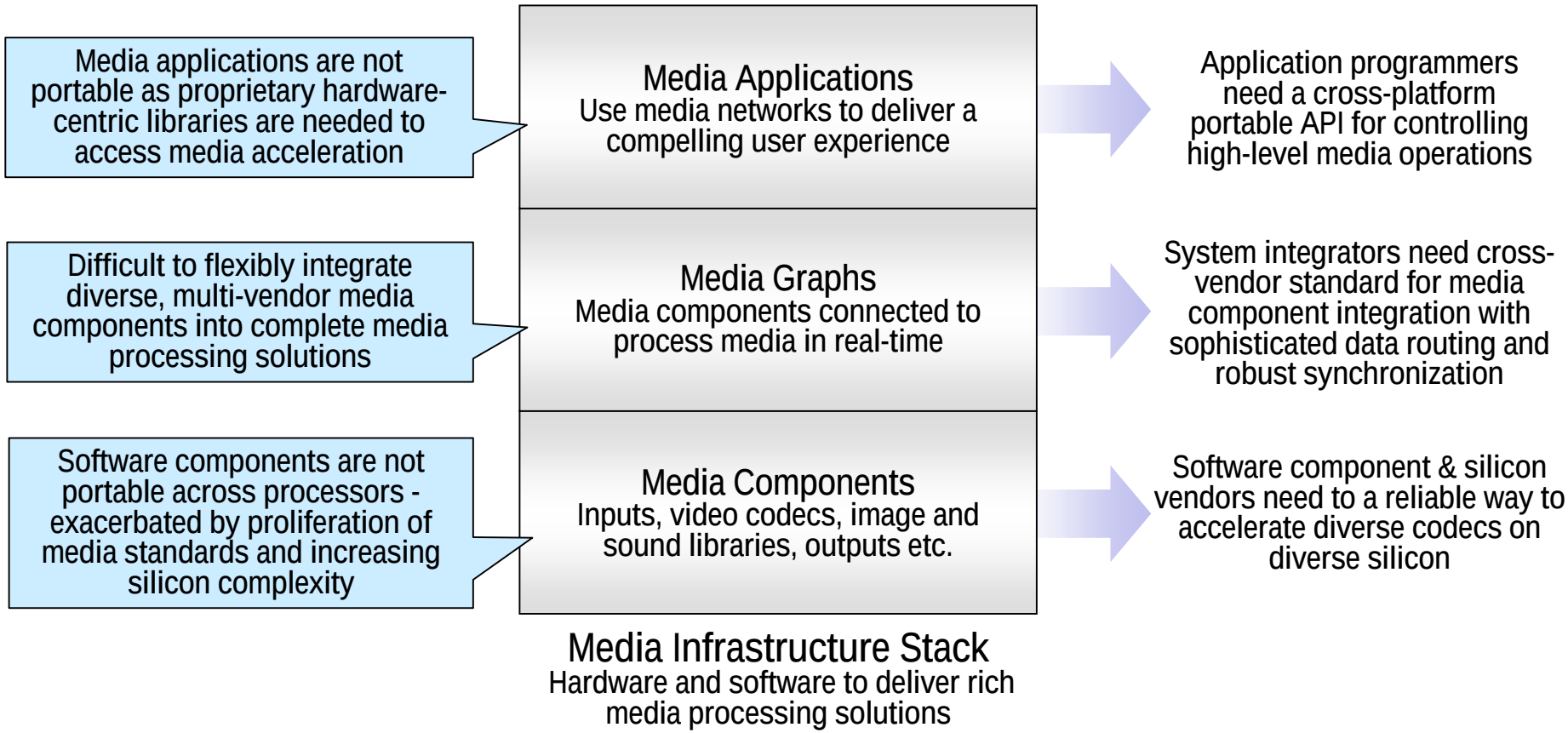
Streaming Media Portability

July 2006

© Copyright Khronos Group, 2006 - Page 1

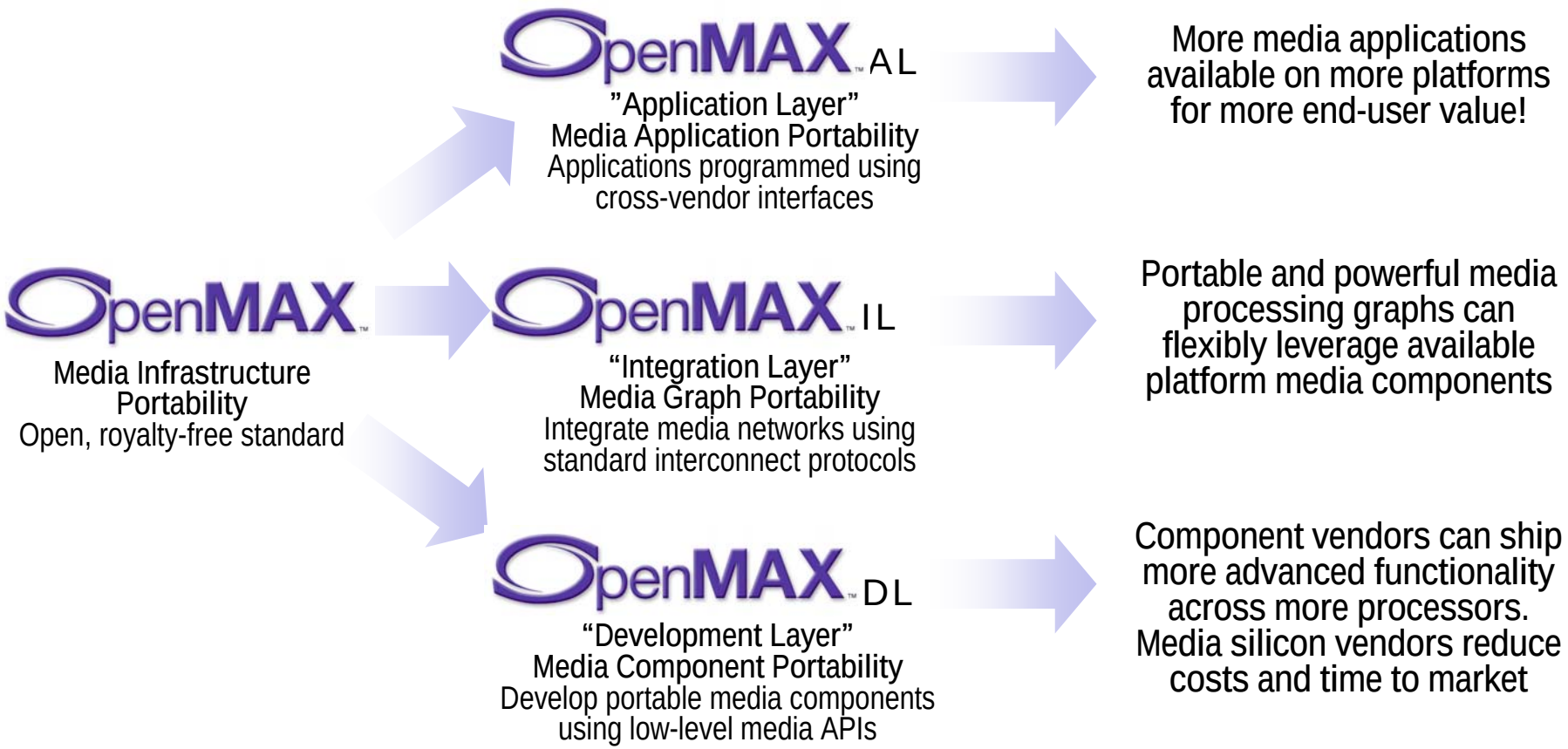
Media Portability Problem

- Media infrastructure portability is a multi-level industry problem
  - Media infrastructure is time-consuming and expensive to develop, integrate and program



