



UDI, a Uniform Driver Interface

By

Project UDI

Kevin Quick, Interphase, Chairman

Agenda

- **What and who are Project UDI?**
- Why portable drivers?
- What will UDI do for me?
- UDI Architectural Overview
- Related Activities
- Status and Futures
- Q & A



What is UDI

UDI is a specification designed to address the Divergence, Inconsistencies, and Deficiencies of current Device Driver implementations and environments.



Who Is Project UDI?

- **Open industry group**
- **Multiple platform and operating system vendors**
Including Compaq, HP, IBM, Intel, SCO, Sun
- **Multiple IHVs**
Including Adaptec, AMI, Bit3, Interphase
- **Integrators and solutions providers**
Including Storage Technology Group (STG) and Lockheed-Martin
- **Broad e-mail reflector community**



Project UDI Goals

- **Design and promote standardized driver interfaces**
 - Portability plus advanced features
- **Publish UDI Specifications**
 - Defines binary and source packaging as well as driver programming interfaces
- **Produce reference implementations**
 - Sample drivers and OS implementations
 - » including Linux, UnixWare, HP-UX, Digital UNIX
 - Validates specification and eases porting effort
 - Jointly developed by Project UDI participants



Agenda

- What and who are Project UDI?
- **Why portable drivers?**
- What will UDI do for me?
- UDI Architectural Overview
- Related Activities
- Status and Futures
- Q & A



The “Driver Problem”

(IHV Perspective)

- **IHV**s have huge matrix of drivers to develop/port
 - # Devices ´ OSeS ´ OS versions ´ platforms ´ features
- **Finite** development and support resources
 - Must choose porting order (target prioritization)
 - Some OSeS and/or platforms not supported
- **Driver** porting not directly related to core business
 - Technology, Performance, Functionality

The “Driver Problem”

(Platform and OS Vendor Perspective)

- **Platform and OS vendors need IHV device support**
 - Need broad device technology coverage to compete
 - Limited internal resources for driver development
 - Must leverage IHV partners
 - Competitive advantage lost if too far down porting order
- **Driver porting not directly related to core business**
 - Features and services
 - Enabling applications
 - Reliability, Availability, Scalability

The “Driver Problem”

(Independent Timelines)

- **OS and platform evolution not under IHV control**
- **IHV device evolution not under OSV control**
- **Independent development**
- **Requires stable, well-defined interfaces**

The UDI Solution

- **The Uniform Driver Interface enables 100% portability**
 - **UDI Specification defines all driver interfaces**
 - **Abstracts H/W and S/W environment**
 - **Not specific to any group of OSes**
- **Single driver source for all UDI-compliant OSes**
- *Co-exists with legacy driver environments*



Key to Portability

- **No OS or platform specific components in driver**
- **No OS policy in driver**
- **All interfaces defined**
- **Room to grow**
 - **OS implementation flexibility w/o changing interfaces**
 - **Support for field-installable extensions**
 - » **Can be supplied by third parties**

Agenda

- What and who are Project UDI?
- Why portable drivers?
- **What will UDI do for me?**
- UDI Architectural Overview
- Related Activities
- Status and Futures
- Q & A



UDI IHV Benefits

- **Portable drivers across platforms & operating systems (no changes to driver source)**
 - Single, fully specified, cohesive interface specification
- **Simpler drivers**
- **More time/resources for core business**
 - I/O technology innovation and implementation
 - Performance & functionality enhancements



UDI System Vendor Benefits

- **Increased device & I/O technology coverage**
 - Leverage drivers developed for other platforms
 - Amortized IHV support costs
- **UDI environment implementation flexibility**
 - High-reliability environments
 - Validation environment
 - Performance environment
- **More time/resources for core business**



UDI: Next-Generation Technology

- **Single model for all device types**
 - “Mixed-model” FibreChannel, USB, etc.
- **Instance independence**
 - Hot plug/hot swap adapters and devices
- **Location independence**
 - Distributed environments and I/O processors
 - No overhead for “local” case
- **Implicit MP synchronization (no lock primitives)**



