



EDGE X FOUNDRY™

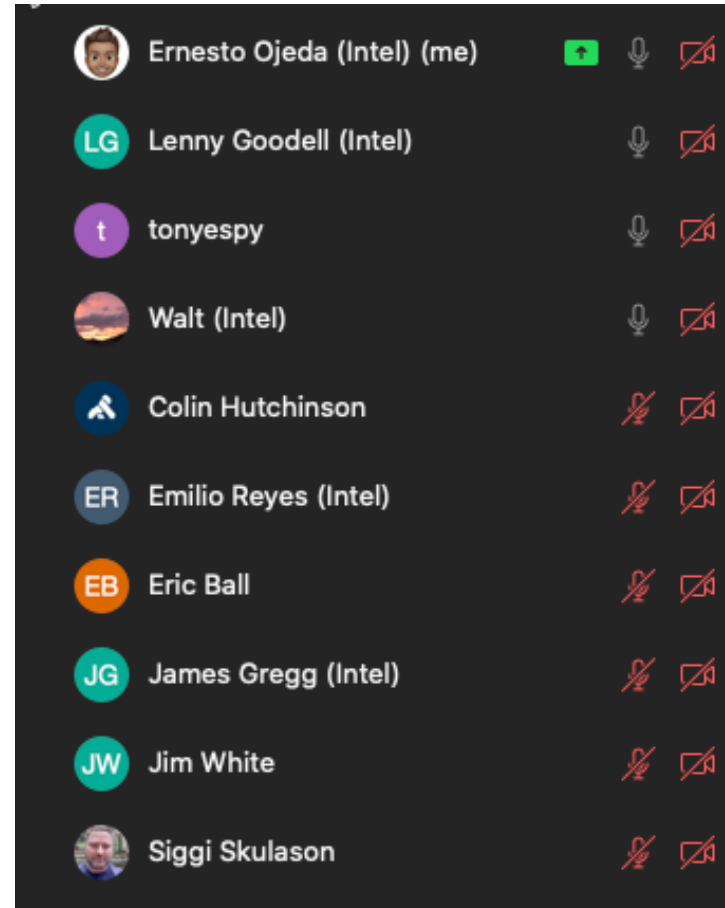
# DevOps Working Group

Thursday July 9, 2020

# Agenda

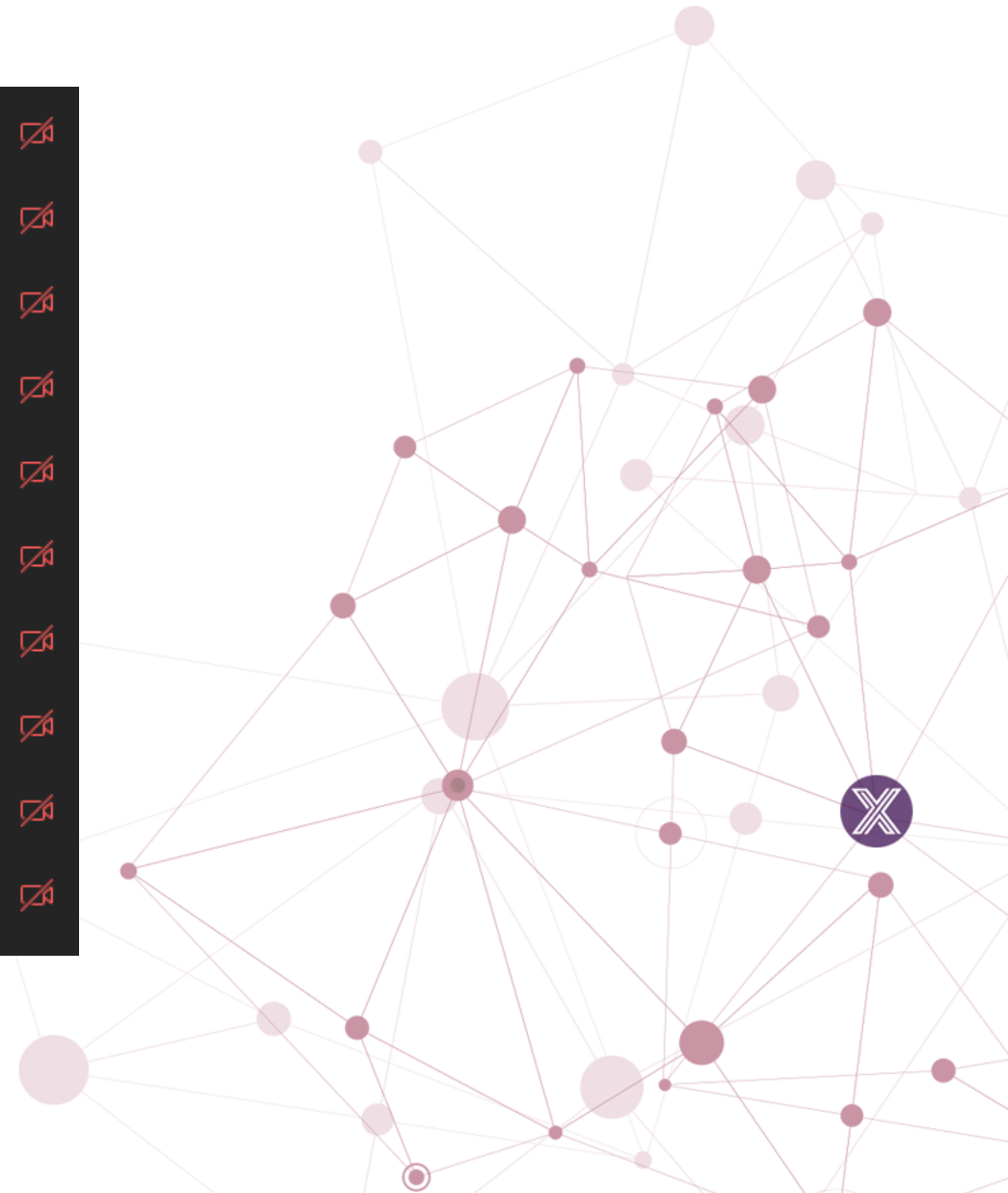
Time	Topic	Owner
10 Min	DevOps Updates (Hanoi)	Ernesto
10 Min	Vetting of Open Source Dependencies	Jim
5 Min	Kanban Board Review	All
5 Min	AOB / Opens	All

