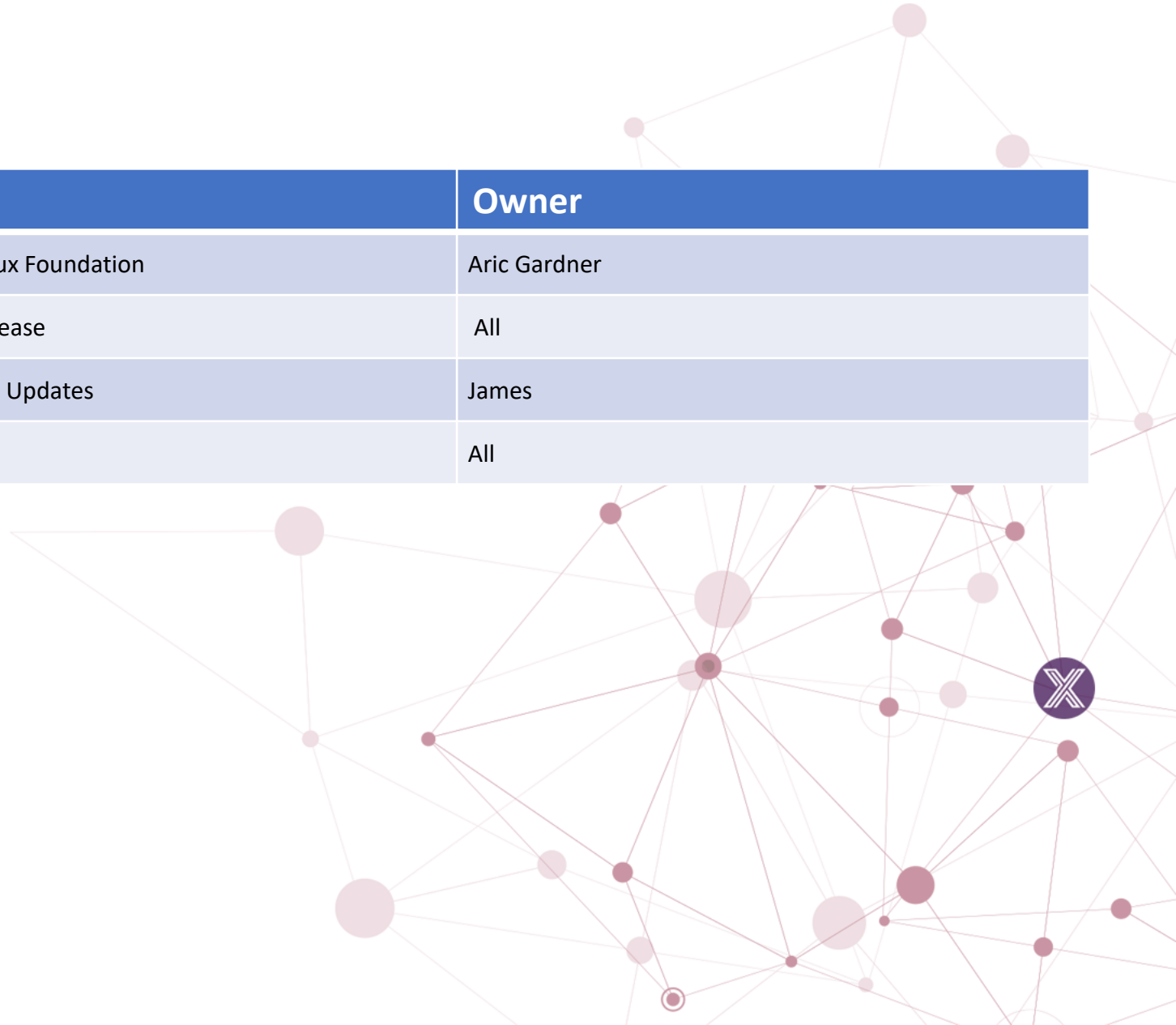
EDGE X FOUNDRY™

DevOps Working Group

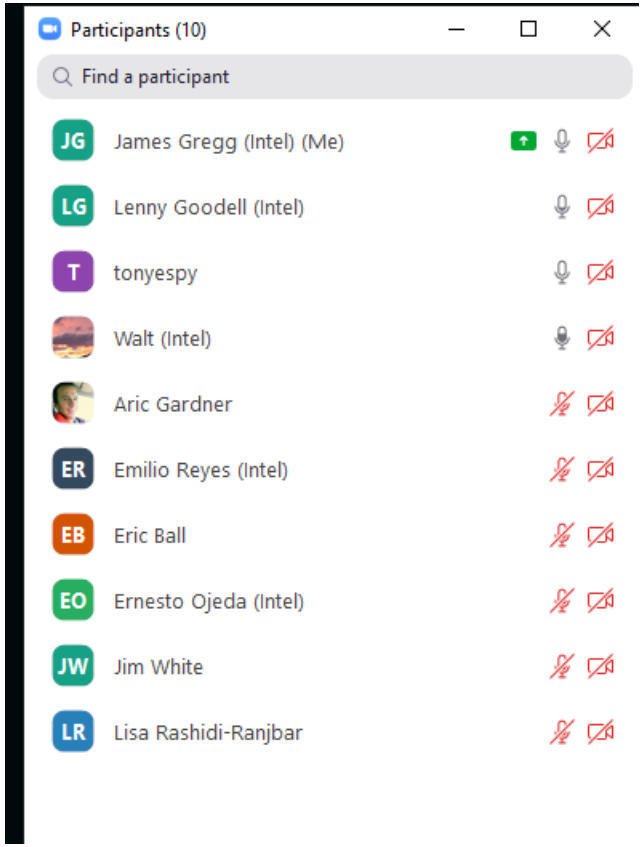
Thursday June 04, 2020

Agenda

Time	Topic	Owner
10 Min	Self-Service Linux Foundation	Aric Gardner
20 Min	Geneva Dot Release	All
25 Min	Hanoi / DevOps Updates	James
5 Min	AOB / Opens	All



Attendees


Participants (10)
Find a participant
JG James Gregg (Intel) (Me)
LG Lenny Goodell (Intel)
T tonyespy
Walt (Intel)
Aric Gardner
ER Emilio Reyes (Intel)
EB Eric Ball
EO Ernesto Ojeda (Intel)
JW Jim White
LR Lisa Rashidi-Ranjbar

DevOps WG Update

Hanoi

- **Performance Optimizations**
  - Build Optimizations for edgex-go
    - Implementation of the optimizations for edgex-go
    - Implementation of image promotions - WIP

- **DevSecOps**
  - Continued explore of options for addressing Issue #1947 - vetting of OSS dependencies
    - Submitted for review and discussion after rewrite of the explore
  - Community Bridge Feature Requests
    - Transitive dependencies for Go modules - findings don't match other tools like Sonatype - ticket still open no updates

