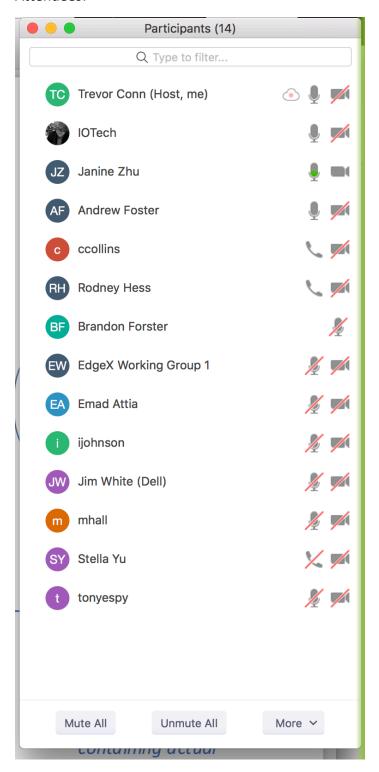
Core Working Group Agenda (19-July-2018)

Attendees:



Old Business

- Update on public cloud Export Connector Endpoints
 - Parent LF account registered with Azure, invoicing and account mgmt. is being configured
 - Suggestion from Brett/Eric that EdgeX community create account in Google Cloud. Once that's done, it can be linked to LF's invoice system.
 - Thoughts? Need someone to take this on → Andy Foster to follow up with Brett/Eric
- Binary Serialization
 - CBOR is the recommendation, vote confirmed by TSC
 - o Trevor deliverable Present diagram for implementing natively end to end
 - South to northbound
 - Account for protobuf blob as reading inside of the reading
 - Overall reading encoded as CBOR (proposed)
 - Hitachi confirmed MQTT would be the primary device protocol, but could also be
 HTTP
 - The attributes shown on the diagram will be part of the binary payload, not individual value descriptors
 - We won't be adding new fields to capture these elements
 - o MIME-Type in a value descriptor? May not be feasible in the existing objects.
 - o Discussion on persistence of binary payload as base64 within Mongo
 - Is this feasible to store in the same catalog with other events?
 - Research capabilities of blob handling in Mongo
 - Possible need for fault tolerance. What about data loss?
 - No store and forward capability right now in EdgeX
 - Hitachi has requirement for this.
 - Replacing 0MQ w/MQTT could add value here
 - Storage size implications. Configurable setting?
 - Will there be queries on the actual binary data?
 - Question regarding just passing through CBOR from MQ to upstream cloud.
 Depends on whether enrichment is required.
 - Incorporate feedback and revisit at next meeting. Discussion of possible timelines.
- Package restructuring is underway for separation of external/internal models and encapsulation.
 - Core/support services have been moved.
 - Export services still need to be moved.
 - Just starting work on domain >external model translation
 - Haven't forgotten Tony asked to review

- Confirm future direction for Docker deployments to NOT comingle dependent platform with EdgeX service into one container
 - Example Consul + Config-seed
 - Example Mongo + Mongo-seed
 - Work item to split these images into separate Docker containers
 - https://github.com/edgexfoundry/developer-scripts/issues/65
 - Tag as help wanted (Jim)
 - Update architectural guidelines to require separation of platform from init services
 - Not done AFAIK
- Export services logging is different from other services
 - See https://github.com/edgexfoundry/edgex-go/issues/328
 - Also, services do not log to the shared volume when deployed via Docker
 - No objection on last call to standardizing logging service integration in Export for Delhi. Make it so!
 - https://github.com/edgexfoundry/edgex-go/issues/328
 - NOTE: local service logging that writes to the file system could be mapped to the volume file system, same as support-logging
- Discuss Java service support and POM dependency mgmt.
 - POM versions weren't even incremented for services not included in California release
 - By Delhi, these services' APIs will be out of sync with RAML and Go implementation.
 - Updates to address this are not on anyone's roadmap
 - Agreed to abandon the platform services (??) Needs TSC vote.
 - Device Service Go/C SDK with examples are required.
 - Agenda Item for DS WG Meeting, possibly TSC
 - Any ideas for additional resources to get us through this push?
 - Jim might have more information next week.
 - Targeting for Delhi is to have sample device services in C/Go, eliminating need for legacy Java device services
 - Remove for now

New Business

- FYI First draft of configuration V2 structure is done. May be possible to present to the community next week for comment.
 - Vault/Consul interaction and handling of secrets is an open question
- Scope of 0.6.1 release
 - Scheduler

- o Notifications
- o DevOps
 - ARM artifacts
- o Code commit end of July, release early August?