UDI as Technology Enabler

- **UDI simplifies support for:**
 - **Future platforms (e.g. IA64)**
 - **Mixed-endian platforms and arbitrary bus hierarchies**
 - **Multi-path I/O**
 - **Real-time Environments**
 - **User-mode drivers, advanced driver debugging tools, and validation environments**



Agenda

- What and who are Project UDI?
- Why portable drivers?
- What will UDI do for me?
- **UDI Architectural Overview**
- Related Activities
- Status and Futures
- Q & A

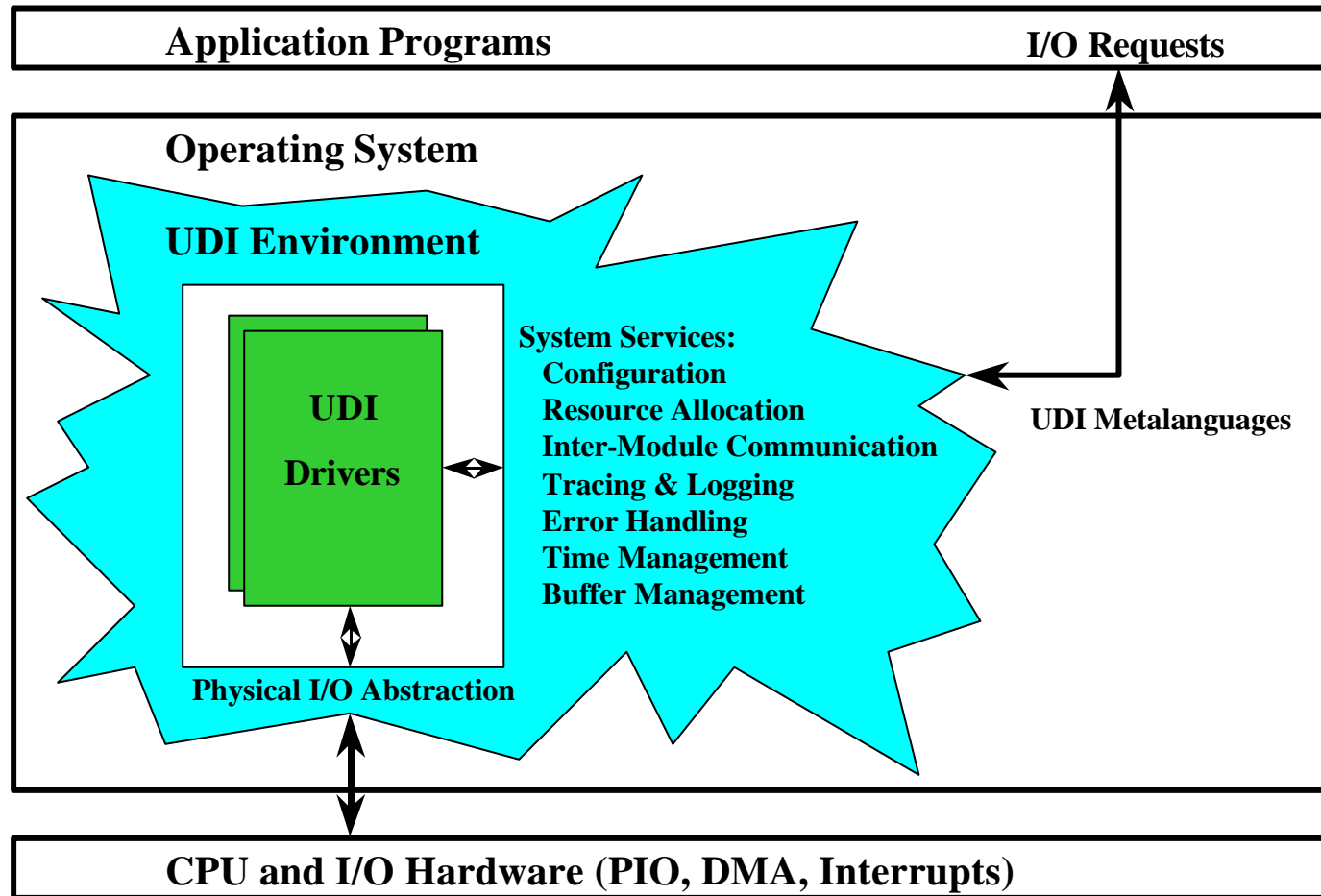


The UDI Technology

- **Complete driver encapsulation**
- **Single model for all device types**
 - Extensible to new device types
- **Instance independence**
- **Non-blocking execution model**
- **Implicit MP synchronization via “regions”**
- **Inter-module communication via “channels”**
- **Explicit resource ownership and transfer**



UDI Fully Encapsulates Drivers



UDI Functional Areas

Core

- Basic data types
- Execution model
- Communication (IMC)
- Configuration
- Memory management
- Buffer management
- Time management
- Tracing and logging
