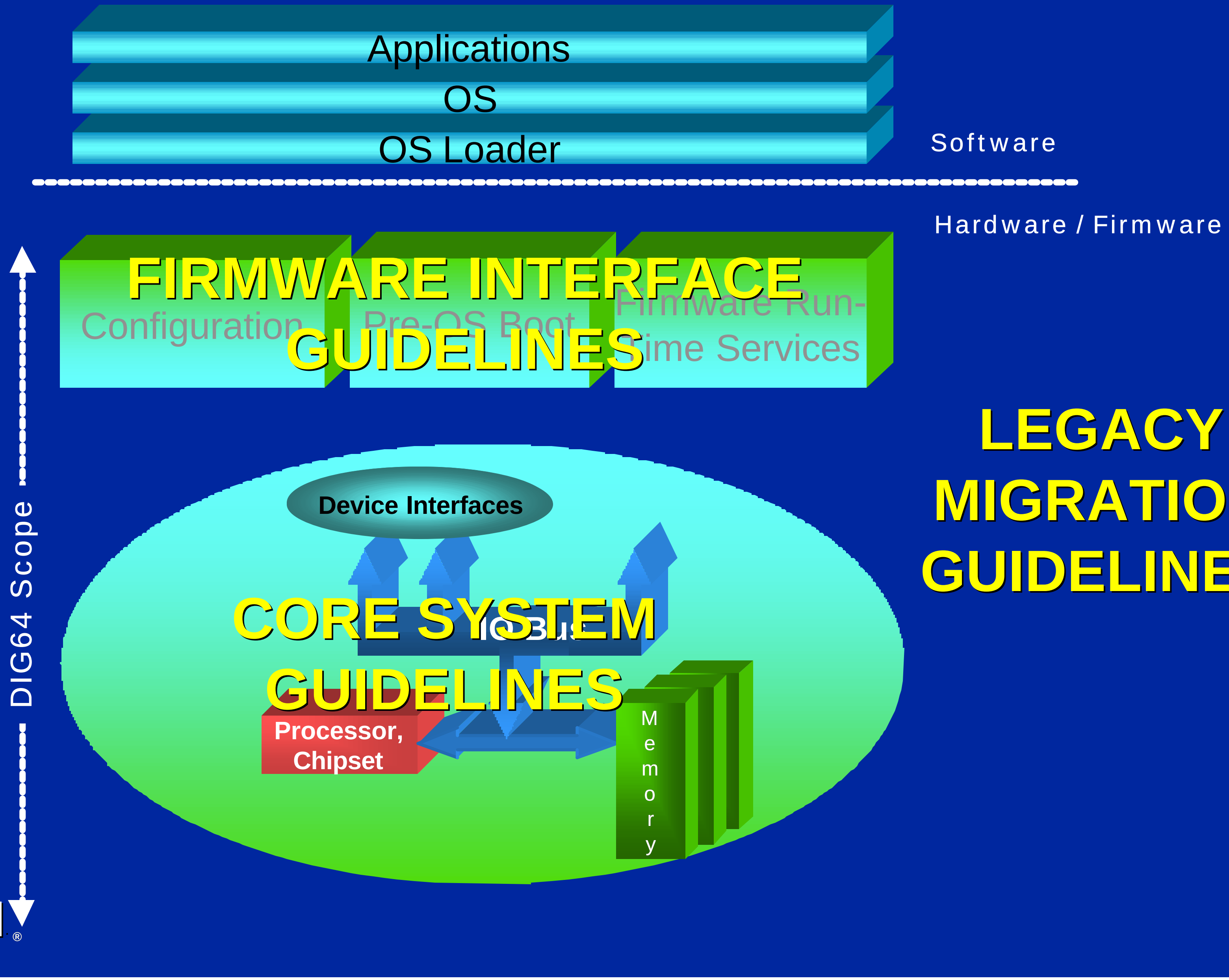
- **Serial, parallel, PS2 ports should all be replaced by USB technology**
  - ◆ USB supports dynamic resource allocation
- **SIO will go the way of the ISA bus**
  - ◆ Lack of usage!
  - ◆ Trend toward “headless” operation is recognized
- **IDE remains as a useful technology**
  - ◆ Removable media
- **Abstraction layers insulate the operating systems**

Conclusions



- Evolutionary Roadmap for legacy removal
- Key enablers
  - Abstraction Layers
  - EFI
  - Optional Guidelines for OS’s
- DIG64 is the opportunity

