Old Business

1. Geneva Scope Review/Status
   a. edgex-go-based system management agent/executor (Michael Estrin, Dell)
      i. Created issues and project board to track Geneva work
   b. CLI (Alex Courouble, VMware)
   c. Open Horizon (Joe Pearson, IBM)
      i. Creating Geneva use cases
      ii. Expect to kick off integration efforts week of January 20th
      iii. (first pass has been done; David Booz to break down use cases into GitHub issues between now and week of January 20th; will need a repository in holding to hold code/issues; Ifty on IBM’s side will work with EdgeX devops to figure out the Open Horizon build/test/deploy story; Joe will be on vacation through the end of the year but will be sporadically available via Slack during that time)

New Business

1. Moving CLI from edgexfoundry-holding to edgexfoundry
   a. What is the current state of the CLI interface (e.g. coverage of EdgeX functionality)?
      i. 0.0.1 in version. Hopes to have 0.1.0 by end of year. To include use of configuration file that has complete URIs.
   b. What missing features/functionality should be completed prior to conversion?
i. 0.1.0 end of year targets
   1. Better testing
   2. Better error handling
   3. Standardized error messages across CLI
   4. Every command/query should have URI in configuration file

ii. 0.2.0 might be target for Geneva release

C. What other work needs to be completed prior to conversion?
   i. See b.i above
   ii. Support for Windows, MacOS? (Anthony)
      1. Will need to be tested (Alex)
      2. Need to define priorities and roadmap (Jim White)

D. Strategy for handling Geneva revisions to core services APIs?
   i. Routes/URIs configured in configuration file – pointed to a command
   ii. Would leverage go-mod-core-contracts for model definitions

E. Pros and cons of moving CLI into the project?
   i. Long-term maintenance – intent is to support it forward as long as CLI is useful to the community
   ii. Is it part of EdgeX or is it a tool to be used with EdgeX? Main repositories are implementation. (Jim White)
   iii. Is CLI as important as an API alternative? Is CLI equivalent to the UI (which is in the main project)?
   iv. What about IDE extensions – would those be included in the main project as well? (Mike)

F. Once converted, how do we ensure changes to EdgeX APIs include corresponding changes to the CLI?
   i. Add new command; trivial – new subcommand
   ii. Add new field to DTO/model – go-mod-core-contracts is used so new field would be reflected there and become immediately available; CLI command implementation selectively outputs DTO/model fields (and would need to be updated to support the output of a new field)

G. A couple of reasons we haven’t moved it yet (Jim White)
   i. Lack of unit test coverage
      1. Have tests for what is easily testable.
      2. Automated testing for commands is hard; easier to manually test
   ii. Developer tool
      1. Alex believes scope goes beyond EdgeX developers
      2. Related; one of the deliverables is to enable CLI to talk through Kong to the rest of EdgeX (i.e. for use by external user)

H. Potential use of CLI as testing tool/usage as testing tool may facilitate testing of CLI (Michael/Alex)

I. Backwards-compatibility
   i. Would mirror larger system
   ii. Would need version alignment with larger system; maybe keep major versions aligned as indication of compatibility
j. Community Input
   i. Should CLI be moved to main org?
      1. Yes (Alex)
      2. Mixed feelings; CLI would need to match version/API – if CLI is broken or
doesn’t adhere to system contracts, would that stop a release? Sees CLI
as power-user tool. (Anthony)
         a. Alex doesn’t think a release should be held if CLI is broken or
not supporting the release’s API/contracts
         b. CLI shouldn’t be in main project if it’s not aligned with the
version of the system components it queries. (Anthony)
         c. Clarification – how is UI versioned by release? It is.
         d. Would lack of support of a releases functionality or breakage
stop a release? (open question).
      3. Mike has no strong feelings. Would lean towards moving it into main
org.
      4. Jim echoed what Mike said.
   ii. What is the gate for going to the TSC with a recommendation to include CLI as
part of the main project?
      1. Alex wants to get the 0.1.0 changes before we do this; expects to have
this done by the end of this year.
      2. Alex – for the next system management working group call, a list of...
         a. What should be done prior to requesting CLI be moved into
main project?; and
         b. What will be done for Geneva release?

2. Other new business?
   a. No meeting next week (US Thanksgiving holiday)
   b. Easy way to interact with system management agent other than REST?
      i. Client in system management agent in go-mod-core-contracts?
      ii. Currently uses general client (https://github.com/edgexfoundry/go-mod-core-
contracts/tree/master/clients)
      iii. Michael Estrin to create an issue to refactor and make it so.