# Attendees



A screenshot of a Zoom meeting attendee list. The list is displayed on a dark background and includes the following attendees from top to bottom:

- Ernesto Ojeda (Intel) (me) - Profile picture of a man, green video icon, microphone icon, and a red 'X' icon.
- Lenny Goodell (Intel) - Initials 'LG' in a teal circle, microphone icon, and a red 'X' icon.
- tonyespy - Initials 't' in a purple circle, microphone icon, and a red 'X' icon.
- Walt (Intel) - Profile picture of a sunset, microphone icon, and a red 'X' icon.
- Colin Hutchinson - Profile picture of a person, microphone icon, and a red 'X' icon.
- Emilio Reyes (Intel) - Initials 'ER' in a blue circle, microphone icon, and a red 'X' icon.
- Eric Ball - Initials 'EB' in an orange circle, microphone icon, and a red 'X' icon.
- James Gregg (Intel) - Initials 'JG' in a teal circle, microphone icon, and a red 'X' icon.
- Jim White - Initials 'JW' in a teal circle, microphone icon, and a red 'X' icon.
- Siggi Skulason - Profile picture of a man, microphone icon, and a red 'X' icon.



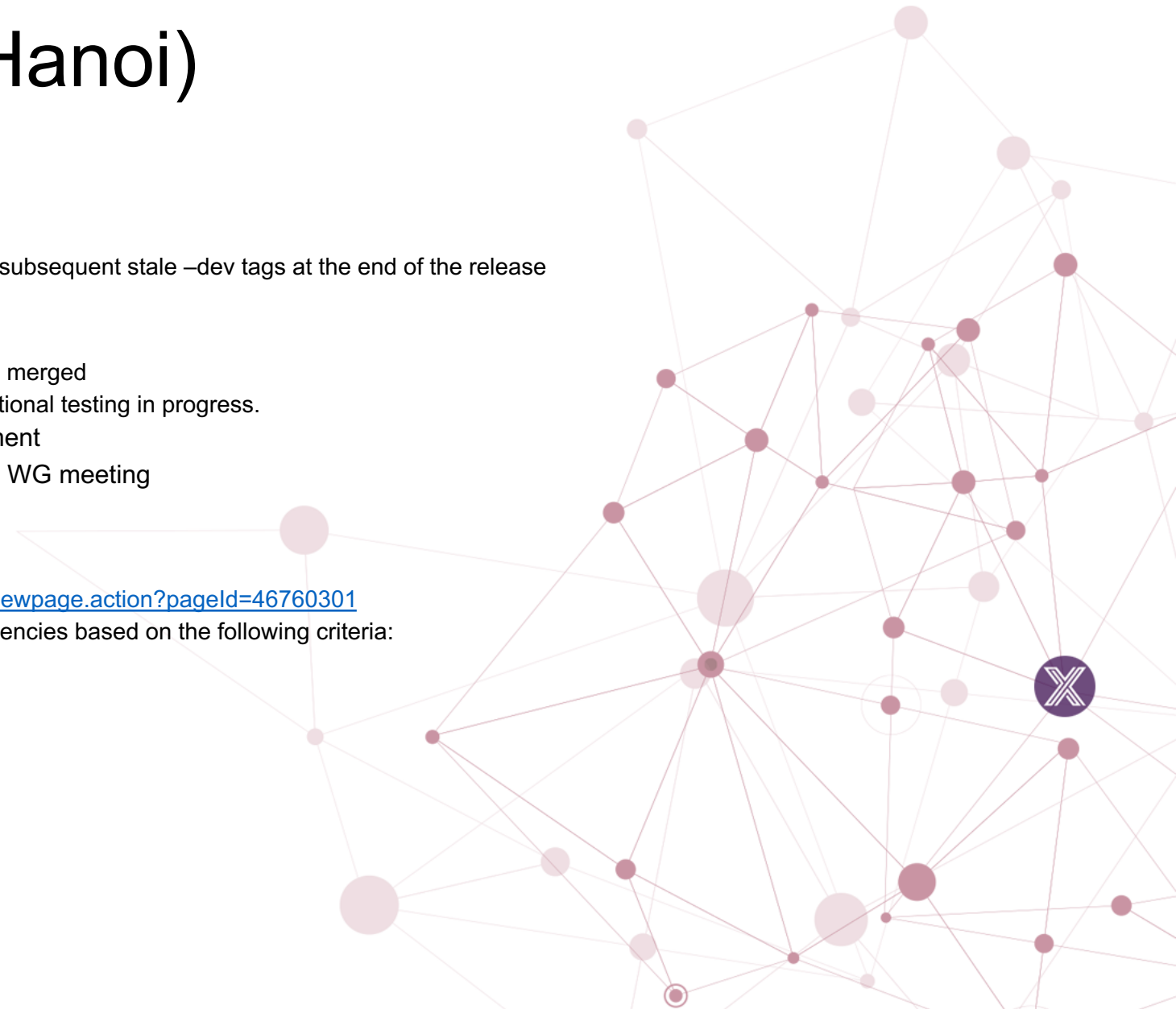
# DevOps WG Update (Hanoi)

