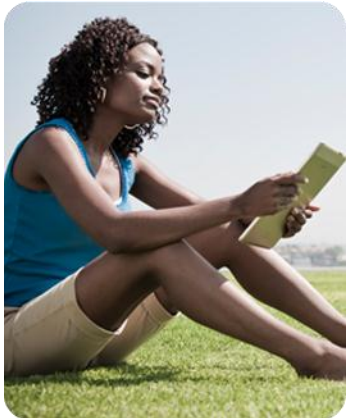


Open Source. Open Possibilities.



The AllJoyn™ Open Source Project

Marcello Lioy
Director, Engineering,
Qualcomm Innovation Center, Inc.

August 30, 2012



Agenda

- AllJoyn Overview
- Architectural Concepts
- Security Concepts
- Performance and Deployment
- Availability and Open Source
- Q&A

AllJoyn Overview

Open Source. Open Possibilities.



What Is AllJoyn?

An Open Source Application Development Framework Enabling Proximity-based, Peer-to-peer Networking that is Platform and Radio Bearer Agnostic



AllJoyn brings proximity awareness to mobile apps, unleashing a whole new set of user experiences to smartphones, tablets, PCs, TVs and more

Why Peer-to-peer (P2P) Is Hard

P2P Friction Developers Face Today



AllJoyn Makes Peer-to-peer Frictionless

DISCOVER

devices and applications around you

ADAPT

to apps and devices coming and going

MANAGE

transports like Bluetooth and Wi-Fi and message routing across them

INTEROPERATE

across disparate programming languages, operating systems, and bearers

EXCHANGE

Information in a secure manner

Why Develop AllJoyn?

Simplify and extend inter-device communication capabilities

Always-on, always-connecting

OS independent platform enabling peer-to-peer application ecosystem

Seed the peer-to-peer application ecosystem

Fuel new and creative application usage models

What New Experiences Can AllJoyn Enable?



ENTERTAINMENT & GAMING EXPERIENCES



MULTISCREEN EXPERIENCES



COLLABORATIVE EXPERIENCES

AllJoyn Ushers in New User Experiences for Smartphones, Tablets, PCs & TVs



GAMES



OFFICE APPS



SOCIAL NETWORKING



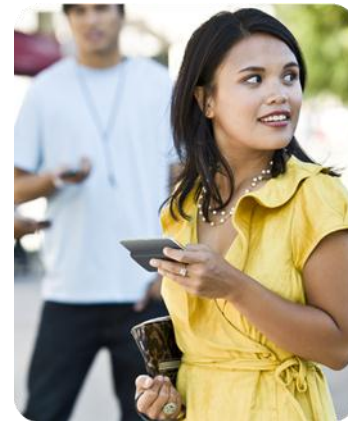
IMAGING APPS



TV CONNECTIONS

Open Source. Open Possibilities.

Architectural Concepts



Overview

Designed to be easily portable to new hardware and OS platforms

AllJoyn is a distributed software bus

- Each device has bus management functionality which is commonly referred to as “the daemon”
 - Daemon functionality can be integrated on the platform or in the app itself
- Applications communicate directly with the daemon
- Daemons handle cross device communication
- A client library is used by applications to interact with the daemon

Bus formation is ad hoc

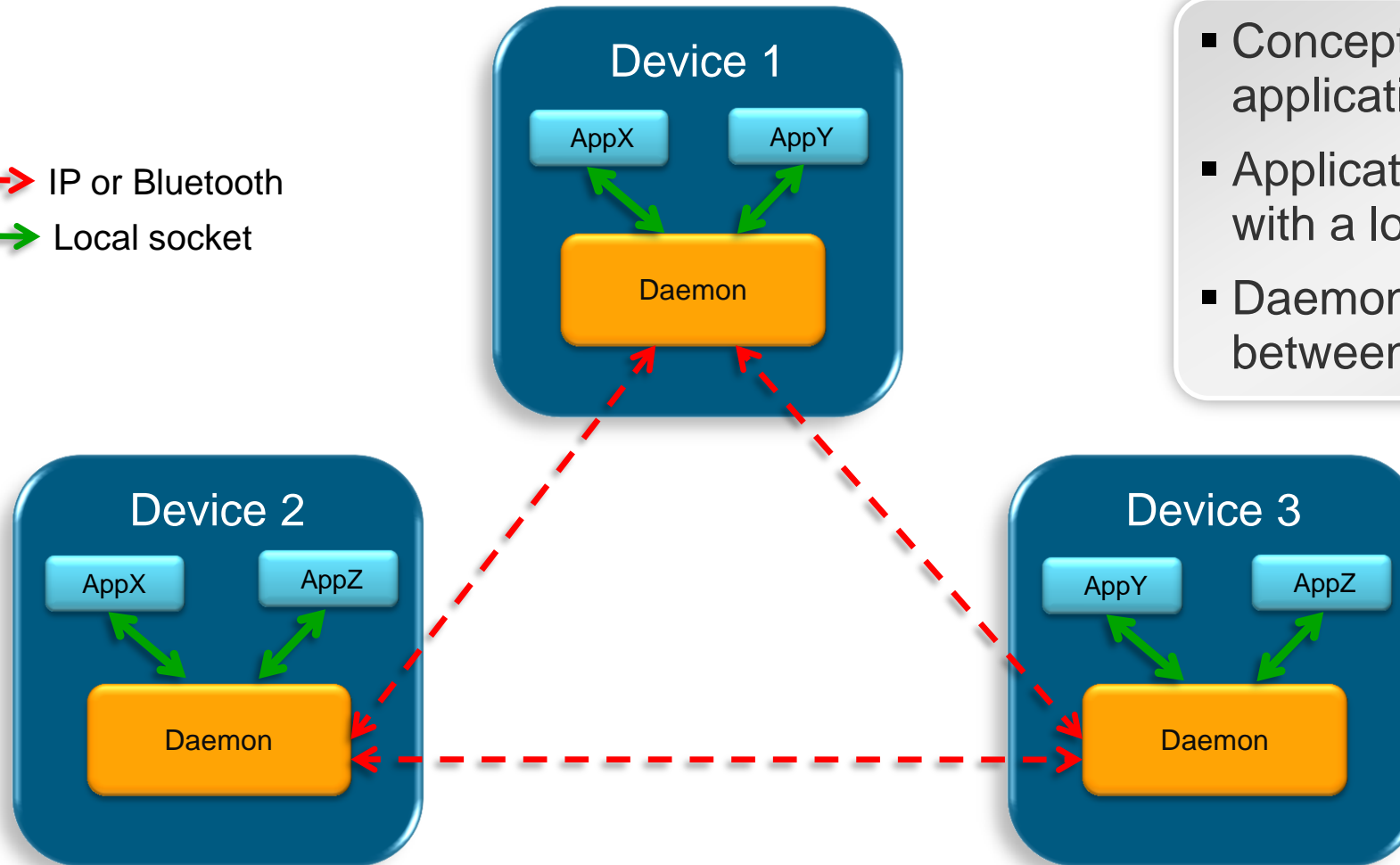
- Based on proximal discovery of applications/services
- Abstracts link-specific discovery mechanisms