- Distribution format

Physical I/O

- DMA and PIO
- Interrupt handling
- Bus bridges

Metalanguages

- SCSI
- Networking
- Pointer
- Other

Bus Bindings

- PCI
- VME
- Etc.

ABI Bindings

- Binary representation

UDI System Services

- **System interface & resource management**
 - Implemented for all UDI environments
 - Provides services typically associated with the OS
- **Non-blocking allocation model**
 - Callbacks used wherever allocation might occur
 - Control blocks provide synchronization and context management

UDI Regions

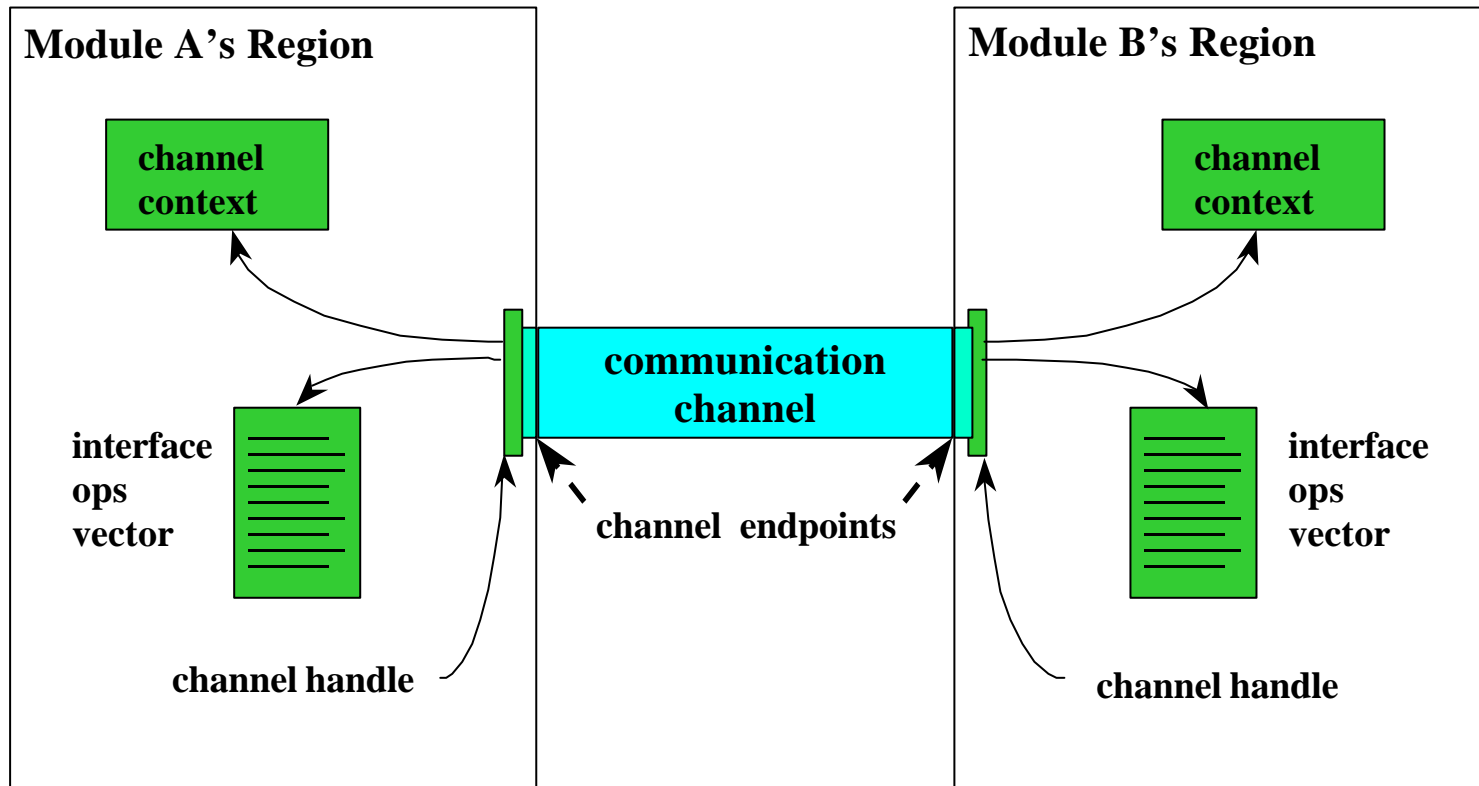
- **Basic unit for execution and scheduling**
 - **Non-blocking with implicit synchronization**
 - » Each call into the driver region is serialized
 - » “Pseudo-threads” interleaved between callbacks
 - **No direct data sharing between regions**
 - » *You have to go through channels*
- **Region attributes (e.g. priority hints) specified at build time**
- **One or more driver regions per device instance**
 - **Each region may operate in a different domain**
 - » e.g. address space, CPU group, network node