## • Pipeline Enhancements

- Build Optimizations for edgex-go
  - git-semver support – Clean up old tags and remove any subsequent stale –dev tags at the end of the release
- edgex-global-pipelines new features:
  - SwaggerHub global library - Code complete, merged
  - Parallelized docker build global library – Code complete, merged
  - Blue/Green deployment of DevOps services – Unit/Functional testing in progress.
- edgex-go new Jenkins pipeline: New code under development
- Audit DockerHub Images – Will present findings in a future WG meeting

## • DevSecOps

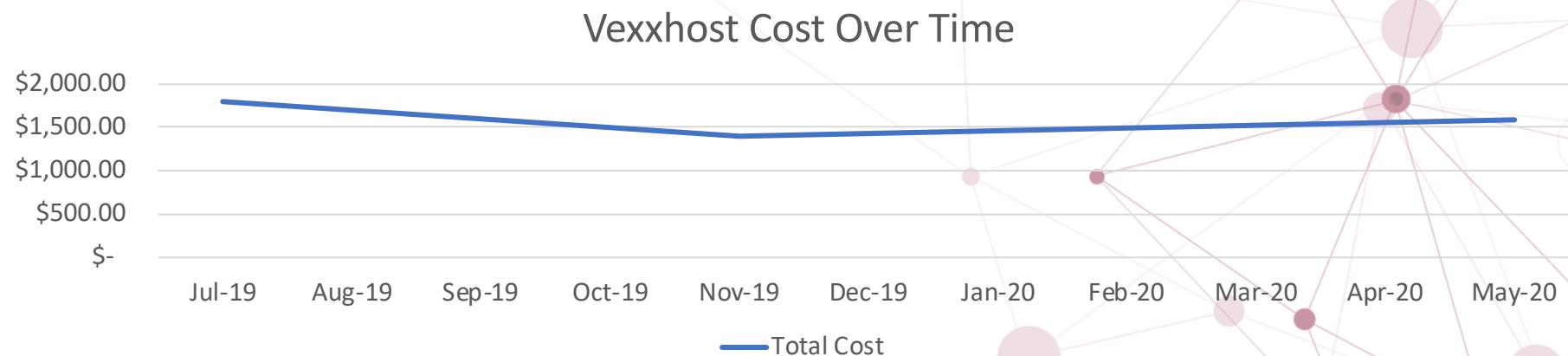
- Vetting of OSS dependencies
  - PDR **on hold** here <https://wiki.edgexfoundry.org/pages/viewpage.action?pageId=46760301>
  - Jim White has done paper study on existing 30+ dependencies based on the following criteria:
    - **Total increase in new imports**
    - **Releases/Tags:** count
    - **Contributors:** count
    - **License** - what is the license, and is it Apache 2.0 compatible?
    - **Stars/Forks/Watchers:** counts
    - **godoc.org metrics:** count
- Community Bridge Feature Requests
  - Snyc API access updates??



# DevOps WG Update (Hanoi)

- **Other**

- James working with LF to add PR Template as part of the repo creation process, using app-functions-sdk-go's as base template
- Code Signing - lftools /sigul replacement
- LFTools / Sigul latest version that supports Python 3.x
  - LF is now considering an alternative signing tool - clarified that this is a future roadmap item - uncommitted
  - LF encountered some of the challenges within their fork trying to refactor sigul to support the latest Python 3.x – **WIP**
- LF Infra Cost analysis
  - We are attempting to start tracking pipeline enhancements to cost savings to the LF infrastructure. According to LF, differences in the efficiency of particular jobs will generally have less effect than the overall number of jobs that are run/number of PR's open.



# Meeting Minutes

## • Notes WW28

- Snyk has given us access to their API. Need a story in the backlog to review the API.
- LF working on a global pipeline library to enable log shipping and cost analysis per build? Eric Ball to find out where cost data is pushed.
- Once cost data source is identified, work with the Project Insights (Community Bridge) team to feed cost data into Project Insights?

## • Vetting of Open Source Dependencies

- Jim found 4 troublesome dependencies (maybe transitive) trying to find source where they are being pulled in
- Issues have been created with working groups to fix the 4 dependencies
- James will set it up a meeting with interested parties to finalize the manual process for developers to follow. Process will be documented on the Wiki. Once the process is more established, we can look to automation in the future.