Protocol is link independent

- Ground-up implementation of the D-Bus wire-protocol with extensions
- Supports Wi-Fi and Bluetooth currently
- Wi-Fi Direct under development

AllJoyn Distributed Software Bus

 IP or Bluetooth
 Local socket



- Conceptually *peers* are applications not devices.
- Applications communicate with a local daemon
- Daemons handle routing between devices

Why the D-Bus Wire Protocol?

Why reinvent the wheel?

- www.freedesktop.org/wiki/Software/dbus

IPC mechanisms used on many Linux distributions

- Deeply integrated with system services and session management
- Supports RPCs as well as unicast and broadcast events

Object oriented

- Objects, interfaces, methods, and properties

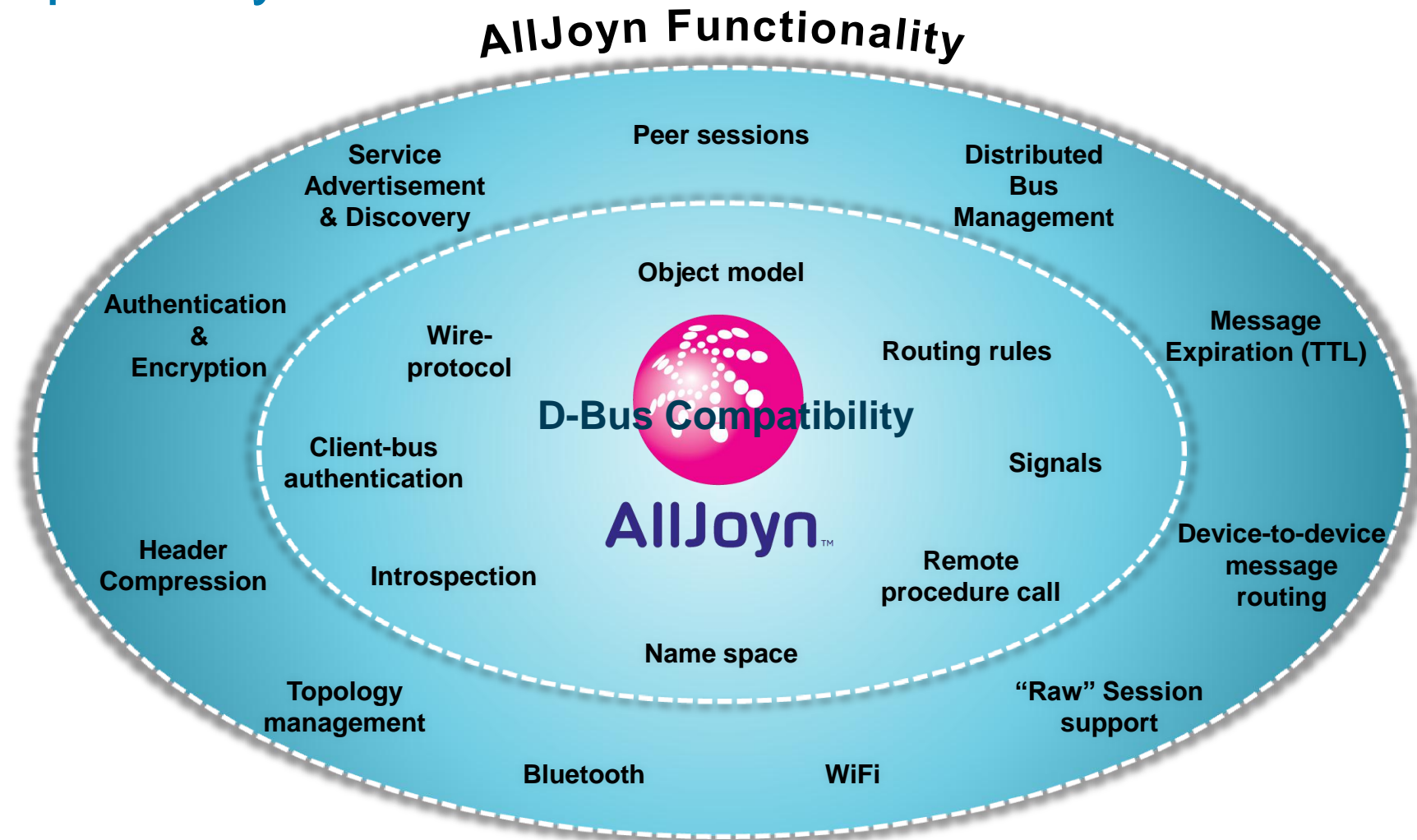
Message bus architecture

- RPC and events implemented as messages
- Publish/subscribe semantics
- P2P came from extending bus cross device; messages flow over bigger bus

