



**ALLSEEN
ALLIANCE**

Core Working Group Meeting

The contents of this document, and the interfaces described, and all the information herein, are the result of collaborative discussions by the Core Working Group. This summary documents the final consensus of the team.



Reminder:
**This call is being
recorded**

Antitrust Compliance Notice

- AllSeen Alliance meetings involve participation by industry competitors, and it is the intention of AllSeen Alliance 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 AllSeen Alliance meetings and in connection with AllSeen Alliance activities are described in the AllSeen Alliance Antitrust Policy. If you have questions about these matters, please contact your company counsel, or if you are a member of AllSeen Alliance, feel free to contact Lee Gesmer or Andrew Updegrove, of the firm of Gesmer Updegrove LLP, which provides legal counsel to AllSeen Alliance.

Agenda

- AJATS discussion
 - TSC approved getting this back in place.
 - Josh investigating/coming up with plan
- C&C discussion
 - Moving forward
- Documentation Testing Proposal
 - General consensus that the proposal is the right way to go
 - Process is that if committers see a change that likely impacts documentation they should file a Jira ticket (task) to investigate, and fix if there is work to do
- Patch release:
 - What is going to be in there other than iOS updates?

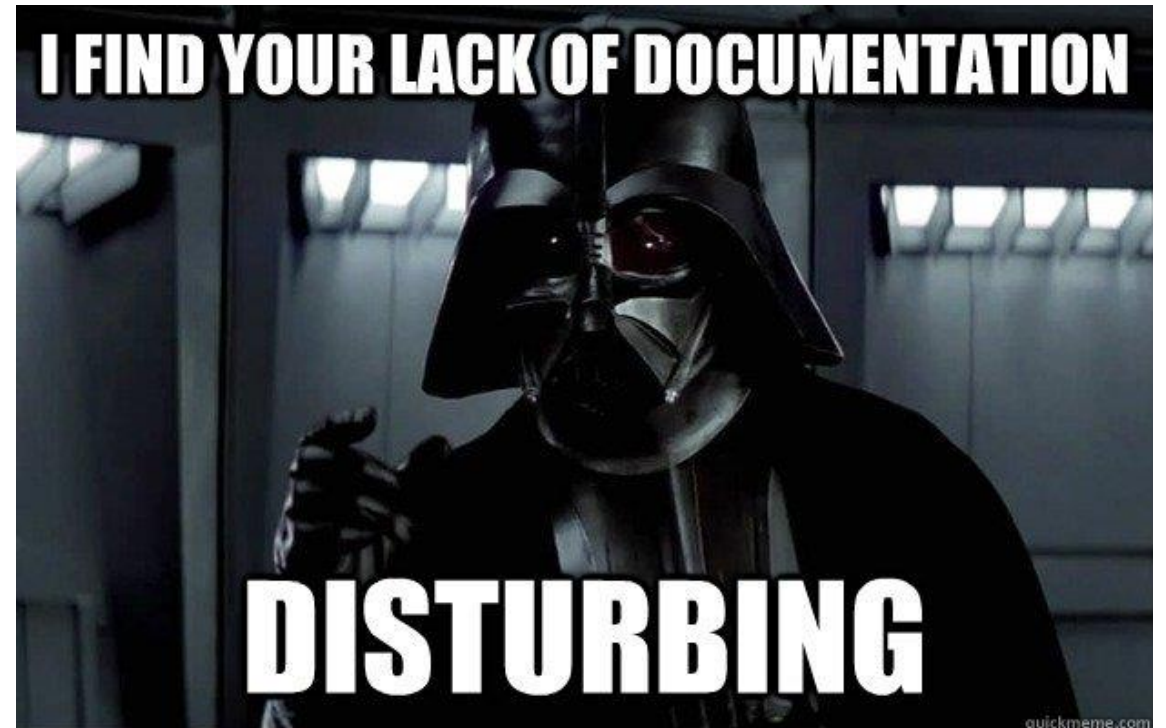


Documentation Testing Proposal

Josh (Affinegy)