© Copyright Khronos Group, 2006 - Page 2

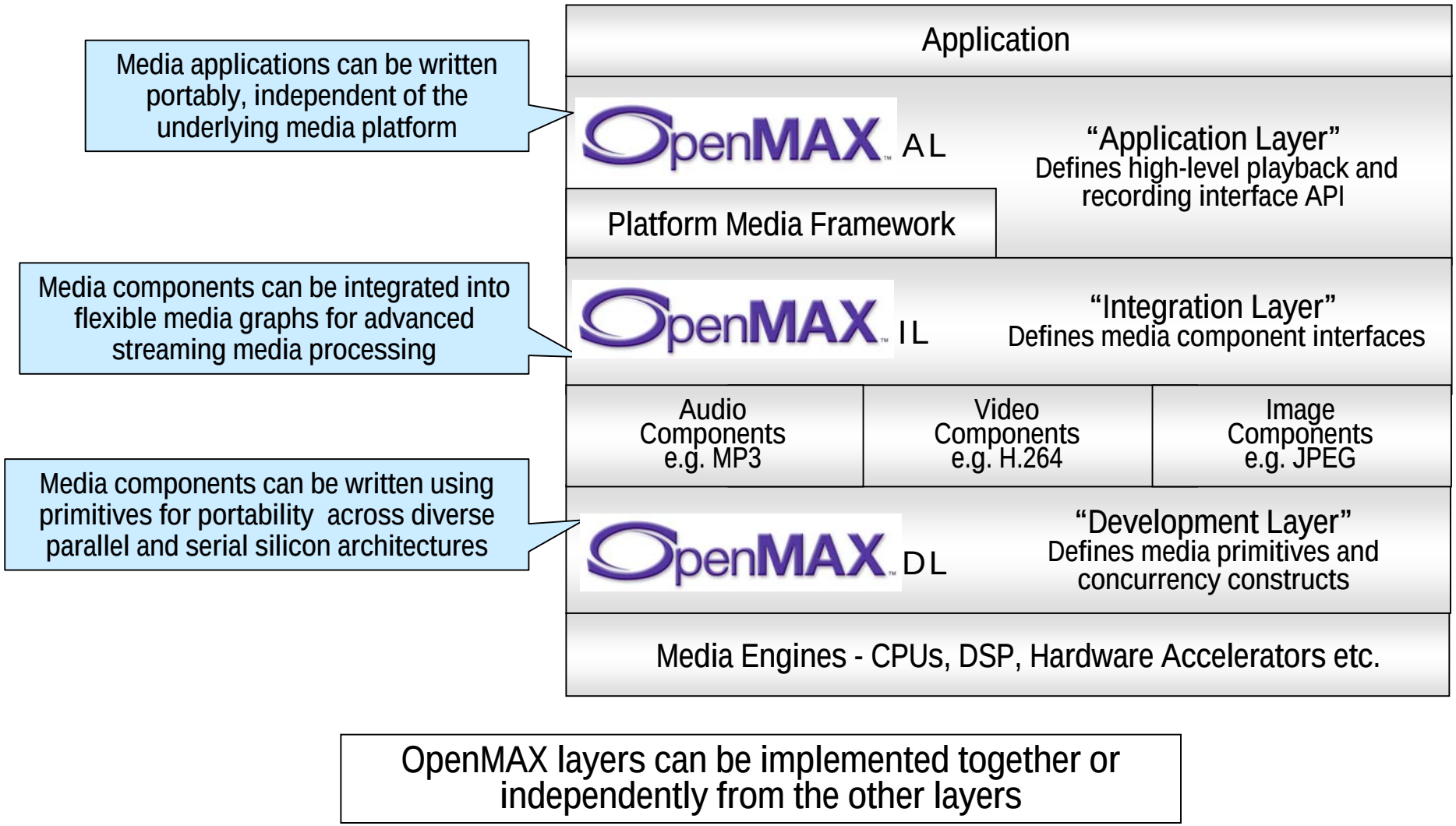
OpenMAX - Three Layer Solution



OpenMAX defines three holistically designed media open standards to provide complete media infrastructure portability