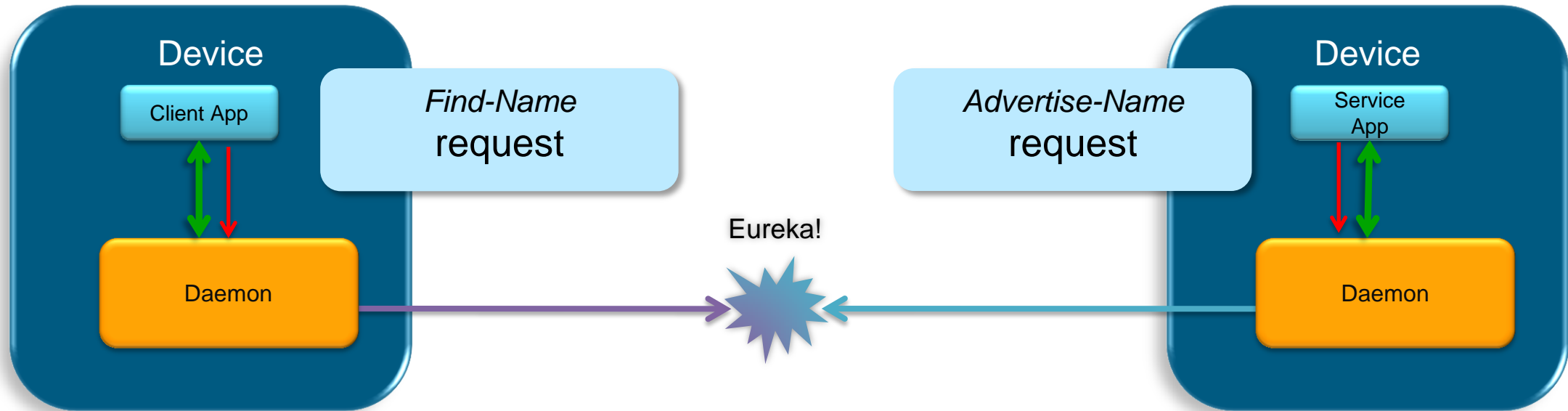
Language neutral

- Bindings for C, Java, Python, Perl, etc.

D-Bus Compatibility



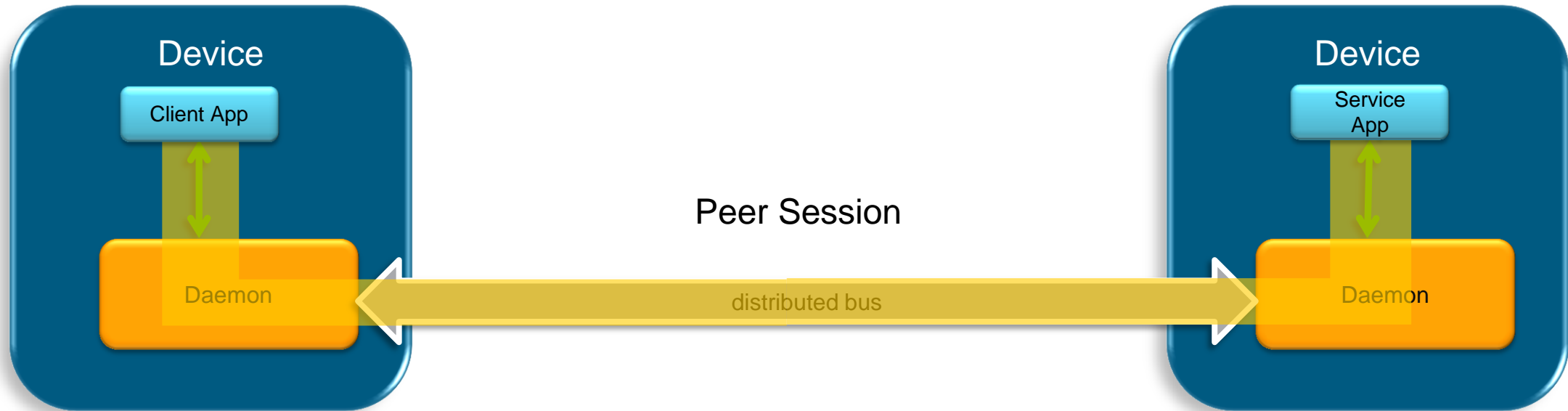
Ad Hoc Bus Formation



Actual discovery mechanism is transport dependent:

- On Wi-Fi a light-weight IP multicast protocol
- On Bluetooth device discovery with EIR and SDP query
- On Wi-Fi Direct pre-association discovery

Ad Hoc Bus Formation



Once connected daemons form a single bus with a shared namespace

- Peers can discover when other peers join or leave the bus
- Peers can make RPC calls and send and receive events
- Session reference counting keeps device-to-device connections alive
- Multicast events can be sent to all peers in the session

Object Model

AllJoyn applications expose their functionality via objects

- These are typically organized in a hierarchy

Objects implement interfaces (one or more)