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

Start Date: 4/22/20



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-label / 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™

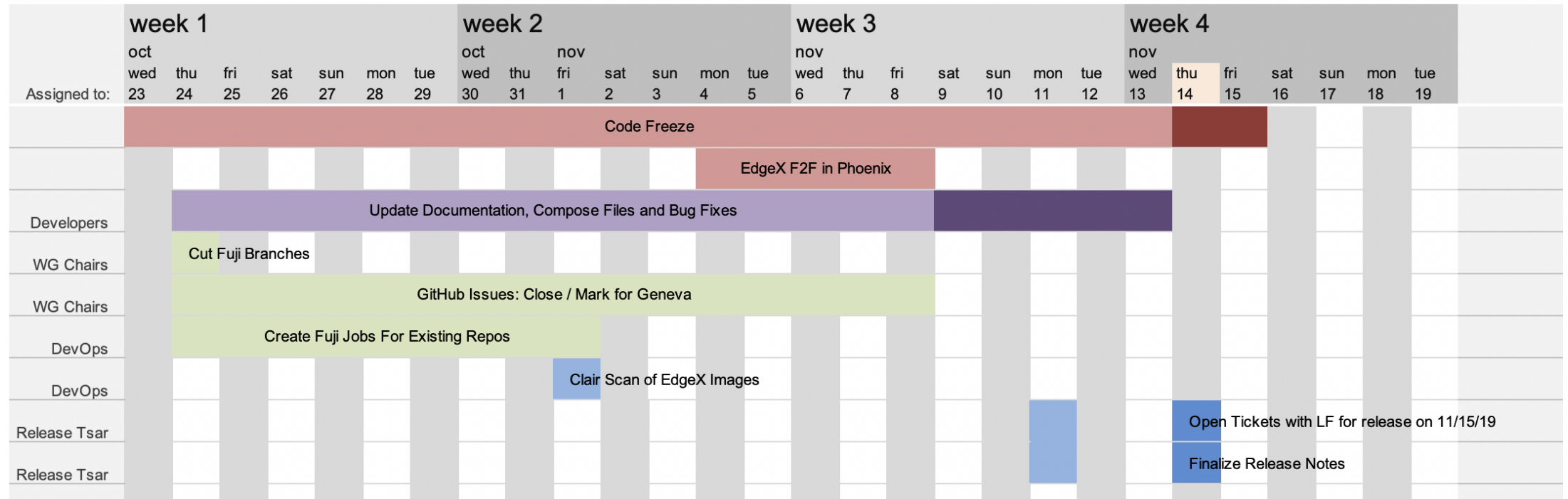
# Geneva Planning

Scope Discussions

# Fuji Release

- Freeze: Oct 23<sup>rd</sup> (Wednesday)
- Release: Nov 15<sup>th</sup> (Friday)

