



Please click on the note callout for speaker notes!

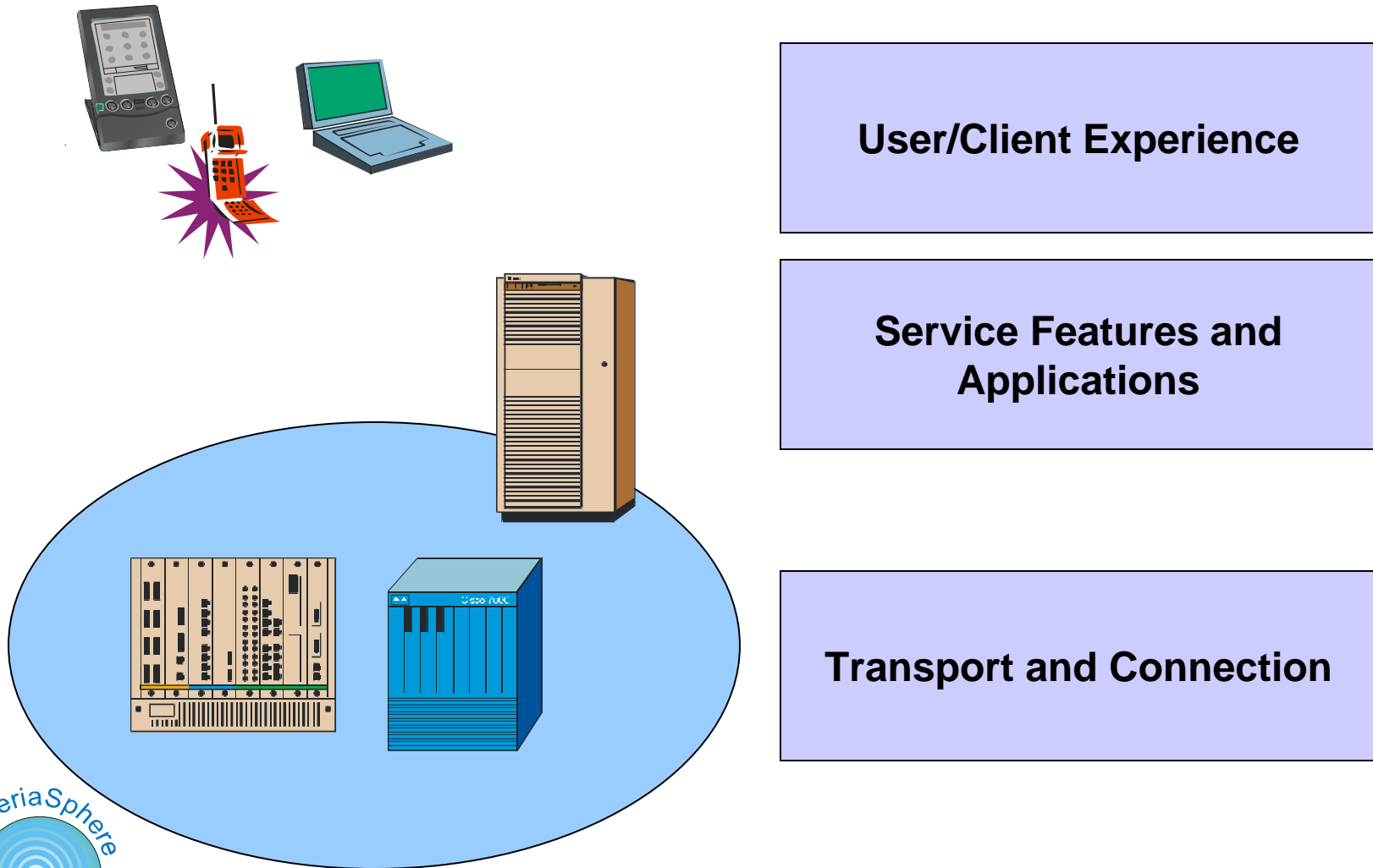


ExperiaSphere™ and Embedded and Open Handset Architectures



