Welcome

To Advance through Presentation
Use Page Up and Page Down Keys


# USB Introduction

Jai Chulani
Technology Manager

# Introduction

- USB is now mainstream across the Macintosh product line
- USB provides great advantages to both developers and users
- Stick to the USB 1.1 Specification

# What We've Accomplished. . .

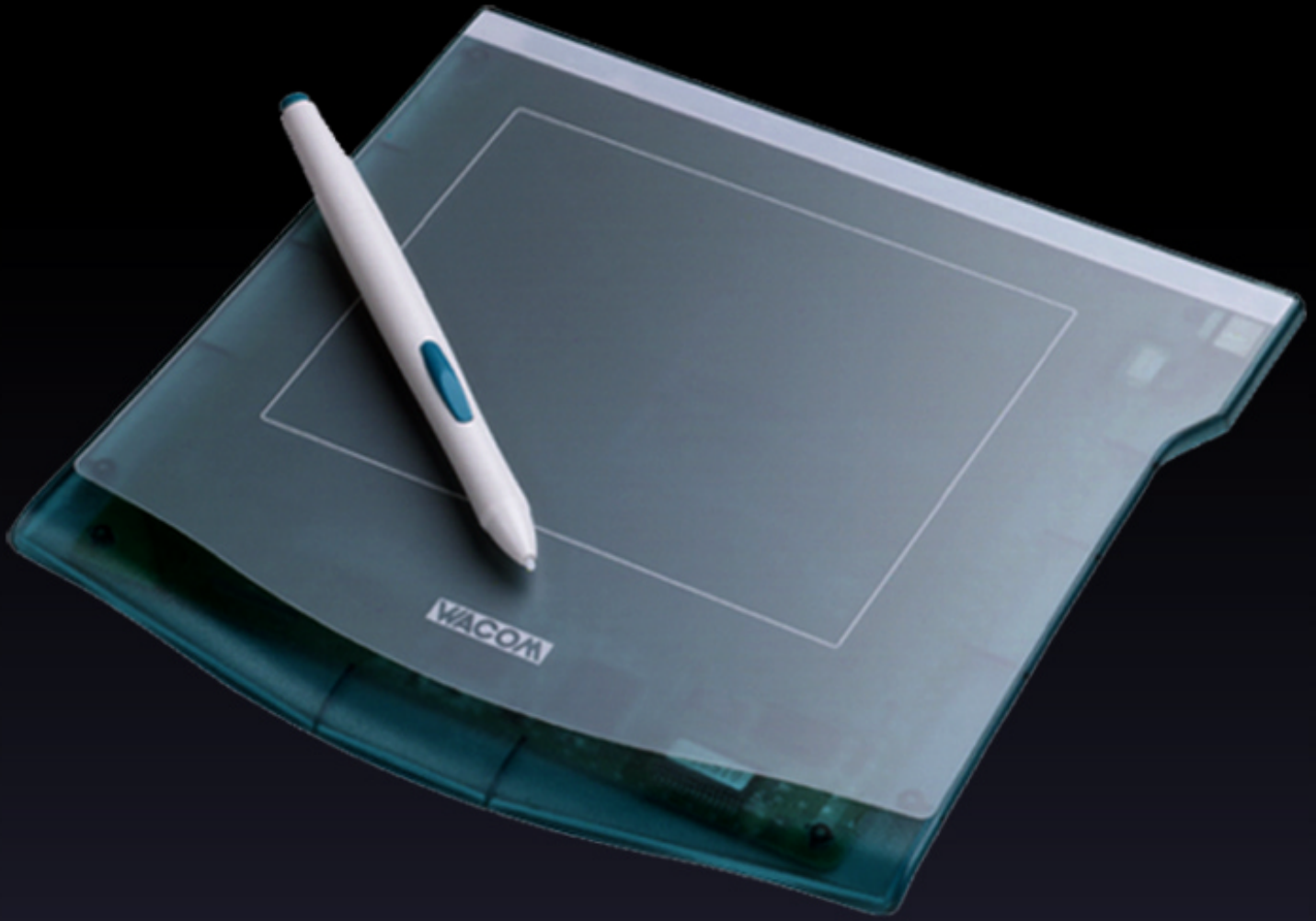
- Brought a new I/O Technology to the Macintosh
- Redefined the way DDKs are delivered to the developers
- Redefined the way Apple S/W engineering works with developers
- Raised the bar on DDK source code content

