



AllJoyn[®]
Open Source Project

Technical Steering Committee

November 28, 2016





Reminder:
**This call is being
recorded**

Antitrust Compliance Notice

- AllJoyn Open Source Project (AJOSP) meetings involve participation by industry competitors, and it is the intention of AJOSP to conduct all of its activities in accordance with applicable antitrust and competition laws. It is therefore extremely important that attendees adhere to meeting agendas, and be aware of and not participate in any activities that are prohibited under applicable US state, federal or foreign antitrust and competition laws.
- Examples of types of actions that are prohibited at AJOSP meetings and in connection with AJOSP activities are described in the Linux Foundation Antitrust Policy: <https://www.linuxfoundation.org/antitrust-policy>



Agenda

1. Approve minutes from previous meeting
2. Core 16.10 update
3. OSU test update
4. Base Services 16.10 update
5. CDM 16.10 Update



Core 16.10 Update

Core 16.10 is complete!

- Delays due to test related challenges and development issues
 - Challenges mainly around the “unknown unknowns”
 - Due to test challenges serious issues surfaced late in the game
 - Many blocked test and were difficult to resolve
- Done now – ready to tag
- Question is: given the change in process around the relationship of AJOSP and OCF it isn't clear what tags should be applied



OSU Testing Update

October 2016

Carrie Hertel, CASS

Resourcing – October 2016 Hours

| Person | Hours Worked |
|-----------------|--------------|
| Carrie Hertel | 92.32 |
| Hayden Anderson | 29.75 |
| Tyler Gilbert | 77.54 |
| Omkar Thakur | 73.51 |
| Total Hours: | 273.12 |

October invoice total: \$ 12,790.40

Invoiced through Oct 31, 2016: \$ 147,432.44

Remaining Balance on Contract: \$ 52,567.56

Contract Expiration: 12/31/17

October Work Items

- Continued execution on 16.10 release
 - 51Jira issues logged
 - 14 of these are test plan issues
- Added Hayden Anderson to the team

November Plan

- Complete 16.10 release testing
- Continue logging test plan improvements in ASATEST Jira project
- Update Test Cases in Test Link from learnings during 16.10
- Continue WebDocs reviews and updates



Common Frameworks

Base Services

16.10 release update

Roles

- Development Lead - Nathaniel Garbutt supported by Brian Hollister
- QA Lead - Brett Rogers

16.10 Base Services milestones

- ~~October 19th: 16.10 Base Services Feature Freeze~~
- ~~November 2nd: 16.10 Base Services Code Freeze~~
- **November 22nd: 16.10 Base Services QA Complete**
 - December 2nd
- **November 23rd: 16.10 Base Services Release**
 - December 5th

16.10 Base Services release plan

- **16.10 Core**
 - ~~Ensure that Base Services builds and runs with 16.10 Core~~
- **Security 2.0**
 - ~~Ensure it is integrated into samples and tests~~
- **Onboarding**
 - ~~Investigate reliability issues~~
 - ~~Fix some interoperability issues between samples on different platforms~~
 - ~~Include Claiming as part of on boarding (Stretch)~~
- **Certification**
 - ~~Update C&C criteria~~

16.10 Base Services issues

- **There was 41 open issues with P3 or above**
 - 15 Deprecation tasks
 - 13 Important bug fixes or tasks
 - 13 nice to have bug fixes (Stretch)
- **There are now 18 open tasks with P3 or above**
 - 6 issues remain (3 are for testing)
 - 10 have been pushed to 17.04 (6 of which are the removal of recently deprecated code)



Smart Spaces

Common Device Models

16.10 release update

Roles

- Development Lead - Aaron Vernon supported by Nathaniel Garbutt
- QA Lead - Brett Rogers

16.10 CDM milestones

- **Dec 2nd:** 16.10 CDM Code Freeze
- **Dec 16th:** 16.10 CDM QA Complete
- **Dec 19th:** 16.10 CDM Release

16.10 CDM features & improvements

- **All new interfaces have been reviewed and +1'd** - Light, Lock, Switch, Door Sensor, Window Sensor, Smart Plug, Speaker, Water Sensor, Smoke Detector, Motion Sensor.
- **New utilities** for both C++ and C code bases to improve ease of use and clarity for OEMs
 - CDM_System - Manages Alljoyn
 - CDM_AboutData - Manages AboutData and announcement
 - CDM_Security - Manages security and loads certificates
- **Improved directory layout** of code base to enhance separation between controller and controllee code.
 - Code is compiled into individual libraries
 - Common for both the framework and interfaces
 - Controller for both the framework and interfaces
 - Controllee for both the framework and interfaces

16.10 CDM features & improvements

- **Controller Framework** has sample implementations for iOS, Android and Desktop (Qt)
 - Can control all the 16.04 and the 16.10 CDM interfaces
 - Preliminary implementations of groups and presets
 - Customisable translation layer used for backwards compatibility with older standards/interfaces
 - LSF is currently used as an example implementation
 - Increased stability with multi-threading fixes and improved object life time management
 - 100% code generation for all features in 16.04 and 16.10 interfaces in C++ (Java is very close too)

16.10 CDM features & improvements

- **Controllee Framework** for standard and thin cores
 - API improvements for OEMs
 - Clearly named methods for emitting change signals on hardware modification
 - Methods can now alter the state of other interfaces and emit change signals to notify controllers
 - Simplified API for creating interfaces
 - No more distinction between CDM interfaces and vendor defined interfaces
 - 100% code generation for all features in 16.04 and 16.10 interfaces in C++ (thin core is close)
 - New samples that simulate hardware integration
 - Upgraded the device emulator

16.10 CDM remaining work

- Finish code generator templates for C and Java
- Upgrading thin core CDM interfaces with new API
- Device Emulator for thin core is in progress
- A few extra samples for both standard and thin core

16.10 CDM breaking changes

- Code layout change breaks previous includes
 - But it drastically improves the ability for more granular linking and paves the way for plugins
- During design discussions, previous naming became an point of confusion so we revisited the names of some classes
- We altered the controllee side *CreateInterface* method. It is now templated instead of using an enum identifier and a factory.
 - This avoids linking in all the unused interfaces
- The public facing API for interfaces has been altered so setting values and emitting change signals are separate ideas
 - This is so the API is capable of emitting change signals from methods that alter state.
 - This is a big change but code generation should help mitigate this
- Removed vendor defined interfaces API

16.10 CDM issues

- **There are 3 open bugs with P3 or above**
- **There are 37 open tasks with P3 or above**
 - 21 are testing related



Thank you