Interfaces are composed of members, which fall into three categories

- Methods – classic OO object interaction
- Signals – asynchronous event notification
 - Can be broadcast, multicast, or point-to-point
- Properties – data members
 - These are accessed by built-in get/set methods

Bus Attachments, Objects, Proxy Objects

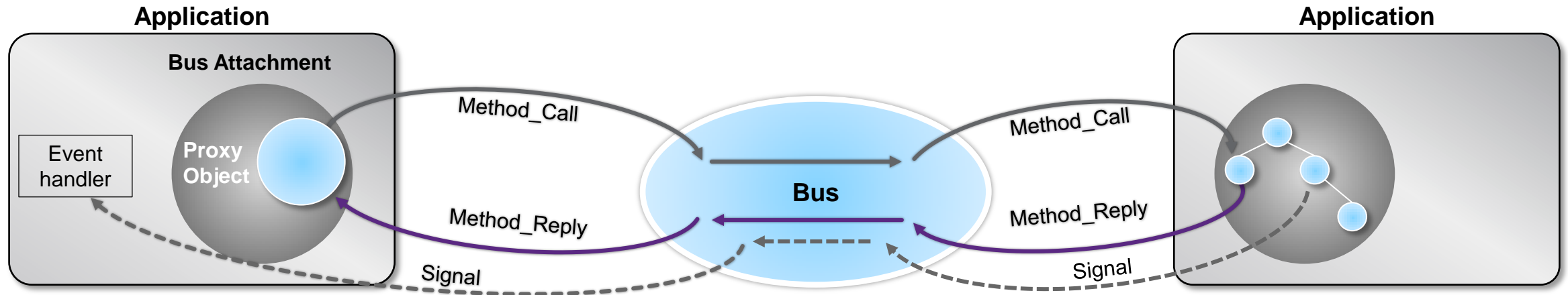
An application needs a Bus Attachment to communicate with/over bus

- Bus Attachments provide a root (/) for the object hierarchy
- Path names look like file paths (e.g., /org/AllJoyn/Games/chess)

Objects live within the Bus Attachments

Proxy Bus Objects are local representations of remote Bus Objects

- Applications use proxy bus objects to make method calls to remote objects



Language Bindings

Native implementation for AllJoyn is C++

Java binding is available for Android

C binding, Unity Binding publicly available

Binding for JavaScript is under development for NPAPI

Objective C binding will be available for iOS

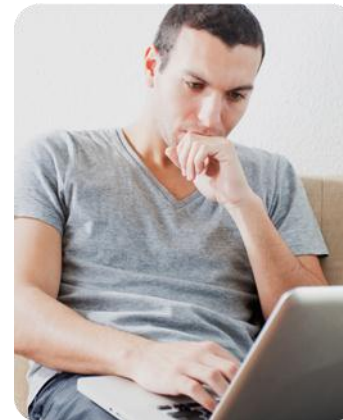
C#, WinJS, and Visual Basic will be available for Windows RT

Object model is similar for all bindings

- Create a BusAttachment to connect to the bus
- Create BusInterface(s) that will be implemented by BusObjects
- Create and register BusObject(s) with the BusAttachment
- BusObjects handle remote method calls and emit Signals
- Applications call methods on remote objects using ProxyBusObjects
- Applications register Signal handlers with BusAttachment

Security Concepts

Open Source. Open Possibilities.



Design of Security Framework

Authentication and encryption is designed to be app-to-app

- Trust relationship established between the applications
- The bus is not involved other than to route
- Device pairing not required unless the transport requires it
- In case of Bluetooth AllJoyn does not normally trigger pairing