Key DIG64 Guidelines



FIRMWARE INTERFACE GUIDELINES

CORE SYSTEM GUIDELINES

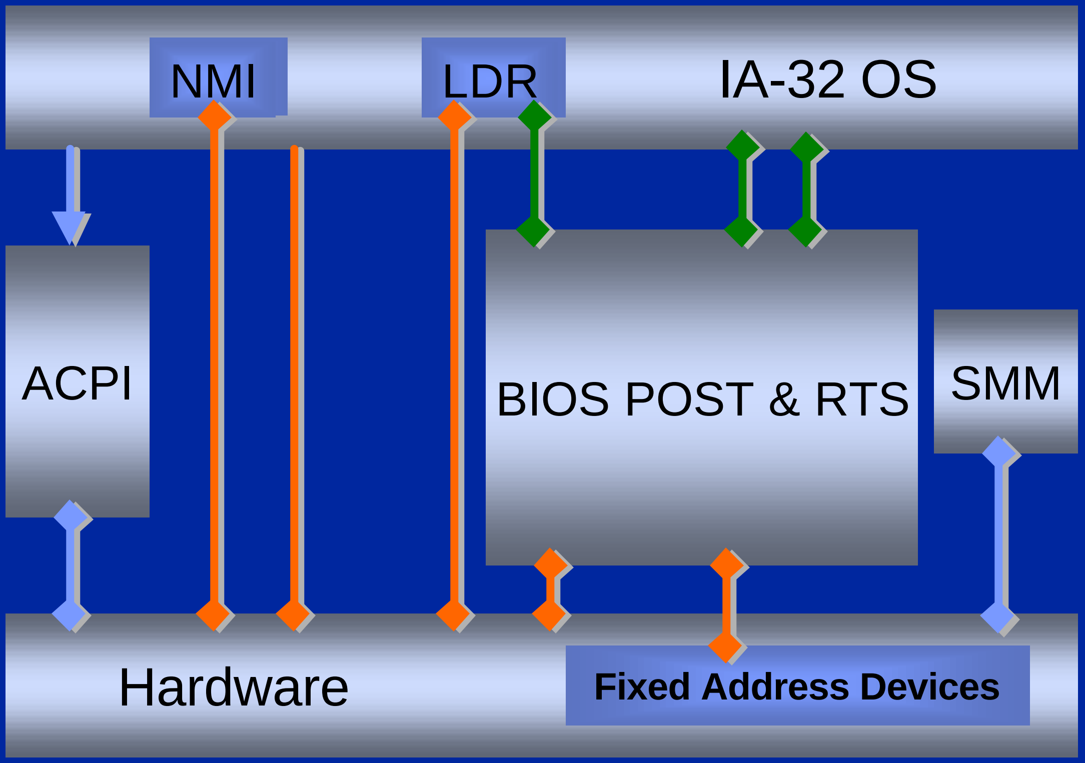
LEGACY MIGRATION GUIDELINES

Guidelines for Firmware Interfaces

Dong Wei - Hewlett-Packard,

Platform Architect

Legacy Firmware Model



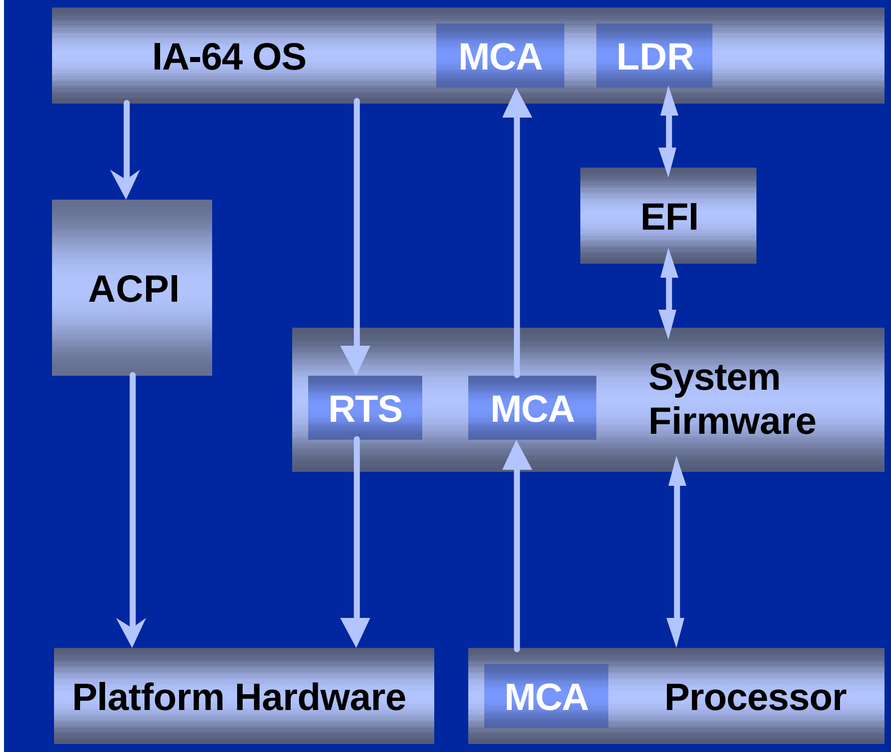
- OS Directly Coupled to HW
- OS Loader Directly Coupled to HW
- 16 Bit OS/BIOS Interface Saturated
- PC-AT Legacy Hardware
- Processor/Platform Coupled

Coupling Slows Down Innovation

DIG64 Firmware

- Objectives
  - Enable Boot of IA-64 OSes
  - Decouple the OS and Hardware Development
  - Decouple the Processor and Platform Development
  - Facilitate Technology Migration
  - Support OEM Differentiation
  - Support Platform Scalability, Reliability, and Availability

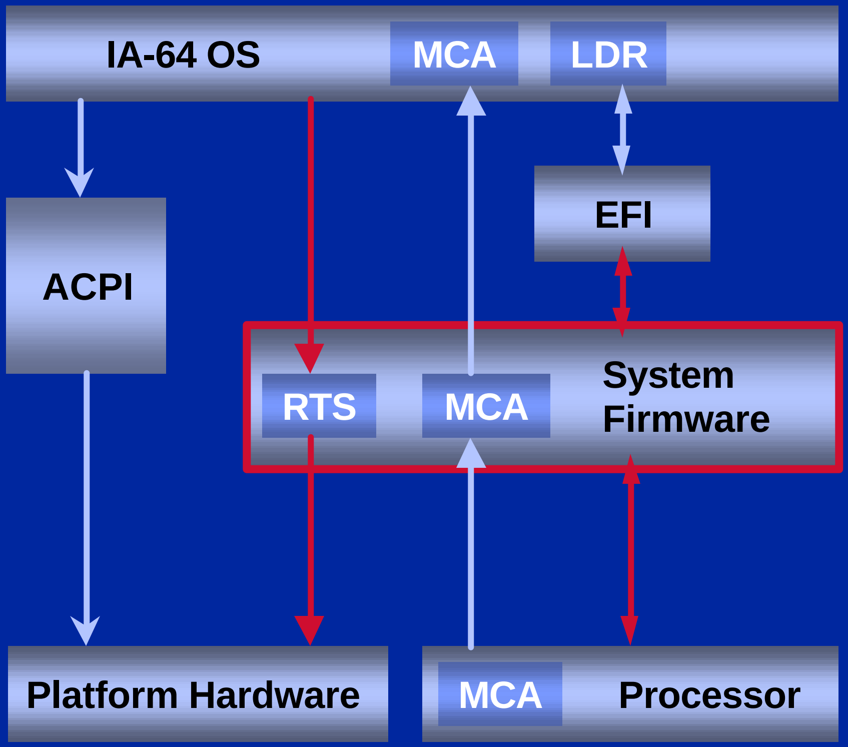
DIG64 Firmware Model



- OS Abstracted from HW
- OS Loader Abstracted from HW
- Uniform Interface
- Separate Processor & Platform Abstraction
- Enable Legacy Removal (e.g., 8259A PIC)

- EFI = Extensible Firmware Interface
- ACPI = Advanced Configuration and Power Interface
- MCA = Machine Check Abort
- RTS = Runtime Services

System Firmware



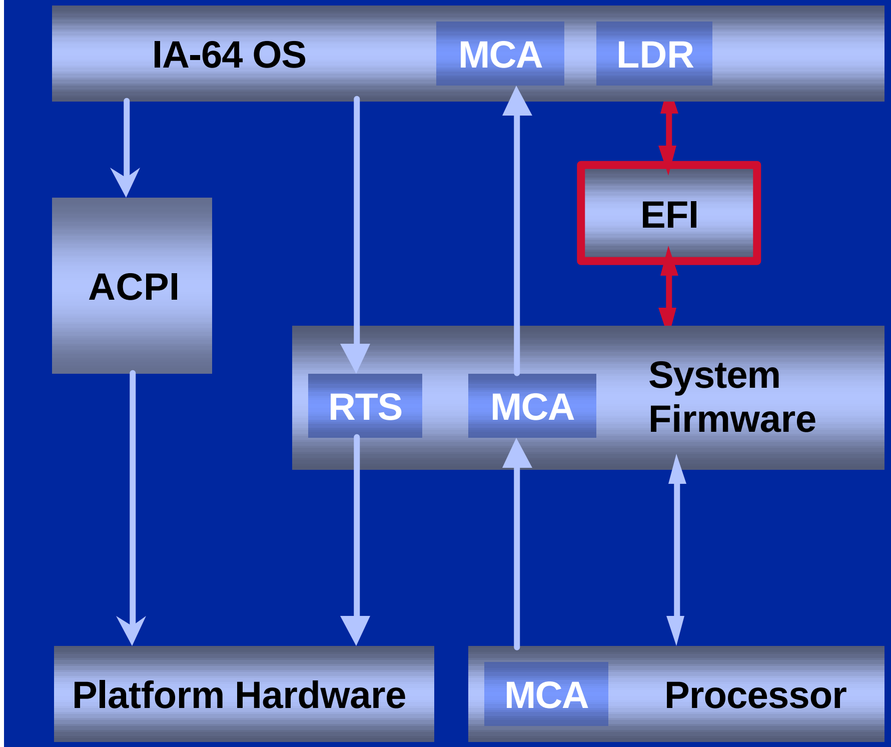
- Initialize, Configure and Test the Platform Hardware (UP and MP)
- Error Recovery (see MCA foils)
- System Table (e.g., platform features descriptor)
- Run-time Procedures (e.g., cache flush)
- Entry Points (e.g., RESET, CHECK, PMI)