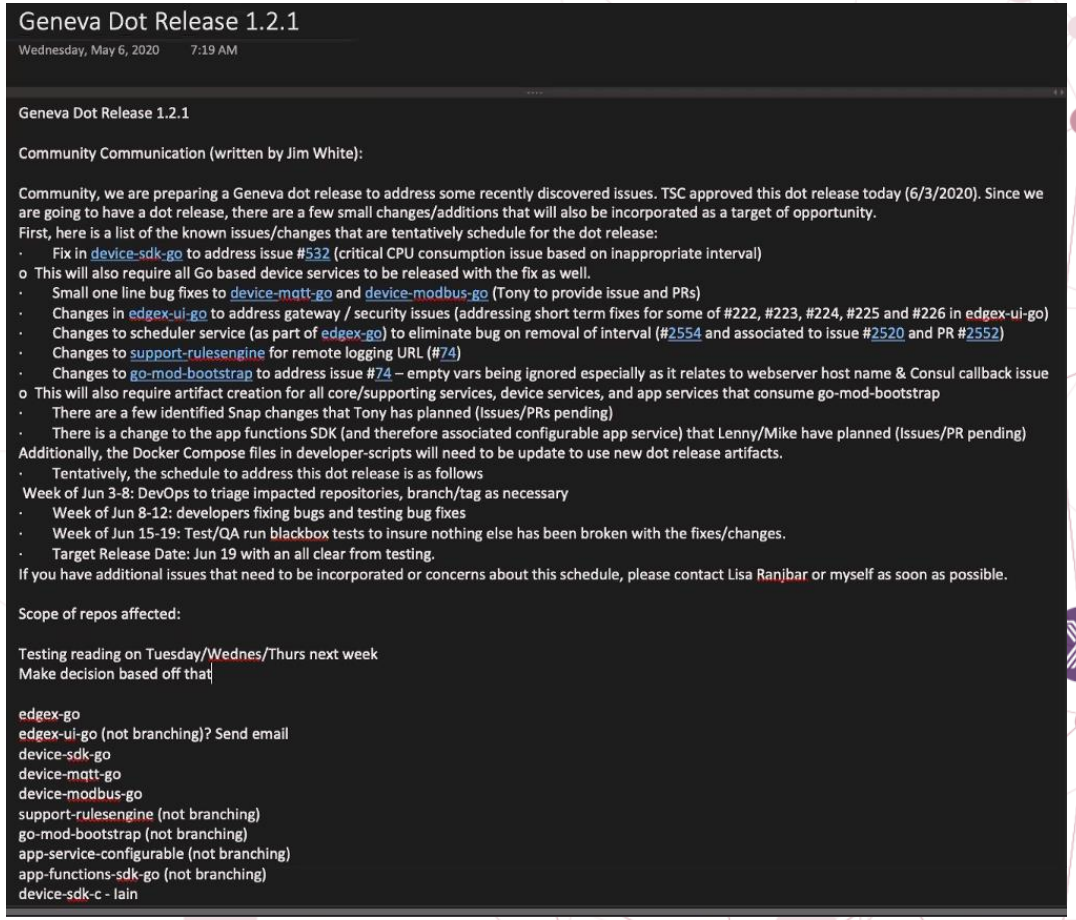
- **Other**
  - Refactor of the GitHub issue label automation to include milestones - WIP
  - Linux Foundation SSO updates completed - Report any login issues via JIRA support request if there's a problem
  - LFTools / Sigul latest version that supports Python 3.x
    - Need input from LF on alternative signing tool
    - At this point, the LF has started working on a fork of it and will own / maintain it

- Self-Service
  - Self-service-committers-management
    - Allows for mgmt of committers on repos 1:1
    - Uses GitHub Actions

- Test Code in Sandbox – via PR – reference the commit hash

Geneva Dot Release 1.2.1

- Branch only if it makes sense
  - If not branching = a mini code freeze
    - Reduces DevOps scope
  - Target Date - Code complete by next Wed (1 Week)
  - BB tests by Tues / Wed
  - Tag on Thurs
  - Release on Friday
    - Lenny updates Compose Files
    - James update Snyk portal



Geneva Dot Release 1.2.1

Wednesday, May 6, 2020 7:19 AM

Geneva Dot Release 1.2.1

Community Communication (written by Jim White):

Community, we are preparing a Geneva dot release to address some recently discovered issues. TSC approved this dot release today (6/3/2020). Since we are going to have a dot release, there are a few small changes/additions that will also be incorporated as a target of opportunity.
First, here is a list of the known issues/changes that are tentatively schedule for the dot release:

- Fix in device-sdk-go to address issue #532 (critical CPU consumption issue based on inappropriate interval)
  - This will also require all Go based device services to be released with the fix as well.
- Small one line bug fixes to device-mqtt-go and device-modbus-go (Tony to provide issue and PRs)
- Changes in edgex-ui-go to address gateway / security issues (addressing short term fixes for some of #222, #223, #224, #225 and #226 in edgex-ui-go)
- Changes to scheduler service (as part of edgex-go) to eliminate bug on removal of interval (#2554 and associated to issue #2520 and PR #2552)
- Changes to support-rulesengine for remote logging URL (#74)
- Changes to go-mod-bootstrap to address issue #74 – empty vars being ignored especially as it relates to webserver host name & Consul callback issue
  - This will also require artifact creation for all core/supporting services, device services, and app services that consume go-mod-bootstrap
- There are a few identified Snap changes that Tony has planned (Issues/PRs pending)
- There is a change to the app functions SDK (and therefore associated configurable app service) that Lenny/Mike have planned (Issues/PR pending)

Additionally, the Docker Compose files in developer-scripts will need to be update to use new dot release artifacts.

- Tentatively, the schedule to address this dot release is as follows

Week of Jun 3-8: DevOps to triage impacted repositories, branch/tag as necessary

- Week of Jun 8-12: developers fixing bugs and testing bug fixes
- Week of Jun 15-19: Test/QA run blackbox tests to insure nothing else has been broken with the fixes/changes.
- Target Release Date: Jun 19 with an all clear from testing.

If you have additional issues that need to be incorporated or concerns about this schedule, please contact Lisa Ranjbar or myself as soon as possible.

Scope of repos affected:

Testing reading on Tuesday/Wednes/Thurs next week
Make decision based off that

edgex-go
edgex-ui-go (not branching)? Send email
device-sdk-go
device-mqtt-go
device-modbus-go
support-rulesengine (not branching)
go-mod-bootstrap (not branching)
app-service-configurable (not branching)
app-functions-sdk-go (not branching)
device-sdk-c - lain

Meeting Minutes

Opens

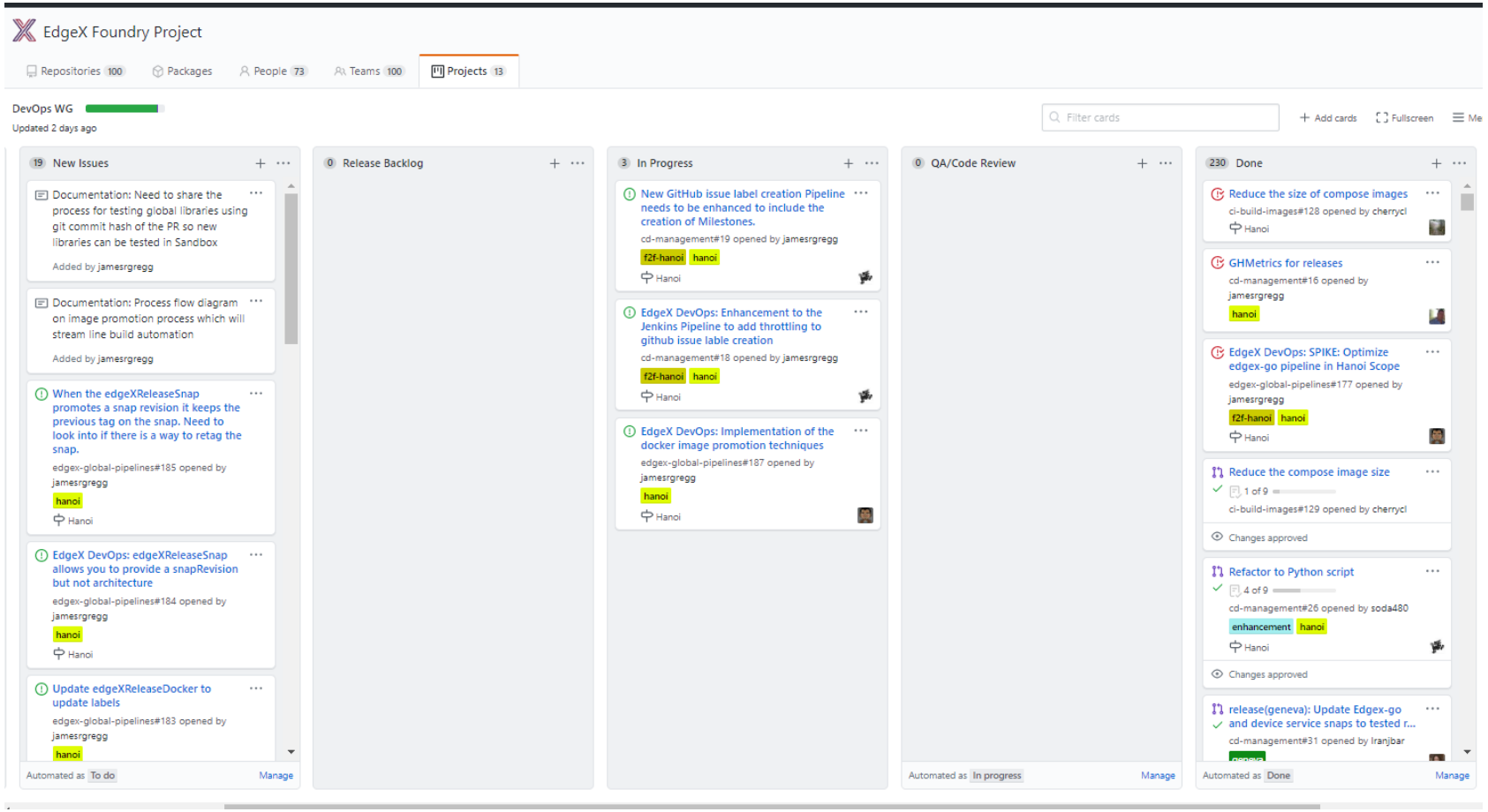
- Snaps push for Geneva Dot Release – Lisa to follow up with Tony
- Documentation Needed – add new stories to backlog
    - Need a process flow diagram on the new image promotion process
    - Need documentation on the new testing process which allows global libraries to be tested on EdgeX Jenkins Sandbox

