

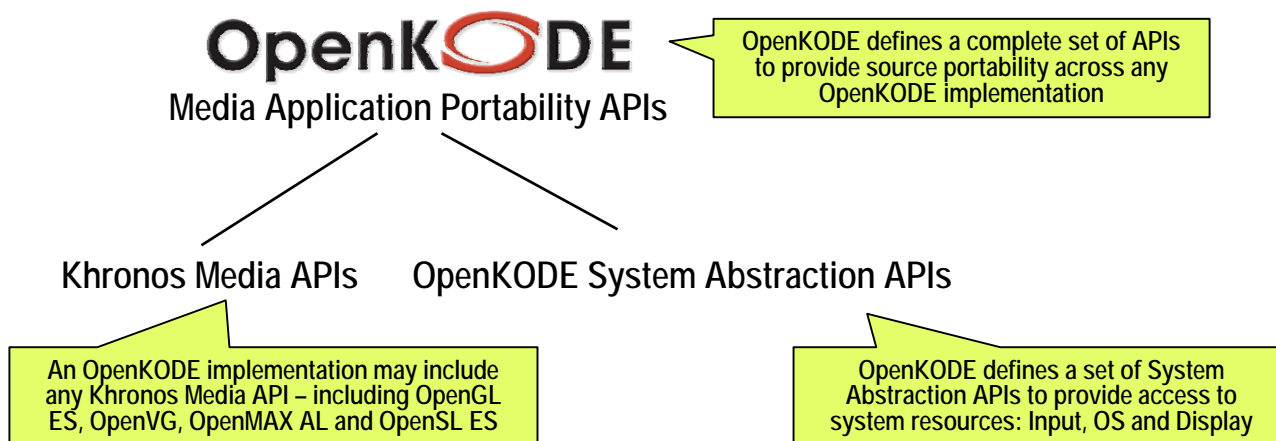


OpenKODE Structure

May 2006

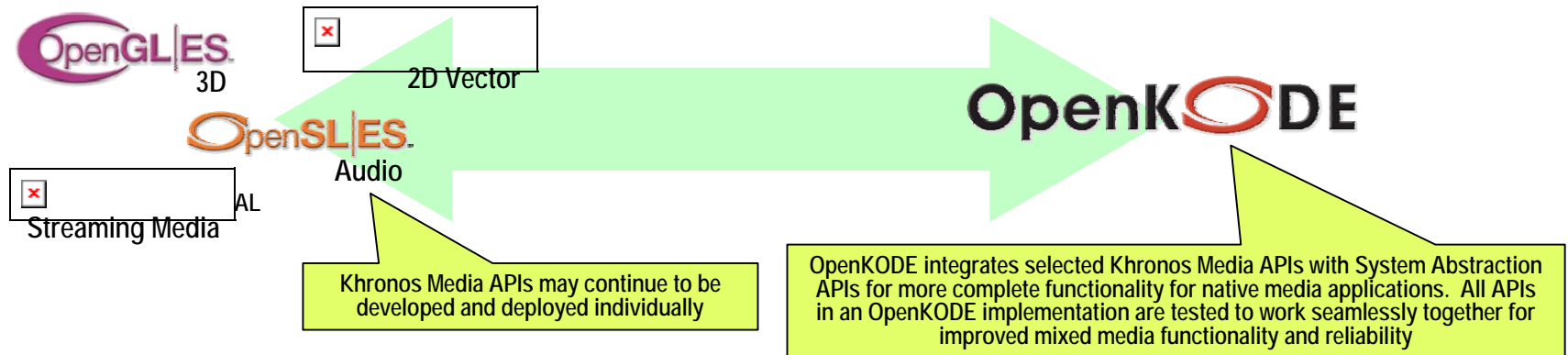
What is OpenKODE

- **OpenKODE is coherent set of Media Application Portability APIs**
 - Enables native media application source portability – significantly reducing fragmentation
 - APIs provide abstracted native access to system resources and media acceleration
 - State-of-the-art APIs – designed to interoperate efficiently
 - All APIs are open standards – available from multiple vendors
- **Conformant OpenKODE implementations will pass Trans-API Tests**
 - Trans-API Conformance Tests encourage seamless cross-media acceleration
 - Makes an OpenKODE platform more than just a “collection of APIs”
- **OpenKODE working group defining implementation tools and guidelines**
 - Encouraging OpenKODE implementations to be efficient and useful



