DevOps Working Group

Thursday May 07, 2020

Agenda

Time	Topic	Owner
10 Min	Geneva - Hanoi / DevOps Updates	James
20 Min	Geneva Release / Discussion re: Blackbox Tests	Lisa /Cloud Tsai
15 Min	Jenkins Build Performance Optimizations – DEMO (edgex-go)	Ernesto
15 Min	AOB / Opens	All

Attendees

DevOps WG Update

Geneva

- Jenkins Transformation to Pipelines
  - Work completed transformation to Jenkins Pipelines
    - git-semver bugs fixed
- Jenkins build issues on 05/04/2020
  - Root cause due to ISV making changes to an endpoint - FIXED by Eric Ball / LF Thank you!!

Hanoi

- Hanoi planning Virtual F2F last week
- Performance Optimizations
  - Build Optimizations for edgex-go
    - Explore completed by Ernesto Ojeda with observable performance improvements in the build time
    - Leverage Docker and Docker Compose within the Pipeline along with caching of the layers
    - DEMO planned for next DevOps WG meeting
  - Explore on use of Kanico as a build container - Completed with decision it's not for consideration with Linux Foundation infrastructure due to requirement for Kubernetes persistent storage
- Community Bridge and Snyk Updates
  - New feature requests submitted in support of the explore work discussed in Hanoi Planning F2F - Security WG
  - Snyk isn't recognizing the change to Go 1.13 within the Snyk.io portal – Snyk support case opened Issue #178
- LFTools / Sigul latest version that supports Python 3.x
  - Need input from LF on alternative signing tool – formal request made as part of IT-19186
- Explore related to enabling validation pipelines with Akraino build infrastructure – IT-19532
  - Decision to not work as scope for Hanoi

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 – <mark>Will not Work</mark>
        - 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*

Geneva Freeze and Release dates

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
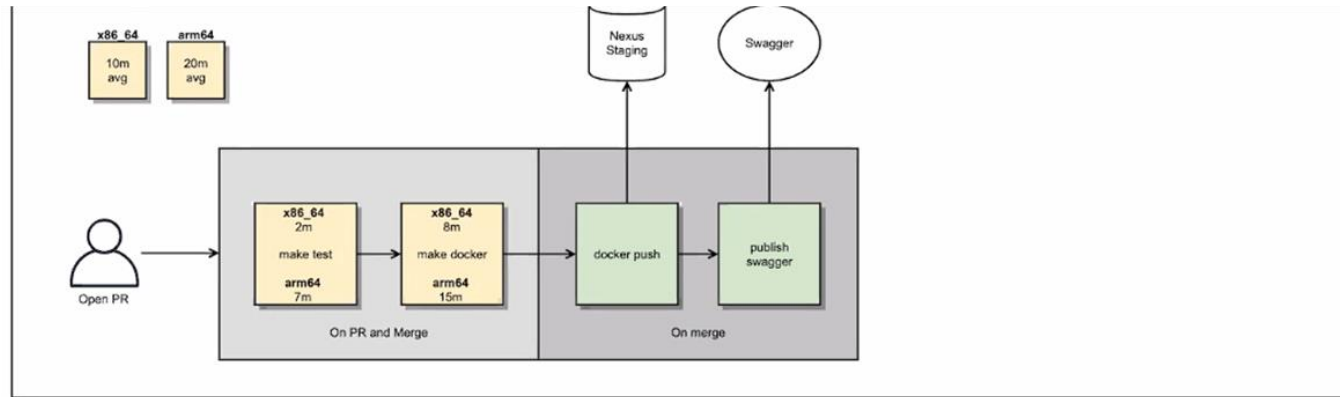
Geneva Release Schedule