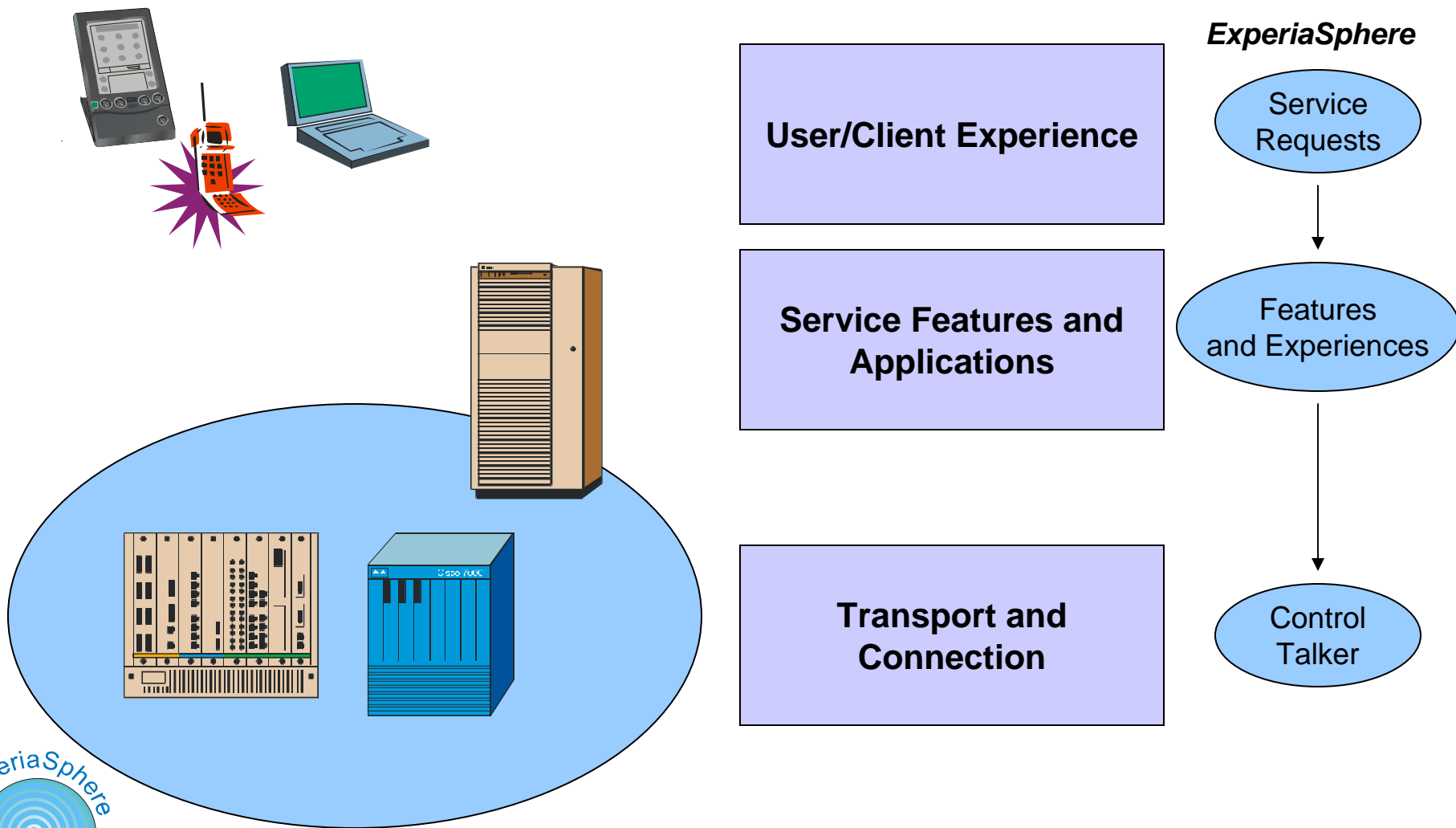


A High-Level View of a “Service”





ExperiaSphere and Services





What's an Embedded System and Why Does ExperiaSphere Care?