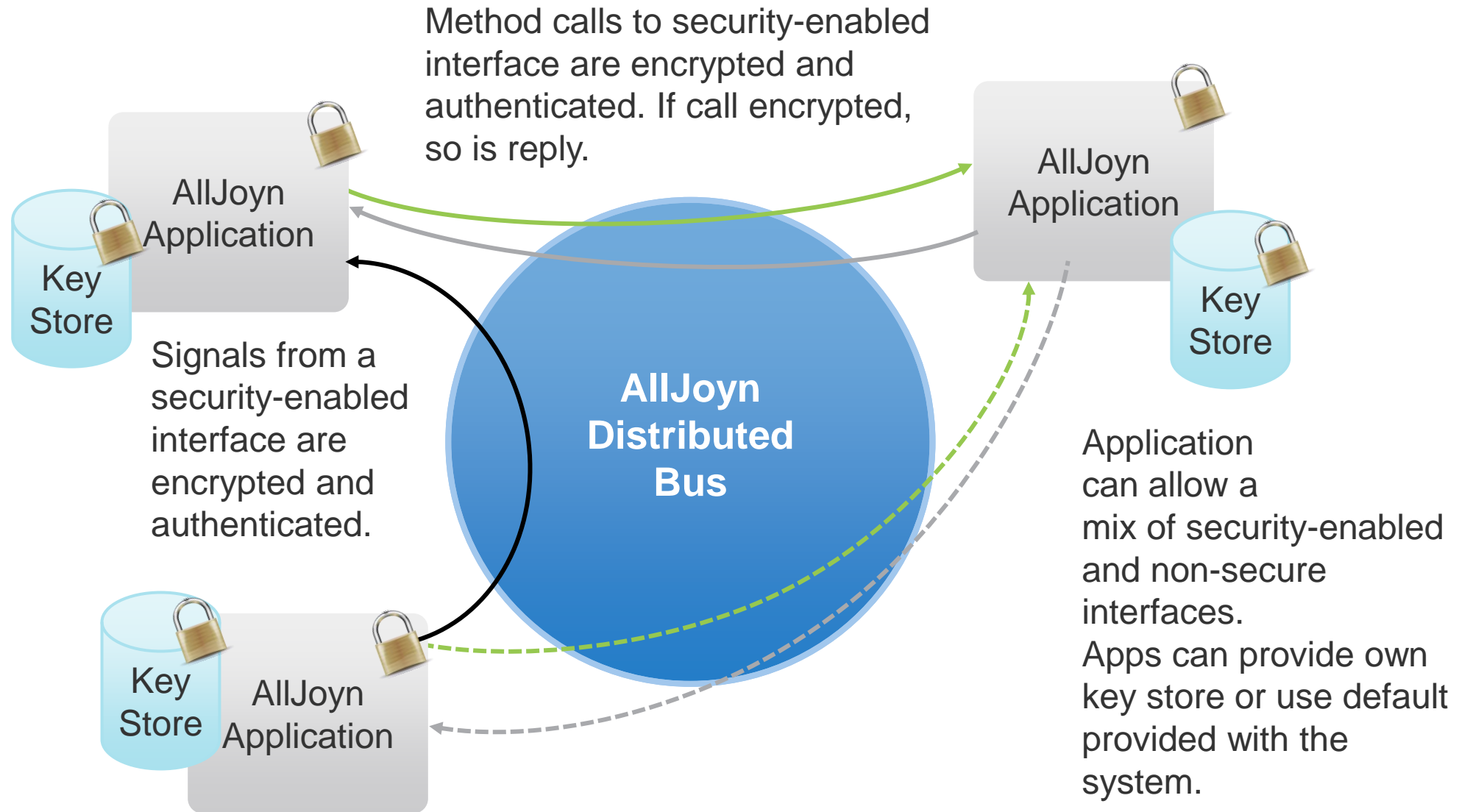
Security is enabled per-interface

- Authentication and key exchange initiated on demand

Security-enabled interface

- Authentication is required to make method calls
- Authentication required to receive signals
- All messages are encrypted

Security Model – Authentication and Encryption



Built-In Authentication Mechanisms

Three mechanisms supported

- Mechanism negotiated using SASL protocol
- Mechanism chosen by Application developer
- Mechanisms adapted from TLS protocols per RFC 5256

Pin-code

- Authentication with a single-use password
- Trust relationship is persistent

Logon

- User name and password type
- Password required every time peers connect

Certificate-based

- RSA public key authentication and X.509 certificates
- Trust relationship lasts while certificate valid

Open Source. Open Possibilities.

Performance & Deployment



Message Optimizations

Header compression

- Designed to significantly reduce the size of message headers

Time to live

- Designed to support isochronous data (e.g. real-time streaming/gaming)

Multipoint sessions

- Bounds the scope of broadcast signals to session members
- Provides mechanism for deciding when radios are no longer in use