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



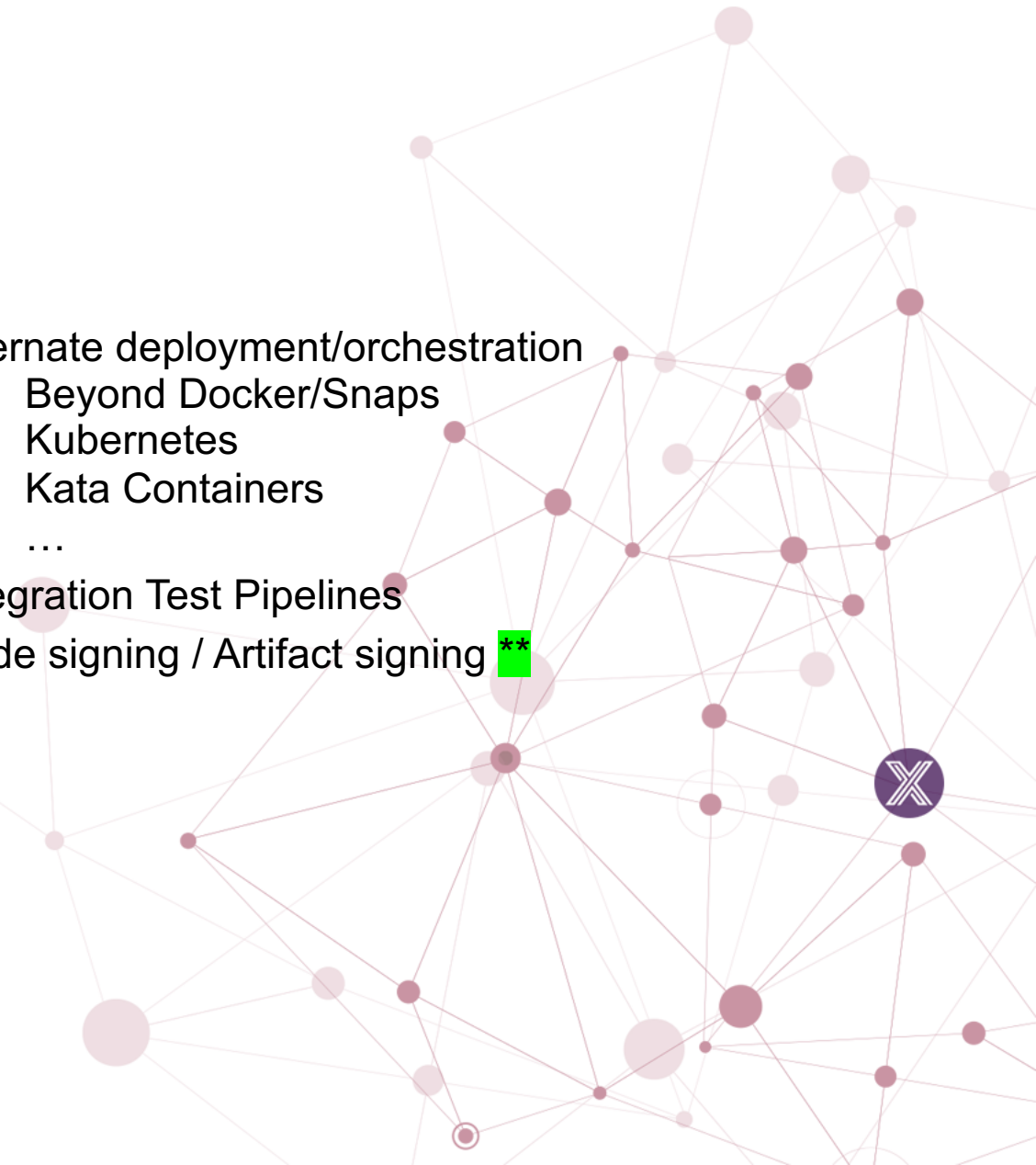
# Geneva – DevOps

## In

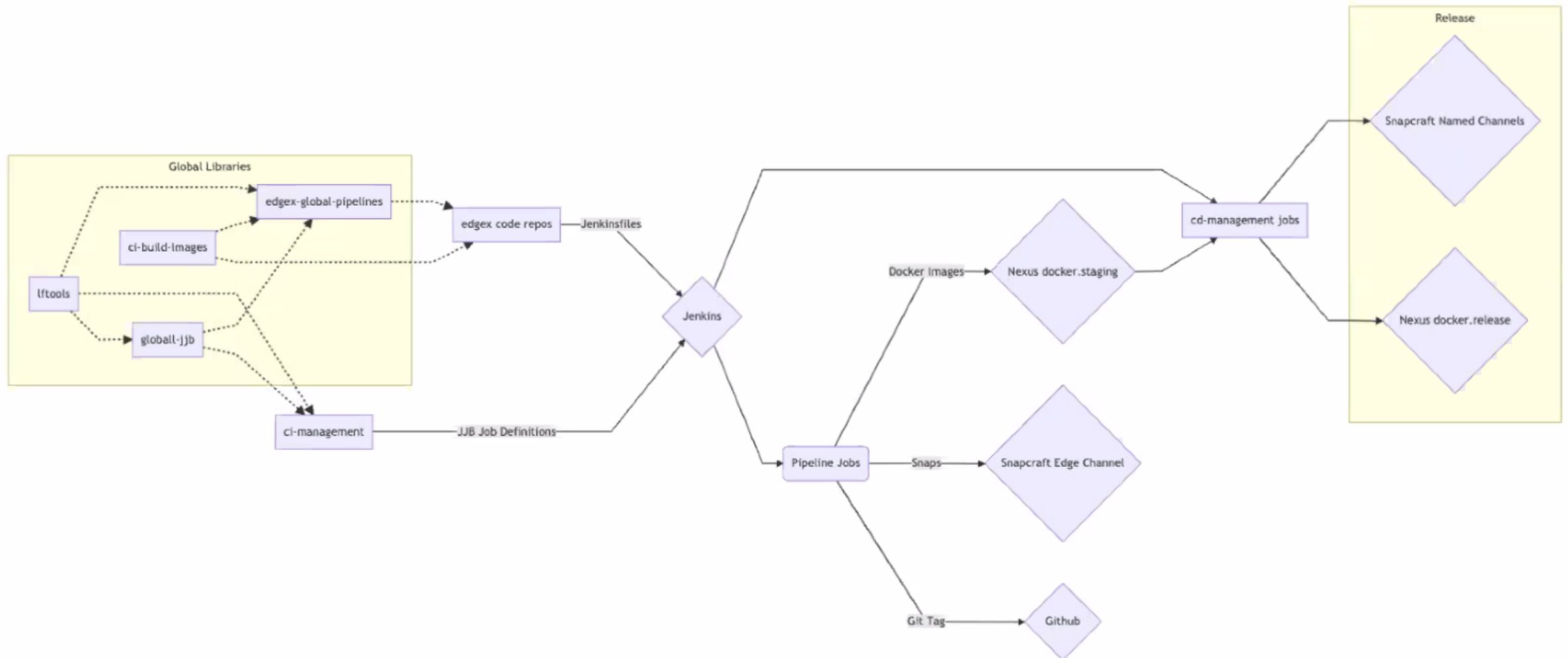
- Full Pipeline transformation for EdgeX services
  - Convert Jenkins JJB Freestyle jobs to Jenkins Pipelines
- Introduce GitHub Org Plugin
- Simplified Jenkinsfile
- Global Libraries to support Jenkins Pipeline transformation
- Add Unit testing to global-libraries (uncommitted) \*\*
- Snyk integration 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 \*\*