© Copyright Khronos Group, 2006 - Page 3

OpenMAX-based Media Stack

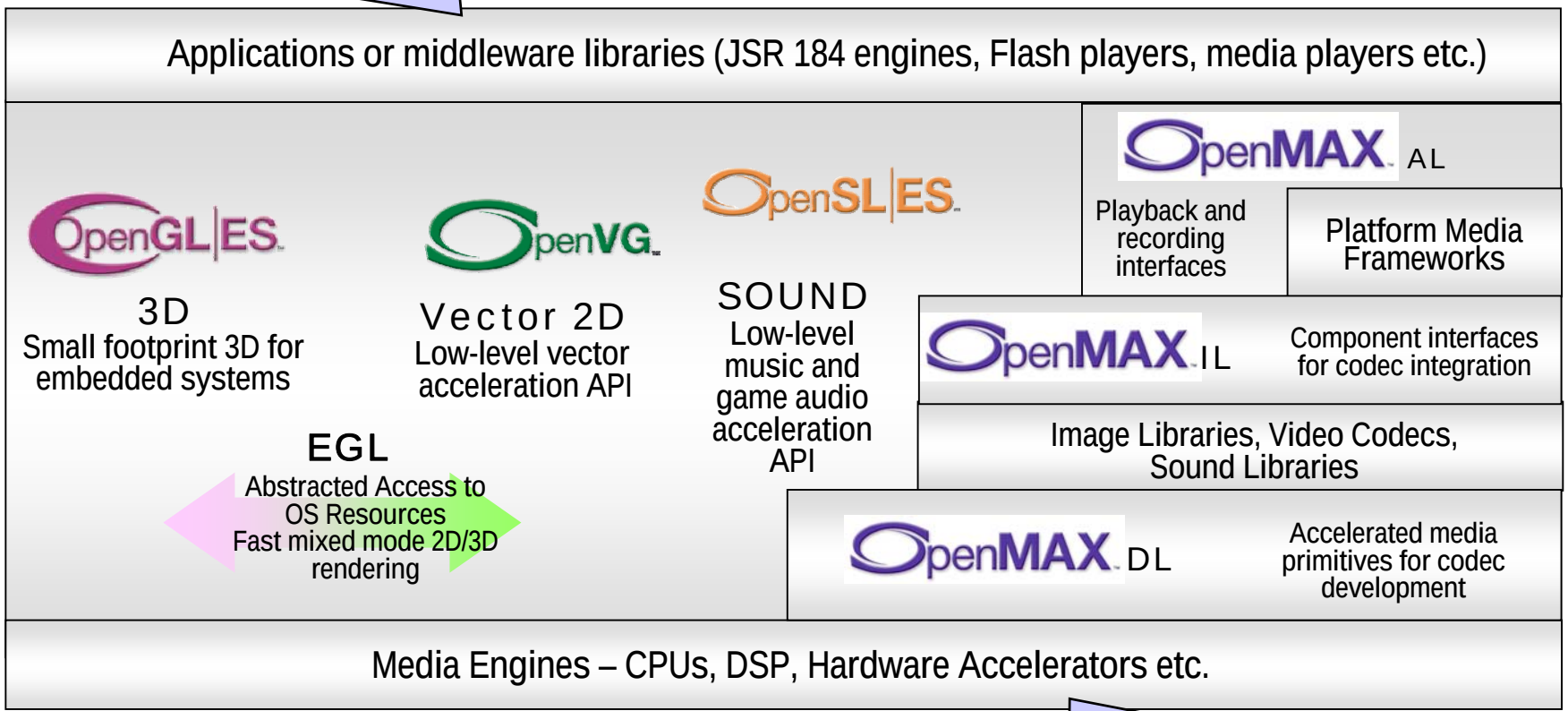


OpenMAX layers can be implemented together or independently from the other layers