Discussion related to blackbox tests need to be considered a "release artifact"
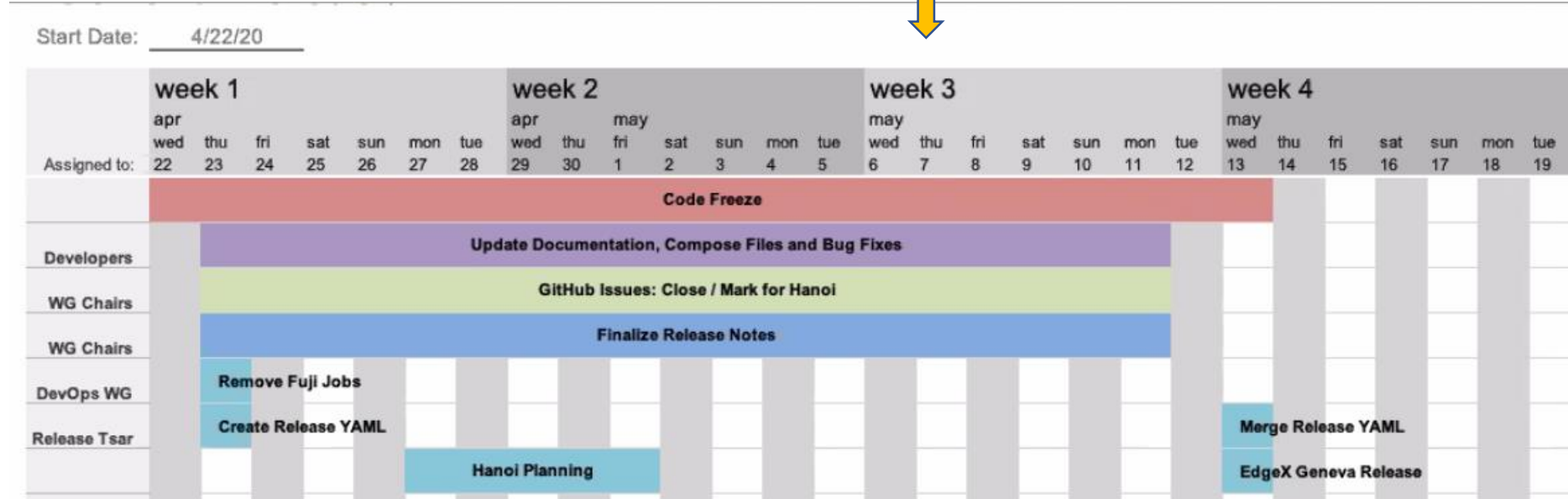
Timeline to be reviewed for Geneva Retrospective

Performance Optimizations for edgex-go
Ernesto Ojeda

Optimizations achieved through use of Docker – parallel and building a docker-compose on the fly

Leverages use of cached layers to speed up the acquisition of dependencies.

Less dependencies on LF network as the build won't have to go download go modules for every service it's building, it will instead use the cache. Only hits the network if the dependency needs to be refreshed to latest version.
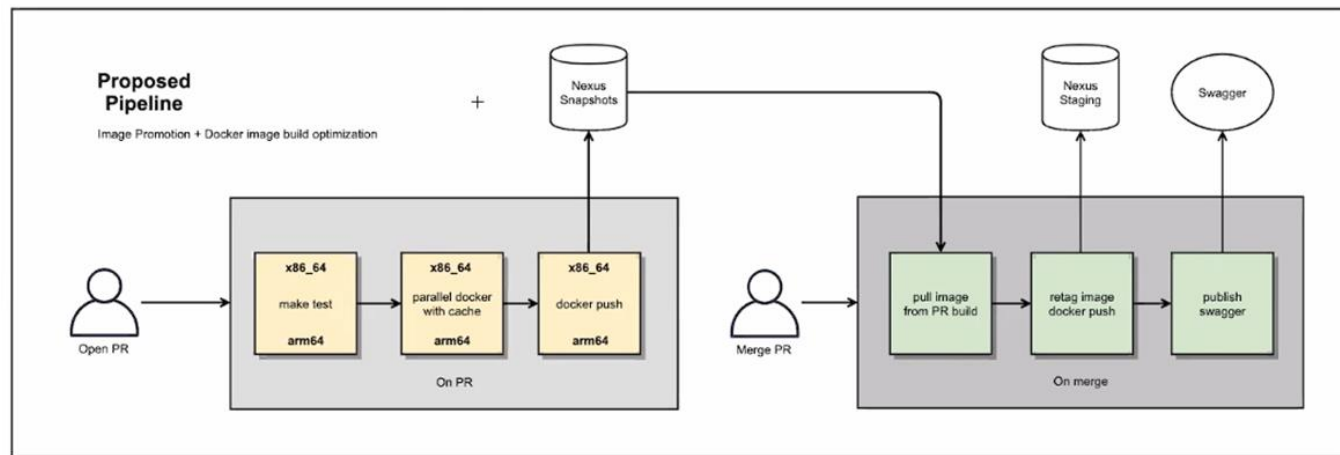
Current image is not Debian based, so –race condition flag not enabled.
Need approval from Core WG to move forward with AS-IS implementation or else would need to change to Debian based build image.. Or wait until Go 1.15 (August)