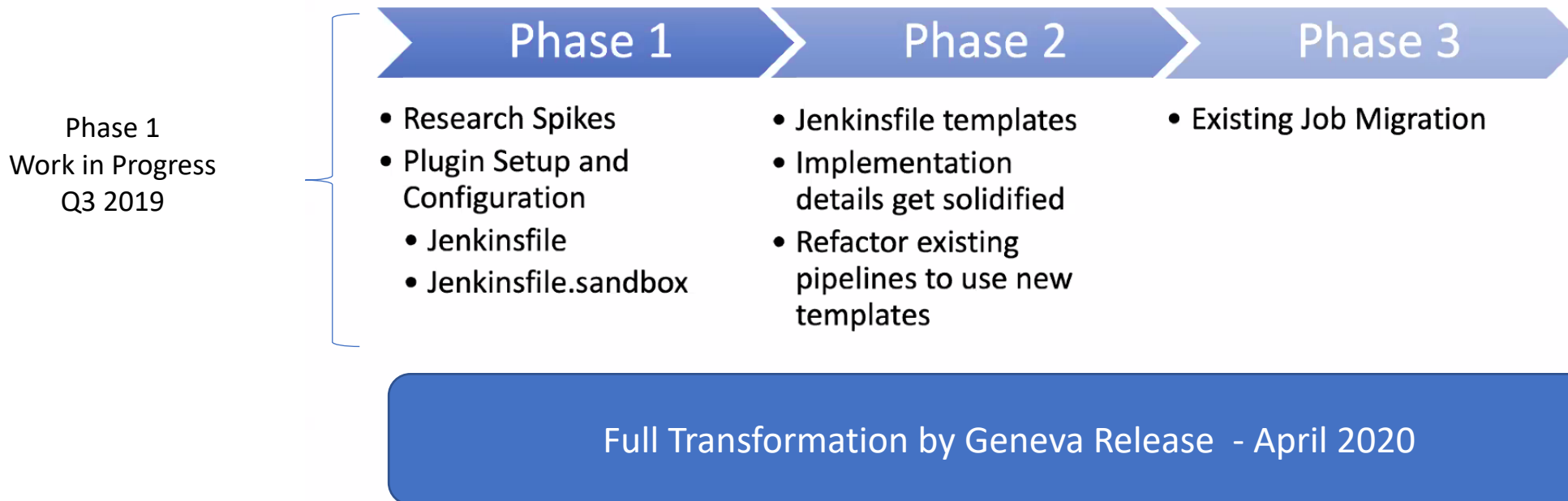
# Geneva Transformation: Architecture





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

## Geneva Transformation





EDGE X FOUNDRY™

# Fuji Planning

Scope Discussions

# Fuji – DevOps

## In

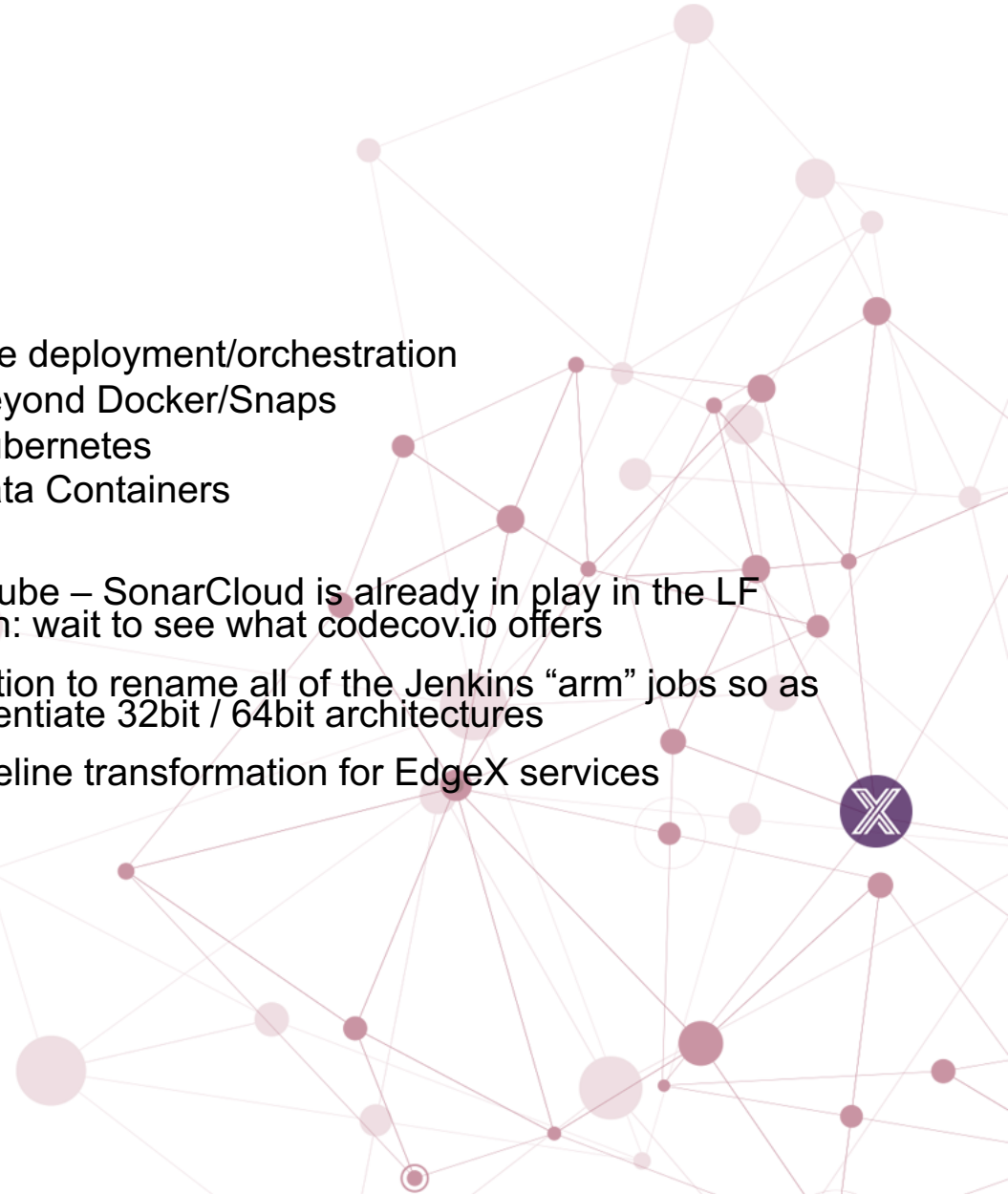
- Static code analysis tool identified and integrated into the EdgeX Jenkins Pipeline for Docker image scanning (Clair Server)