UDI Channels

- **Used for Inter-Module Communication (IMC)**
- **Control blocks sent over channels**
- **Strongly typed function-call interface**
- **Bi-directional**



UDI Channel Communications

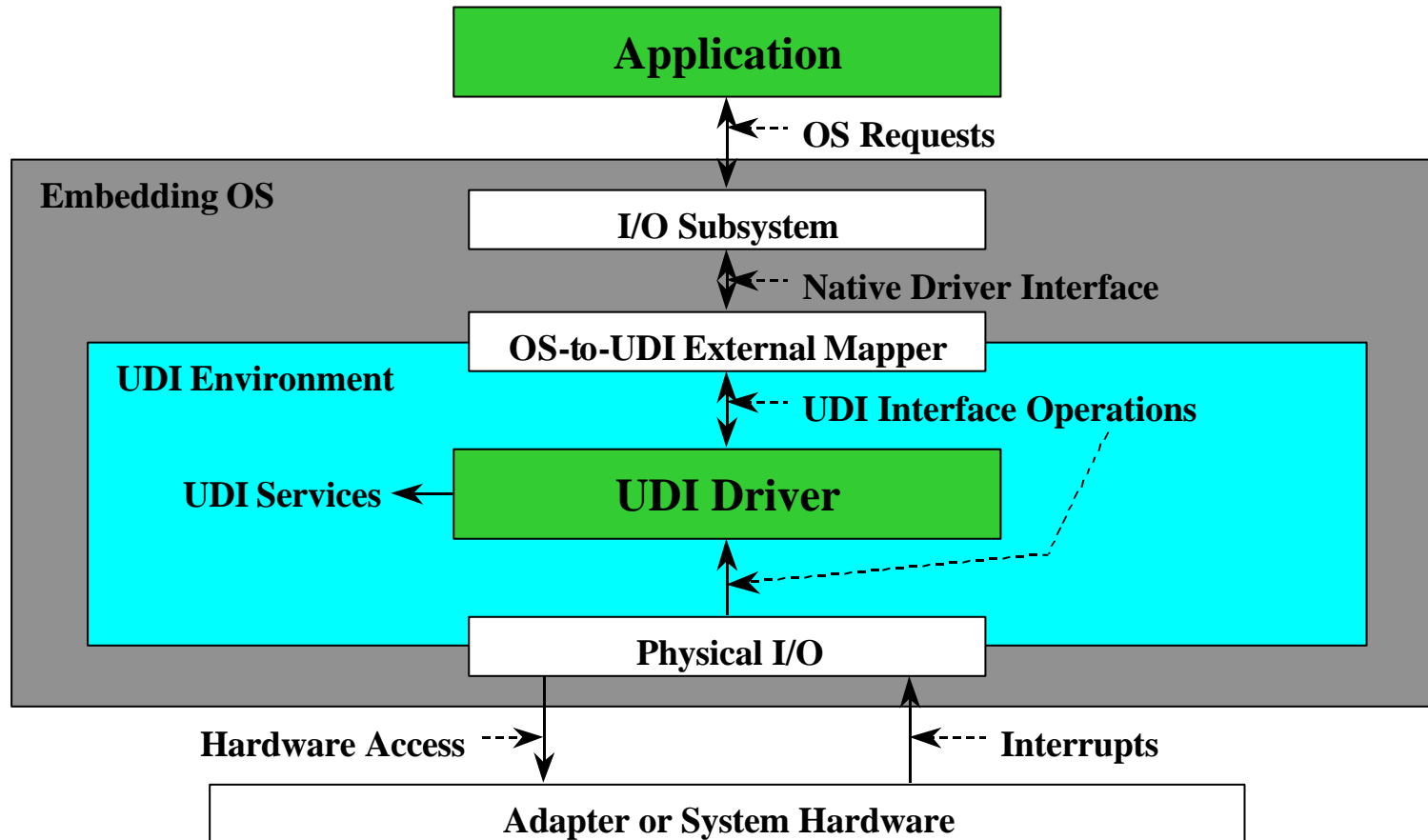


UDI Metalanguages

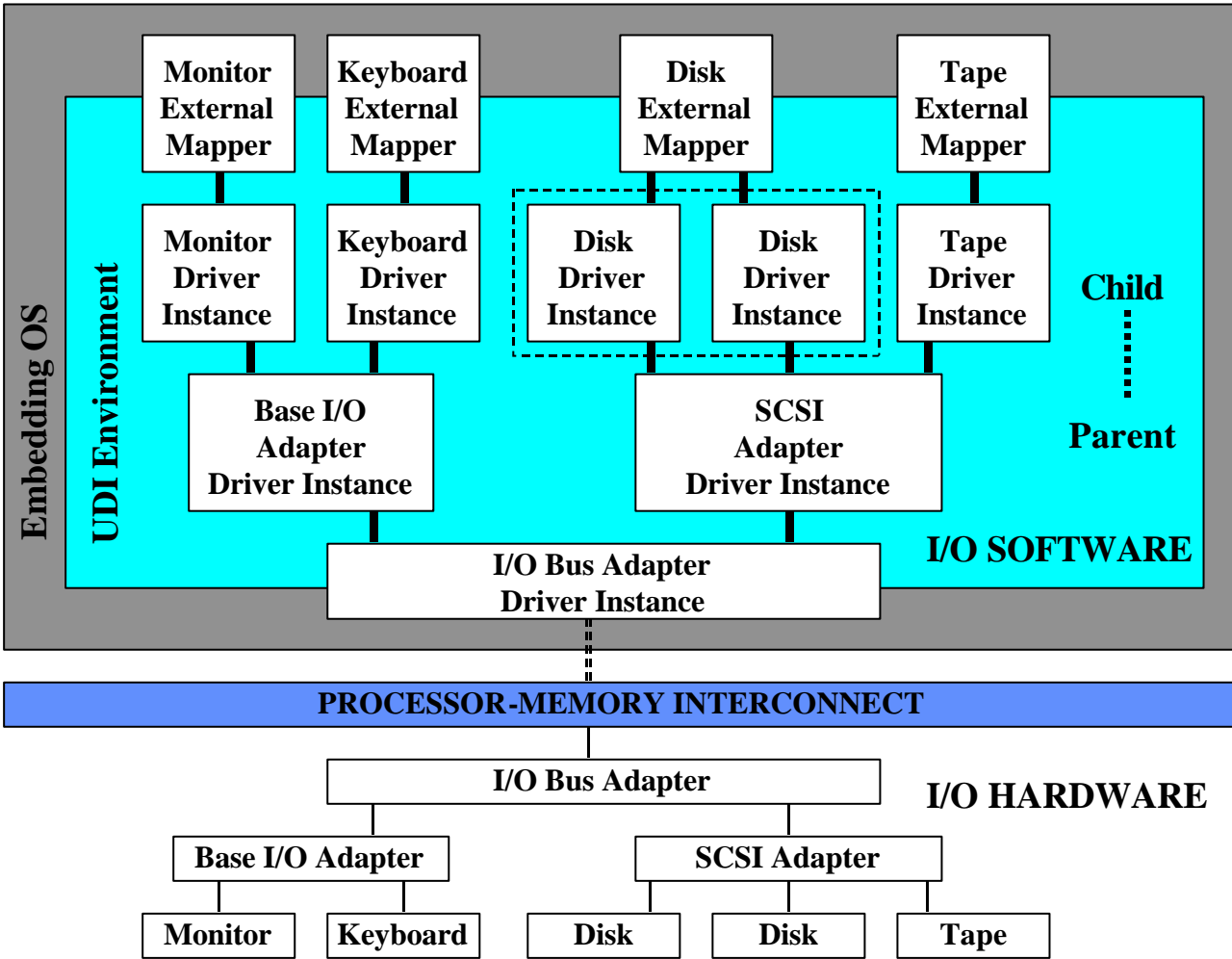
- **Device-type specific communication**
- **Defines communication paradigm between cooperating modules**
 - **Number and type of channels**
 - **Interface operation types on each channel**
 - **Operations and sequences to implement technology-specific functionality**
- **Analogous to SCSI CAM, DLPI, etc.**



