Jakarta Pre-Wire Summary

June 4, 8am-11am PDT

Attendance

Some people may have joined the meeting after attendance was first captured.

Ireland High Priority Items

None to discuss. Release looks to be going smoothly so far.

Ireland 2.0.1 Patch Release

- Initially agreed upon by the TSC for later this summer (tentatively August) – per 3/24/21 TSC meeting.
- While there could be a patch release near the middle/end of August, the TSC will evaluate month to month on the necessity of a patch release.
- The initial driver for a patch release was the creation of tools/documentation for any migration and to release bug fixes which were assumed to be needed given the 2.0 nature.
- Documentation on migration is a must on config, migration, profiles.
    - This is considered an independent deliverable and needs to be accomplished ASAP.
    - Documentation is needed for changes (and migration) on configuration files (TOML), device files (new configuration for devices), database migration (from V1 to V2 document stores), and device profiles
- Development of migration tools, outside of documentation, will be considered by the TSC once the documentation establishes the size, scope and potential needs for tools.
- TSC to evaluate release of patches and other independent grouping of releases month to month until 2.1 is released
- Bugs that are deemed high level of severity or better (high is something that doesn't have a work around and critical are bugs that causes system failures) will trigger a patch release. Low severity bugs are cosmetic in nature (ex: spelling mistake). Medium severity bugs are issues where there is a work around for the problem.
- The following independent items will likely release this summer (2021). They can be released as needed in coordination with the working group chair, release czar and product manager.
    - Any Device Services not upgraded to V2 APIs (GPIO, CoAP, LLRP)
    - Any Application Services not upgraded to V2 APIs (LLRP)
    - EdgeX examples not upgraded to V2 APIs (in edgex-examples repos)
        - Those not going to be upgraded should be annotated explicitly as 1.x only

- o New GUI
  - o CLI V2 (refactor and upgrade to V2 APIs)
  - o C device services released will include the following updates:
    - Event tagging in C Device Services (done for Go DS)
    - Security implementations for C DS (secure Consul, secure message bus)
- If there are a number of independent release items and a patch release to go out, this might be coordinated so that everything releases together.
- If there is a patch release, additionally there may be removal of V1 or otherwise "dead" code that was not handled for the Ireland release.
  - o This should not be a trigger for the patch release.
- Two pieces of work that we are hoping will still get done by V1 release, but if they don't would get pushed to Jakarta or a patch release
  - o Early seeding of secrets without calling the endpoints (which isn't ready yet) – work in progress even in design yet along with implementation
  - o Readings by timestamp and name (per Slack channel). (#3519)
- IOTech's DCT tool for use with EdgeX Ready – working with V2 profiles; may not even be ready by August – not part of the patch release anyway

Jakarta Release

General Scope Question

- What is this release about? What is the major theme(s)?
- It is desired that the Jakarta release be an LTS release.
  - o There are questions if LTS is the right term for this?
  - o There are questions about how long we support, and what support means.
  - o **This will be a topic at the planning meeting. As part of this discussion we will look at:**
    - **What is our LTS cadence?**
    - **How long we support a release**
    - **This is documented but the planning meeting will open reconsideration or reaffirmation of the documented approach.**
    - **For homework the community is asked to review https://wiki.edgexfoundry.org/display/FA/Long+Term+Support**
    - **Mike Johanson also provided these additional links for consideration as part of the planning meeting**
      - **https://github.com/kubernetes/community/issues/2720**
      - **https://nodejs.org/en/about/releases/**
- In addition to the Jakarta release as an LTS, the focus of this release cycle will be on "stability." This means:
  - o No major architectural changes: adding new endpoints, enhance a service (but not how a service works, not how configuration is organized)
  - o Non-backward compatible features are excluded
  - o More focus on testing on areas that are not the default
  - o And robustness -> restarting of services, or environmental (ex OS, or hardware) conditions may show issues with 2.0