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: <ul style="list-style-type: none"> <li>• Adding Pipelines for EdgeX Foundry base build images</li> <li>• Allow base build images to be managed locally within Nexus</li> <li>• Leverage PyPi Proxy for local pip dependencies</li> </ul>	
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

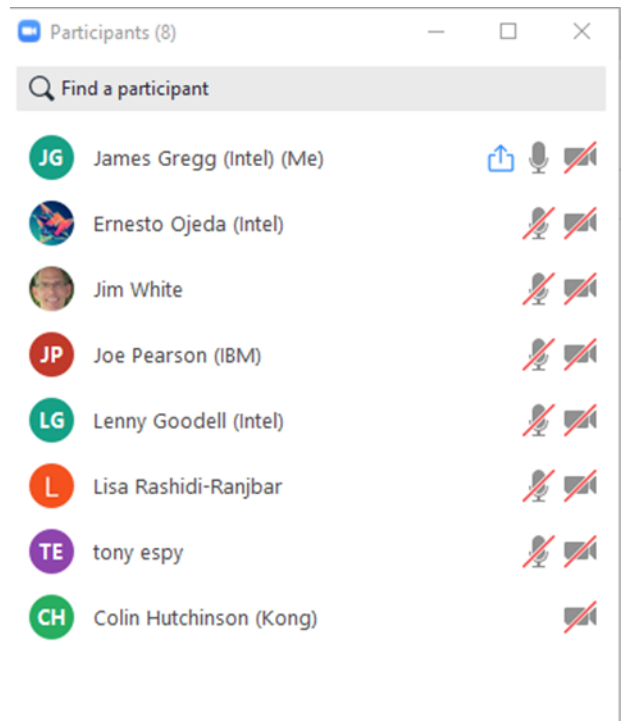


EDGE X FOUNDRY™

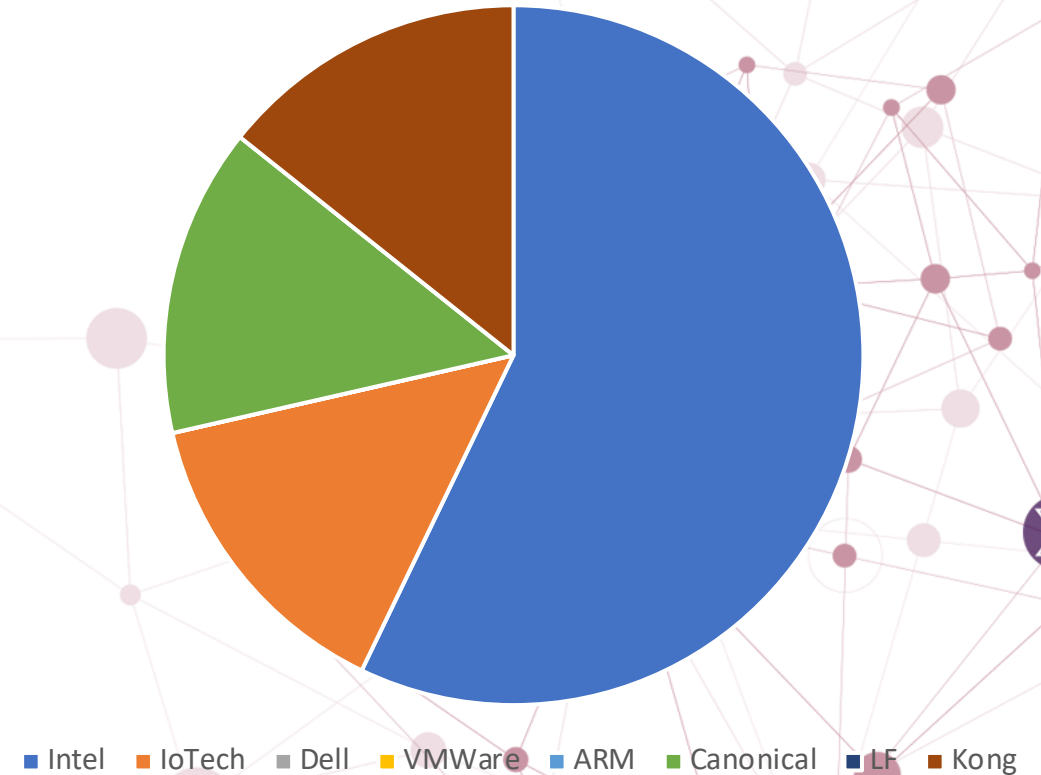
# Past / Future Agenda Topics

	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 Kaniko
	Snyk Dashboard Review