EdgeX-Go Pipeline Demo
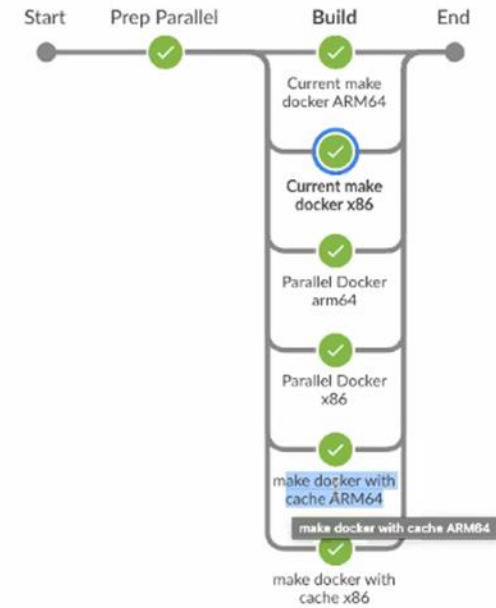
Geneva release

Completed	Work In Progress	Release Backlog
209	0	0

Meeting Minutes

- Need to update bot - https://github.com/apps/semantic-pull-requests
  - Completed

- Alternative to use something other than Docker – Kanico << <mark>WONT WORK</mark>
  - Requires use of persistent storage within K8s which isn't scoped for Hanoi

- Blackbox tests will now be considered a "release artifact" and be tagged at the end of a release

- Core WG will want to know about the proposed optimizations for edgex-go and give the approval to move forward without the –race condition flag

DevOps WG Update (Geneva)

Geneva (~Apr 2020) Focus:

- DevOps Jenkins Pipeline Transformation completed
  - 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
  - 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

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 – <mark>Will not Work</mark>
        - 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
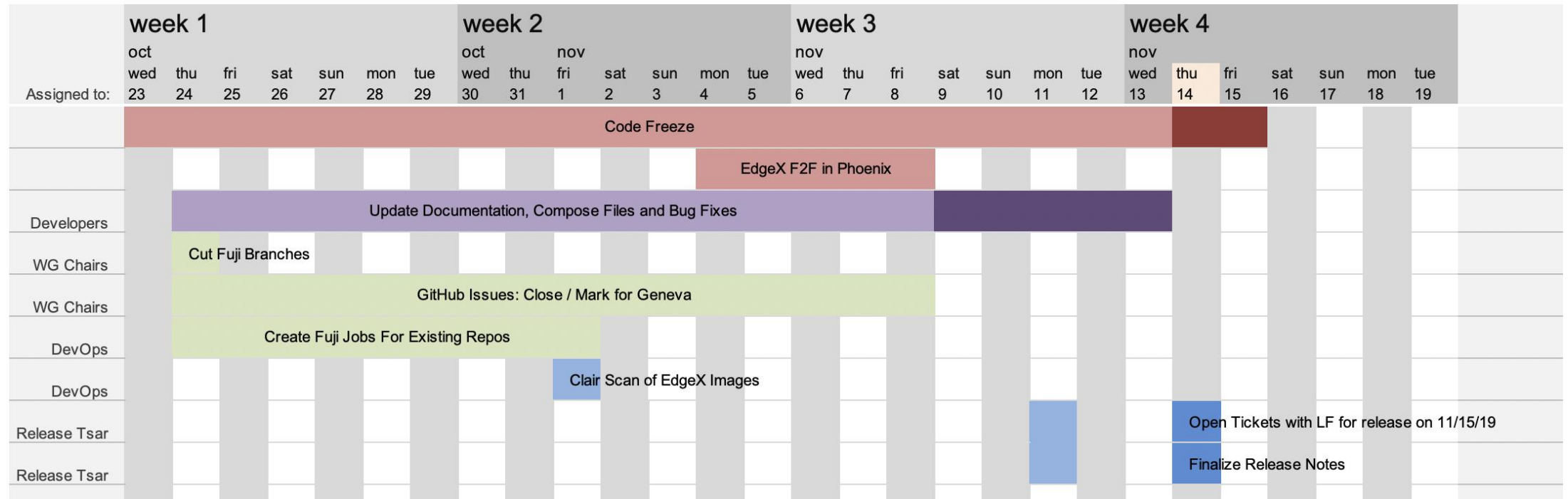 - Support for –race flag with goals to address with Go 1.15 …*but there are options*

Geneva Planning

Scope Discussions

Fuji Release

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

Start Date: _10/23/19_ (with extension)

Assigned to:	week 1 oct wed 23	thu 24	fri 25	sat 26	sun 27	mon 28	tue 29	week 2 oct wed 30	thu 31	nov fri 1	sat 2	sun 3	mon 4	tue 5	week 3 nov wed 6	thu 7	fri 8	sat 9	sun 10	mon 11	tue 12	week 4 nov wed 13	thu 14	fri 15	sat 16	sun 17	mon 18	tue 19
	Code Freeze																											
								EdgeX F2F in Phoenix																				
Developers		Update Documentation, Compose Files and Bug Fixes																										
WG Chairs		Cut Fuji Branches																										
WG Chairs		GitHub Issues: Close / Mark for Geneva																										
DevOps		Create Fuji Jobs For Existing Repos																										
DevOps								Clair Scan of EdgeX Images																				
Release Tsar																						Open Tickets with LF for release on 11/15/19					
Release Tsar																						Finalize Release Notes					