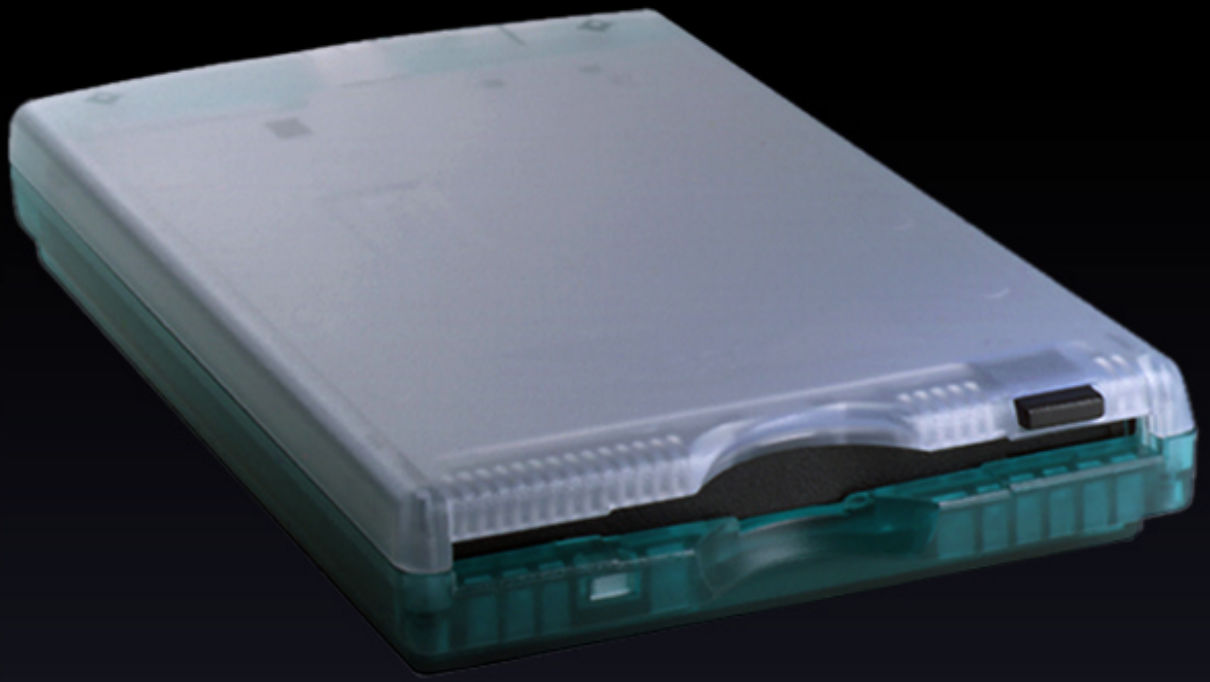
# From 0 to 175 Devices in 12 Months Flat!

- Enhanced mouse drivers
- Floppy, SuperDisk, and ZIP drives
- Cameras
- Flash and Smart Card Readers
- Printers and Printer adapters
- Serial adapters

# From 0 to 175 Devices in 12 Months Flat! (Cont.)

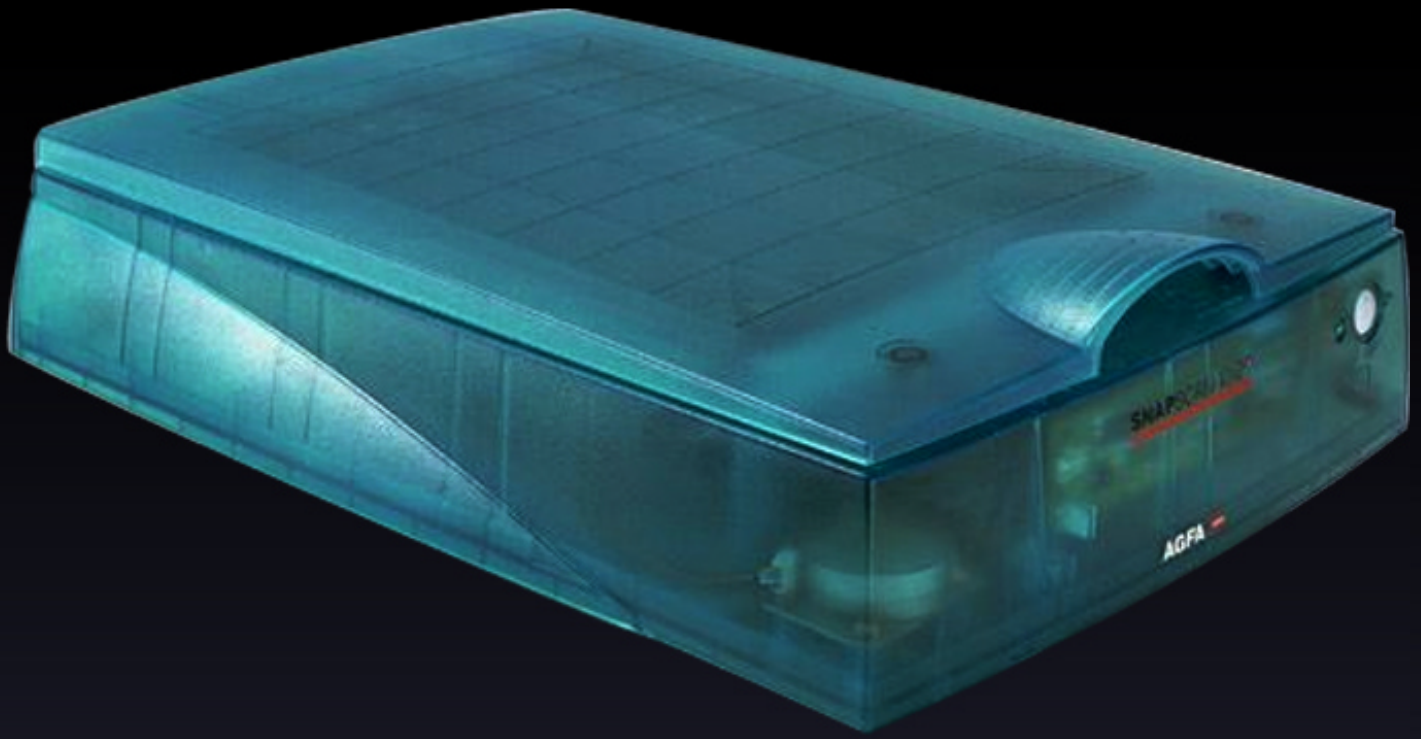
- Game controllers
- Modems
- Monitor and display control
- Barcode scanners
- MIDI adapters
- TV Tuners and video capture