Abstract Platform Implementation-specific Features

System Firmware Guidelines

- Implementation
  - Provided by OEMs/IBVs
- Use Run-time Services (RTS) to Access PCI Configuration Space
  - OS Access to PCI Configuration Space not Made Directly to HW
    - Abstract from the CF8h/CFCh Implementation
    - Provide Resource Locking
  - Enable PCI Segments to Handle >256 Buses

EFI



- Boot & Run-time Services
- Protocols (e.g., SERIAL_IO)
- System Partition
- Boot Manager (Boot Order Menu, ...)
- EFI Drivers
- EFI Apps (EFI shell, DIAG, MFG,..)
- Support Legacy-API Free Env.
- Evolutionary Path for Legacy Removal

**Provide an Abstracted Interface to the OS
and
Enable Technology Migration**

EFI Guidelines

- EFI Implementation
  - ◆ Provided by OEMs/IBVs
- NV Storage must be Provided for EFI
  - ◆ Supports EFI Environment Variables
- Implement Serial Protocol
  - ◆ For Debug Applications and Console Redirection
- Recommend PXE for Remote Boot
  - ◆ Uniform Remote Boot Protocols

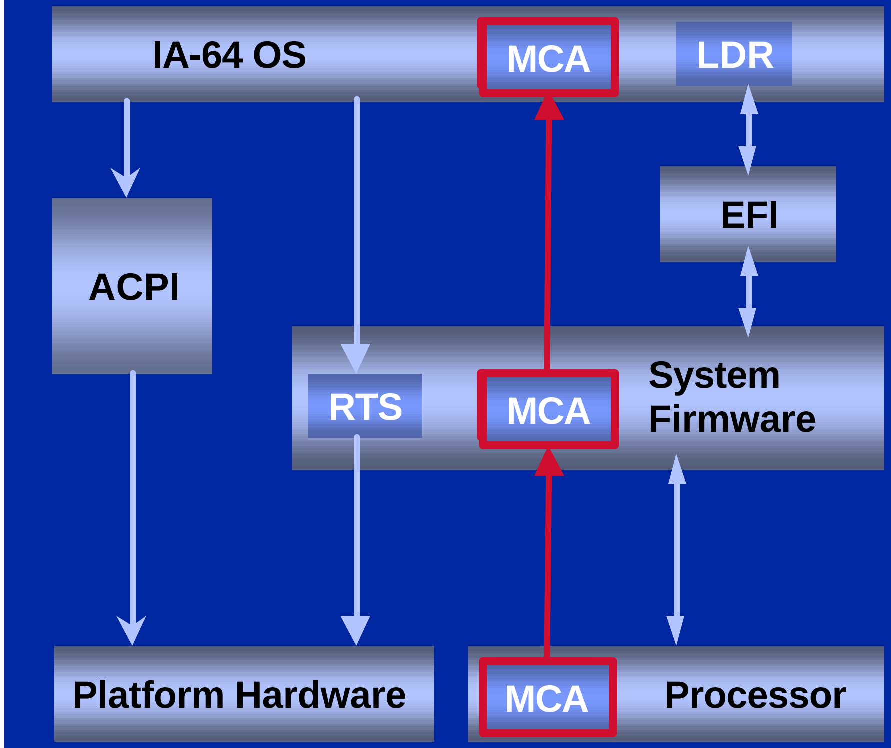
EFI Boot Drivers

- EFI Provides the Migration Path
  - Non-EFI Option ROM Support is Optional
    - PC-AT Hardware Dependency
  - EFI Option ROM Support is Recommended
    - Enables Legacy Hardware Removal
    - IA-64 Native Instruction Set Model Defined
    - Portable Byte Steam Model is under Investigation

Please review EFI presentation at:

http://developer.intel.com/design/ia64/devinfo.htm

MCA



- Check for Error Conditions
- Log Errors
- Error Recovery
- Error Notification

Support More Robust Error Checking and Recovery

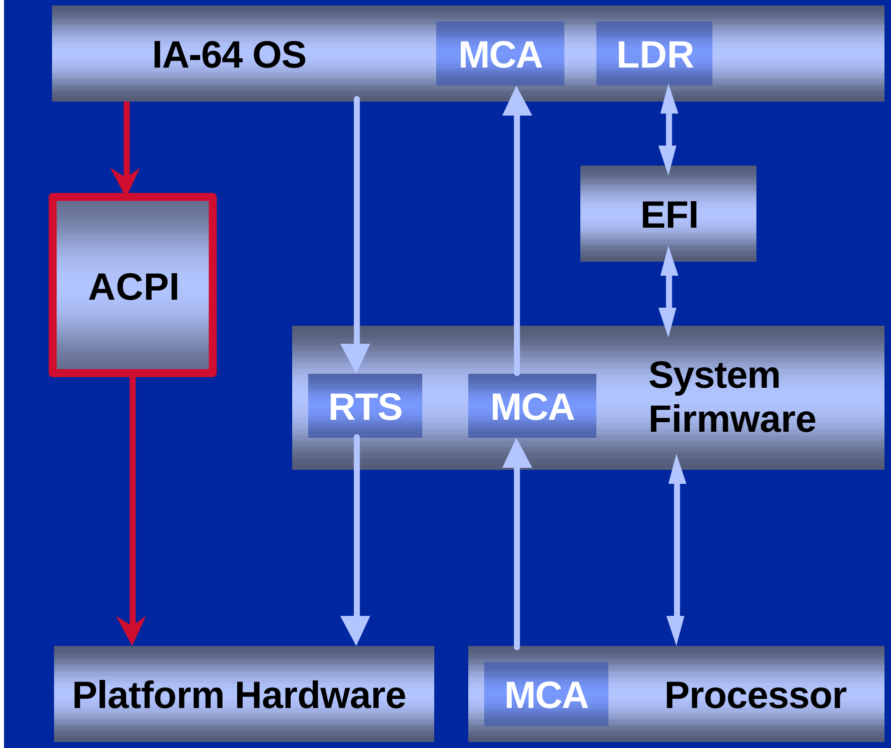
MCA (Cont.)