- While this Jakarta release is expected to be a "stability release", the community did agree that this release could be used to design out/experiment with new implementation features so long as they are feature flagged, could not impact the stable implementation, and do not bloat the service size.
- There was discussion around the question of if we have an existing feature with a bug that needs a non-backward compatible fix, what does that mean?
  o This does trigger a major release (due to non-backward compatibility).
  o To solve it without a major release, we may have to have create a feature flag patch (in other words fix with proverbial bailing wire, chewy gum and duck tape) that is backward compatible for current release.
- There was also a discussion on consideration of the premise that Jakarta is LTS, is Kamatura (or any minor release) an LTS by extension as well?
  o Once Jakarta is out we will have a 2.1
  o If we make Jakarta an LTS (desired), we will have to tag this release and branch at the time there is any bug that needs to be worked.
  o At that time, any bug fix work would have to be done on the 2 forks (main branch and the LTS Jakarta branch).
  o Any patch release on Jakarta (2.1) would be LTS by default
  o We start Kamakura work on the main branch
  o Whether it is 2.2 or 3.0 or whatever – it is not, by default, LTS
  o It would include any patches from the Jakarta patch fork, but is not LTS itself unless TSC says it is

Release Timeline
- We'll set formal target dates in the planning meeting.
- General agreement with an early November release timeline for Jakarta.

Potential Scope Exploration
Based on exploration of Iceboxes, Roadmap lists, etc.

Items in red strikethrough are postponed beyond Jakarta.

Items in **green bold** are considered and more likely to be implemented for Jakarta at this time

Items in gold are to be considered but need more information, context and consideration during the planning meeting. If you will, these are tasks that are more generally "on the bubble" and may be excluded for Jakarta.

Architecture Topics for Consideration

- Implement Device Service Filter ADR – No, resourcing issue
- Implement Service List ADR
- Implement Registry ADR
- Implement Metrics ADR – No, this is big and would be pushing things on stabilization
- Implement UoM ADR
- Solve System Management issues
- Digital Twin – probably out unless DTC just needs what we have with small tweaks

- For the two ADR's under consideration for implementation with Jakarta, there is a concern that we can address these in backward compatible fashion? Have we missed the boat until 3.0?
  - Perhaps we use Jakarta to provide "Beta" – first get an actual design, then try an implementation in beta form (experimental feature/prototype) with priority (in order) to:
    - Service list
    - Registry
  - We should get feedback from community if anything else rises to level of Jakarta scope
  - There is also a question on whether there would resources to complete any of these efforts.

New Features

- General
  - **Enable CORS (for API Gateway and service APIs in general)**
- DevOps
  - **DevOps list –Ernesto to provide prior to planning**
- Core/Supporting
  - **MaxEvent Size a config setting – there now but not being used (DS use?)**
  - Improve ability to add interval/interval actions statically
  - Send commands to group of devices based on label
  - ~~Add NATS implementation for message bus – too big~~
  - ~~Add Websocket support for notifications~~
- Device Services
  - Support automatic migration of devices between device services
  - **Add record and replay function (new device service) – low impact with new device service**
  - Allow for dynamic updates to device profiles
  - ~~Event Filtering in Device Services (per ADR)~~
  - Implement Validation of ProtocolProperties – without breaking change?? May be hard for backward compatible
  - ~~Allow multiple events with readings to be created from DS~~
  - ~~Enable resource command chaining – nesting capability of resource groupings – no push for this~~
  - Support regular expressions in assertions
  - ~~Overwrite for profiles / devices loaded from disk – want a reload profile option (without delete and new add)~~ – **Action Item: Jim close this issue**
  - **Always send CBOR for performance (test first??) – do for testing; then depending on results change the default**
  - DS SDK Go – update last connected – **Action Item: Jim research for planning meeting**
  - **Device stanza in configuration – but not implemented; should drop them (init commands and remove commands as example)**
  - ~~From Brad C – use device camera enhancements~~ – but for consideration/contribution by Intel/IOTech (stretch goals by contributing companies)