Designed for an optimized experience on mobile embedded devices

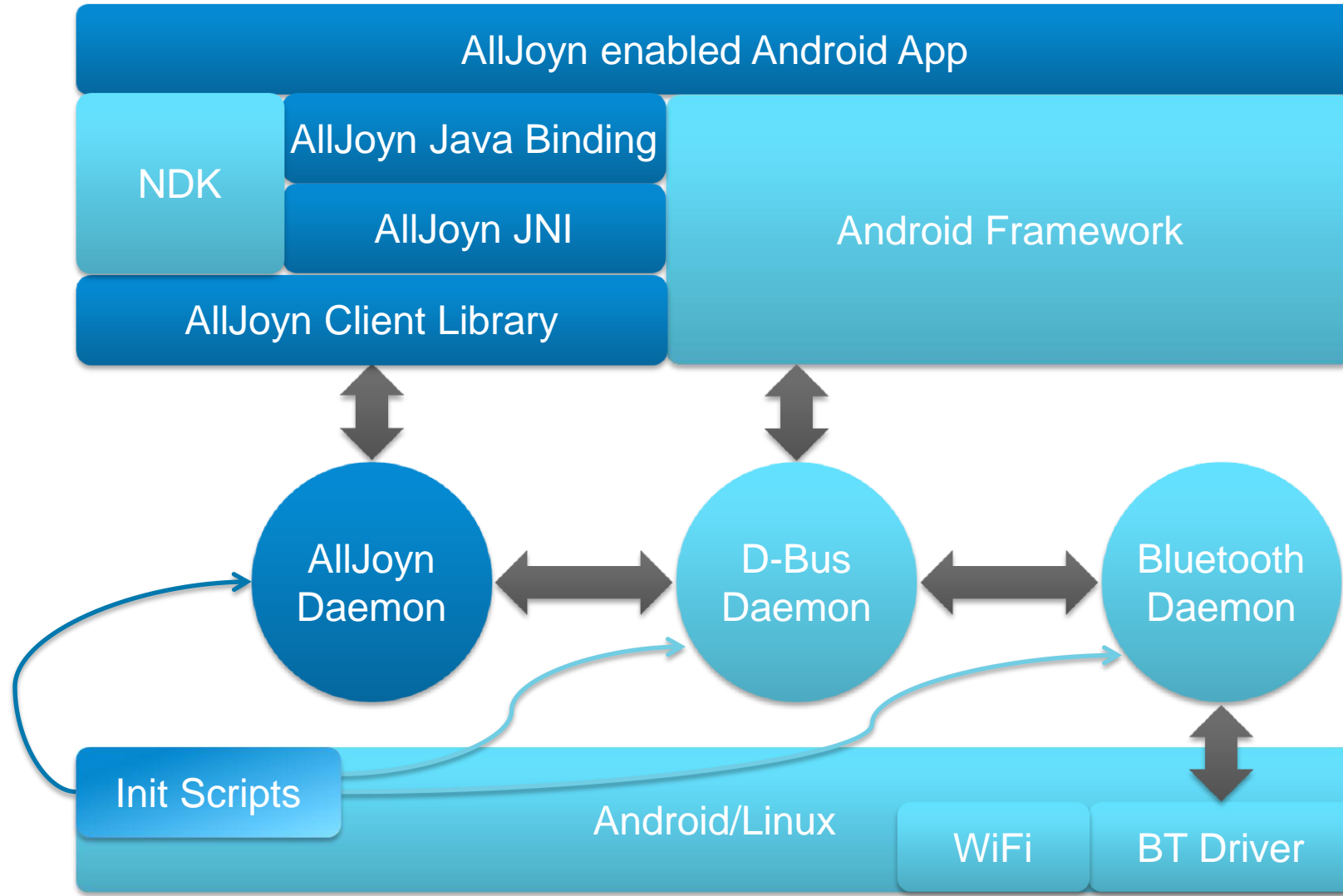
Deployment Options

AllJoyn applications require the daemon functionality

There are three options for deployment on Android:

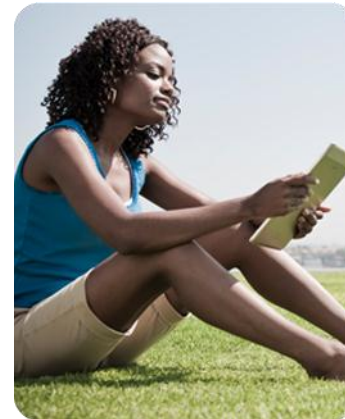
- Platform Integration
 - Standalone daemon for system, can use WiFi as well as BT
 - Started at system startup via initialization scripts
- Downloadable APK Daemon
 - Single daemon for system, restricted to WiFi only
 - Launched via intent
 - Available in 2.3 release, but deprecated with 2.5 release
- Application Integration
 - Daemon functionality resides in the application
 - Also launched by intent
 - Will only be used if neither of the other two is available
 - Like the APK: restricted to WiFi only, but will support WiFi Direct in near future
 - Each application will full Alljoyn functionality

AllJoyn Platform Integration



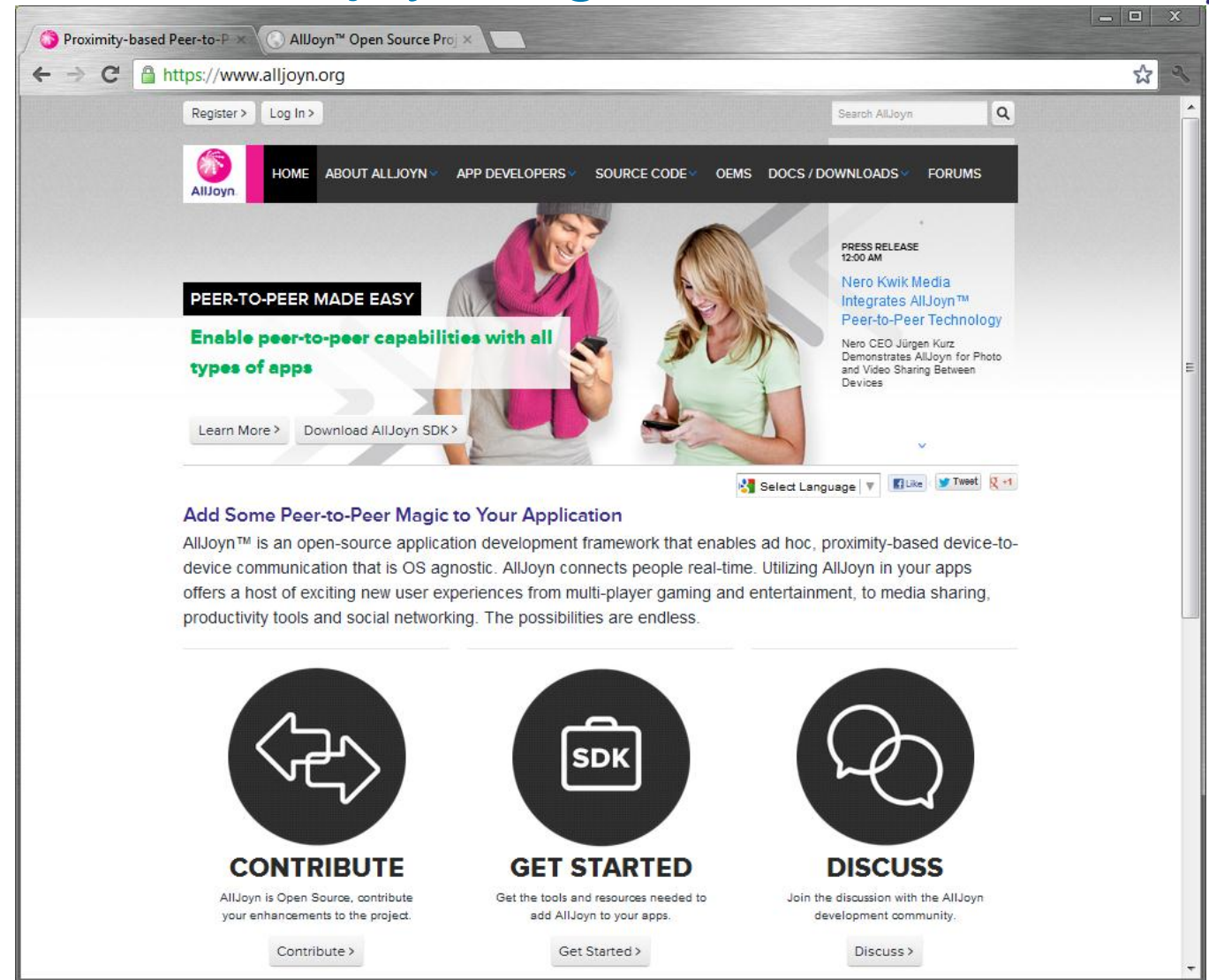
Availability and Open Source

Open Source. Open Possibilities.