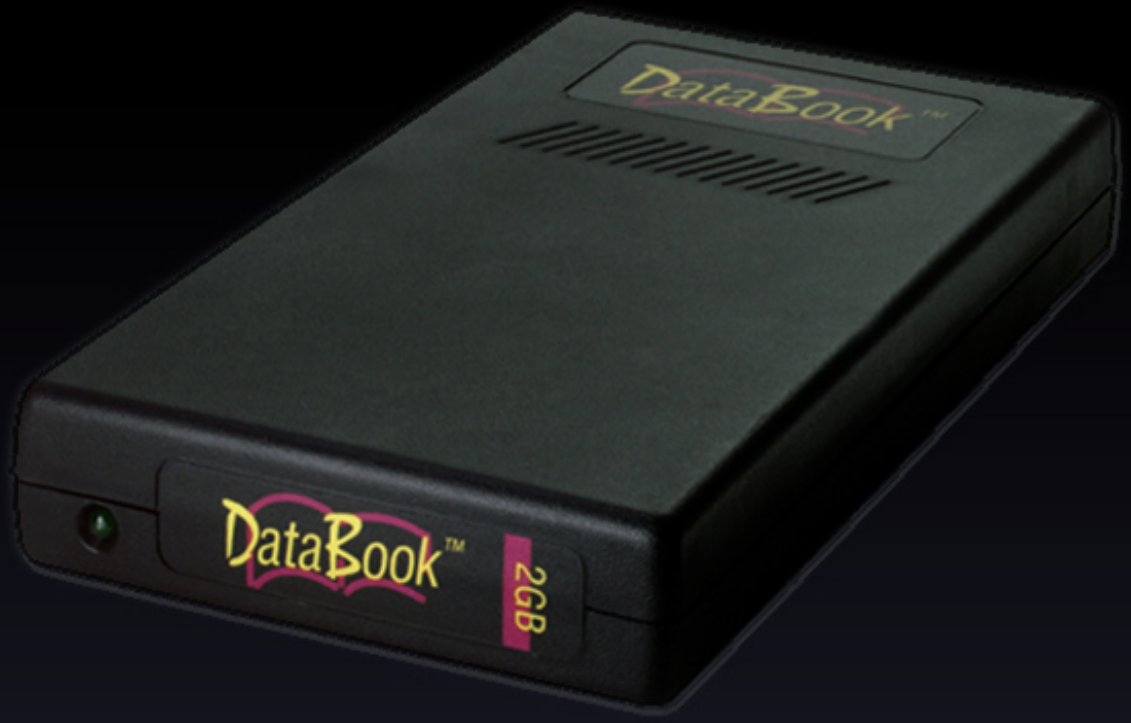

WACOM



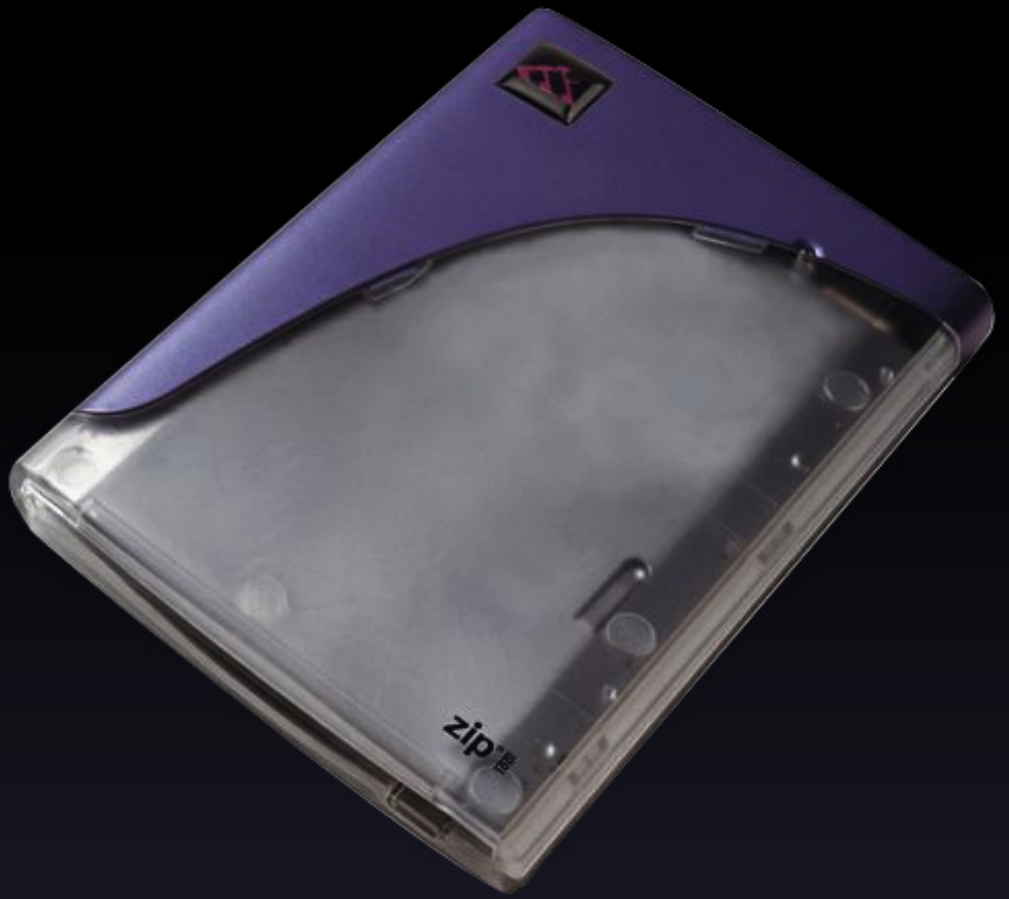

iomega
zip


CYBORG 3D
USB
F1
F2
F3


AGFA


DataBook™
DataBook™
2GB


FUSION
THRUSTMASTER


zip

# What You Will Learn

- USB Overview
- Mac OS USB Software Architecture
- Resources available to you


# USB Overview

Craig Keithley
Technology Manager

# USB Overview

- What is USB?
- The Mac OS USB Architecture
- USB and the Mac OS—Present
- USB and the Mac OS—Future
- Demonstrations
- Futures

# Origins of USB

- USB Specification created by a group of companies
- USB-IF manages USB plugfests and DWGs
- Device Working Groups refine and revise device class specifications
- http://www.usb.org

# USB Implementers Forum

- Referred to as the “USB-IF”
- Assigns USB Vendor IDs
- Organizes plugfests and other USB events
- USB Developers should join and participate in the USB-IF

# USB Device Class Specifications

- Provide detailed specifications for each class of device
- Available for a variety of device classes
- Apple follows these specifications when developing standard Apple drivers
- Your devices should follow these specifications