- MCA Implementation Benefits
  - Improved error containment for internal processor errors
  - Many errors are correctable without processor interruption
  - Many errors are recoverable with OS support
  - Abstractions for system error handling
  - Robust error logging for processor and system errors through well-defined structures
  - Enhanced error handling in multiprocessor environments.

MCA Guidelines

- Provide Support for Code Resources
  - Robust Error Checking and Recovery Mechanism
    - Use MCA Hooks in Processor, System Firmware and OS
    - Discontinue the PMI (SMI) and NMI Usage Model
- Provide NV storage for MCA Logging
  - Enable Error Analysis and Recovery

ACPI



- Server Configuration Mechanism (tables, control methods, IA-64 extensions)
- Obsolete PnP BIOS and MPS Tables
- Relax ACPI Specification Requirements for Servers (e.g., power button, power policy)

Abstract platform HW from the OS

ACPI Guidelines

- ACPI Implementation
  - Provided by OEMs/IBVs
- Use ACPI for Platform Configuration
- Use ACPI for Power Management if PM is Supported
- Provide Multiple SAPIC Description Table
- Provide PCI Interrupt Routing Info
- Use ACPI Control Methods for PCI Hot Plug if PHP is Supported

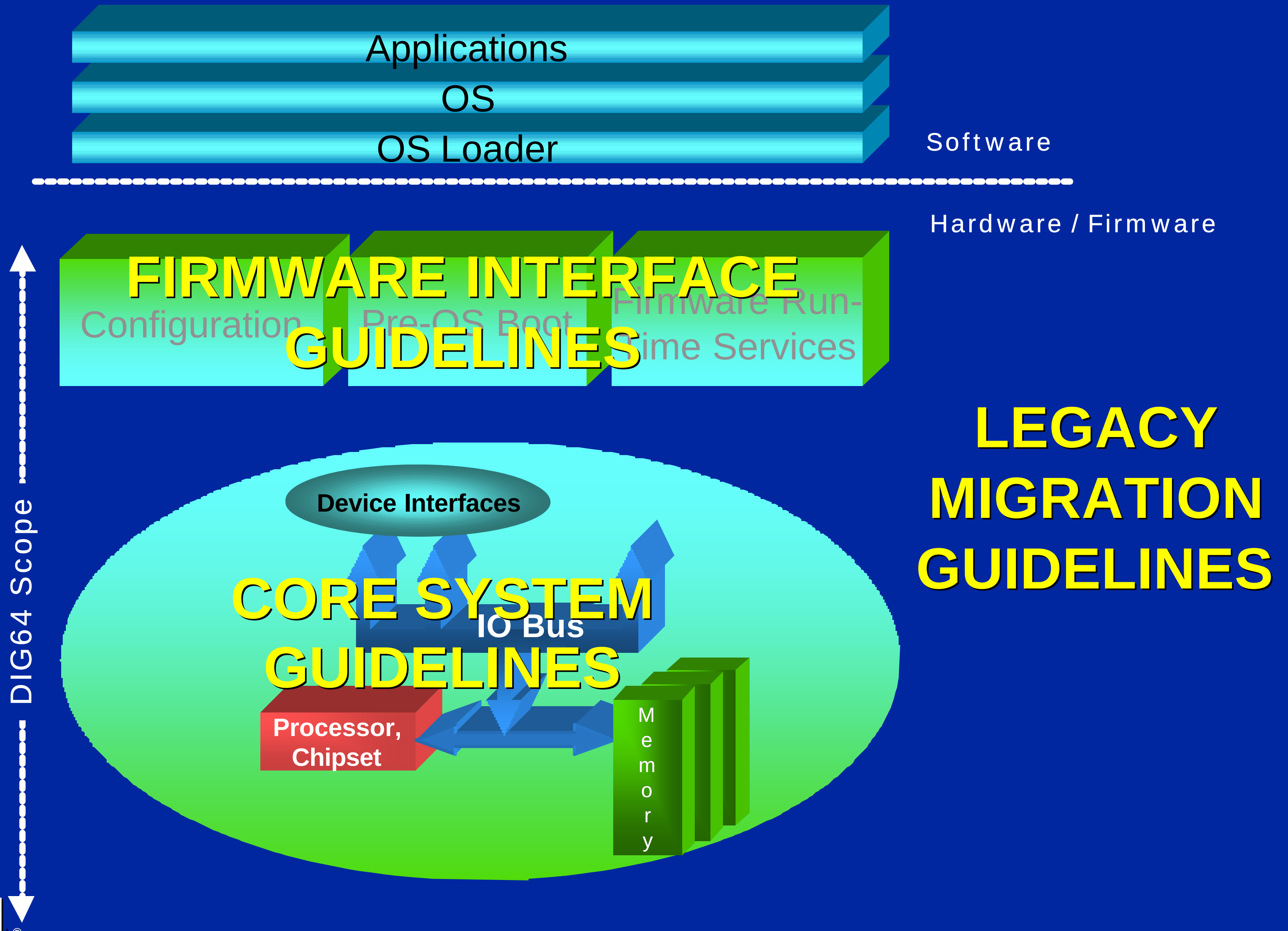
IA-32 Support Guidelines

- Provide IA-32 Emulation in System Firmware if:
  - IA-32 Option ROM is Used
  - Portions of POST are Implemented in BIOS

Conclusions

- Support DIG64 Firmware Model
  - Decouple the OS and Hardware Development
    - Abstraction of Hardware Implementation and Uniform Interface
  - Decouple the Processor and Platform Development
    - Separate Processor and Platform Abstraction
  - Facilitate Technology Migration
    - Legacy API Removal
    - Legacy Hardware Removal

Key DIG64 Guidelines



Core System Guidelines

Tomm Aldridge - Intel,

Architecture Manager

Key elements of a DIG64 Platform

- OS independent
- Memory addressability >4GB
- Designed from ground up to be MP
- Technology migration enabled in orderly fashion
- Firmware abstraction of hardware
  - EFI/Firmware Services layers
  - Legacy free boot process
- Uniform interrupt handling
- MCA error handling
- ACPI based configuration
- Dynamic resource allocation