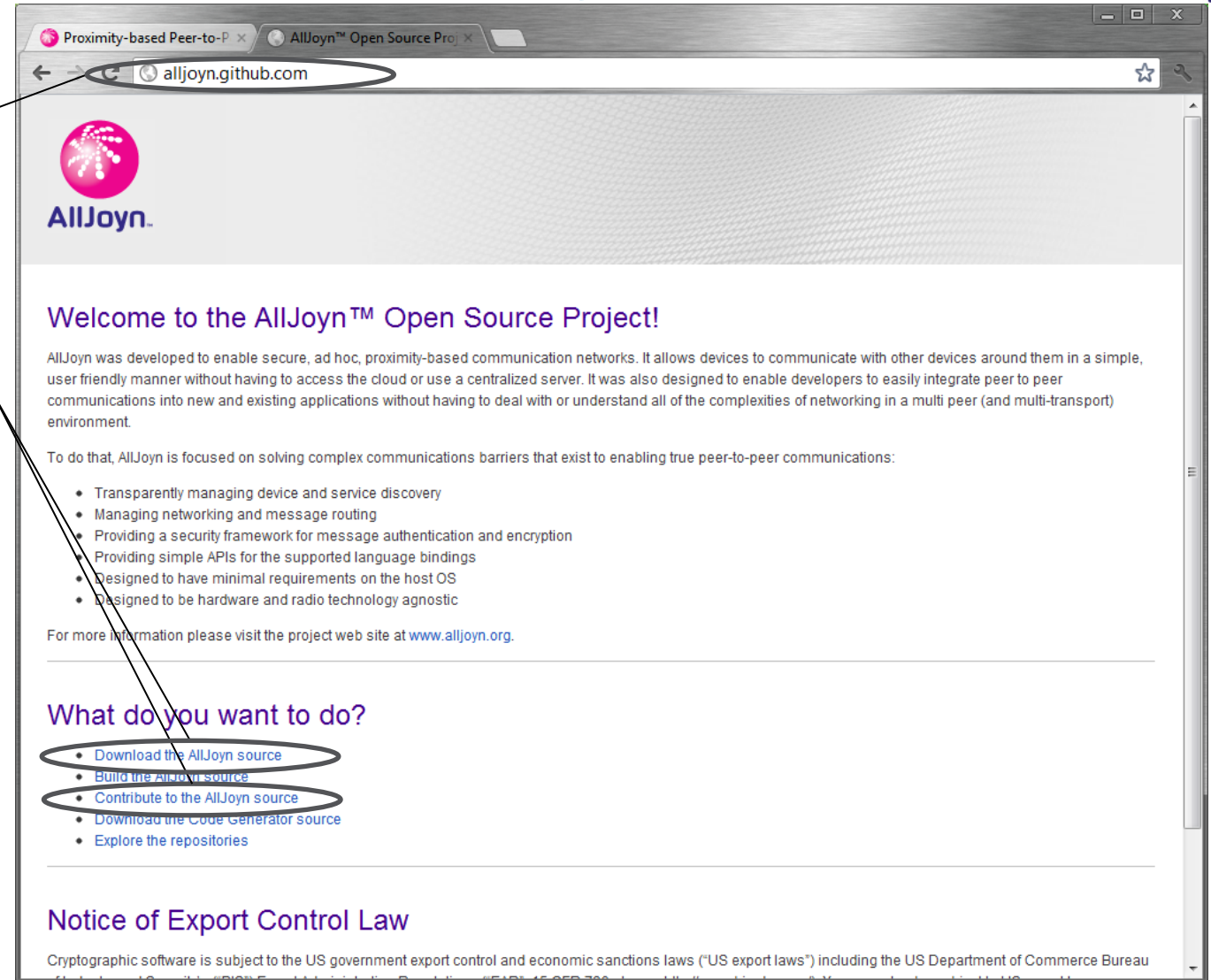
AllJoyn Open Source Project: www.alljoyn.org

- AllJoyn is an Open source project
- Source available on GitHub
 - <http://alljoyn.github.com>
- Accepting 3rd party contributions
- Binary SDKs available on alljoyn.org
 - Currently have Android, and Windows
 - Recently released 2.6
- Licensed using Apache 2.0
 - Flexibility to use and modify