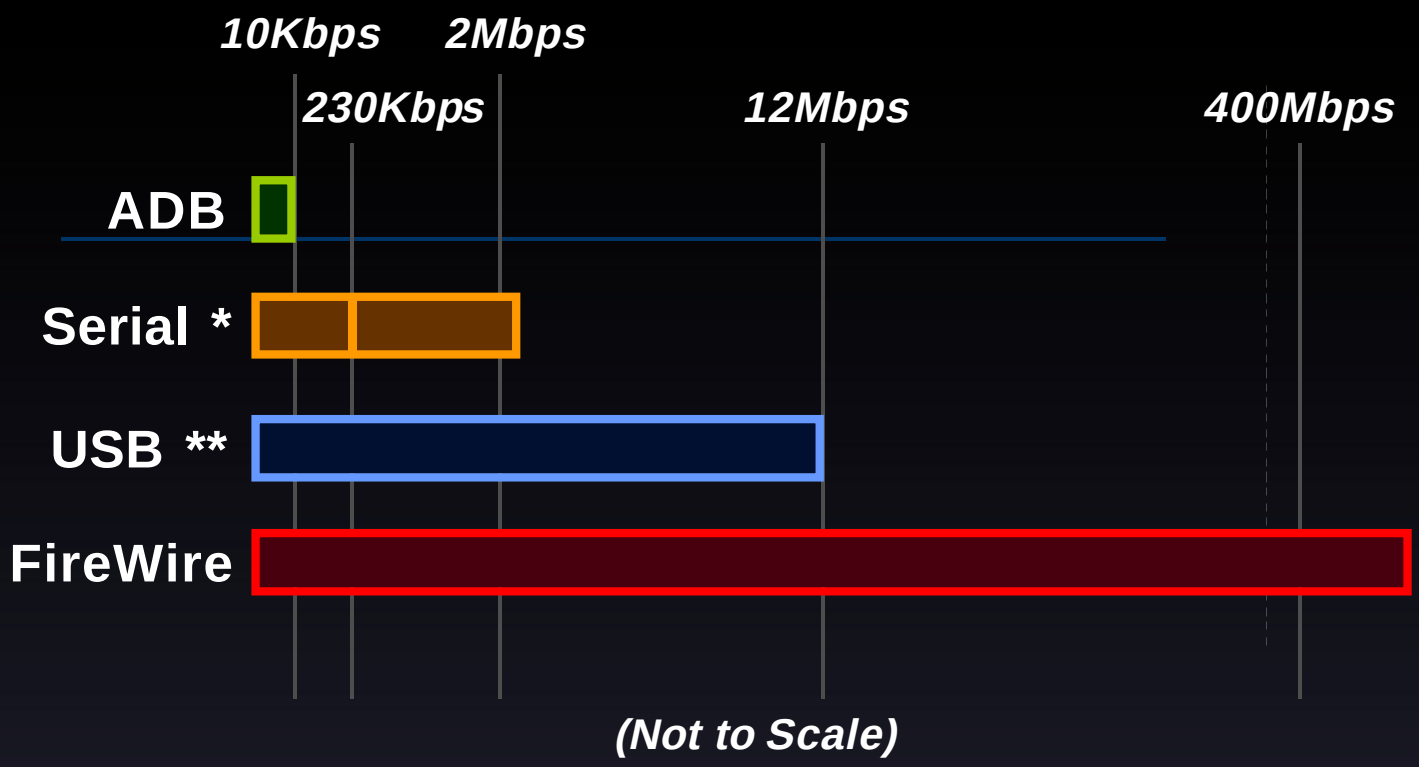
# USB Device Classes

- Hubs
- Human Input
- Printers
- Audio
- Mass Storage
- Communication
- Monitors
- Imaging
- Power
- IrDA Bridge

# What Is USB?

- Is reasonably fast (1.5Mbps and 12Mbps)
- Supports up to 127 devices per bus
- Devices are plug & play
- Devices must plug into “hubs” and cannot be daisy chained
- Bus supplies enough power that many devices don’t need AC adapters

# Performance




*(Not to Scale)*

*Externally clocked serial devices, such as GeoPort, can reach 2Mbps

**USB supports 1.5Mbps and 12Mbps modes

# USB Fundamentals

- Two types of devices
  - Vendor specific
  - Class specific
- Communicating with devices
- Devices get power from…
  - The bus
  - A wall adapter

# Types of USB Devices

- Vendor specific
  - Defined by the vendor
- Class specific
  - HID, Composite, and Hub are defined by the USB Specification 1.1
  - Other devices defined by the USB-IF device class specifications