Decisions

- Standard naming convention of Milestones – “Hanoi”
- We will punt on the use of Nexus IQ for now
    - Nexus IQ needs the go.sum file
    - EdgeX Foundry developers do not use go.sum and due to differences in the way the hashes are generated (differences between Windows / Linux)

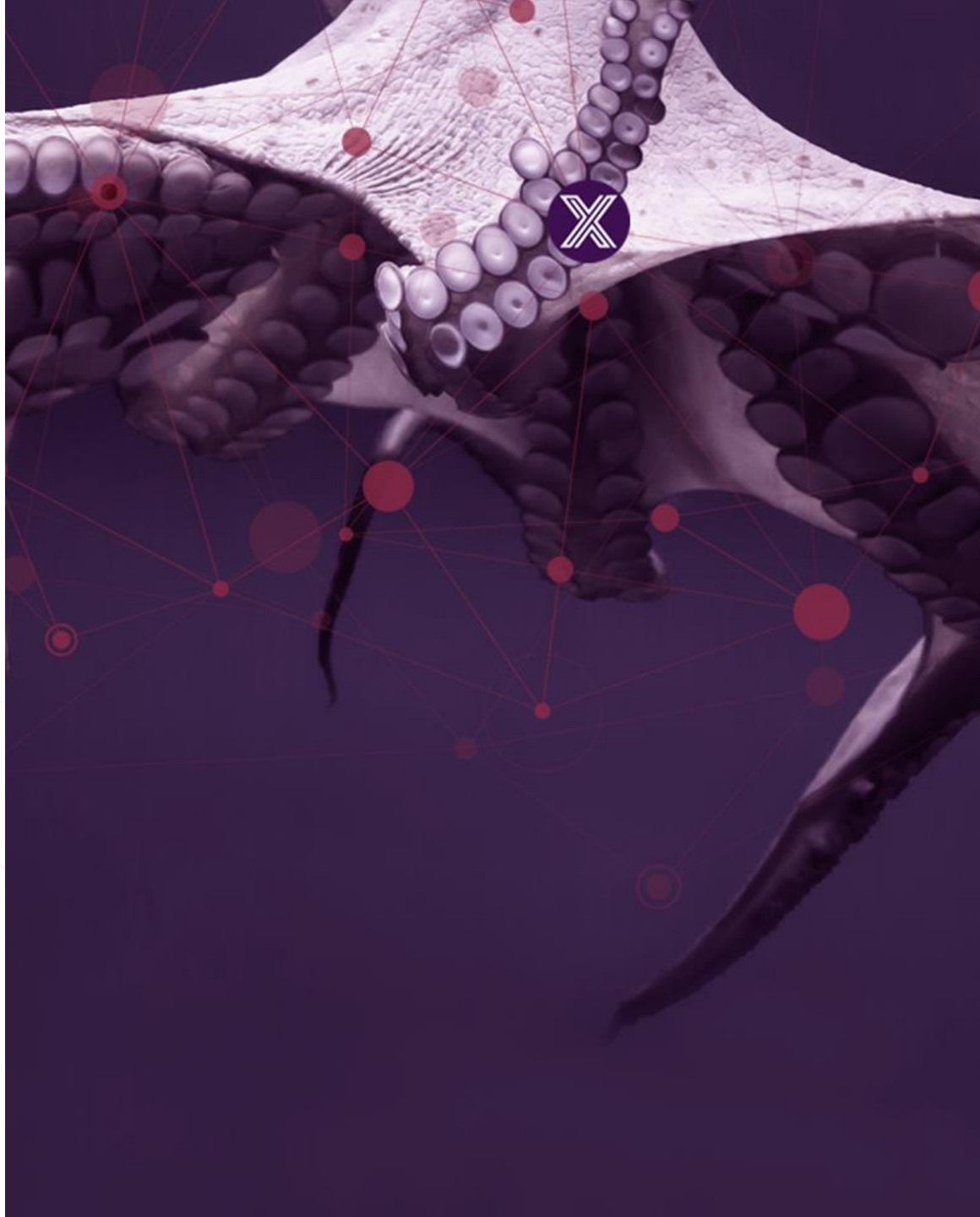
Hanoi release

Completed	Work In Progress	Release Backlog
3	3	0



EDGE X FOUNDRY™

Backup



EDGE X FOUNDRY™

Hanoi Planning

Scope Discussions

DevOps Scope of Work - Hanoi

- Performance Optimizations
  - Jenkins Pipeline optimizations for edgex-go
  - Explore options from LF for supporting Jenkins on K8s – completed roadmap review within Geneva
  - Explore alternatives to containerization within the builds
    - Explore use of BuildKit to simplify creation of x86/ARM build images so they share a single manifest when published to Docker Hub / Nexus
    - Explore use of Kanico
      - Explore Complete – Will not Work
        - Requires use of K8s persistent volumes and dedicated build agents which are long lived
- Performance of the Build Environment
  - ~~Monitoring / Alerting optimizations (Continuous Improvement Opportunity)~~
- Technical Debt
  - ~~Caching Dependencies – speed it up (upstream dependencies)~~
    Reference Linux Foundation roadmap
- ~~Open Horizons Enablement~~
  - ~~Shared Infra with Open Horizons~~
  - ~~Build Automation for OH~~
- Stretch Goals
  - Code Coverage for Jenkins Global Libraries (codecov.io)
  - Snap improvements – build optimizations
  - Support for –race flag with goals to address with Go 1.15 …*but there are options*

DevOps WG Recap (Geneva)

Geneva (May 2020):

- DevOps Jenkins Pipeline Transformation
  - Introduced new Jenkins Global Libraries for build automation
    - Includes test framework for Groovy code
    - Explore underway to look into code coverage of Groovy code using Codecov.io
  - Semantic Versioning using Intel contributed utility (git-semver) enhanced to include test framework
  - Continuous Delivery via "release-kraken"
  - Developer Enablement – GitHub Project Tracker, GitHub Issue label creation automated, gitcommit linter implemented *
  - New ci-build images and global libraries developed to support Jenkins Pipelines
  - New life cycle policies implemented on Linux Foundation Nexus repositories
  - Automation of the labels across the project
  - GitHub Tracker (Kanban board) – utilized weekly with built in workflow
  - Developer Documentation created for new Jenkins Pipelines
  - Improved performance of all builds to include collaboration with Linux Foundation to drive performance improvements for ARM builds (~15 mins build performance improvements using a new flavor of LF build nodes)
    - X86 build nodes (VM) uses 4cpu – 2gb
    - Arm64 build nodes (VM) now uses 4 cpu – 16gb