- Embedded systems are devices that contain computer logic but which are not intended to be general-purpose computers
- Many communications devices, including mobile handsets, PDAs, electronic books, game systems, etc. are embedded systems
- ExperiaSphere should be able to add value to these devices in some useful way to be fully market-relevant





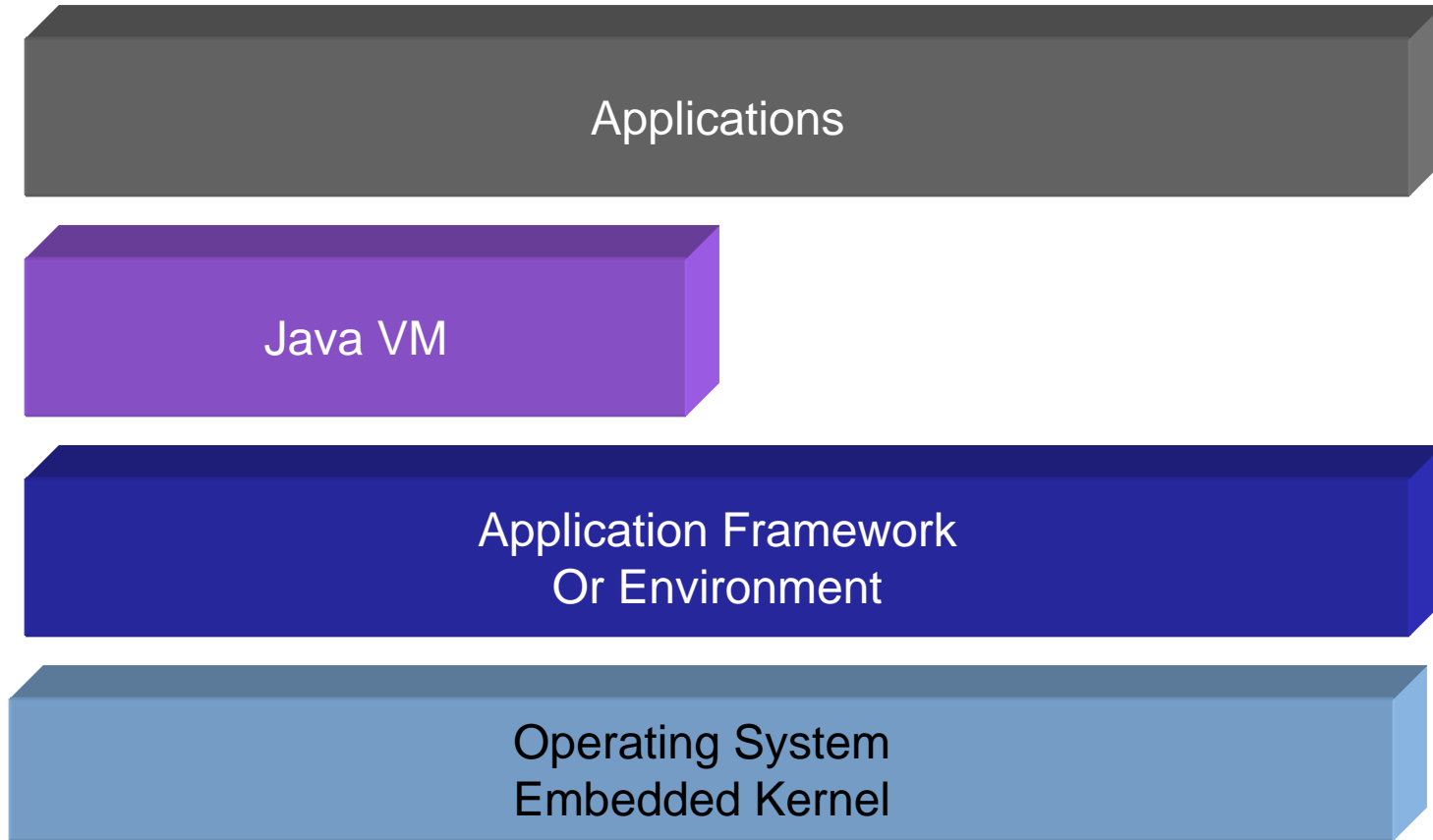
An Overview of the Issues

- ExperiaSphere is a Java-based architecture that is based primarily on a class called “Experiam” which represents services or experiences.
- Experiams are “Plain Old Java Objects” or POJOs, and we encourage them to be developed using the basic J2SE Version 6 capabilities.
- J2SE V6 is generally available on computer systems like desktops, laptops, and servers running Windows, Linux, Solaris, etc.
- Special-purpose devices may not offer complete J2SE V6 support and will thus present special issues with ExperiaSphere





Embedded System Architectures





Supporting Embedded Architectures

- There are three approaches:
 - Develop Experiams on the embedded device in essentially the “normal” way
 - Develop Experiams on the embedded device that are proxies for “real” Experiams that are hosted elsewhere
 - Develop an Event Source on the embedded device that links to an Experiam hosted elsewhere





The “As Usual” Approach

- If the embedded device supports full J2SE V6 capabilities there are no restrictions in developing Experiams for it and all properly designed Experiams should work on it
- If the embedded device supports some reasonable subset of J2SE V6, it may be possible to develop and use Experiams freely as long as they use only the available Java libraries and facilities
- **There may be performance issues with this approach if the embedded device is constrained in resources or processing power**





The “Proxy Approach”

- Proxy Experiems require only minimal Java capabilities to develop, and need only a communications link available to bind them to their host
- Considerable processing can be offloaded onto the host Experiems reducing resource load on the embedded device





The “Event Approach”

- Experiems require a Java VM and if none is available they cannot be developed
- The recommended approach for this is the binding of an Experiam on another Java-compatible platform to a signal set from the embedded platform
- A Bind Experiam class provides a link between Experiems and the embedded system by acting as an event relay from the embedded signal set to Java and ExperiaSphere
- The “signal set” can be a browser-generated event set if the embedded device supports a browser (AJAX, for example)