# Device Descriptors

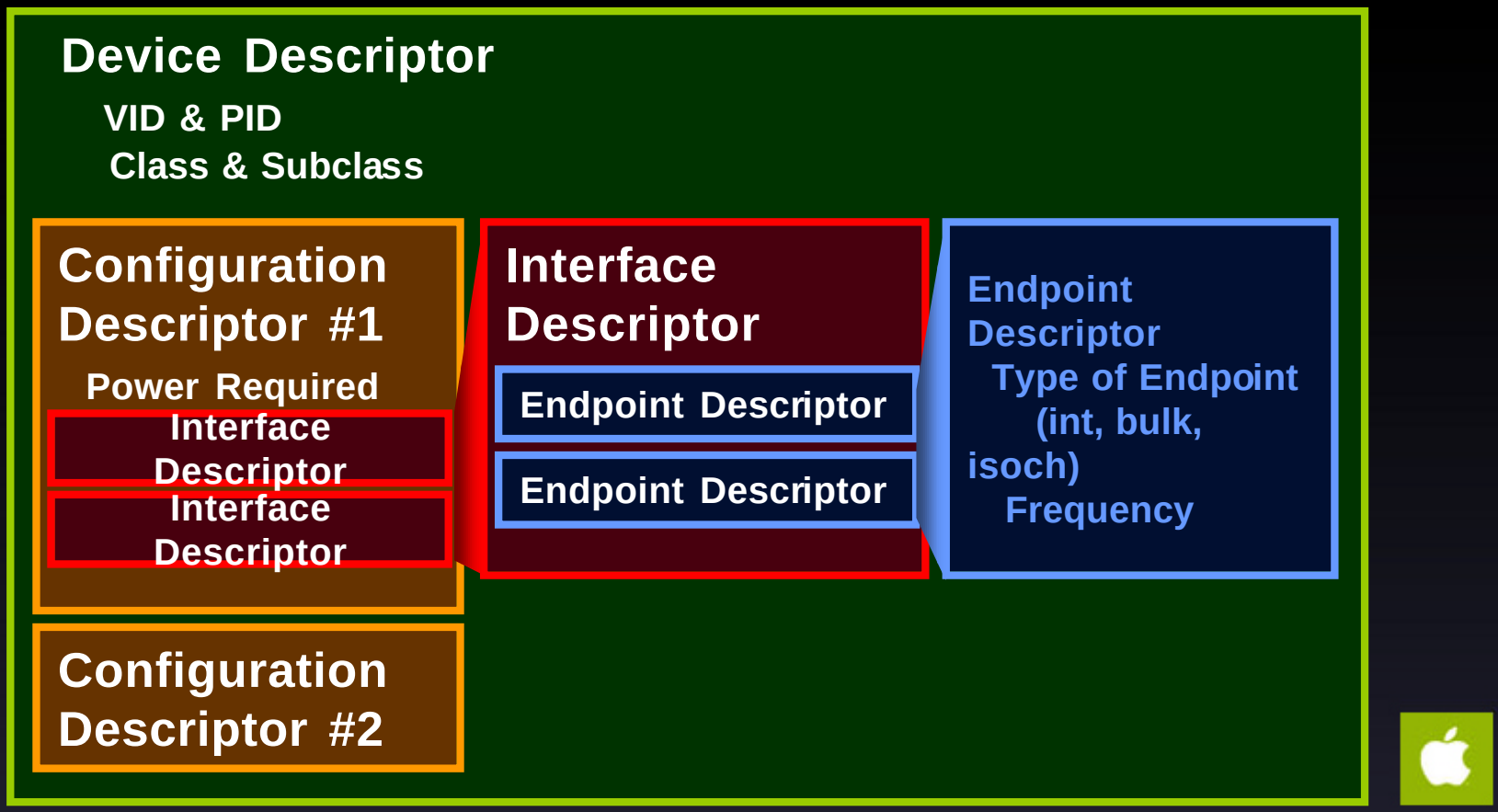
- A Device Descriptor
  - Vendor and Product IDs
  - Device Class and Subclass
  - Number of configurations
- One or more Configuration Descriptors
  - Power required for each configuration
  - Interfaces in the configuration

# USB Devices Also Contain. . .

- Logical connections called “endpoints”
- All devices contain a “control endpoint”
- Each interface contains one or more endpoints

- *Note:* the word “pipe” is used when referring to the communication channel between the host and endpoint

# USB Device Descriptors




# Demo

## USB Prober

# Powering Devices

- Ports on self powered hubs provide 500ma to each device
- Ports on bus powered hubs provide 100ma to each device
- The hub in the Apple USB Keyboard is “Bus Powered”
- *Note*: “Self Powered” means the hub has it’s own power supply, and doesn’t get power from the bus