DevSecOps scope includes:

- Snyk Advanced Reporting via Community Bridge - $8K savings on licensing for developer licenses
- Snyk Docker Hub image scans with weekly reports of new vulnerabilities
- Snyk CLI of Go integrated into scan stage of Jenkins Pipelines
- Clair image scans within scan stage of Jenkins Pipelines
- DevOps contributed code fixes to address CVEs found in images based on Snyk reporting
- Lftools updated to use latest version – code signing, git tag signing, Docker image signing

Geneva Freeze and Release

TSC approved

- Freeze: 12pm GMT, April 22 (Wed, week before planning meeting)
- Release: 12pm GMT, May 13 (Wed two weeks after planning meeting)

See Geneva release notes for details (on Slack)

REMINDER:
We will NOT be branching off master for the Geneva release.

Includes **EVERYTHING**

Will not be versioning go modules

Do we need blackbox tests to be an "artifact" of a release?

- **QA/ Test WG doesn't require signed tags, but since release kraken can be used to automate the creation of the tag, it would be a signed tag**
- **If there's a need to patch Geneva, the tagged blackbox tests would be used**
- **Since blackbox tests wasn't previously considered a "release artifact" does it get tagged? – YES it does**

Decision: We now need to consider blackbox tests as a formal artifact. Tag would be generated at the time of the formal release

Geneva Release Schedule

New scope – consider blackbox tests as artifact of the release

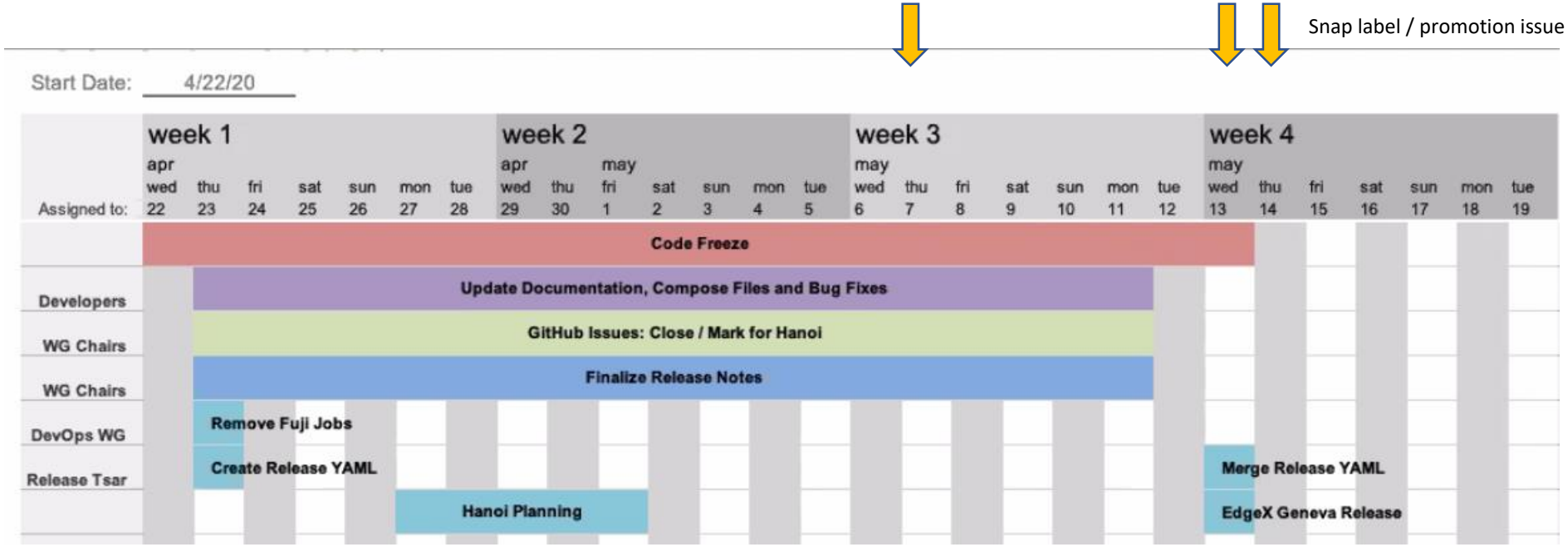
- should have been considered within review of ADR007

Green light decision to release

- TSC meeting late in the day
- Multiple issues worked throughout the day

support-rules-engine

Snap label / promotion issue identified



Timeline to be reviewed for Geneva Retrospective

Geneva Retrospective

What went right?

- Smoother release – no branching at code freeze equated to efficiency
- Whole DevOps team was responsive
- Developers embraced the opportunity to create the Jenkinsfiles themselves
- Great collaboration and cross pollination of the information
- Linux Foundation was very helpful and responsive in the release – easier and supported well
- Use of JIRA tickets helped with response times on support / help needed from LF release engineer
- Andrew Grimberg came into the DevOps wg for a roadmap discussion
- Automation of the release went well – good coordination
- Phased approach of the work helped align to sprint cadence
- ADR practice helped with communications across the project
- Ernesto recognized for work on the snaps
- Lisa recognized the good communication / teamwork with Emilio / Ernesto
- Tony / Ian helped with review of the snap automation code – THANK YOU!!
- Risk acceptance / Risk taken - It worked!!
- Dry Run on release automation functionality
- Tony / Ian were responsive wrt Snap store issues – THANK YOU!!

What could be improved?

- Communication gaps
  - support-rules-engine issue related to a change in plan
    - DEPRECATION (Define process needed ??)
- Snap release process could be better understood
  - Need full path to production for snap release process
    - release to beta candidate channel >> stable
    - Time crunch in the end could be root cause for the snap release issue
    - Might need TAF testing for snaps
    - No real hw testing (Akraino community lab – University of New Hampshire)
      - Canonical presentation on how they do hw testing with snaps
  - Need functional testing for snap automation
    - Need to figure out an example service (sample-service)
  - Inability to properly test in a sandbox, test environment
    - Help needed from LF to support ability
  - Release Kraken Improvements (re-lable / tagging)
    - Idempotency
    - Need to specify a commit (might be an edge use case but better)
    - Set up of the snap YAML
  - Manual release of documentation needs fixed
  - Snap store issues (503 error) – length of time to build snaps