Documentation Needs Testing

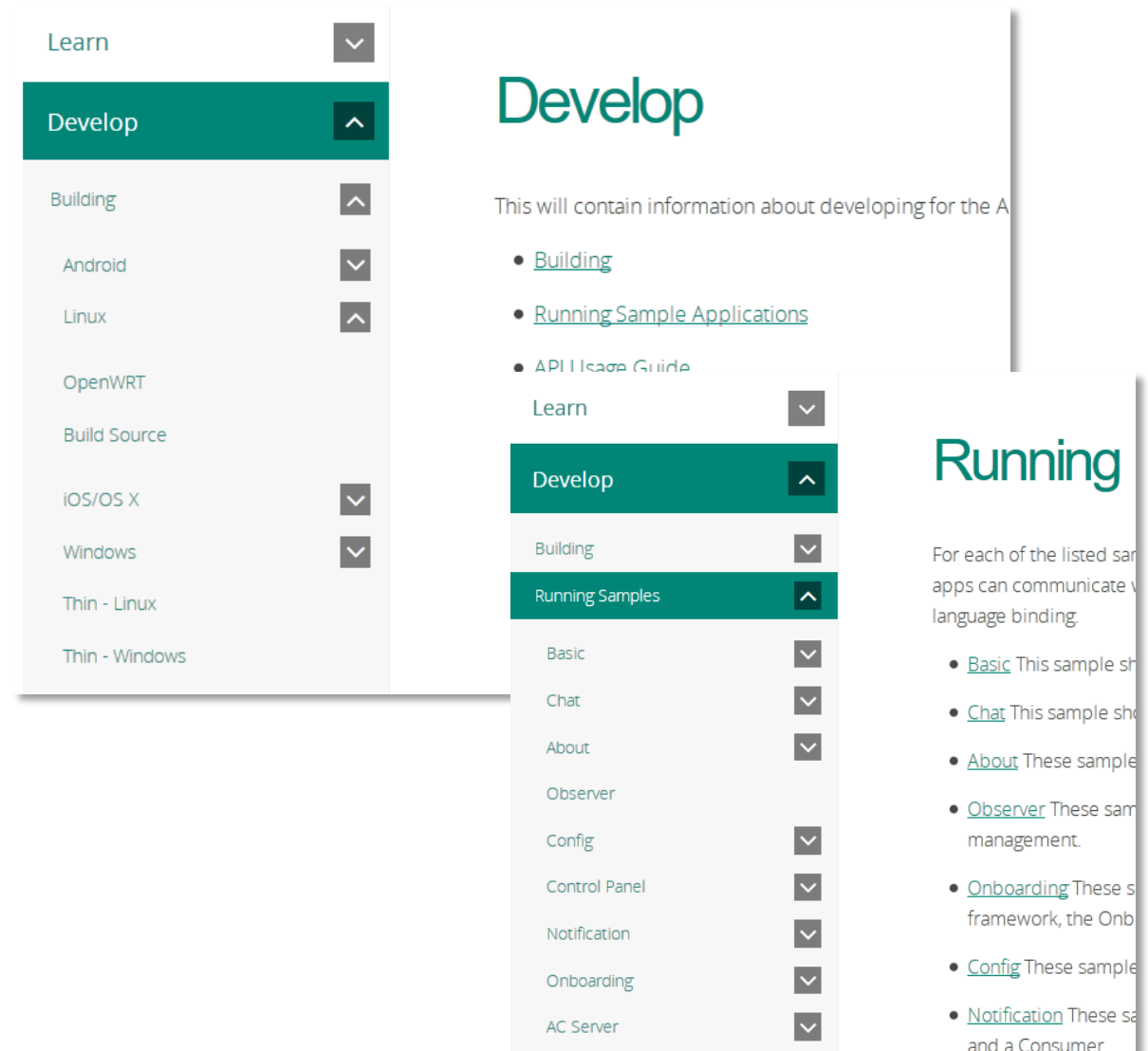
- Documentation is IMPORTANT!
 - It's the first tangible experience that a developer has with AllJoyn
 - It needs to be professional and up to date
- Our current documentation is woefully out of date!
 - Still apparently assumes Ubuntu 12.04
 - None of the tutorials/recipes work “out of the box”
- We need someone reviewing it as part of our dev/test cycle.



Barriers

- Time during release testing
 - There is *a lot* of documentation
 - Even the specific subset of Tutorials/Step-By-Step instructions is significant
 - QA team cannot afford to focus their efforts on documentation during normal release testing
- Current test agreement does not directly include documentation
 - The closest is this:

“The Test Team will also work with the Alljoyn community to identify gaps in test processes and scripts, and help the community write and implement more complete and robust test procedures.”



Proposal

- Documentation pass *after* every release
 - Begins immediately after the release is completed
 - Documentation bugs are filed as issues are discovered
 - Full documentation pass
 - One or two AllJoyn developers read through the latest documentation for a “sanity check”
 - OSU reviews/tests the “hands-on” documentation (e.g., “building for xyz”, tutorials, etc.)
- Work with the TSC to get approval for OSU to add documentation to their test procedures





Action Items

Action Items

- Proposal to outline the process for changing APIs
 - Has been handed over to Brian Rockwell
 - will craft proposal for review by Core WG and then presented to TSC
 - Expectation that this will be reviewed with WG in January.
- Arvind (MSFT) to create Wiki process page linked off of Core WG wiki
 - Remove references to java from mandatory binding list for 16.04 timeframe
 - Hope to have something by the end of Jan.

Action Items

- Add a way in JIRA to track compatibility issues and proposals
 - Proposal is the use a label: “compatibility impact” in the Jira ticket
 - Committers should require for these sorts of change that the release notes be updated
 - **Marcello** to document this process in the Core Process wiki
- Define process to require regression/unit tests for bugs
 - **Marcello** to add description of Committers requiring unit test for bug fixes to Core Wiki
- Define process for handling “Technical Debt”
 - Example: Took a shortcut to make a release and not loosing track of this to fix the shortcut
 - Jira items to track when things are missing – e.g. feature added, but missing from Java binding
 - If shortcut is taken: could create a Jira task to track the problem



Thank You

Follow Us On      

- For more information on AllSeen Alliance, visit us at: allseenalliance.org & allseenalliance.org/news/blogs