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 **
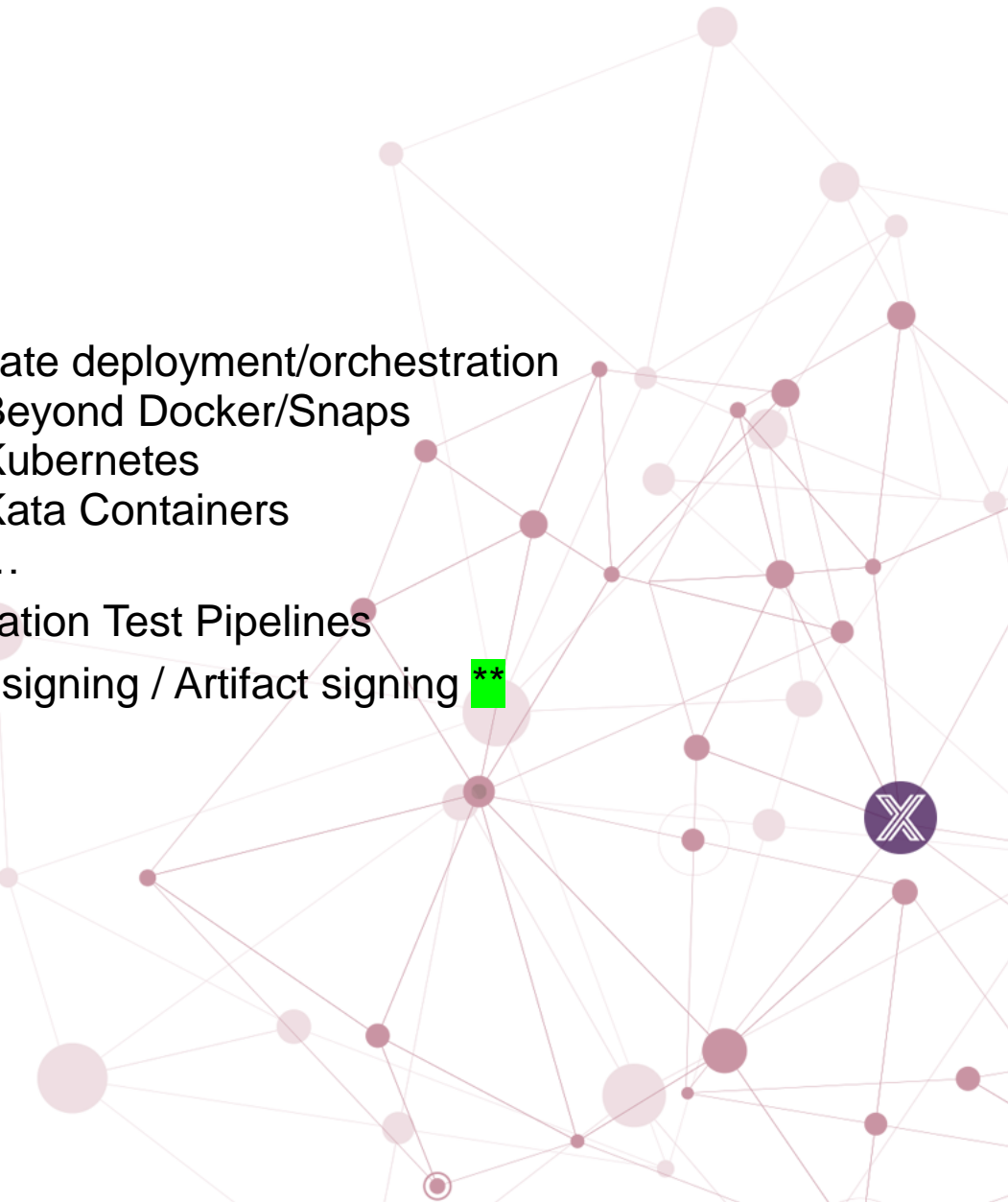