- o ~~Digest authentication on device camera~~ – requires use of external digest auth package, but need to add issue to track this
- **Security**
  - o **Micro service authentication – ADR first; implementation is out**
    - ▪ **Build a secret store unsealing daemon (#1944)**
    - ▪ **Build micro service communication security (service to service - #1950)**
    - ▪ **Enable Vault identity Secrets (#2456) – another way to create JWTs**
  - o Enable signing of EdgeX images for Docker Hub (#2470)
    - ▪ Would require work with sigul
  - o **Static code analysis with Go code (LGTM replacement)**
- **App Services**
  - o Fork the pipeline
  - o Flogo integration (awaiting Flogo team)
  - o ~~CloudEvent as TargetType in Pipeline configuration~~
    - ▪ ~~Relook size of CloudEvent libs~~
  - o ~~Provide example live camera watching Application Service~~ – **Action: Jim to close this**

Tech Dept/cleanup

- **General**
  - o **Go next versions**
  - o **Upgrade of 3rd party stuff – Action: Jim to research possible upgrades**
  - o **Go through dependencies for any LTS versions of 3rd party stuff**
    - ▪ **What if we have to go backwards**
    - ▪ **Wrap into LTS discussion**
  - o **Go Sum – committed in repositories**
  - o **Upgrade YAML library to v3 (vs v2) – research needed and impact understood.**
  - o Replace poor 3rd party libs based on paper study (not supported, not used, etc.)
    - ▪ Includes ZMQ, logfmt, goburrow for modbus etc.
    - ▪ Possibly on a case-by-case basis; those with low impact to stabilization
- **DevOps**
  - o **Do we need to archive (store in an EdgeX repo) anything so we can build in the future (availability of artifacts like Go, Consul, etc.) – Ernesto to do some research /thoughts on this for planning**
- **Core**
  - o ~~Allow separate PublishHost and SubscribeHost in go-mod-messaging MQTT support – probably not for Jakarta~~
  - o ~~Fix SMA's default service list (also part of general system management and service list ADRs)~~
- **Security**
  - o ~~Address defect in secret store paths (#2465)~~ – Action: Jim to close
  - o ~~Configuration of root CA file is inconsistent across services (#2495)~~ – Action: Jim to close
  - o ~~Tag service location and secretstore config to not be in Consul (#2868)~~ – Action: Jim to close

- o ~~Refactor the anonymous function inside secretstore-setup for init Vault (#3182)~~
  - ▪ ~~Critical piece of code we don't want to touch unless we have to~~
- o **Service restarting (security services at large)**
  - ▪ **Address Dynamic services**
  - ▪ **Address stability concerns**
- o **Phase 3 of Consul implantation – (Security team to provide details on what this is – and how we'll do it for dynamically added services)**
  - ▪ **Implement the ACLs**
- App Services
  - o **Integration tests for Store and Forward**
  - o **SDK hangs when calling docker-compose stop – can't recreate – need to disposition whether this is an issue or not**
  - o ~~Unit testing error cases rely on string literals – low priority~~
- Snap
  - o Redis in the snap triggers apparmor denials
  - o Modify install hook to drop usage of SNAP_INSTANCE_NAME

Future considerations – not for Jakarta but for the community to consider and scope long term

- Get rid of ZMQ – discuss at planning
- Get rid of YAML – discuss at planning

Project Governance, Processes, Organization Topics to Discuss at the Planning Meeting

- **PRC Evangelism, how to support, how to leverage, etc.**
- **Monthly Architects Meetings (continue or alter or stop)**
  - o **Meeting schedules in general**
- **Outreach organization (working, not-working, room for improvement)**
  - o **Vertical Liaison (more Agstacks?)**
- **Certification (Proposed)**
  - o **Add support for v2 device profiles acceptance to EdgeX Ready**
  - o **Expand supported device profiles for EdgeX Ready for which:**
    - ▪ **An EdgeX reference implementation device service exists**
    - ▪ **We have a validator**
  - o **Continued support for v1 device profiles in EdgeX Ready**
  - o **Improve website for EdgeX Ready approved device profiles**
  - o EdgeX Certified program on hold until an LTS candidate identified
- Adopter Series (continue, alert, stop) – keep it, but no additional discussion necessary