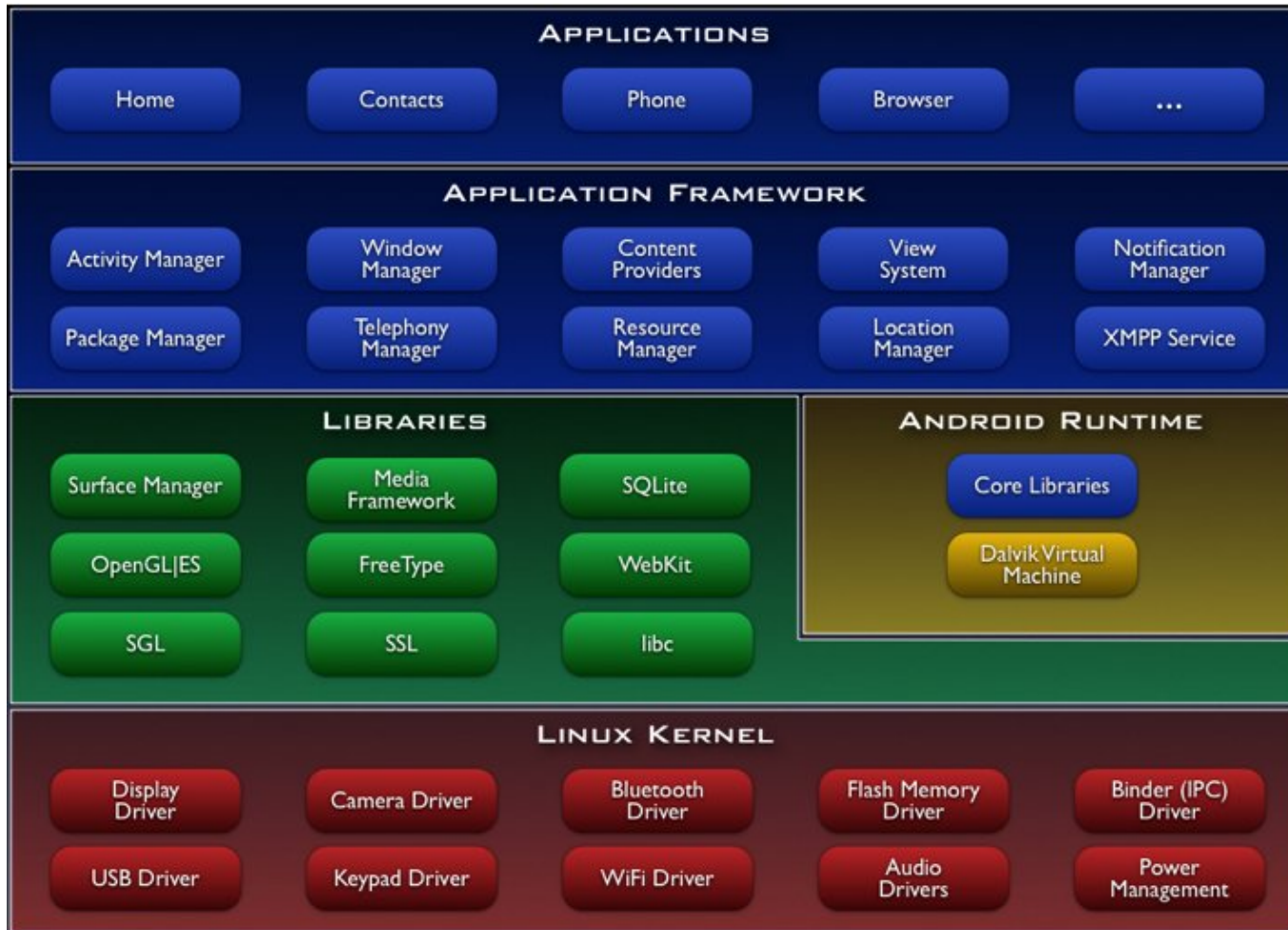


Some Key Examples

- Google Android, a Linux-based system with strong Java support
- Apple's iPhone, OS X (Unix-based), some Java browser support but no developer support for Java
- LiMo, Linux-based, limited data capabilities, no Java support

