

Core Working Group Agenda (30-Aug-2018)

Attendees:

Old Business

- Work to merge core-config-seed-go into edgex-go is underway. Targeting current feature parity before making Config V2-related changes.

New Business

- Golang 1.11 released
 - When do we update?
 - Pre/post-Delhi
 - Effect on CI, if any?
- External pkg/clients and /models file name inconsistencies
 - What guidelines should we adhere to for file names?
 - Single go file to contain one type only (per Java) or multiple related types (per intent)
 - Lots of guidance around package names, not so much file names
 - https://golang.org/doc/effective_go.html#names
 - <https://github.com/golang/go/wiki/CodeReviewComments#package-names>
 - @ GopherCon only thing I've heard so far is avoid "stutter"
 - Schedule/schedule.go
- Recap Delhi deliverables
 - <https://wiki.edgexfoundry.org/display/FA/Delhi>
 - Completed
 - Pending
- Updates from Persistence working group
 - 1) For Edinburgh release, the core dev team (that would be Trevor's) works to implement a layer/abstraction system for databases. Allowing any database to be used under the covers more easily.
 - Largely in place via interfaces, however decoupling of Mongo from models and the interfaces themselves is required.
 - 2) For Edinburgh release, we implement 2 options for the EdgeX reference implementation: a MongoDB and a Redis set of core services (Core Data and Metadata). (Trevor's team)
 - 3) For Edinburgh release, we document how to replace the database for others (Trevor's team).
 - This should be easy if we utilize plugins.

- If not, then the customer will have to fork the repo and put their implementation into the fork
 - Recommend we do not accept any more direct DB integration contributions.
- 4) For Edinburgh release, we have a certification process for core services – insuring any new core service adheres to the API sets and work appropriately in an EdgeX environment (Jason’s team with help from our teams to take care of the technical work)
- 5) For Edinburgh release, we have a marketplace for core services (Jason’s team and LF marketing)
- 6) For Edinburg release, we have a performance test harness for core services that provide statistics on memory usage, CPU usage, speed of queries, writes, etc. (Andy’s team)
- 7) For the Fuji release, we make a determination of which of the reference implementations we want to keep long term (may be one or both or even take one out of the marketplace if it is shows itself to be better)