Geneva Transformation: Architecture
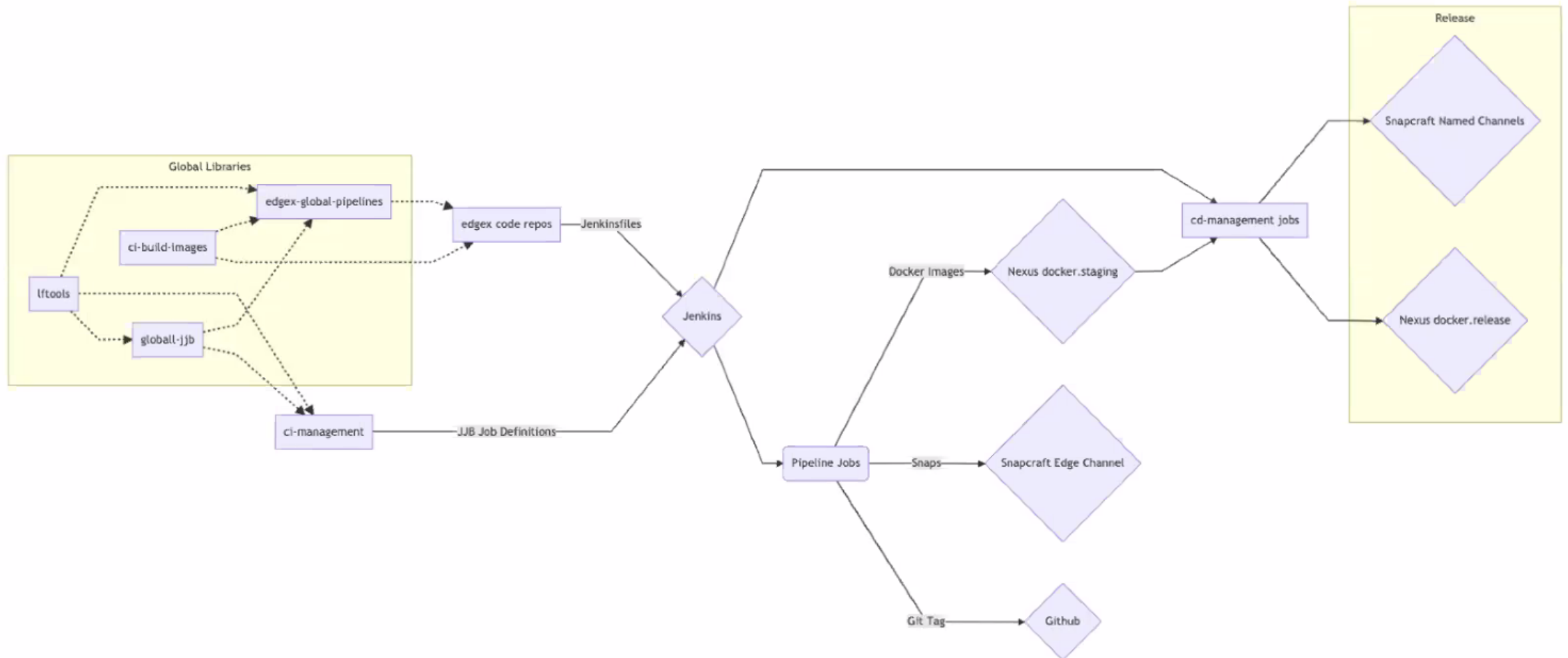
How long does it take? Is this all Geneva scope?