Path From Application to UDI Driver



Example Driver Hierarchy



Control Block Tracking

- **Control Blocks represent I/O Operations**
 - Exchanged across channels
 - Basic request/command/response unit
 - » Extended in a metalanguage specific fashion
- **Environment Tracking**
 - Origin encodes OS policy in UDI-neutral manner
 - Origin preserved and replicated throughout I/O chain
 - Environment OS-specific implementation tracking
 - » Priorities, Quotas, etc.

Environment Protection

- **Different environments have different levels of trust in drivers**
- **UDI environments can:**
 - **detect misbehaved drivers (e.g. bad pointers, state violations, etc.)**
 - **track resource ownership and transfers**
 - **abruptly terminate (“region-kill”) driver instances**
 - » **Frees all resources**
 - » **Shuts down device**

Agenda

- What and who are Project UDI?
- Why portable drivers?
- What will UDI do for me?
- UDI Architectural Overview
- **Related Activities**
- Status and Futures
- Q & A



Related Activities

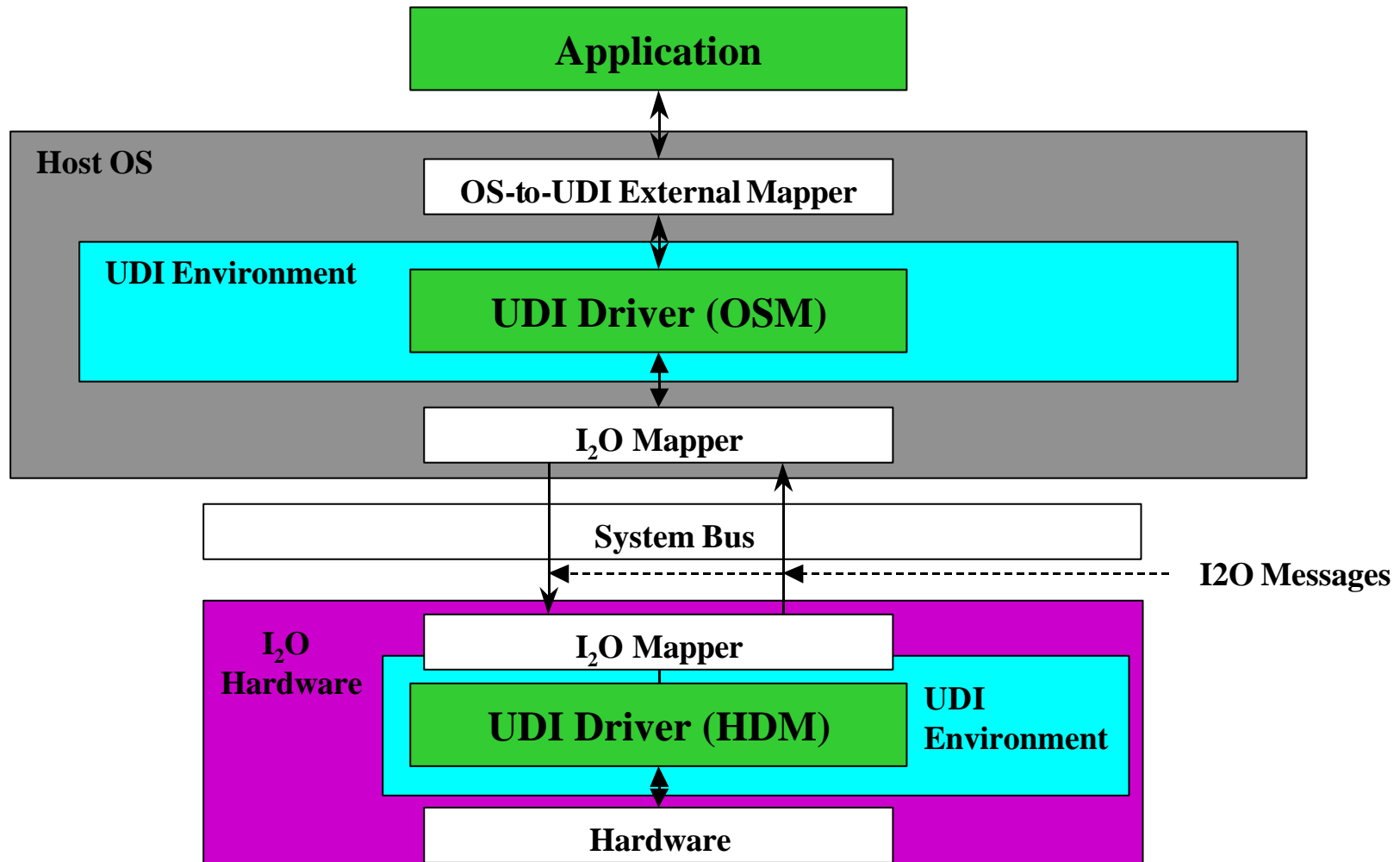
- **USB drivers**
 - **UDI forms basis of OpenUSBDI (USB driver interface) to be standardized by the USB Device Working Group**
- **Developing a UDI Metalanguage to support the I₂O messaging protocol**
- **SCI Metalanguage in progress**
- **UDIG (UNIX Developer's Interface Guide for Intel Servers)**
 - **IA32 and IA64 UDI ABI's**



(More) Related Activities

- **Real-Time Support**
 - ANSI R1.1 Technical Committee
 - Developing recommendations for use of UDI in real-time operating systems
- **“HomeGate” residential gateways**
 - JTC1/SC25/WG1
- **Graphics Drivers**
 - GGI re-basing onto UDI (www.ggi-project.org)

Example UDI & I₂O Combination



Agenda

- What and who are Project UDI?
- Why portable drivers?
- What will UDI do for me?
- UDI Architectural Overview
- Related Activities
- **Status and Futures**
- Q & A