© Copyright Khronos Group, 2006 - Page 4

Complete Khronos Media Stack

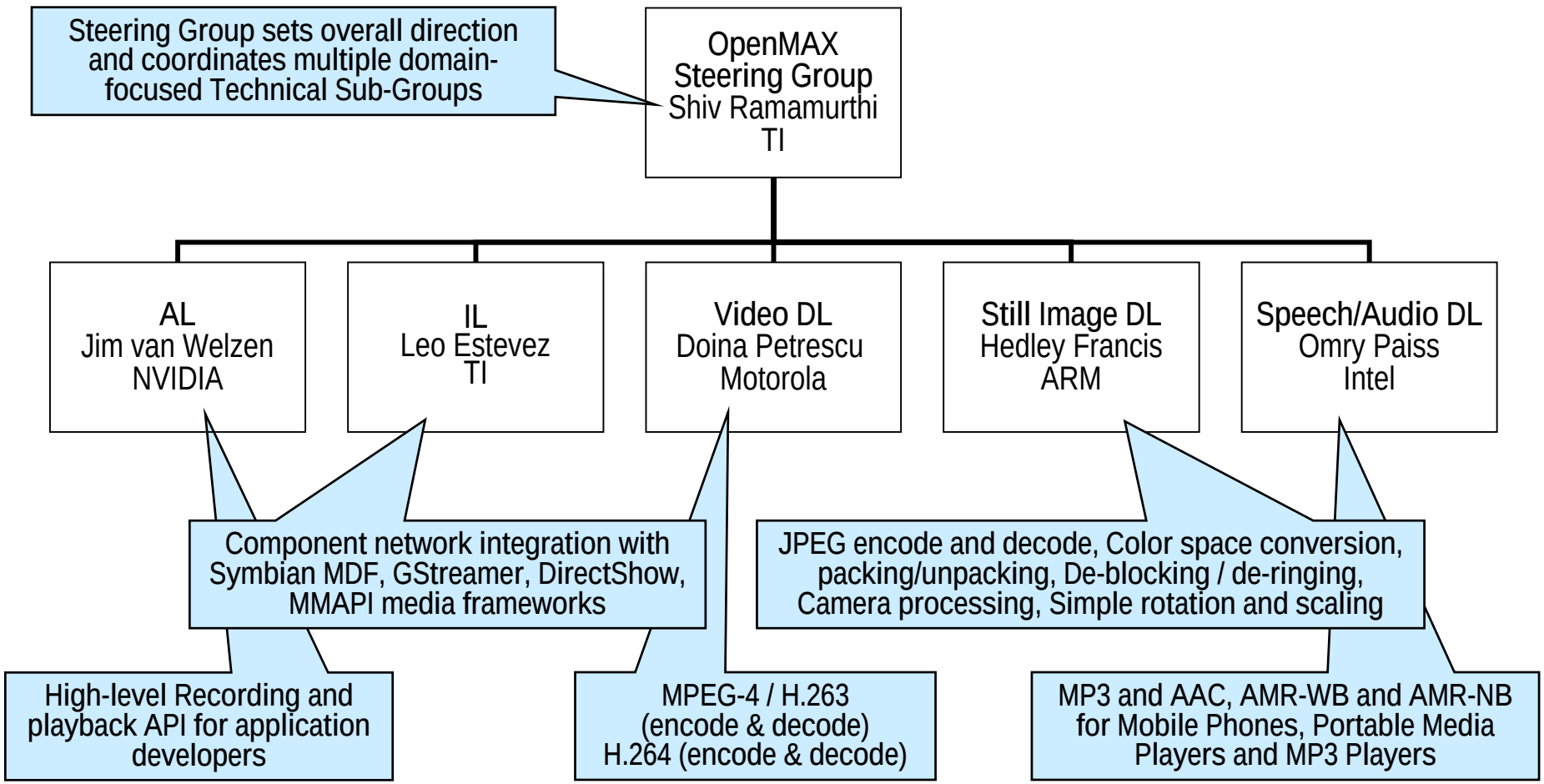
The Khronos API family provides a complete ROYALTY-FREE, cross-platform media acceleration platform



Khronos defines low-level, FOUNDATION-level APIs.
"Close to the hardware" abstraction provides portability AND flexibility