# Suspend and Resume

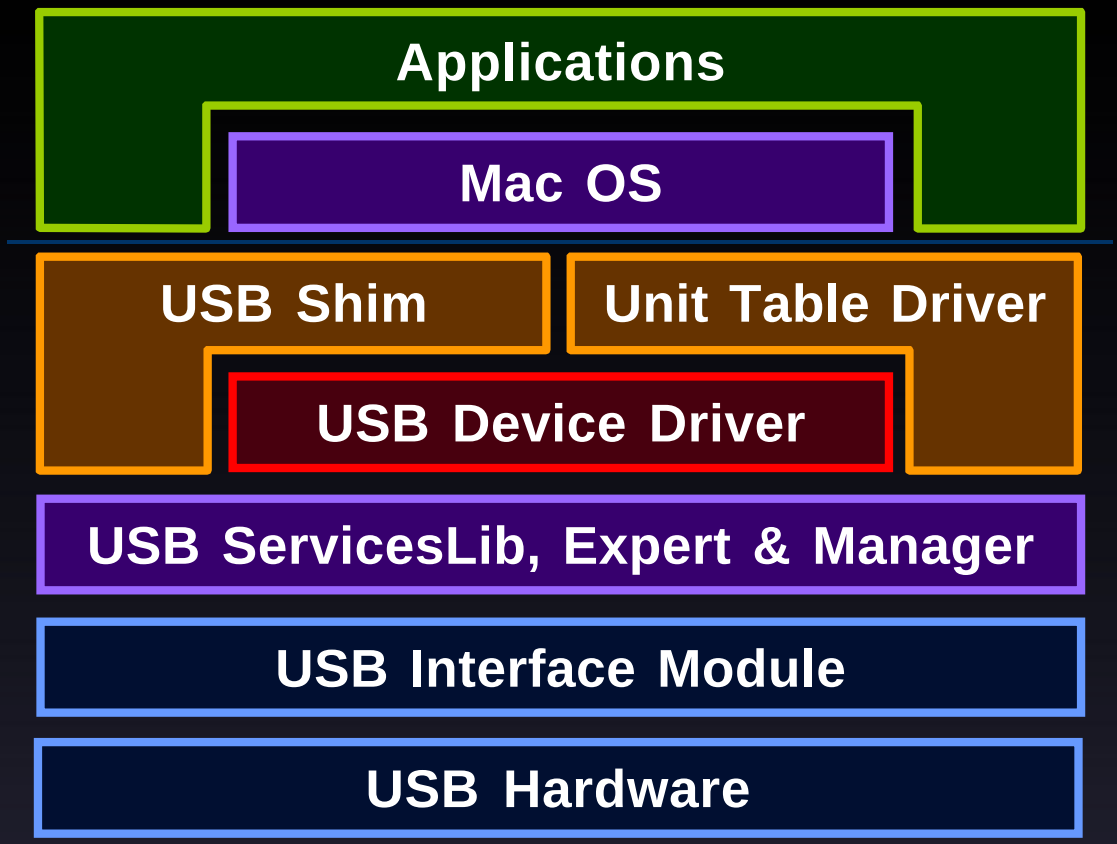
- Suspend places the device in a non-operational mode that consumes less than 50µA
- Devices needing attention may alert the host by placing a resume signal on the USB bus
- Support for Suspend & Resume is planned for Mac OS USB 1.3


# Mac OS USB Architecture

# Mac OS USB Architecture

- USB Manager and Expert
- USB Services Library
- USB Interface Module (“UIM”)
- Standard Apple Drivers
  - Hub and Composite
  - Keyboard and mouse
  - Printing Support

# USB in the Mac OS



# USB Manager and Expert

- Handle requests by the hub and composite drivers to load class drivers
- Examines drivers and determines if they should be loaded
- Provides notification services that Shims can use to learn when devices have been connected or disconnected

# USB Driver Loading

- The Hub driver…
  - Detects a device attach and “enumerates” the device
  - Asks the USB Expert to load a driver
- The USB Expert loads a matching driver
- The USB Driver is initialized
- The USB Expert sends device notifications

# USB Driver Matching

- Drivers contain data that describe what types of devices they support
- The Expert uses this data to determine if the driver is Vendor Specific
- Vendor Specific drivers take precedence over Standard Apple Drivers

# USB Driver Internals

- Driver Description Structure
- Driver Dispatch Table
- Driver code

# Driver Description Structure

- Vendor and Product ID of driver
- Device Class, Subclass, and Protocol
- Interface Class, Subclass, and Protocol
- Driver loading options indicate. . .
  - If the driver is Vendor Specific
  - If the driver should only load as an interface driver


# Demo

## USB Prober

# USB Services Library

- APIs used by class drivers
- Covered in more detail in Session 204 —USB In Depth

# Driver Code

- USB Drivers are *not* Unit Table Drivers
- USB Drivers do not have predefined entry points (besides the dispatch table)
- Additional entry points defined by vendor specific dispatch tables, or by class specific dispatch tables

# USB Shims and Unit Table Drivers

- Shims are intended to act as a layer between legacy code and USB drivers
- Shims can be written to install a unit table driver
- Installing a Unit Table Driver is appropriate when you need to emulate a legacy driver

# Apple Recommends That. . .

- Devices meet the USB Specifications
- Devices contain the developer's Vendor ID, not the chip maker's Vendor ID
- Devices have Manufacturer, Product name, and Serial Number strings

# Vendor Specific Drivers

- Write Vendor Specific drivers only when the Standard Apple Drivers don't support your device's feature set
- Use Shims *or* Unit Table Drivers to provide an interface between applications and USB drivers
- Consider a "no-restart" installation


# Mac OS USB: Present

# Mac OS USB—Present

- Mac OS USB 1.2
- USB Printing Support
- Desktop Printing Utility supports many PostScript™ USB Printers
- Mac OS USB 1.3 in development

# Mac OS USB 1.2

- Isochronous read and write
- Printing Class supported
    - Currently named “USBPrintDriver”
- Interrupt write now supported
- Numerous bug fixes