EDGE X FOUNDRY™

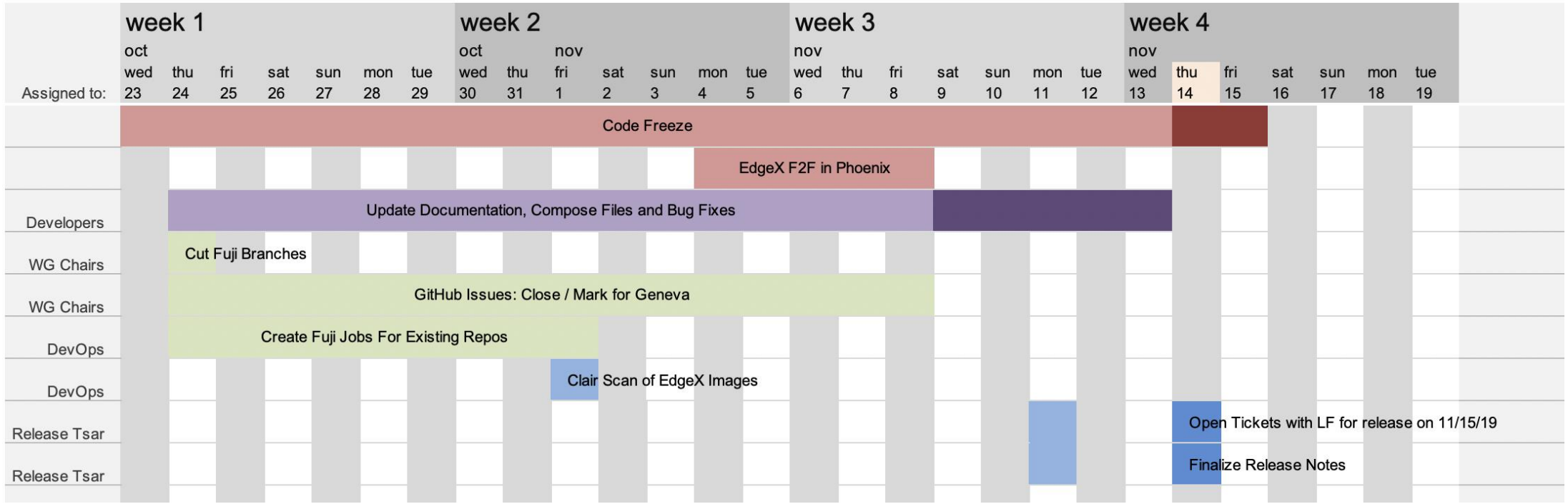
Geneva Planning

Scope Discussions

Fuji Release

- Freeze: Oct 23rd (Wednesday)
- Release: Nov 15th (Friday)

Start Date: 10/23/19 (with extension)



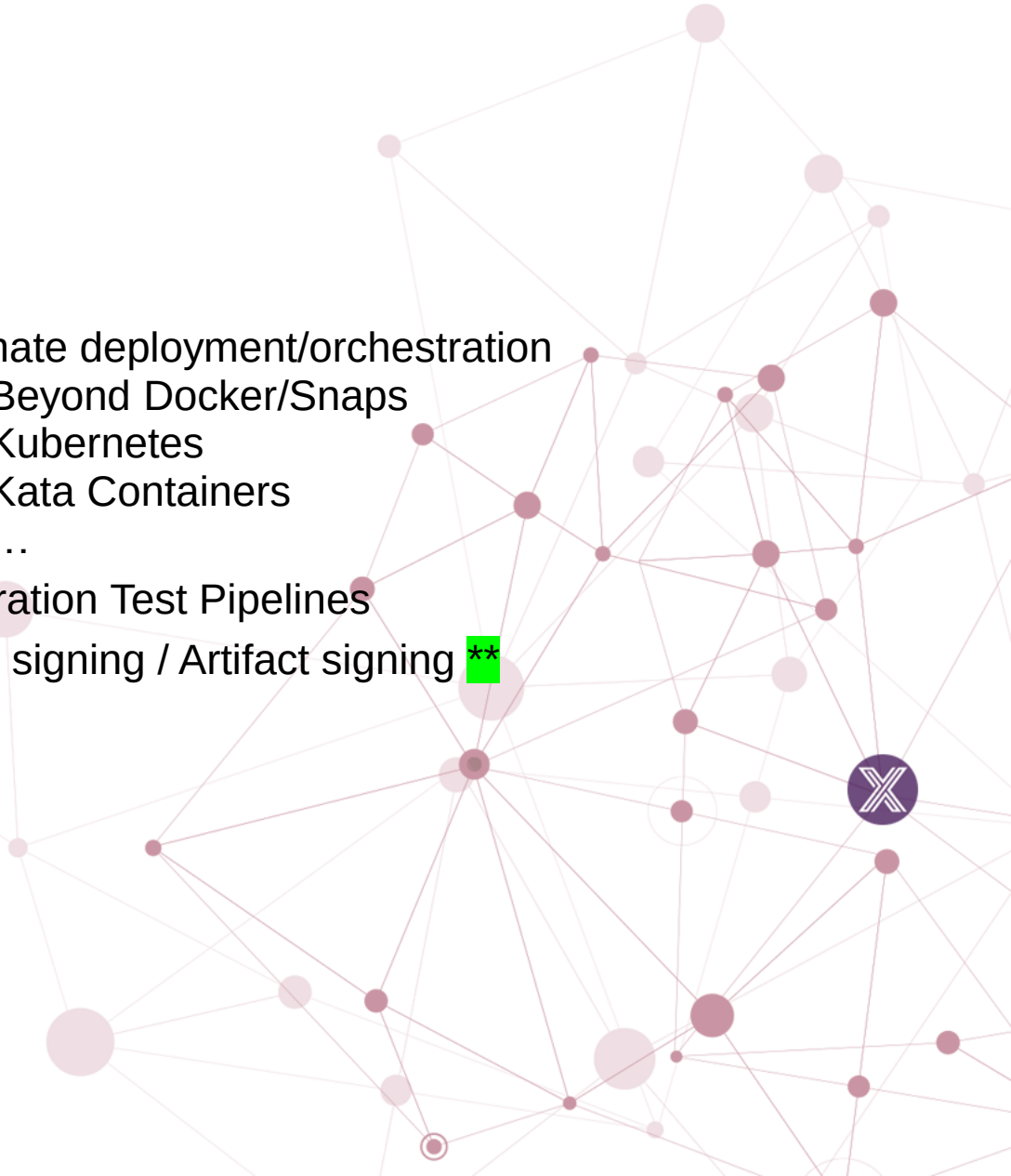
Geneva – DevOps

In

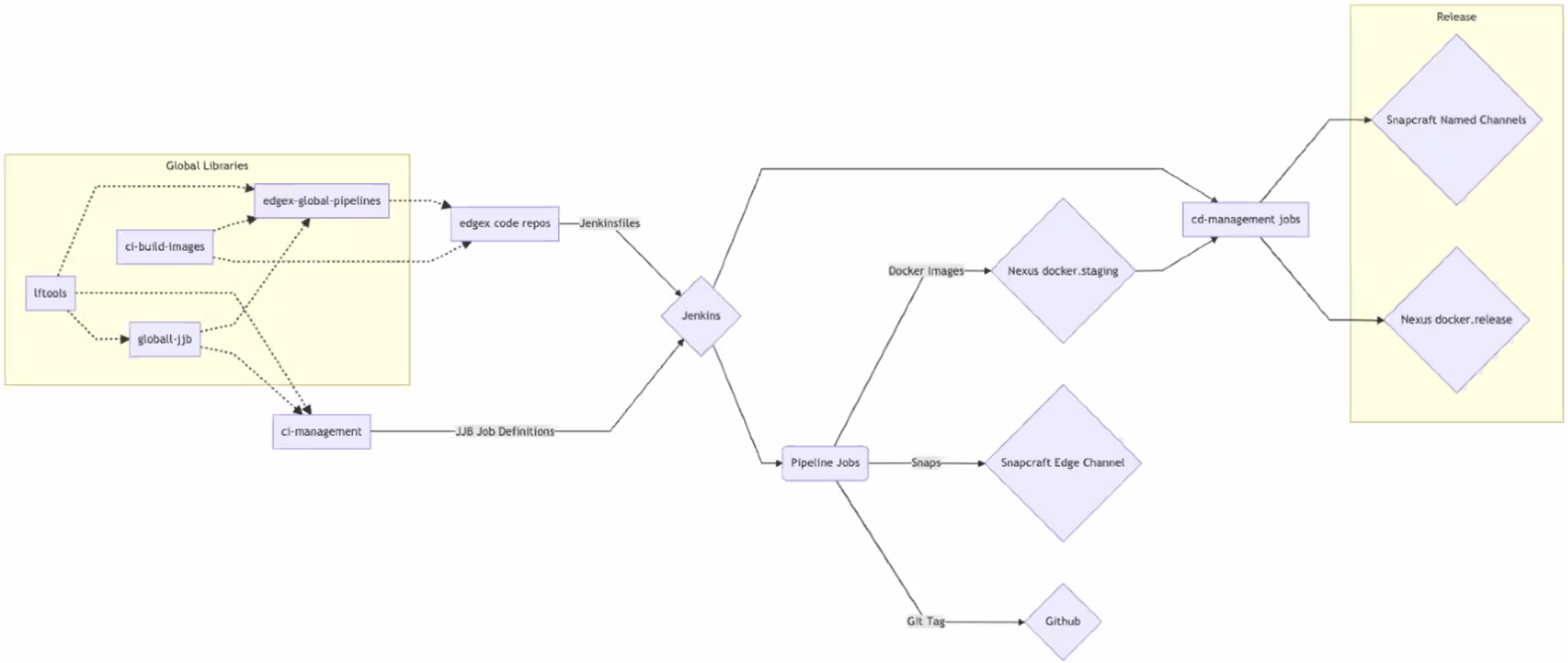
- Full Pipeline transformation for EdgeX services
  - Convert Jenkins JJB Freestyle jobs to Jenkins Pipelines
- Introduce GitHub Org Plugin
- Simplified Jenkinsfile
- Global Libraries to support Jenkins Pipeline transformation
- Add Unit testing to global-libraries (uncommitted) **
- Snyk integration for edgex services
  - As part of Jenkins Pipeline conversion
- Slack integration with Jenkins pipelines
- Nexus Cleanup / Lifecycle Policy