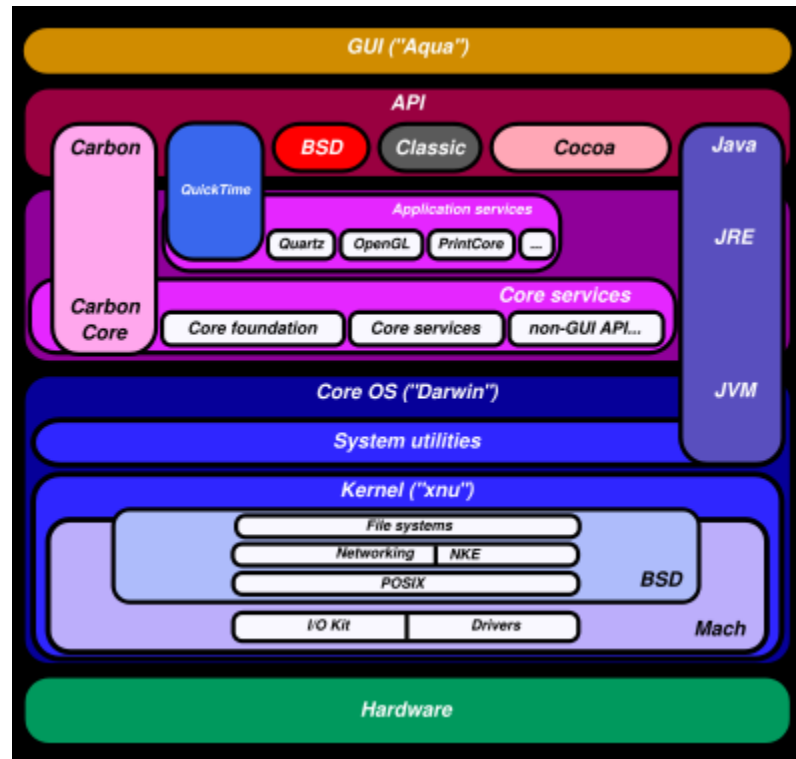


Google's Android



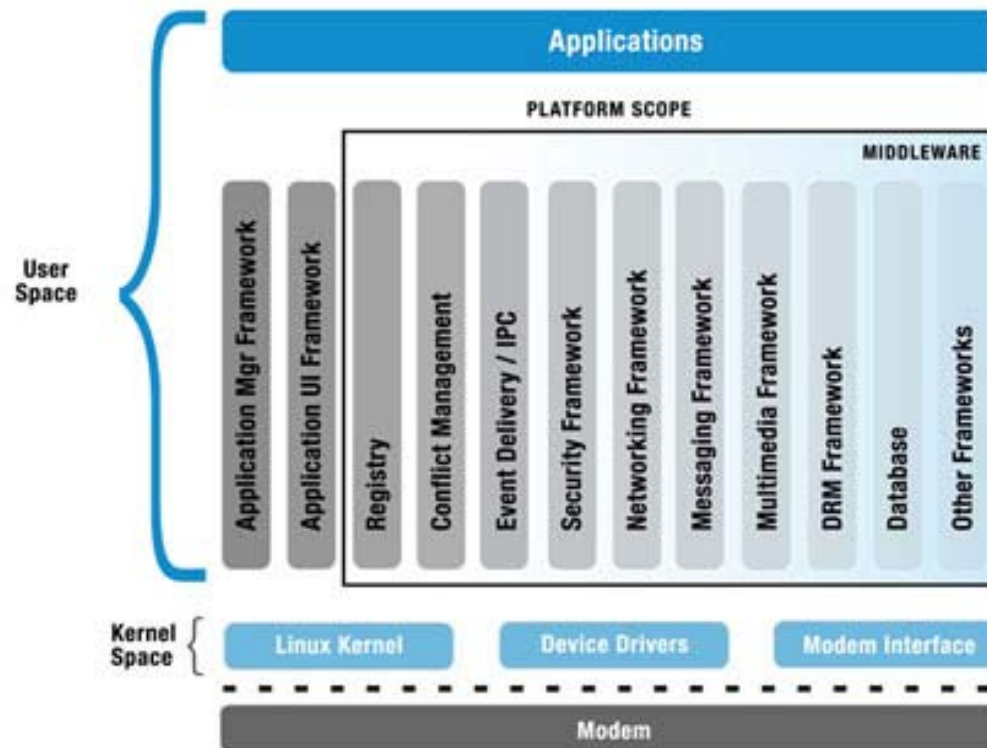


Apple's iPhone Architecture





The LiMo Architecture





In Summary...

- ExperiaSphere's “experiences” are represented by Java objects that can be triggered in a variety of ways
- Where Java is runnable on a remote device, that device can run Experiams and use at least some ExperiaSphere mechanisms directly
- Where Java cannot be run, any communications channel can be used to activate an experience through ExperiaSphere mechanisms





We Need Help!

- Equipment vendors who are prepared to work to expose management APIs via Experiams
- Software/application vendors who are prepared to create Experiams around their interfaces and elements
- Enterprise equipment and software vendors interested in an ExperiaSphere model of collaboration and Unified Communications, or other applications
- Java programmers who want to participate in the service convergence revolution
- Contact tnolle@cimicorp.com for details on how to get involved!



Thank you!