© Copyright Khronos Group, 2006 - Page 5

OpenMAX Working Group Structure

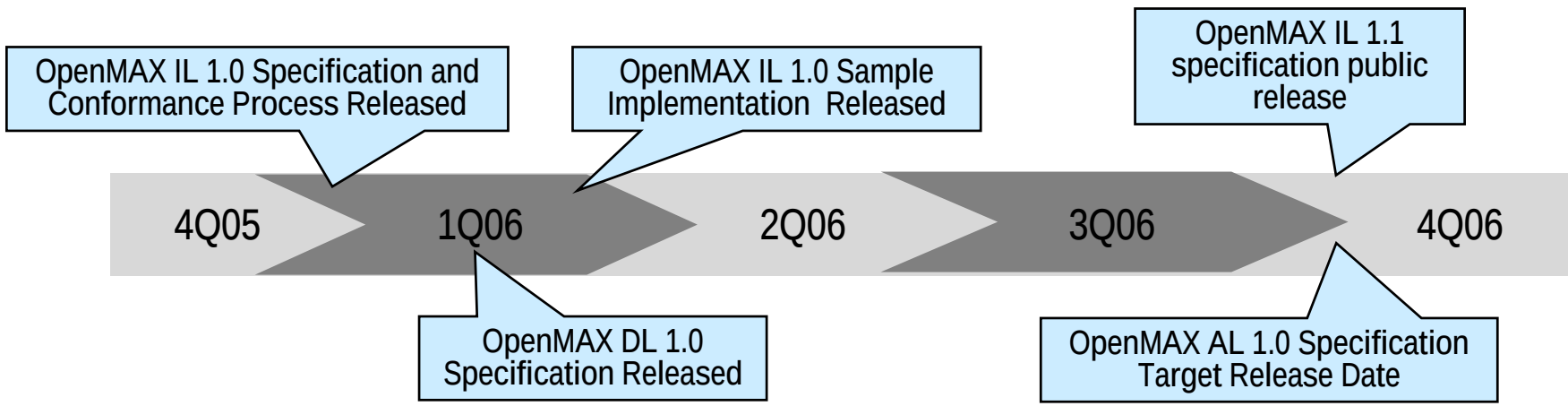


Technical Sub-Groups (TSG) and Chairs as of February 2006

© Copyright Khronos Group, 2006 - Page 6

OpenMAX Summary

- Three layer standard for media infrastructure portability
  - Media component Development, Integration and Application programmability
- Created with strong industry consensus and participation
  - ARM, ATI, Beatnik, Broadcom, Emuzed, Fraunhofer, Freescale, Infineon, Intel, Motorola, Nokia, NVIDIA, Philips, SKY MobileMedia, Samsung, Sasken, Siemens, STMicroelectronics, Symbian, Texas Instruments
- Specification is open and royalty-free using Khronos IP framework
  - Delivered with sample implementations and conformance tests
- Available on wide variety of architectures and operating systems
  - To enable true streaming media portability



