

Core Working Group Meeting Notes (30-Aug-2018)

Attendees:

Participants (16)

Search: Type to filter...

Participant	Audio	Video
TC Trevor Conn (Host, me)	On	Off
13039688842	Off	Off
JW Jim White (Dell)	On	Off
RH Rodney Hess	Off	Off
AA Akram Ahmad (Dell)	On	Off
AF Andrew Foster	On	Off
AS André Srinivasan	On	Off
BF Brandon Forster	On	Off
EA Emad Attia (Intel)	On	Off
EC Eric Cotter	On	Off
F Fede	On	Off
IOTech IOTech	On	Off
JP J Phelps - LF	On	Off
Markus Junginger	On	Off
MH Michael Hall	On	Off
t tonyespy	On	Off

Old Business

- Work to merge core-config-seed-go into edgex-go is underway. Targeting current feature parity before making Config V2-related changes.
- Scheduler status
 - *Follow-up next week from Tony w/r/t addressable, schedule event interaction*

New Business

- Golang 1.11 released
 - When do we update?
 - Pre/post-Delhi
 - Effect on CI, if any?
 - *Individual contributors can update locally*
 - *CI – provision instances with 1.11 for trial, target a couple jobs*
 - *Wholesale adoption driven via Ansible templates*
 - *Adoption most likely post-Delhi, decision to be made @ Edinburgh*
 - *Rodney is already upgraded with no issues yet.*
- 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
 - *Filename guidelines*
 - *Try to keep it to one word*
 - *Failing to do that, use snake case "schedule_event.go"*
 - *Multiple types in one file is okay if they're logically grouped*
 - *Avoid stutter*
 - *Tony to create an issue for cleaning up the pkg/models*
- Recap Delhi deliverables
 - <https://wiki.edgexfoundry.org/display/FA/Delhi>
 - Completed
 - Pending
 - *Discussed organizing issues grouped by milestone (Delhi, for example)*
 - *Responsibility of Release Manager*
 - *For Delhi, wiki is sufficient for released items*
 - *Working groups leads should either coordinate with Jim w/r/t deliverables for Delhi and/or go through issues and assign Delhi milestone.*

- *For Edinburgh,*
 - *Issues in Github should be assigned a milestone and label indicating working group ownership*
 - *Release Manager should stay on top of this and make sure leads are accountable.*
- 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)
- *Tony brought up issue of PR rebase required when PR #2 doesn't have any overlap with PR #1*
 - *Controlled by a setting in the project, can we loosen the restriction?*
 - *Might be more of a DevOps item*
 - *Considered as a request from Core WG to DevOps WG*
-