Out

- Alternate deployment/orchestration
  - Beyond Docker/Snaps
  - Kubernetes
  - Kata Containers
  - …
- Integration Test Pipelines
- Code signing / Artifact signing **

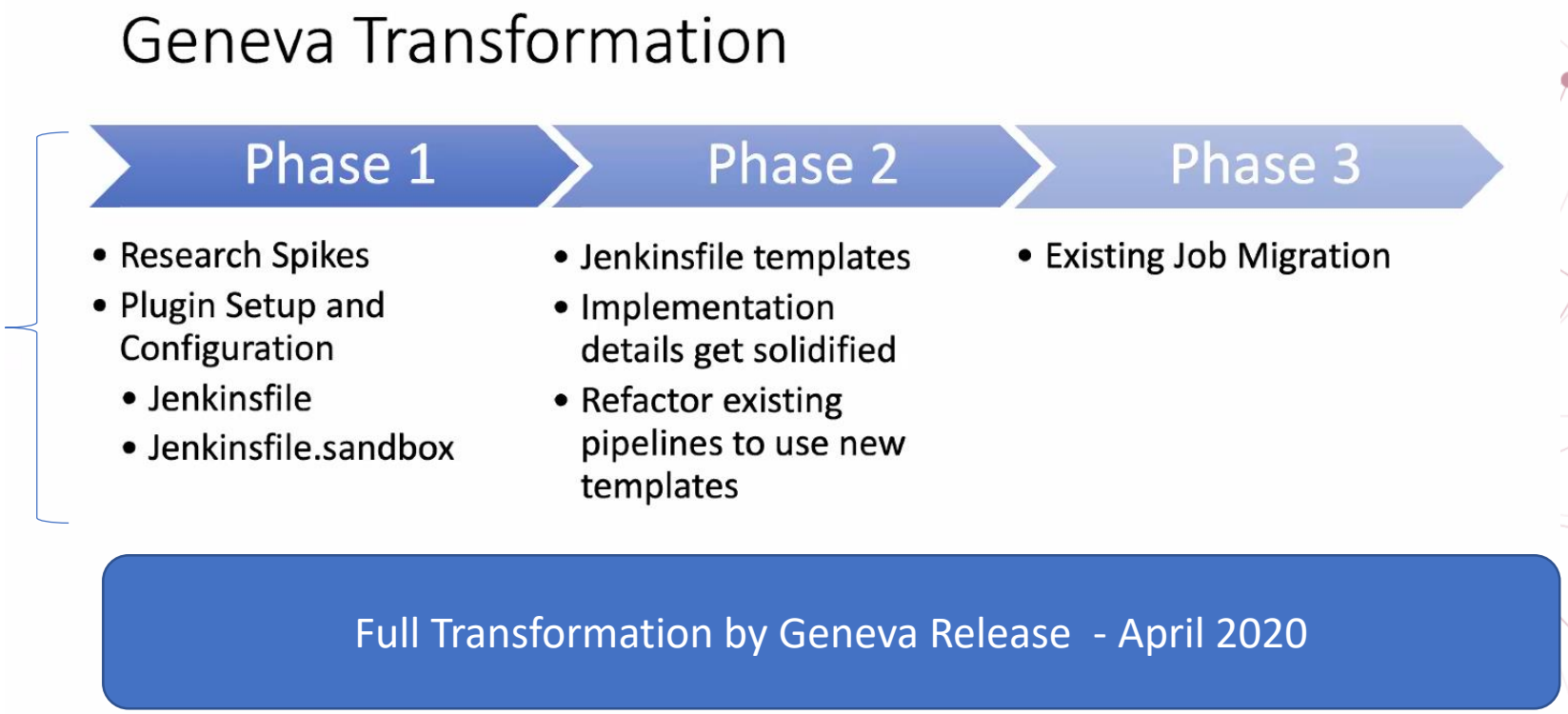


Geneva Transformation: Architecture



How long does it take? Is this all Geneva scope?

Phase 1
Work in Progress
Q3 2019



EDGE X FOUNDRY™

Fuji Planning

Scope Discussions

Fuji – DevOps

In

- Static code analysis tool identified and integrated into the EdgeX Jenkins Pipeline for Docker image scanning (Clair Server)

Explore SAST for true static code analysis to include additional tooling such as Fortify / Coverity

- Code and artifact signing with semantic versioning
- Fix Documentation – edgex-go
  - Create a new repo for edgex-docs
- Build Performance Optimizations
  - Pipelines for EdgeX Foundry base build images
  - Basebuild images managed locally within Nexus
  - Leverage PyPi Proxy for local pip dependencies
  - ARM builds – optimization leveraging different high CPU build nodes / OS (ARM Team)

Out

- Alternate deployment/orchestration
  - Beyond Docker/Snaps
  - Kubernetes
  - Kata Containers
  - …
- SonarQube – SonarCloud is already in play in the LF Decision: wait to see what codecov.io offers
- Suggestion to rename all of the Jenkins “arm” jobs so as to differentiate 32bit / 64bit architectures
- Full Pipeline transformation for EdgeX services

EdgeX DevOps Commitments (Fuji)

Scope of Work	
Add static artifact analysis into the EdgeX Jenkins Pipeline (analysis of Docker /runtime artifacts, not the source code)	
Add code and artifact signing with semantic versioning	
Conduct build performance optimizations by: • Adding Pipelines for EdgeX Foundry base build images • Allow base build images to be managed locally within Nexus • Leverage PyPi Proxy for local pip dependencies	
Explore static code analysis like Checkmarx, Coverity, GuardRails, Synk, SonarQube	

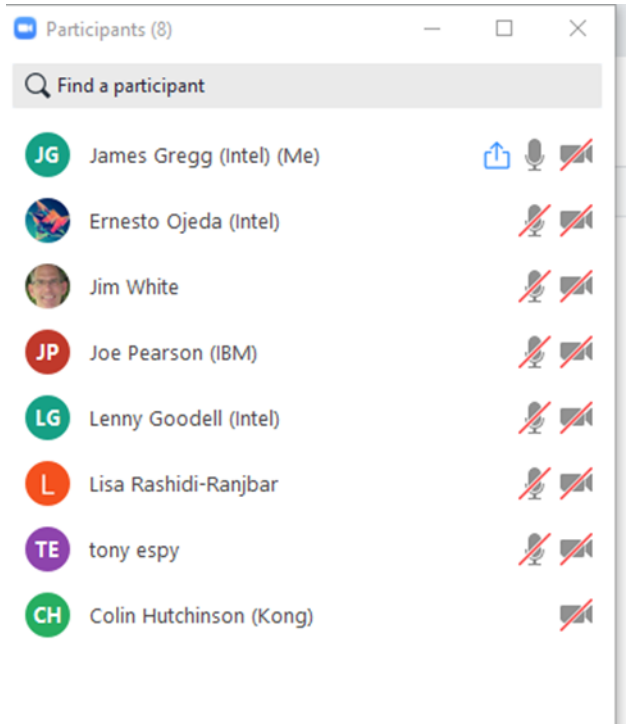
- Clair Server landing no longer at risk for Fuji
  - LF committed to implement on AWS and fund with expected completion next week
- gitsemver along with lftools used for artifact signing and semantic versioning
- Jenkins build performance optimizations for base build images completed
- All base build images will now be stored in Nexus (Snapshot):10003
- PyPi enabled as part of Edinburgh scope
- Initial review of GuardRails showed that the product was identifying issues which were not applicable for microservices architecture