© Copyright Khronos Group, 2006 - Page 7

Integration Layer

© Copyright Khronos Group, 2006 - Page 8

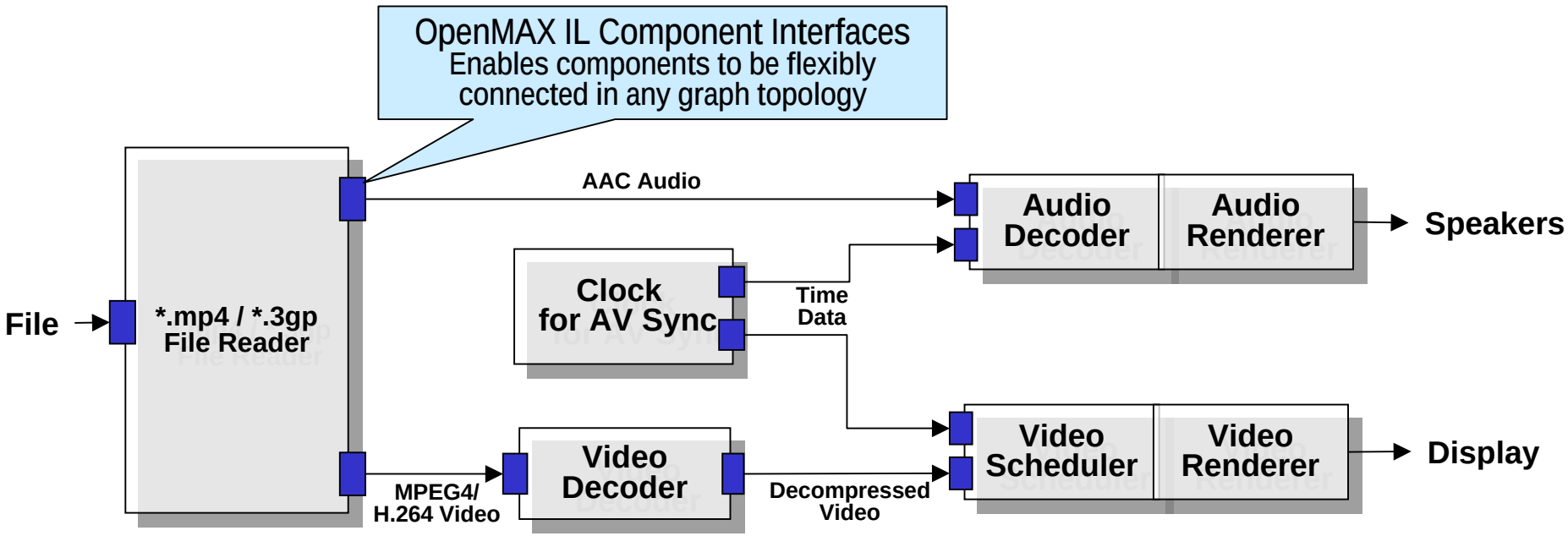
OpenMAX IL - Integration Layer

- Defines component interfaces to construct portable media graphs
  - OpenMAX IL graphs are consistent across systems
- Abstracts hardware architecture
  - Processor specific code is encapsulated within components
  - Intelligently built components maximize system utilization
- Reusable integration with major media frameworks
  - Provides a uniform interface for framework integration across many architectures
  - Designed to sit below major frameworks - e.g. Symbian MDF, GStreamer, DirectShow, MMAPI
  - Defines a low level initialization and communication protocol
- Extensible
  - API extensions can be used to expose non standard features with only minor tweaks
- Media graph use cases can be reused
  - Use cases can be debugged in parallel on different projects and then shared
- Enables Performance Comparisons and Optimization
  - Common API allows benchmarking of different architectures, implementations and frameworks
  - Performance differences can be used by vendors to find areas for further optimization

© Copyright Khronos Group, 2006 - Page 9

OpenMAX IL Example Graph

- Standardized component interfaces enable flexible media graphs
- Includes multi-stream synchronization



Example: MPEG-4 video synchronized with AAC audio decode

© Copyright Khronos Group, 2006 - Page 10

OpenMAX IL Deliverables

- OpenMAX IL 1.0 specification
  - Publicly released
- Conformance Tests
  - Component based with two profiles
    - Base Profile - to test the component's basic operation
    - Interop Profile - to test the component's interoperable behavior with a test component
  - Conformance tests will be validated on independently developed sample implementations
- Implementation Whitepapers
  - Examples of how to implement Microsoft DirectShow, Symbian MMF, and GStreamer
- Linux sample implementation (coded by TI)
  - Video - H.263
  - Audio - Narrow Band AMR
  - Image - Baseline JPEG
- Bellagio OSS implementation (coded by ST)
  - ALSA and MP3 components (based on ffmepeg)
  - Available on Sourceforge

© Copyright Khronos Group, 2006 - Page 11

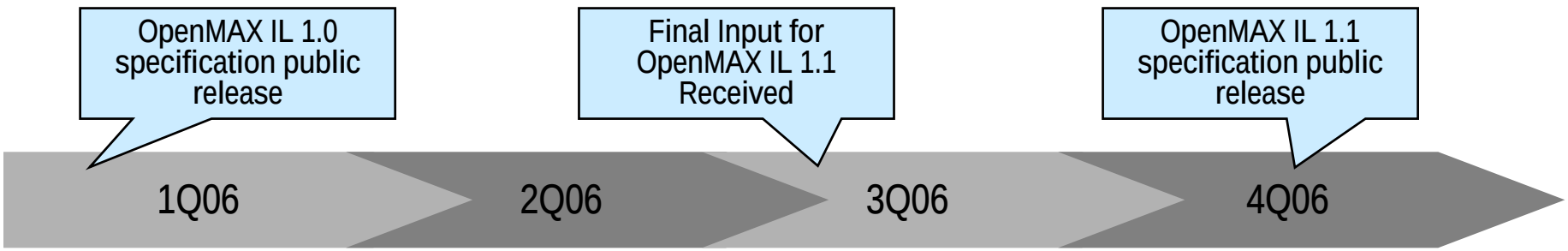
OpenMAX Architecture & Features

- Objectives & Profiles
- System Architecture
- Component Architecture
- Component Registration
- Component States
- In-Context/Out-Context Behavior
- Buffer Allocation & Sharing
- Port Reconnection
- Buffer Queue Flush
- Buffer Marking
- Buffer Payload
- Buffer Flags
- Synchronization
- Rate Control
- Resource Management
- Future Features

© Copyright Khronos Group, 2006 - Page 12

Tentative OpenMAX IL Roadmap

- Standard Components
  - Set of group defined components
- OS Services
  - File I/O, Network I/O, Scheduling, Memory Management
- Security
  - DRM, Platform
- Power Management
  - Metrics, Hooks
- Resource Management
  - Metrics, Hooks