Geneva Transformation

Phase 1
Work in Progress
Q3 2019

Phase 1	Phase 2	Phase 3

• Research Spikes
• Plugin Setup and Configuration
  • Jenkinsfile
  • Jenkinsfile.sandbox
	• Jenkinsfile templates
	• Implementation details get solidified
	• Refactor existing pipelines to use new templates
		• Existing Job Migration

Full Transformation by Geneva Release - April 2020

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
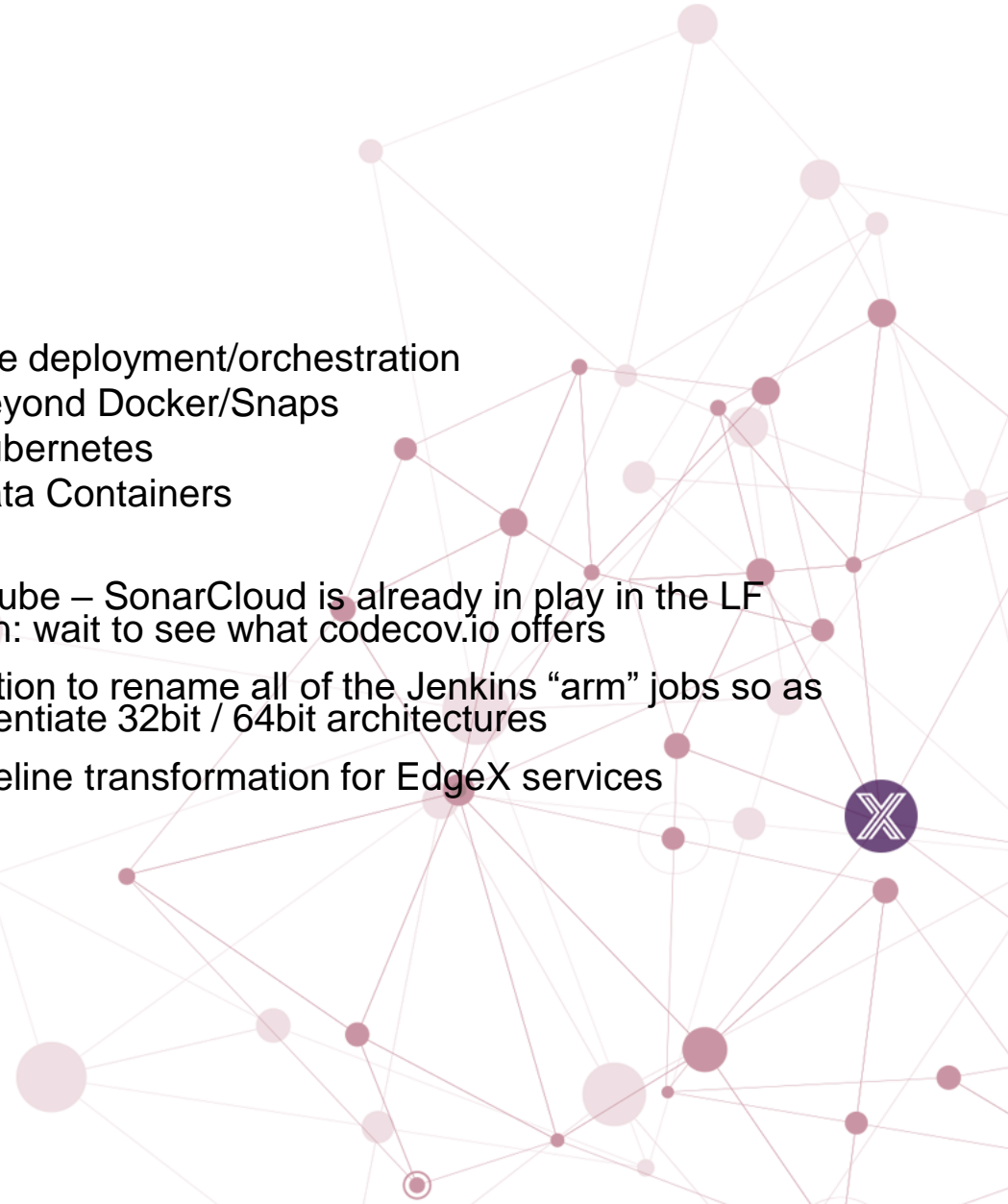
EdgeX DevOps Commitments (Fuji)

Scope of Work	
Add static artifact analysis into the EdgeX Jenkins Pipeline (analysis of Docker /runtime artifacts, not the source code)	🚦 (green)
Add code and artifact signing with semantic versioning	🚦 (green)
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	🚦 (green)
Explore static code analysis like Checkmarx, Coverity, GuardRails, Synk, SonarQube	🚦 (yellow)

- 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

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

Participants (8)

Find a participant

- JG James Gregg (Intel) (Me)
- Ernesto Ojeda (Intel)
- Jim White
- JP Joe Pearson (IBM)
- LG Lenny Goodell (Intel)
- L Lisa Rashidi-Ranjbar
- TE tony espy
- CH Colin Hutchinson (Kong)