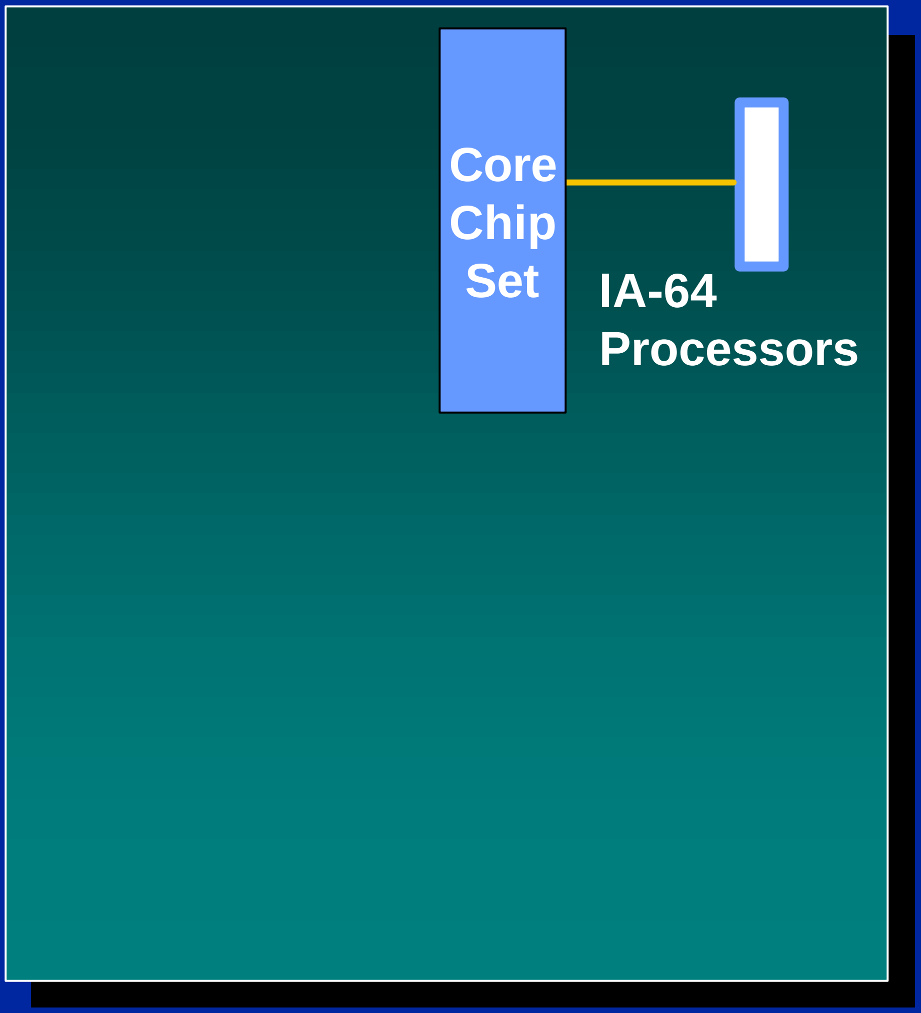
System Processors

IA-64
Processors

Guidelines:

- System processors must execute the IA-64 instruction set

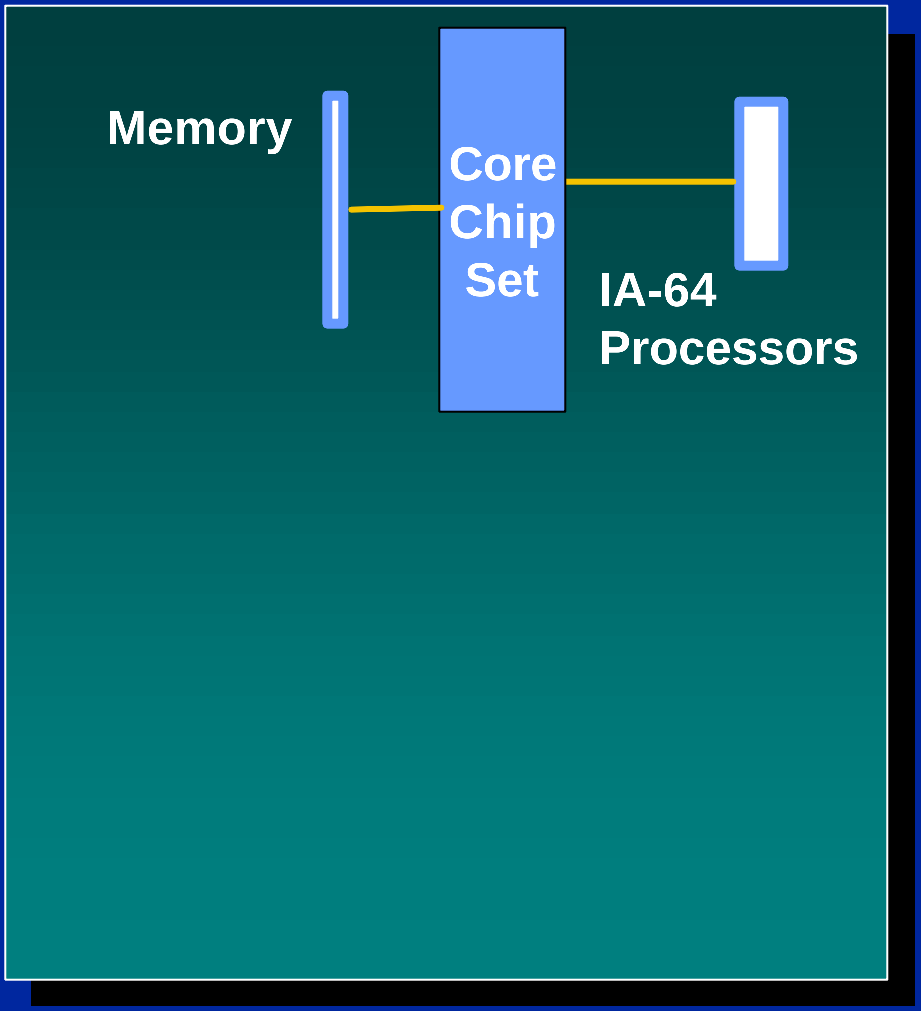
Core Chip Set



Guidelines:

- Must use Symmetric view of system I/O and memory
- Interrupt delivery mechanism must use the SAPIC compatible programming model
- Must permit processor to processor interrupt delivery