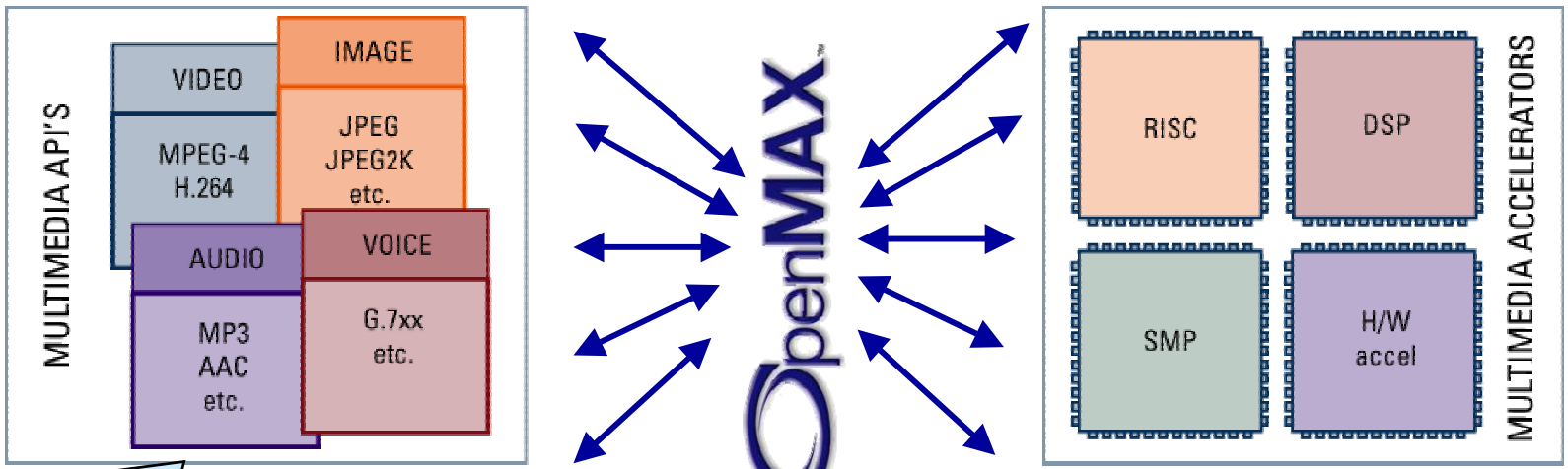
© Copyright Khronos Group, 2006 - Page 13

Development Layer

© Copyright Khronos Group, 2006 - Page 14

OpenMAX DL – Low-Level Media API

- OpenMAX DL is a library of key static primitive functions
  - Designed to cover 80% of the processing required in a multimedia codec
- Abstracts the ISA from the multimedia codec
  - Enables faster codec development time and faster porting of existing codecs
- Enables third party codec vendors to sell processor-agnostic codecs
  - Multi-core architectures (i.e. ARM + DSP) gain greater code reuse between cores



An increasing number of multimedia API codecs for video, audio, graphics and images

Silicon vendors supply optimized OpenMAX DL library for rapid porting of codecs across multiple accelerators

A wide range of media acceleration silicon using many diverse architectures

© Copyright Khronos Group, 2006 - Page 15

OpenMAX DL Domains

- Video Domain
  - MPEG-4 SP/H.263 BL (encode and decode)
  - H.264 (encode and decode)
- Image Codec Domain
  - JPEG (encode and decode)
- Image Processing Domain
  - Color space conversion
  - Pixel packing/unpacking
  - De-blocking / de-ringing
  - Rotation, scaling, compositing, etc.
- Multimedia Audio Domain
  - MP3
  - AAC
- Signal Processing Domain
  - FIR
  - IIR
  - FFT
  - Dot Product

© Copyright Khronos Group, 2006 - Page 16

OpenMAX – Asynchronous DL (aDL)

- API to group or chain multiple DL primitives together
  - To form a single executing block
- Enables vendors to accelerate key groups of primitives through:
  - Specialized hardware
  - Co-processors
  - Hand-coded ISA optimizations
- Enables a standard migration path between platforms
  - With pure software and tightly coupled hardware
- OpenMAX iDL
  - Achieves same effect as OpenMAX aDL using OpenMAX IL constructs

© Copyright Khronos Group, 2006 - Page 17

OpenMAX DL Video Domain

- Computationally intensive “hotspots” for video applications
  - Basic video processing building blocks
- Typical devices
  - Digital still cameras, PDAs, Mobile Phones, Portable Media Players, Set-top-boxes, PCs, etc.
- Example video primitive functions in OpenMAX DL 1.0
  - 8x8 Add, Sub and 16X16 Add, Sub
  - 8x8 DCT+Q+Scan and 8x8 IDCT+Q+InvScan
  - MPEG-4 Variable Length Decode
- Merged functions for improved performance on some architectures
  - Motion Estimation, Motion Compensation, Deblocking
- Video codecs covered by OpenMAX DL 1.0
  - MPEG-4 SP/H.263 BL (encode & decode)
  - H.264 (encode and decode)
- Can use aDL and iDL for video processing
  - OpenMAX DL 1.1 will publish standard DL chains for aDL wrappers