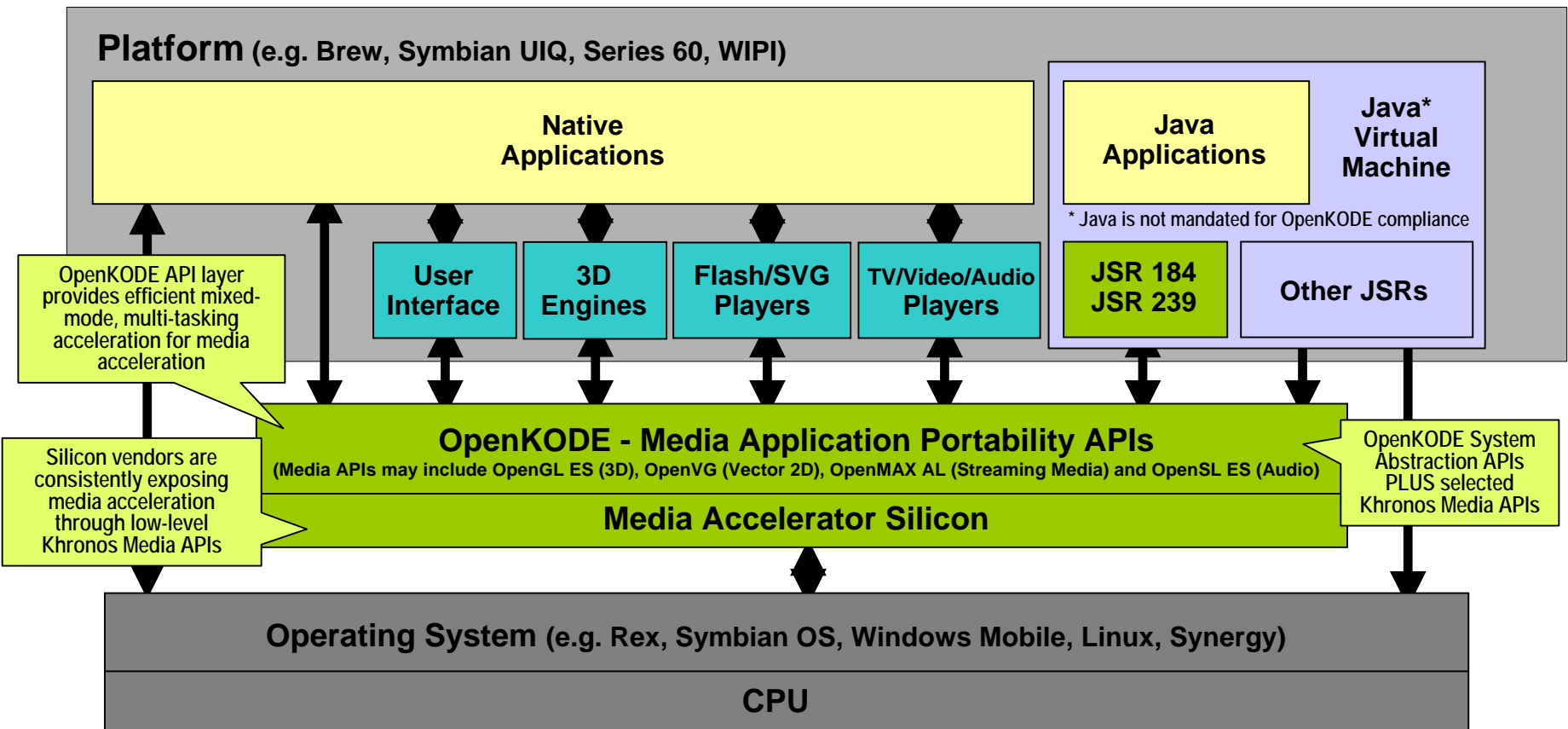
Who Benefits?