# USB Printing Support

- A new class driver for Mac OS USB 1.2
- Supports devices which conform to the USB Printing Spec 1.0
- Requirements
    - Print responds to the “Get 1284 Capability String” request
    - The 1284 Capability String complies with the 1284 specification

# Desktop Printing Utility

- Supports many PostScript™ USB Printers
- Allows users to create desktop icons for USB PostScript™ Printers

# Developing for PowerBooks

- Mac OS USB 1.2 turns devices off during PowerBook sleep
- Mac OS USB 1.2.1 adds support for USB PC Cards
- Mac OS USB 1.3 is expected to suspend and resume devices when in PowerBook sleep

# Handling PowerBook Sleep

- Update your USB driver to recognize sleep messages in the notification routine
- Be aware that a Sleep Demand message means that the Mac OS *will* perform a device disconnect
- Notify the user that they should unmount the volume when sleeping

# Mac OS USB 1.3

- Current plan is that devices will be suspended during PowerBook sleep
- This would allow USB device drivers to suspend their device to conserve power
- Be aware that desktop systems may someday suspend devices to conserve power


# Mac OS USB: Future

# Mac OS USB—Future

- More Standard Apple Drivers
  - Mass Storage Devices
  - Communication Devices
  - Audio Devices
  - HID Library and InputSprocket
  - Net Download
  - USB in Portables

# Mass Storage Class

- USB Mass Storage support extension supports the following devices:
  - RBC (Reduced Block Command) devices
  - UFI (USB Floppy Interface) devices
  - SFF-8070i (Atapi Floppy) devices
  - SCSI Transparent command set devices

# Mass Storage (Cont.)

- Support for standard HFS, Extended HFS, and DOS Volumes
- Support for single and multiple partitioned disks
- Handles auto and manual eject devices
- Does not support CD or Tape drive devices

# Communication Class

- USB Communication Support extension
- Supports USB Comm Device Class 1.1
- Abstract Control Model
- V.25ter (Hayes)
- Legacy applications via Serial Shim Library

# Audio Class

- USB Audio Support extension
- Supports the following:
  - 44.1KHz @ 16 Bits
  - PCM Format
  - Stereo
- Works as a standard Sound Manager device (no special USB Sound APIs needed)

# Audio Class Requirements

- Devices should correctly report their delay in the audio class descriptors
- Important that developers not use the chip maker's Vendor and Product IDs
- Using the chip maker's Vendor ID will cause problems for Net Download

# Audio Class Requirements (Cont.)

- Vendor unique implementations are not supported
- Apple is willing to work with developers of these types of USB Audio Devices

# HID Library and InputSprocket

- A new library to support USB HID Devices
- Parses USB HID reports and report descriptors
- InputSprocket 1.7—uses the HID Library to support many USB gaming devices automatically

# Net Download

- Checks the web for a driver on hot plug of an unknown device
- User asked before the download occurs
- Apple web site contains the database
    - Developers register their drivers in this database

# Schedule. . .

- USB Mass Storage Driver—Now
- USB Communications Driver—Sonata
- USB Audio Driver—Sonata
- Net Download—Sonata
- HID Library and InputSprocket—Sonata


# Demo

## Net Download

# Final Thoughts

- Follow the USB Specifications
- Join the USB-IF
    - http://www.usb.org
- Provide product strings in your device
- Be aware of how PowerBook sleep may affect your devices
- Work on “no-restart” installs


# Getting Started

Jai Chulani
Technology Manager

# Getting Started

- USB DDK from Apple developer site
- USB mailing list
- USB web site (www.usb.org)
- New PowerMac G3 or older G3 with a PCI USB card is your development platform

# Is It Too Late To Start?

- USB Installed base is growing rapidly
- There's a huge market for devices
- Current vendors can't meet demand
- Hardware devices are cross-platform
- No driver development needed for Class-compliant devices

# Things To Do

- Register with the USB-IF and get your vendor ID
- Design your device to be class-compliant
- Use the ‘Mac’ and USB logos
- Enter your product in the Mac Products Guide




# Related Sessions

| | |
|---|---|
| **204 USB In-depth:**<br>More USB details - developing drivers | Hall A2<br>**Wed., 4:00pm** |
| **119 I/O Drivers—Mac OS X:**<br>USB on Mac OS X and I/O Kit | Hall A1<br>**Thur., 1:00pm** |
| **915 USB Feedback Forum:**<br>Tell us what you think | Hall J2<br>**Thur., 9:00am** |
| **916 FireWire Feedback Forum:**<br>Tell us what you think | Hall J2<br>**Thur., 2:30pm** |

Think different.™

Welcome

To Advance through Presentation
Use Page Up and Page Down Keys