© Copyright Khronos Group, 2006 - Page 18

OpenMAX DL Image Domain

- Computationally intensive “hotspots” for imaging applications
  - Basic image processing building blocks
- Typical devices
  - Digital still cameras, PDAs, Mobile Phones, Set-top-boxes, PCs, Printers etc.
- Example image primitive functions in OpenMAX DL 1.0
  - JPEG - encode and decode, 8x8 DCT and 8x8 IDCT, Quantization
    Merged DCT & quantization functions, Huffman encoding and decoding
  - Image Processing - color space conversion and packing/unpacking
    De-blocking / de-ringing filtering, Filtering, Moments, Block copy, rotation, mirroring and scaling
- OpenMAX DL 1.1 will widen image functionality
  - JPEG2000
  - Image Blending
  - Raw Camera data processing etc…

© Copyright Khronos Group, 2006 - Page 19

OpenMAX DL Speech / Audio Domain

- Computationally intensive “hotspots” for audio applications
  - Speech codecs are not supported since the standards are bit-exact
  - Other speech applications are supported indirectly with some signal processing APIs
- Typical devices
  - PDAs, Mobile Phones, Portable Media Players etc.
- Example speech / audio primitive functions in OpenMAX DL 1.0
  - Audio API - Unpacking of headers and bit-streams, Huffman decode, IMDCT and MDCT Polyphase filter, TNS and PNS processing
  - Signal Processing API - FFT and IFFT, FIR, IIR and Median filters, Dot product, Block exponent (finding minimal sign bits in array elements)
- Example uses
  - MP3 decoder, including low frequencies extensions, MPEG4-AAC decoder (LC/L TP profiles), Signal processing (FFT, digital filters, some math)
- OpenMAX DL 1.1 will widen functionality
  - Audio encoders
  - EAAC, EAAC+
  - LMS filters
  - Voice Recognition front-end …

© Copyright Khronos Group, 2006 - Page 20

Application Layer

© Copyright Khronos Group, 2006 - Page 21

OpenMAX AL - Application Level

- Enabling application developers to easily leverage OpenMax acceleration
  - A simple high-level interface for common multimedia playback and capture use cases
- Typical applications are found in:
  - Mobile Phones
  - Mobile Music/Video Players
  - PDAs
  - Digital still cameras
  - Digital Media Adapters
  - STBs, PCs, etc…

© Copyright Khronos Group, 2006 - Page 22

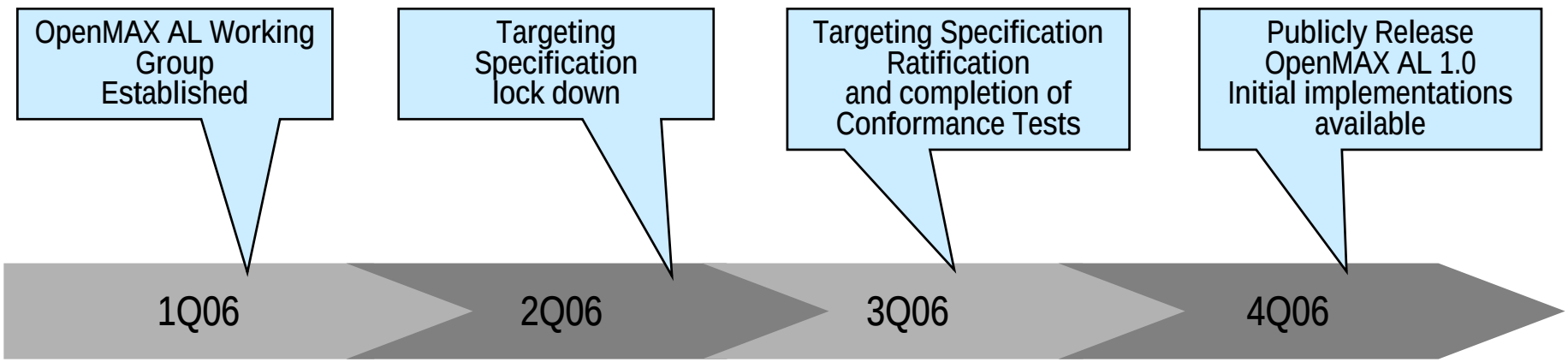
OpenMAX AL 1.0 – Scope

- Standard use cases
  - Playback: play a video file, play a music file, display an image file
  - Recording: record a video file, record an audio file, capture an image file
- Operational controls
  - Playback: play, pause, stop, FF, RW
  - Recording: record, stop
- Configuration control
  - Audio output: volume, channels, etc
  - Video output: video window position, size, etc
- Metadata controls
  - Extract metadata from a playing stream
  - Insert metadata into a recording stream

© Copyright Khronos Group, 2006 - Page 23

OpenMAX AL - Milestones

- OpenMAX AL Taskforce formed in November 2005
  - Membership included: ATI, Beatnik, Freescale, Nokia, NVIDIA, Symbian, SkyMobile Media, TI
  - Scoped intended functionality and investigated alternative solutions
  - Recommended formation of an OpenMax AL working group
- OpenMAX AL Working Group formed in December 2005
  - Call for widened working group participation
  - Official scope/requirements definition at face to face meeting in January 2006



© Copyright Khronos Group, 2006 - Page 24