- **Media Application Developers**
 - Can easily port advanced media applications across multiple devices
- **Silicon Vendors**
 - Need to implement just one set of media acceleration APIs
- **Software Platform Vendors**
 - Can easily adopt an effective, vendor-neutral, media acceleration API layer
- **Handset OEMs**
 - Can bring more accelerated media applications to their platform
- **Carriers**
 - Can raise ARPU by charging more for compelling media applications

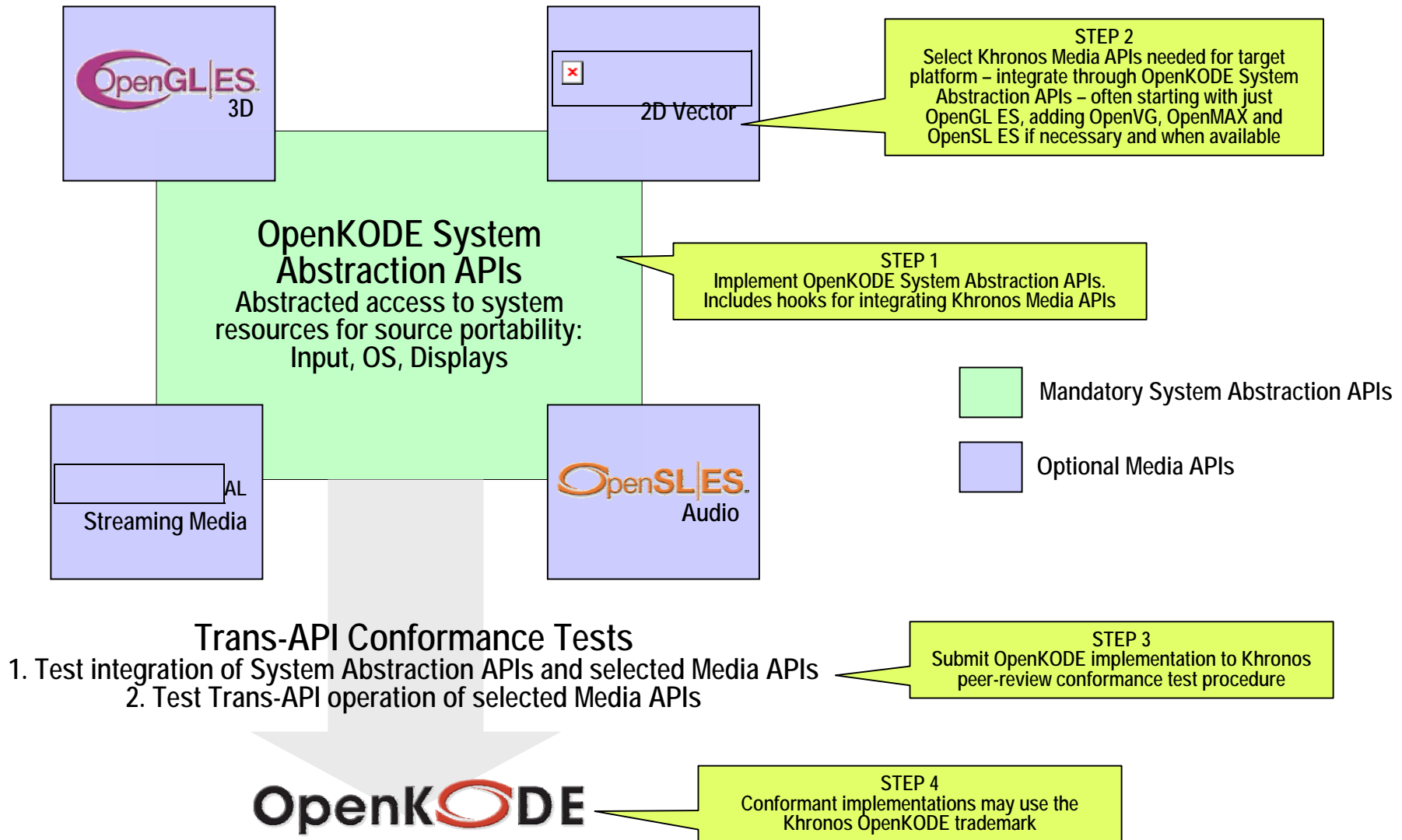


OpenKODE, OS's and Platforms

- OpenKODE - a coherent media API layer for integration into any platform
 - Sits between the Platform and the Operating System



Building an OpenKODE Implementation



OpenKODE Working Group Deliverables

- **System Abstraction APIs**
 - Run-time APIs for source portability deployed in a production device: Display, Input, OS
- **Trans-API Coordination**
 - Trans-API Design Coordination – designing all OS and media APIs to work together coherently
 - Trans-API Conformance Tests – testing OS and media APIs work together correctly
- **Implementation Infrastructure**
 - Debug and Profiling API – enabling better authoring tools
 - Device Discovery Utility – enabling content to scale with device capabilities/throughputs
 - Platform Integration Guidelines – how to access platform resources (e.g. Java)