UDI Documents

Normative

- UDI Core Specification
- UDI Physical I/O Specification
- UDI SCSI Driver Specification
- UDI Network Driver Specifications
- UDI Metalanguage Library Specification

Informative

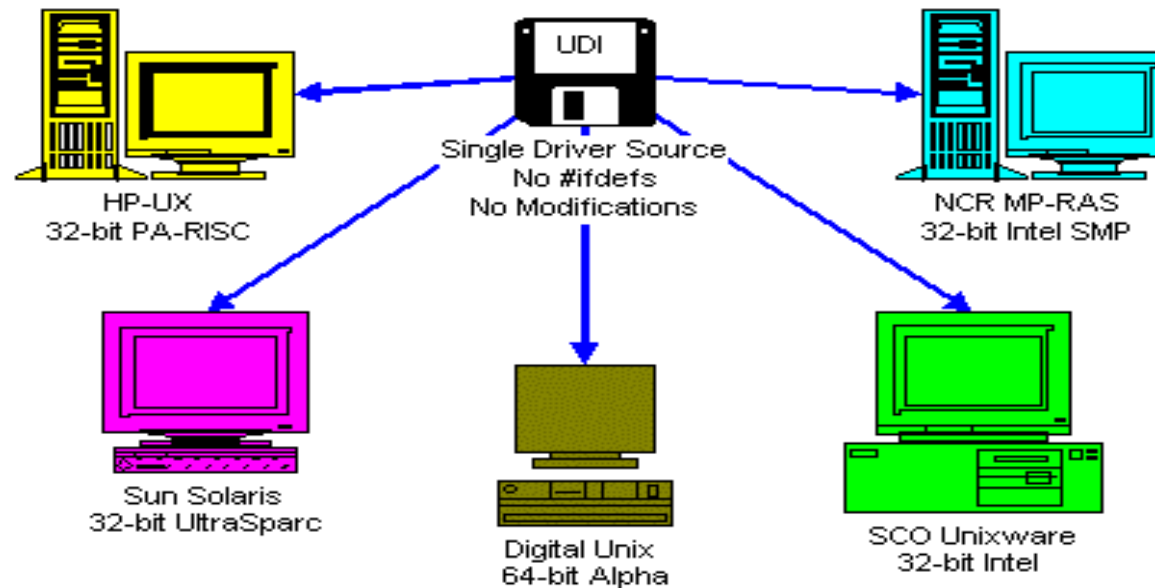
- UDI Technical Overview/White Paper
- UDI Marketing Information
- UDI Implementation Guides

Specification Schedule

- **Revision 1.0 Finalized**
 - **Corrections Document Available**
 - » **Revision 1.01 w/Corrections at completion of Prototype Implementation**
 - **Downloadable from UDI web page**
- **Future Extensions Planned**
 - **New Metalanguages and Device support**

UDI In Action

- **Prototype completed 12/9/97**
 - Based on revision 0.75; updated to 0.80 in '98
 - Adaptec SCSI, Interphase NIC
 - Six OS ports on various CPUs and platforms



Reference Implementation

- **Linux reference port being developed by Intel and STG**
- **Other OS environments in Progress**
 - **UnixWare 7.1 (available in Beta)**
 - **Monterey**
 - **AIX**
 - **Solaris**
 - **HP-UX**
 - **Open Server/5**
- **1.0 reference source to be released publicly**



UDI Information

Web page

<http://www.project-udi.com/>

Project UDI contacts

Chair: Kevin Quick, +1 214 654 5173, kquick@iphase.com

Vice Chair: Mark Evenson, +1 408 447 5601, mevenson@cup.hp.com

Secretary: John Lee, +1 650 786 5323, john.lee@eng.sun.com

Editor: Kurt Gollhardt, +1 908 790 2277, kdg@sco.com

Advisor: Mark Bradley, +1 303 684 4753, markb@btc.adaptec.com



Agenda

- What and who are Project UDI?
- Why portable drivers?
- What will UDI do for me?
- UDI Architectural Overview
- Related Activities
- Status and Futures
- **Q & A**