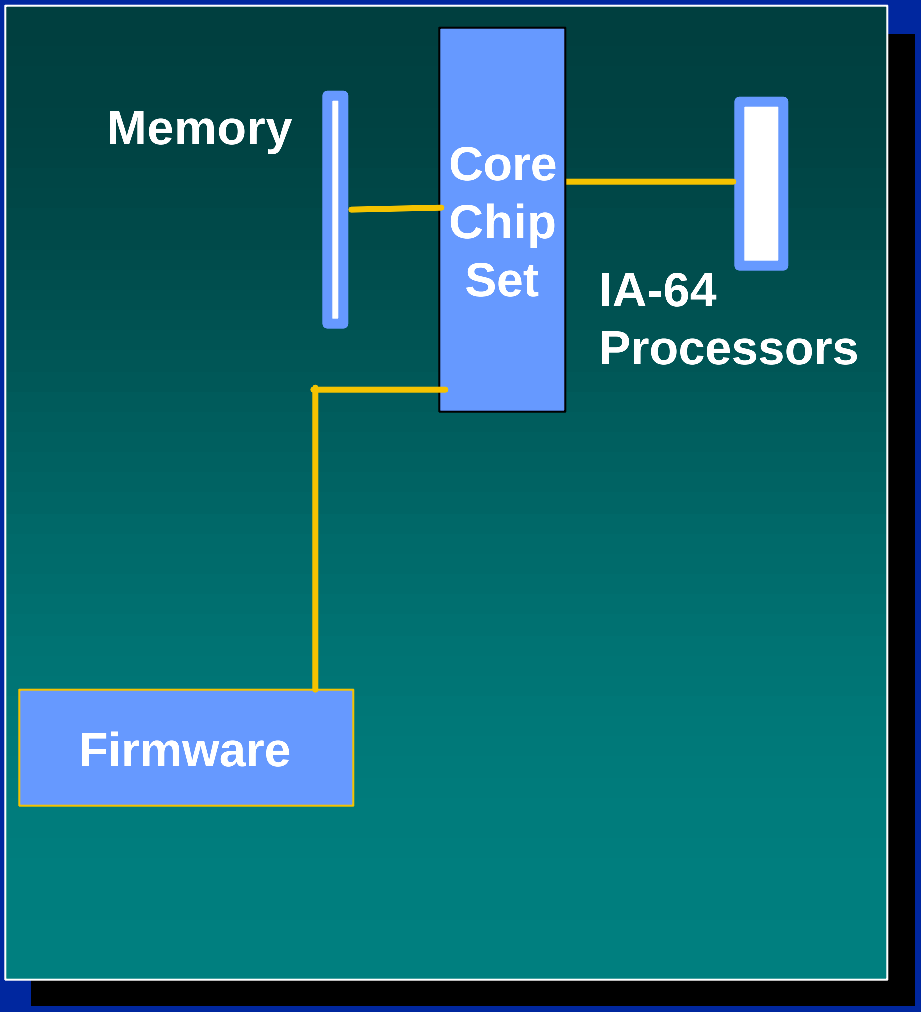
Memory



Guideline:

- System should provide for at least 256MB of system memory per attached processor

Firmware

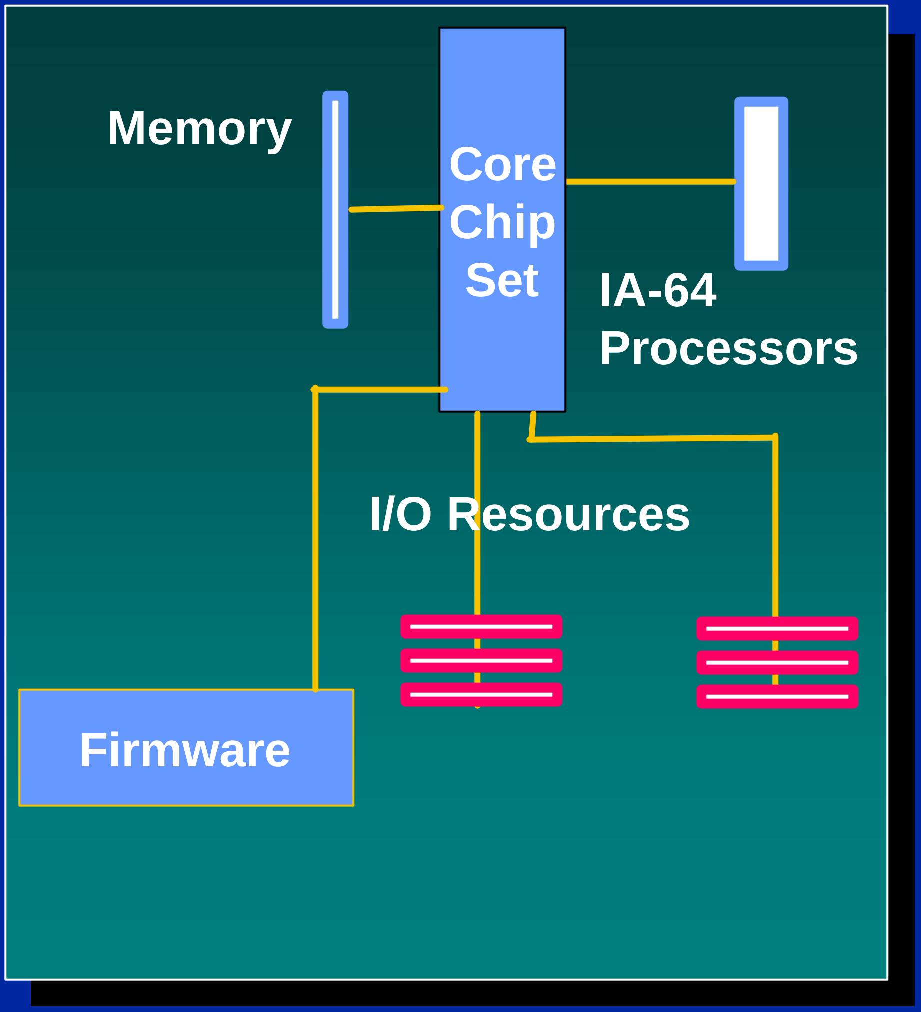


Guidelines:

- Use EFI (Extensible Firmware Interface) compatible firmware components for IA-64 servers