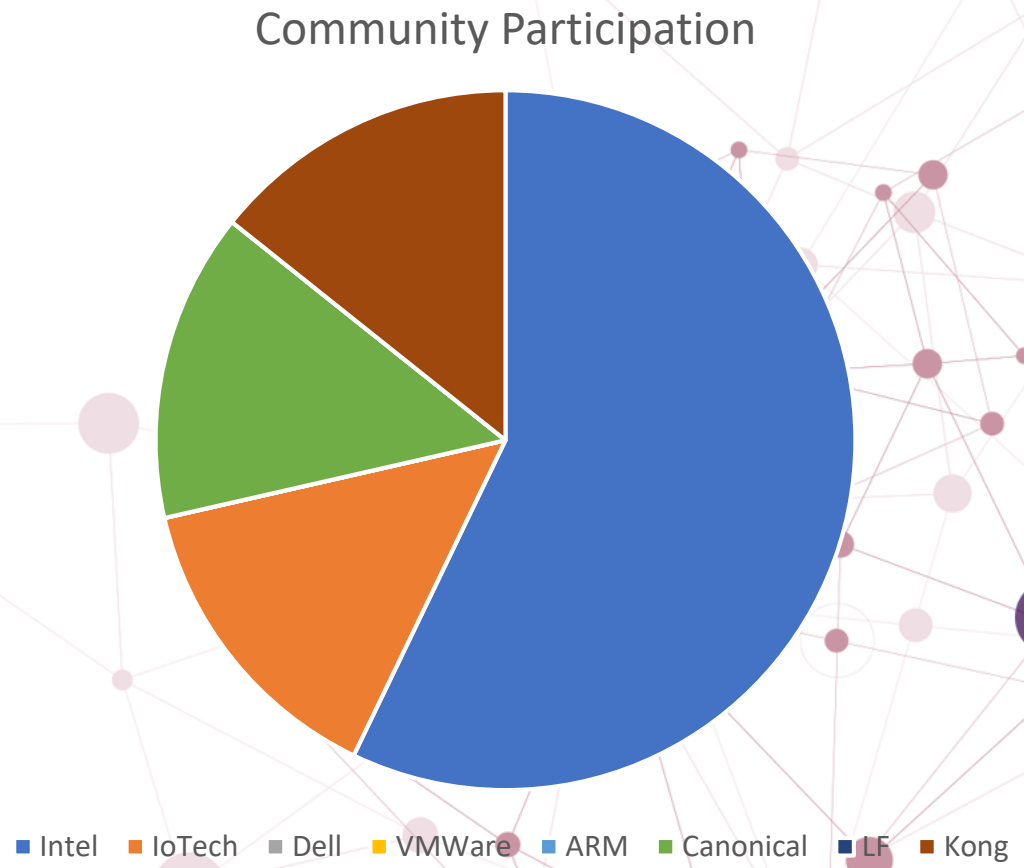
edgexfoundry.org | @edgexfoundry

Past / Future Agenda Topics

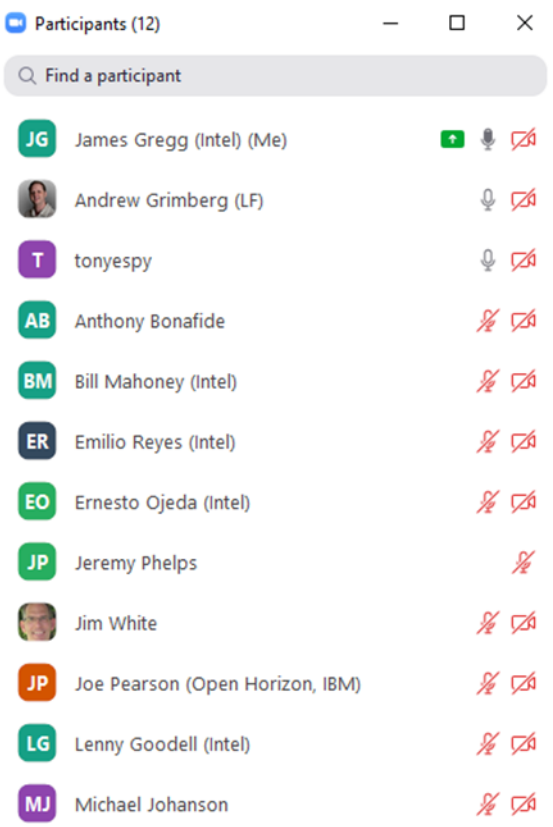
	Size change to use Ubuntu / Debian base build images to support –race flag for Go Lang
	Clair scan findings – Discussion developer community if we want to break the build when there's findings - Bring into Security WG for discussion
	Open Horizons enablement
	Alignment to new LF roadmap self-service offerings – EdgeX use case for handling holding repositories
	Release automation - key learnings and sharing with LF
	Explore use of Buildkit
	Explore use of Kanico
	Snyk Dashboard Review

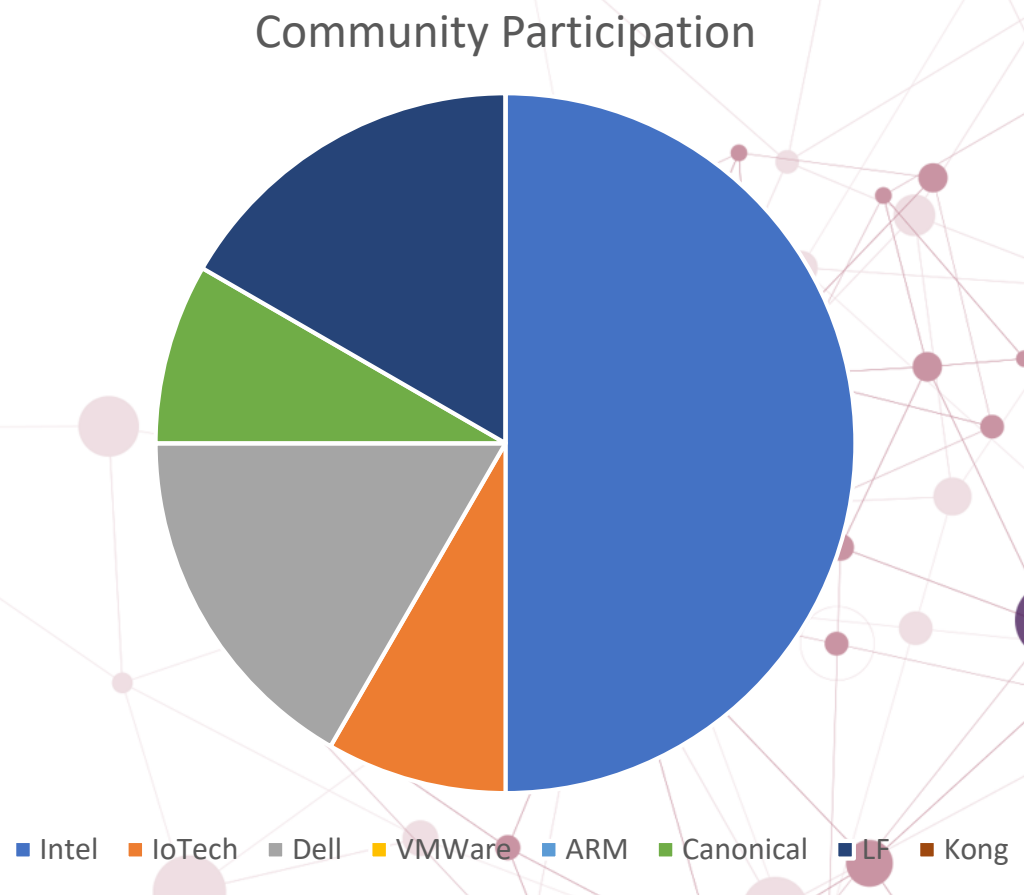
Attendees & Community Participation – ww14





