# Go Working Group – 11/14/2017

Attendees: Jeremy(LF) , Chencho, Drasko (Mainflux), Fede & Gorka (Cavium), Konrad & Tony (Caonical), Lcapra, Ryan & Jim (Dell), Peter Miron (guest from nats.io)

## Guest Speaker

Pete Miron (for Brian Flannery) from Apcera to speak on Nats messaging infrastructure (Nats.io)

Pete’s presentation is available at: <https://www.slideshare.net/nats_io/simple-secure-scalable-messaging-for-the-cloud-native-era-allthingsopen-2017-talk-81195772>

## Old business

### Style Guidelines

* Please check Wiki for any final updates that need to be made regarding styles, contributor guidelines, etc.: [https://wiki.edgexfoundry.org/display/FA/Contributor%27s+Guide+-+Go+Lang](https://wiki.edgexfoundry.org/display/FA/Contributor%27s%2BGuide%2B-%2BGo%2BLang)

### ARM/CI Build

Working with Jeremy on first 3 tasks per our Go/Arm build special meeting

* Set up Go build with assistance from Dell/Cavium/Mainflux based on the research/work they have done in the LF existing CI environment for x86 (includes unit tests for x86).
* Create Go x86 Containers with same
* Since Go can be cross compiled, set up cross compilation for ARM and generate ARM artifacts via LF CI environment on x86
* Other tasks beyond these 3, but this is our starting point

### Go Lang Version

* Deferred to next meeting.

### Architecture/design Topics

Single versus multiple repos (as with Java)

* Deferred to next meeting, but general concensus from past discussions is to allow Go to use single repos per layer (i.e. – export, device services, core, etc.) if desired. This discussion also has ties to new business topic below.

## New Business

Is it time to fold this project group back under core working group? Are our tasks / issues aligned at this time?

* Tony – meeting is still useful a little while longer – til F2F in January

Additional New Business

* Tony – at last TSC, export service and naming of repos that we need to pick up that in this group
* Go packages don’t have dashes, dots in them. How do we insure we get the names right? How does the export service repos get created?
* Drasko – the repos names do not have to identical package names. Export services in one repos could be export-go, for example.
* **Action**: Reopen this discussion during the core working group meeting and carry it over to the next Go meeting.