intel®

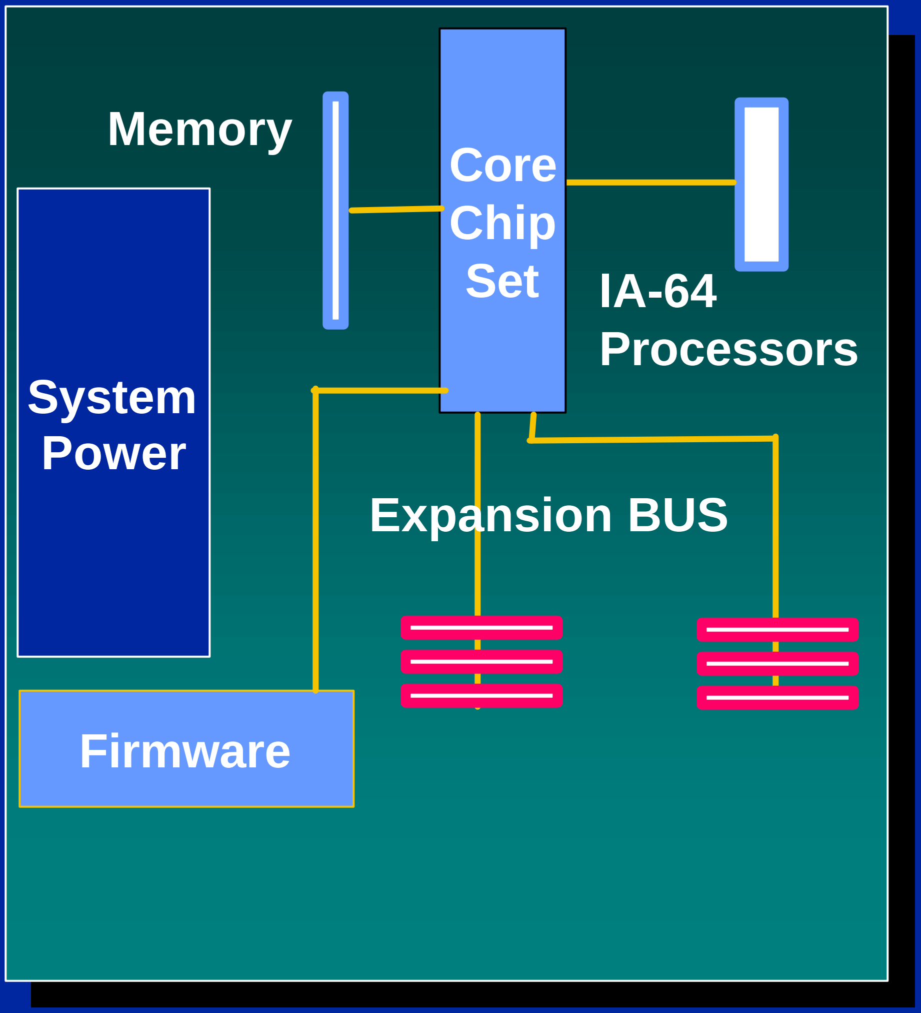
I/O Resources



Guidelines:

- The hardware implementation of all expansion busses must provide a means for bus agent discovery and identification
- Use base address registers (BARS) and extent registers to assign system resources

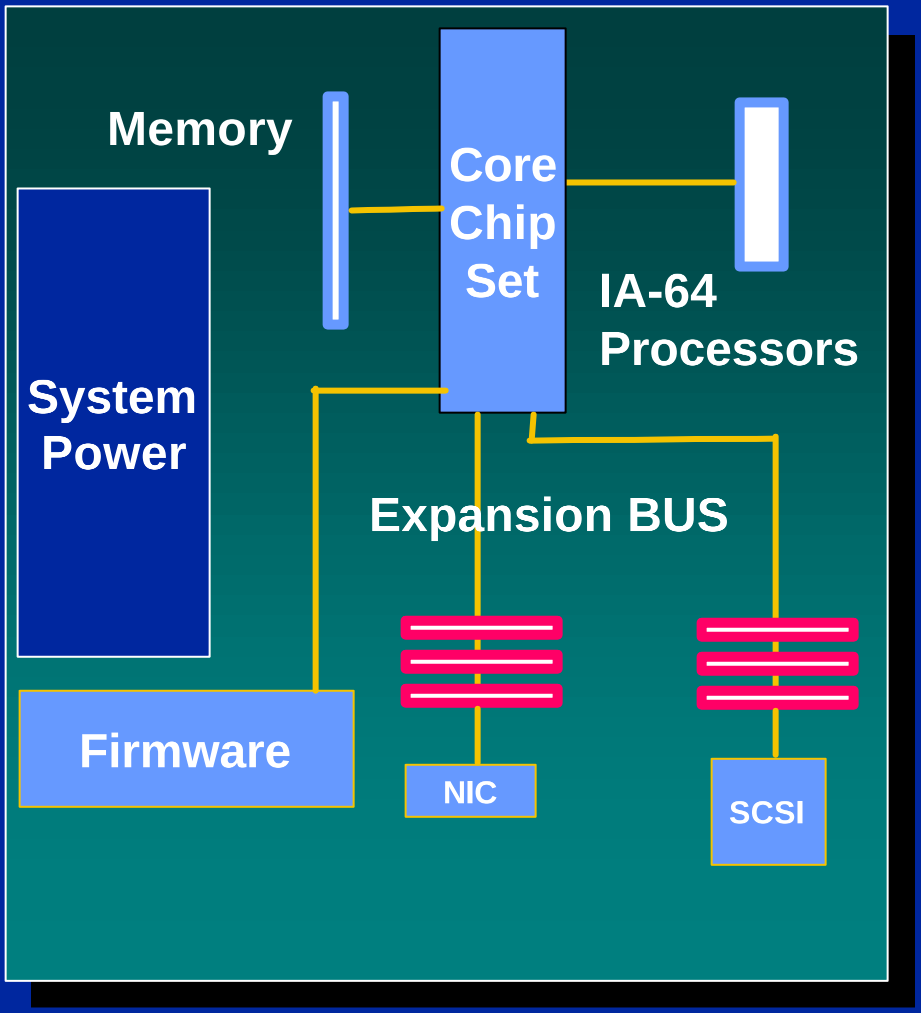
System Power Considerations



Guideline:

- The platform must monitor the status of the primary power source
- The primary power subsystem should provide hot plug capability
- The primary power subsystem should provide redundancy

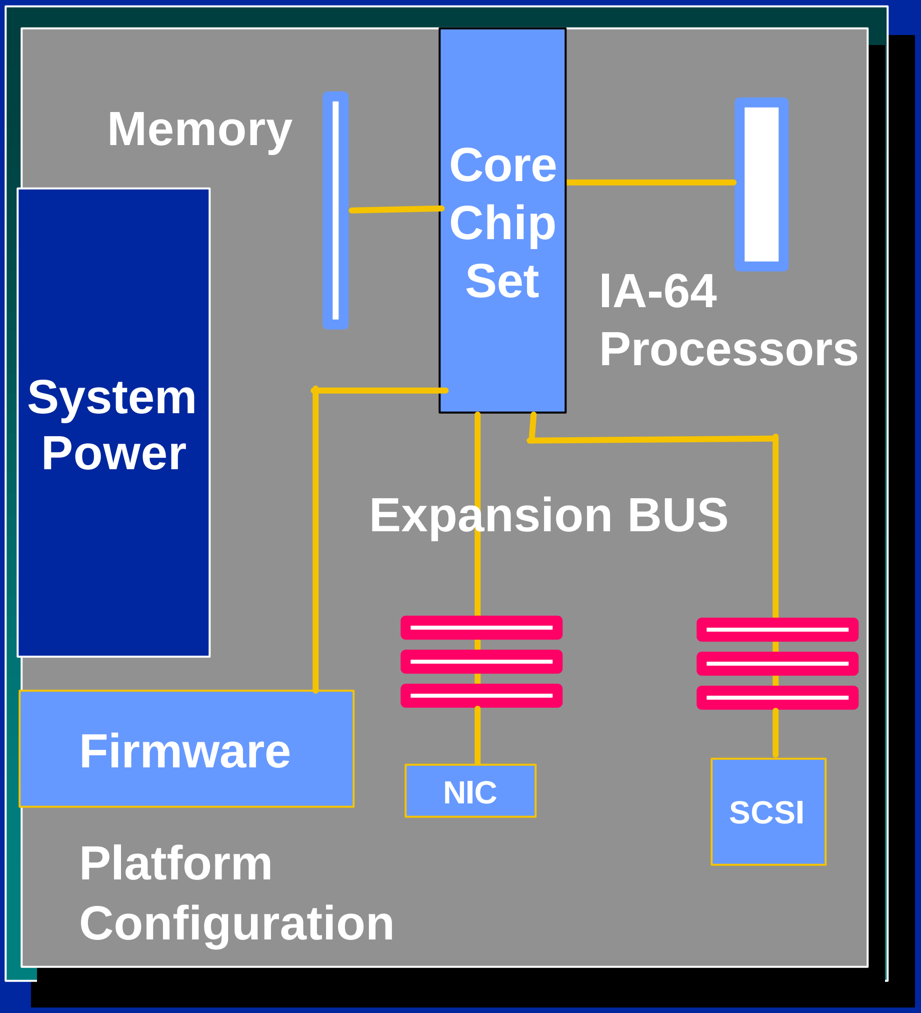
Networking, Storage and Management



Guidelines:

- Net adapters must be able to transmit packets from buffers aligned on any boundary
- Option ROMs should use the EFI implementation
- The platform must provide the capability to log all critical events

Platform Configuration

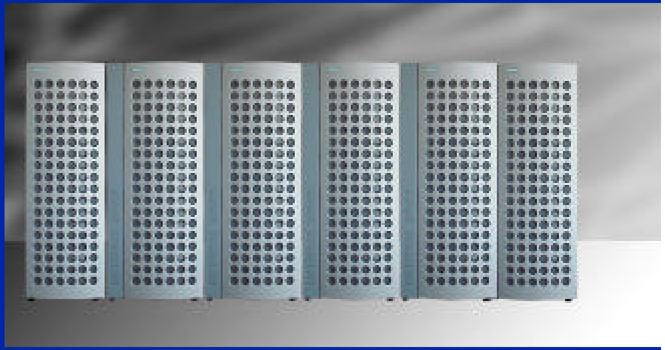


Guideline:

- Provide ACPI hardware for system configuration
- ACPI control methods to configure devices must be included
- Support ACPI sleep states : S0 and S5.
- Recommended to support additional states S2 through S4

Conclusions

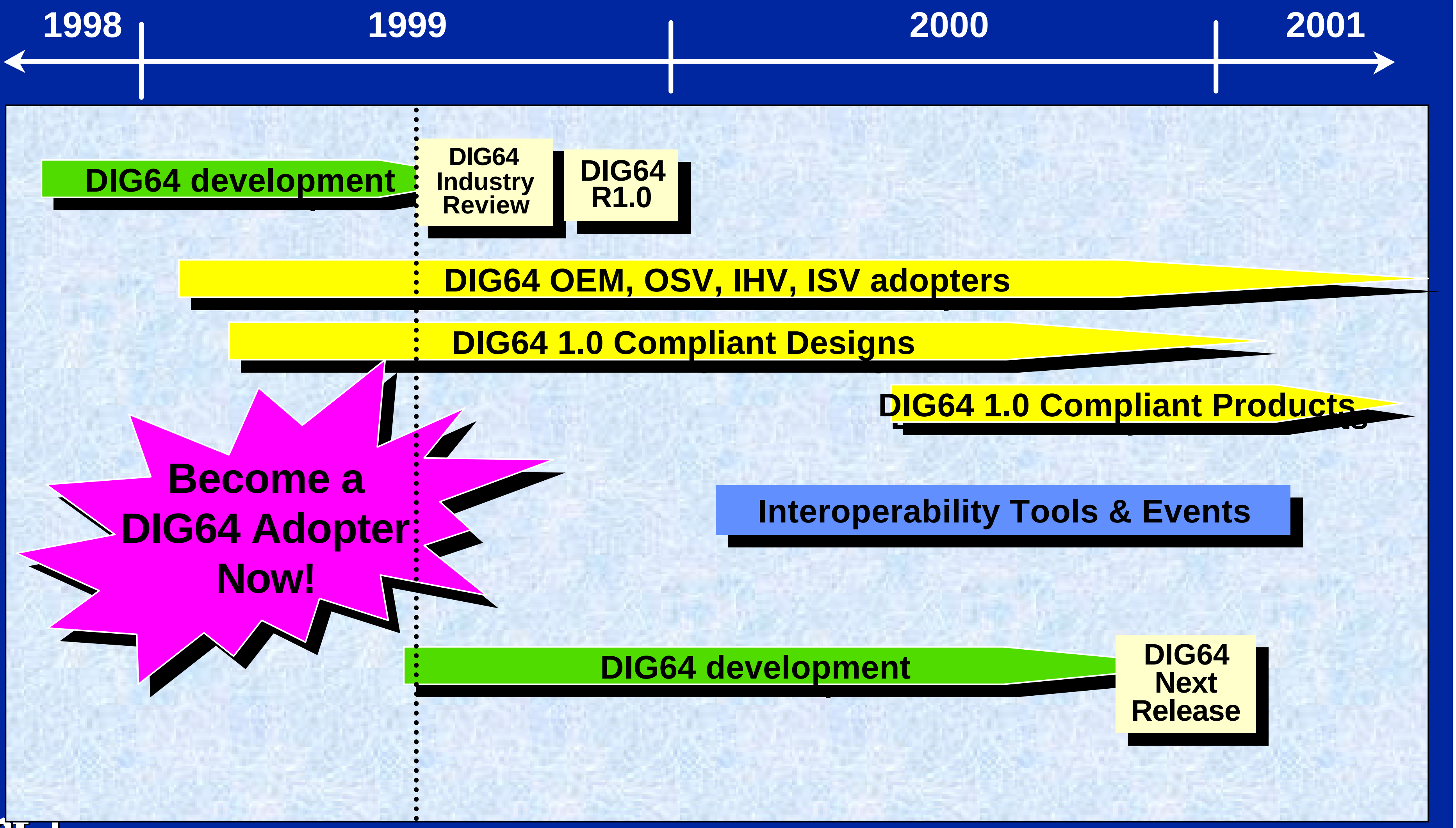
- **Goals supported by core components**
  - Common system model which promotes interoperability across various OS's
  - Make migration of applications to IA-64 as easy as possible
  - Enable technology migration
- **DIG64 will intersect the first generation of IA-64 system**
- **Many compliant systems will be available at launch of Itanium™**



Agenda

- Introduction to DIG64
- Guidelines for Migration of Legacy Technology
- Guidelines for Firmware
- Guidelines for Core System
- Getting Involved with the DIG64

DIG64 Target Timeline



1998

1999

2000

2001

DIG64 development

DIG64 Industry Review

DIG64 R1.0

DIG64 OEM, OSV, IHV, ISV adopters

DIG64 1.0 Compliant Designs

DIG64 1.0 Compliant Products

Become a DIG64 Adopter Now!

Interoperability Tools & Events

DIG64 development

DIG64 Next Release

Becoming a DIG64 Adopter

- **DIG64 Adopters Receive:**
  - Early access to future drafts of DIG64 Guidelines
  - Participation in DIG64 interoperability events
  - Promotion of compliant products on DIG64 web site and at DIG64 events
  - Access to DIG64 Adopters web site and tools
- **What’s required:**
  - Execution of DIG64 Adopters Agreement (one per company)

DIG64 Guidelines Now Available for Industry Review

To review the DIG64 Guidelines document and give your input to the DIG64 working group, go to:

http://dig64.org

Industry review period runs from September 1 through October 31, 1999