# Attendees & Community Participation – ww14





























Community Participation



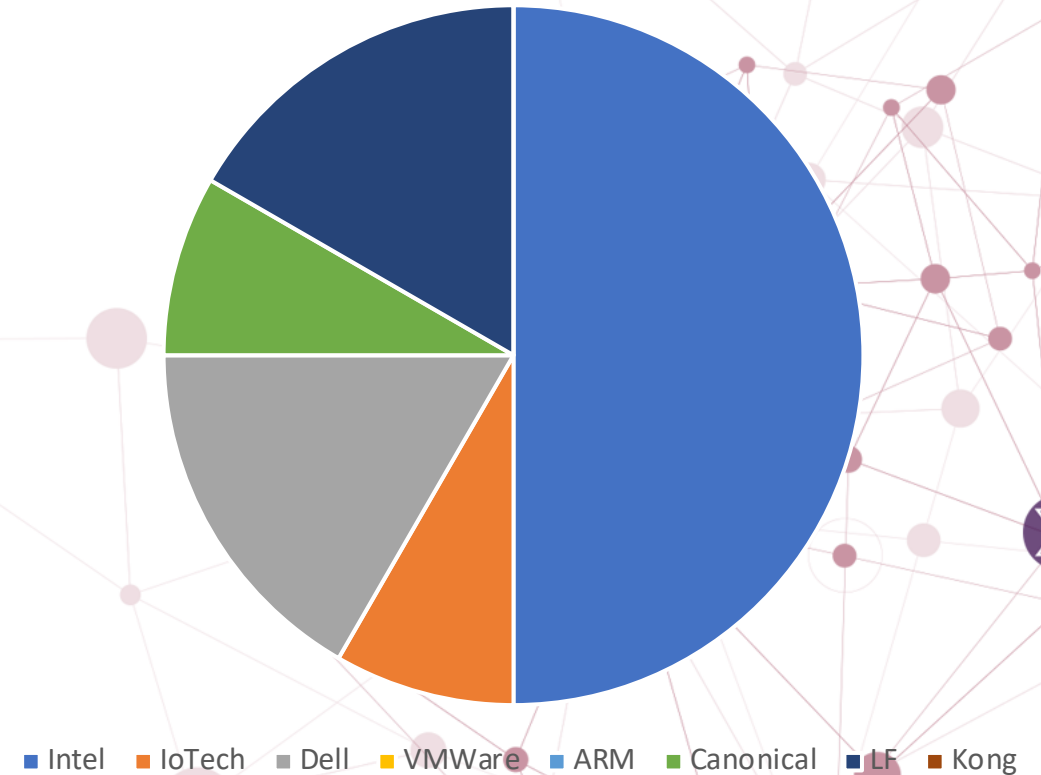
# Attendees & Community Participation – ww15

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  

Community Participation





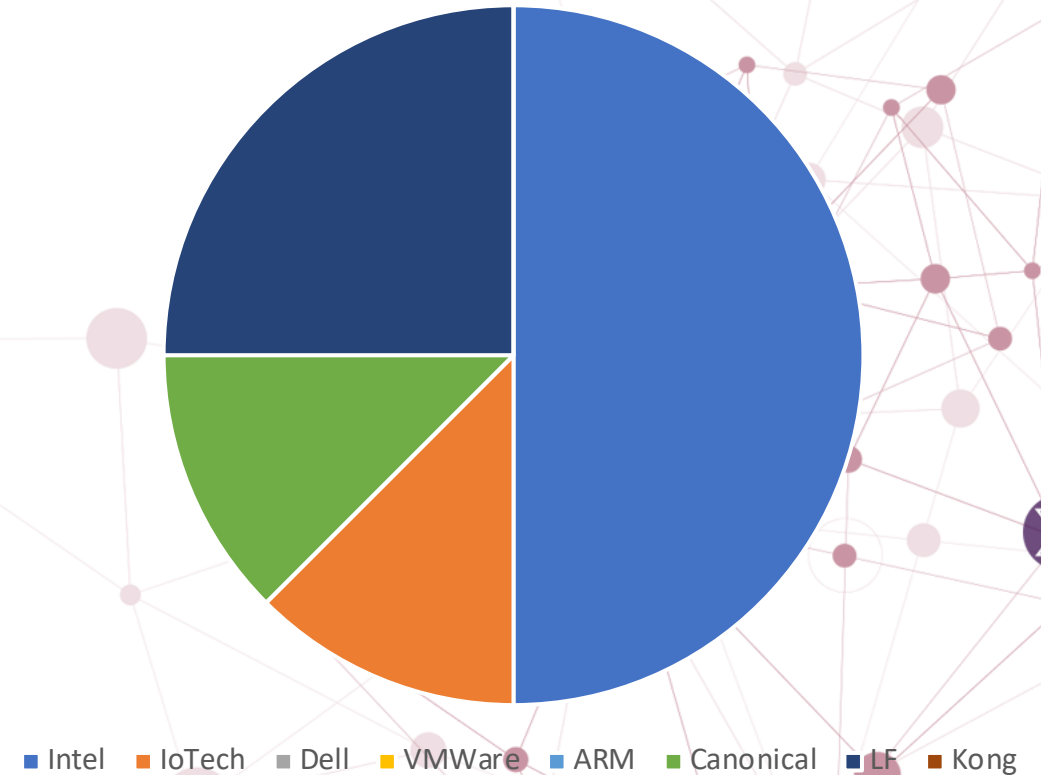
# Attendees & Community Participation – ww16

## 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



# Attendees & Community Participation – ww17

Participants (8)

Find a participant

- 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

Community Participation