Community Participation

Legend: ■ Intel ■ IoTech ■ Dell ■ VMWare ■ ARM ■ Canonical ■ LF ■ Kong
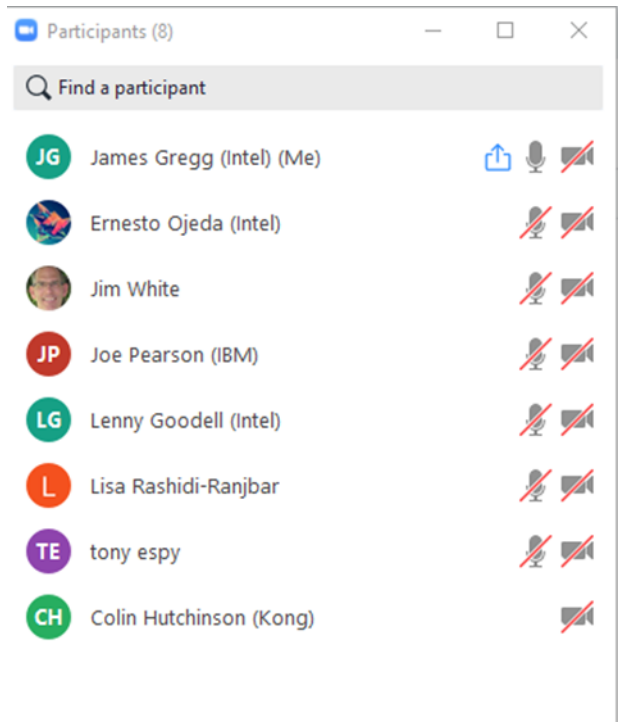
Attendees & Community Participation – ww15
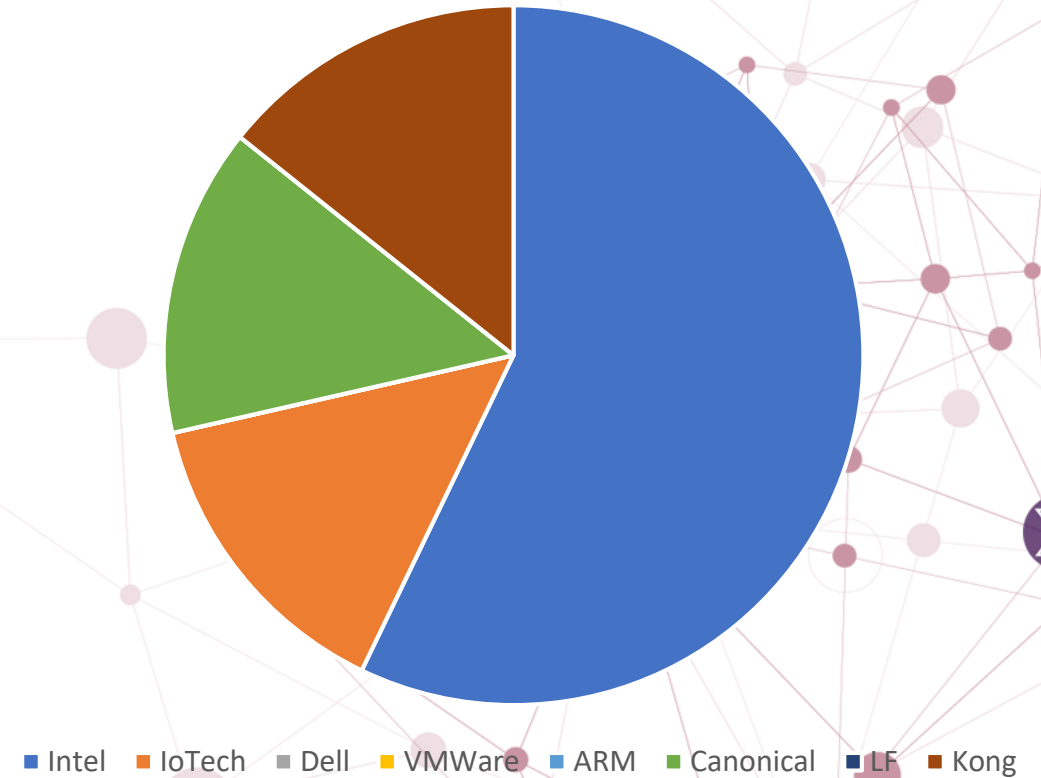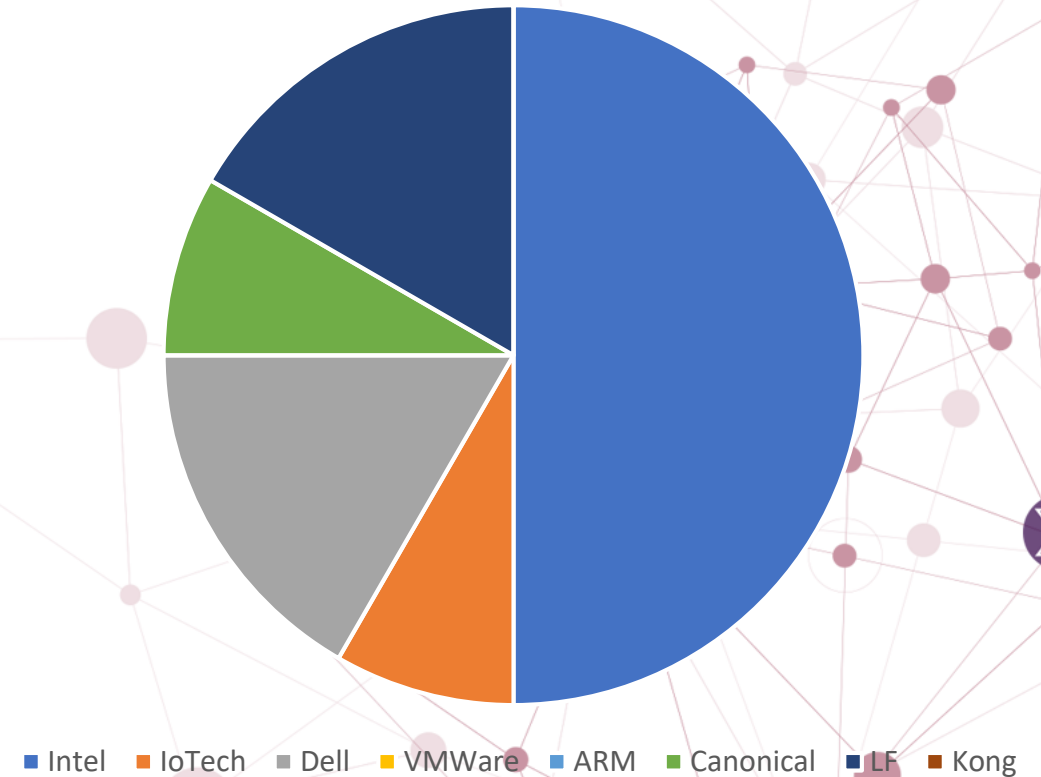