AllJoyn Open Source Project: www.alljoyn.org

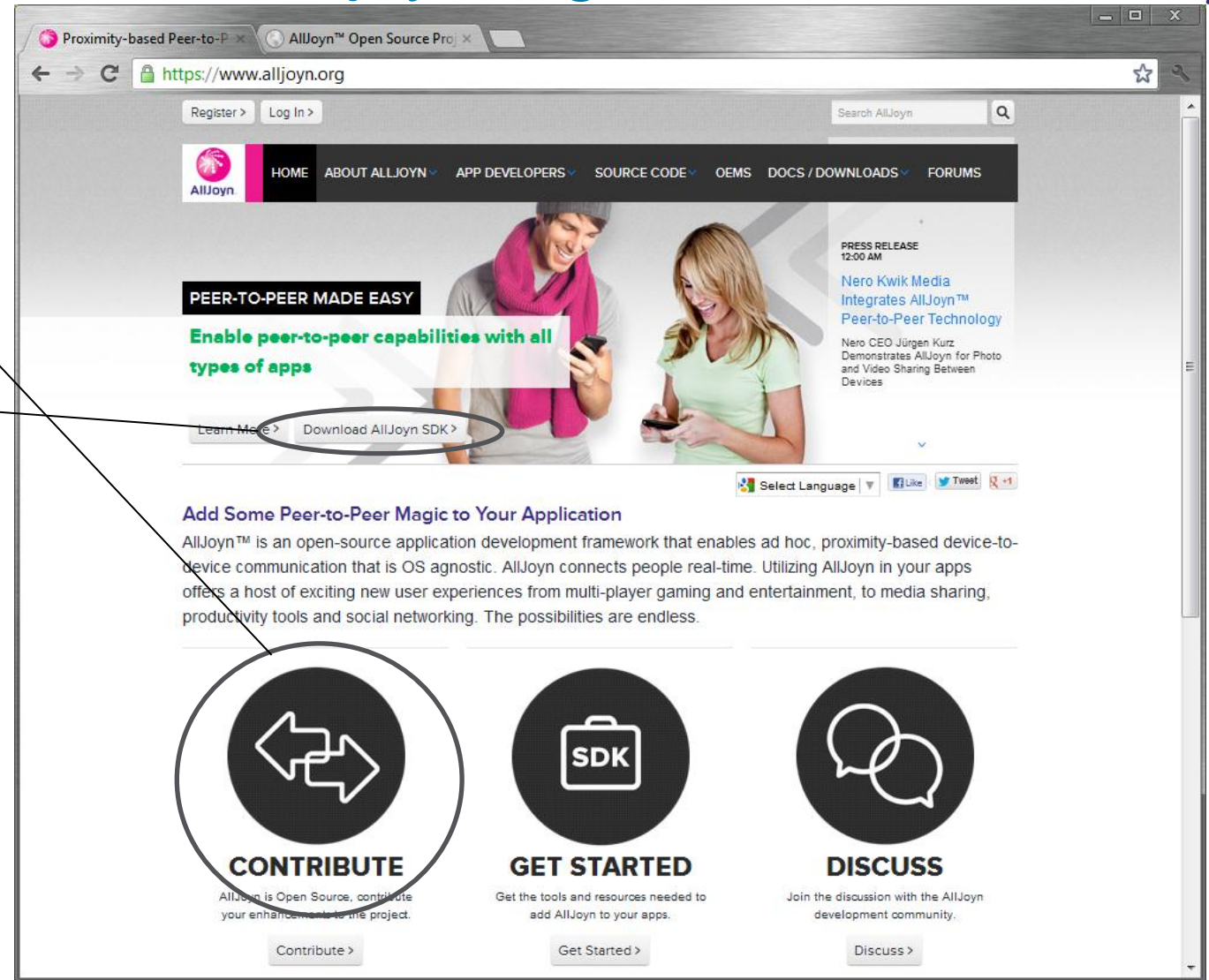
- AllJoyn is an Open source project
- Source available on GitHub
 - <http://alljoyn.github.com>
- Accepting 3rd party contributions
- Binary SDKs available on alljoyn.org
 - Currently have Android, and Windows
 - Recently released 2.6
- Licensed using Apache 2.0
 - Flexibility to use and modify



The screenshot shows a web browser window displaying the AllJoyn Open Source Project page on GitHub. The browser's address bar shows alljoyn.github.com. The page features the AllJoyn logo and a welcome message. A list of features is provided, followed by a link to the project website. Below this, a section titled "What do you want to do?" lists several actions: "Download the AllJoyn source", "Build the AllJoyn source", "Contribute to the AllJoyn source", "Download the Code Generator source", and "Explore the repositories". The first three items are circled in blue. At the bottom, there is a "Notice of Export Control Law" section.

AllJoyn Open Source Project: www.alljoyn.org

- AllJoyn is an Open source project
- Source available on GitHub
 - <http://alljoyn.github.com>
- Accepting 3rd party contributions
- Binary SDKs available on alljoyn.org
 - Currently have Android, and Windows
 - Recently released 2.6
- Licensed using Apache 2.0
 - Flexibility to use and modify





AllJoyn

質問は?

Questions?

问题?

Ερωτήσεις?

Questions?

Fragen?

Вопросы?

Spørsmål?

Questions?

Domande?

Questions?

Vragen?

Questions?

질문이 있습니까?

Cwestiynau?

Spørsmål?

問題?

FRÅGOR?

¿Preguntas?

Kysymyksiä?



Thank You!

Open Source. Open Possibilities.



Qualcomm Innovation Center, Inc.

5775 Morehouse Drive
San Diego, CA. 92121-1714
U.S.A.

Copyright © 2012 Qualcomm Innovation Center, Inc.

All rights reserved.

Not to be used, copied, reproduced in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Innovation Center, Inc.

This technical data may be subject to U.S. and international export, re-export, or transfer ("export") laws. Diversion contrary to U.S. and international law is strictly prohibited.

AllJoyn, QuIC and the QuIC logo are trademarks of Qualcomm Innovation Center, Inc. Other product and brand names may be trademarks or registered trademarks of their respective owners.

Nothing in these materials is an offer to sell any of the components or devices referenced herein. Certain components for use in the U.S. are available only through licensed suppliers. Some components are not available for use in the U.S.

[6/20/2012]