Attendees & Community Participation – ww15

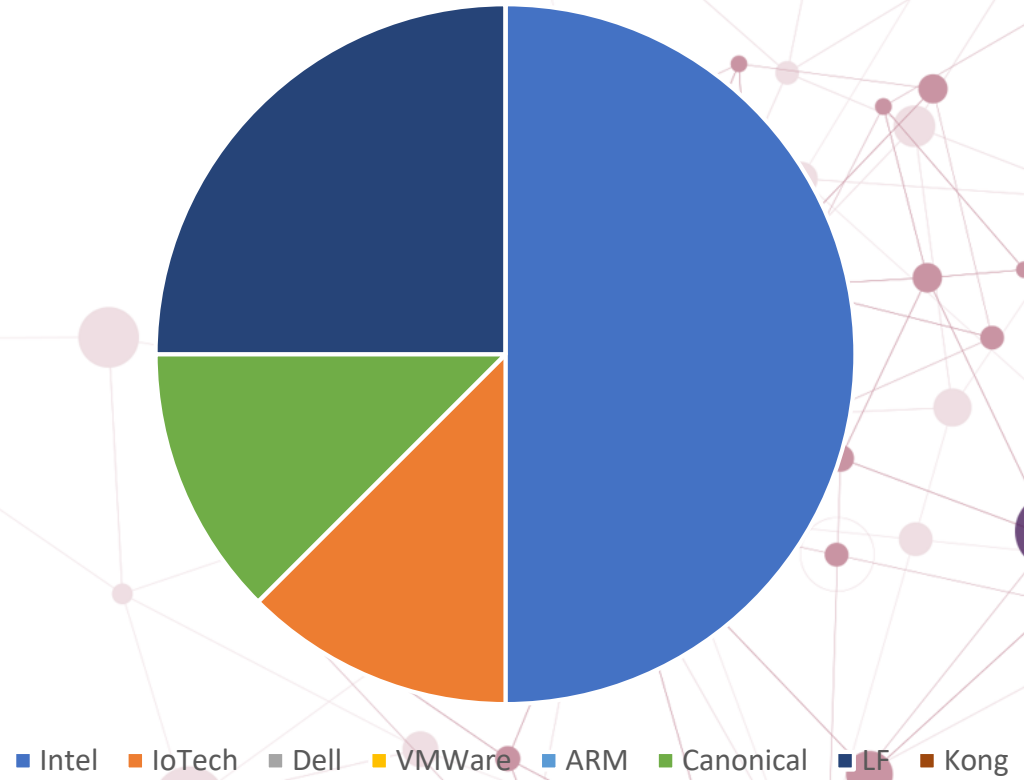
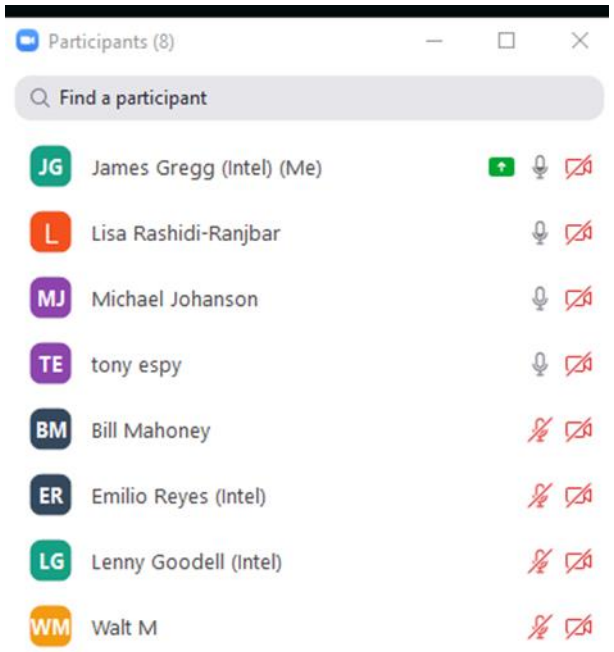
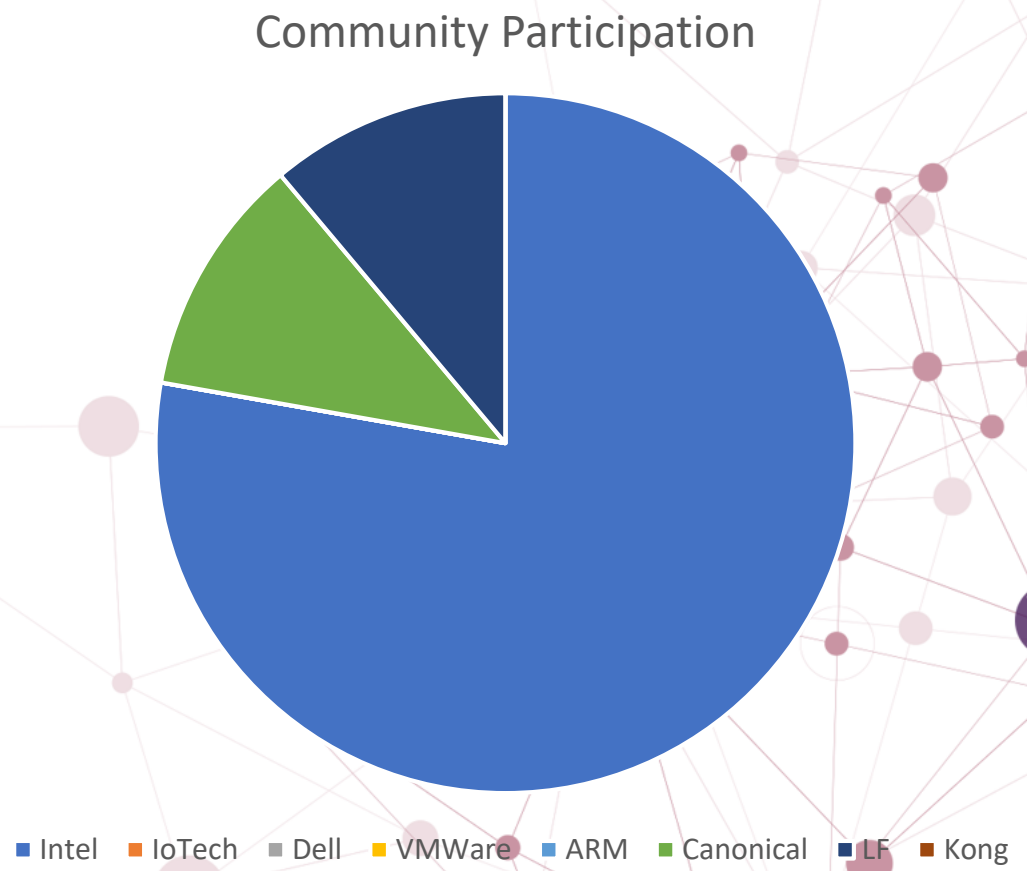




Attendees & Community Participation – ww16

Attendees

Community Participation



Attendees & Community Participation – ww17