Community Participation

Participants (12)

Find a participant

- JG James Gregg (Intel) (Me)
- Andrew Grimberg (LF)
- T tonyespy
- AB Anthony Bonafide
- BM Bill Mahoney (Intel)
- ER Emilio Reyes (Intel)
- EO Ernesto Ojeda (Intel)
- JP Jeremy Phelps
- Jim White
- JP Joe Pearson (Open Horizon, IBM)
- LG Lenny Goodell (Intel)
- MJ Michael Johanson

■ Intel ■ IoTech ■ Dell ■ VMWare ■ ARM ■ Canonical ■ LF ■ Kong

Attendees & Community Participation – ww16

Attendees

Participants (5)6

- JG James Gregg (Intel) (Me)
- EO Ernesto Ojeda (Intel)
- TE tony espy
- ER Emilio Reyes (Intel)
- JW Jim White
- Lisa Rashidi-Ranjbar

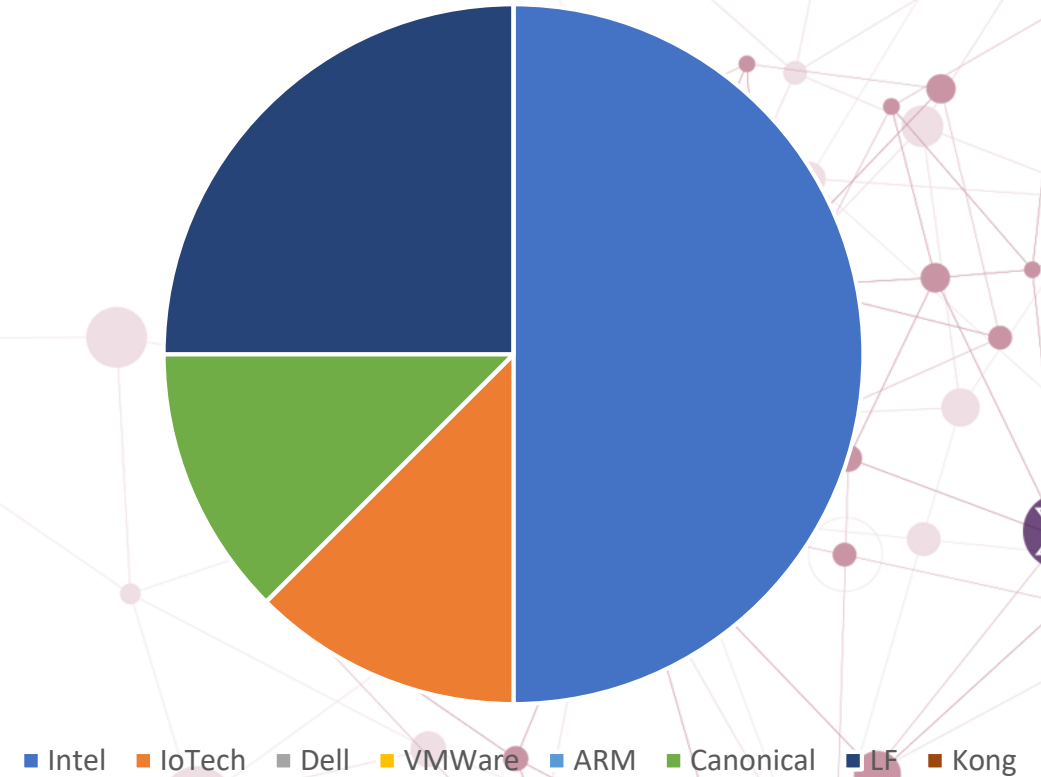## Community Participation

Legend: Intel | IoTech | Dell | VMWare | ARM | Canonical | LF | Kong

Attendees & Community Participation – ww17

Community Participation

Legend: Intel · IoTech · Dell · VMWare · ARM · Canonical · LF · Kong

Participants (8)

- JG James Gregg (Intel) (Me)
- L Lisa Rashidi-Ranjbar
- MJ Michael Johanson
- TE tony espy
- BM Bill Mahoney
- ER Emilio Reyes (Intel)
- LG Lenny Goodell (Intel)
- WM Walt M