



# EdgeX Planning Conference Action Items and Final Scope List

Kamakura Release

Nov 1-4, 2021

# Jakarta Items

- CORs enablement needs to be documented with information on how to pick/select secure (Kong) or unsecure CORs enablement
  - **Jim** to create doc issue (Done✓)
- One more update to Bootstrap
  - **Lenny** (Done✓)
- App service AES problem
  - Leave old transform in place to keep backward compatibility
  - Have fix in place that people can opt to use (Done✓)
  - New feature to be merged (**Bryon and Lenny and Alex**) (Done✓)
  - **Bryon** will publish CVE once we have remediation in place
  - Issue for docs created (pending PRs)
- New issue in Provision Watcher (Lenny discovered)
  - Watcher is trying to pull it from cache when it has not been added there
- Docs
  - Need binary data sending/consuming example (**Jim** to create issue) (Done✓)
  - **Jim/Lenny** to clean up Jakarta docs project board; determine what is in scope (Done✓)
  - URLs in repository READMEs and TOML files point to different versions (**Jim** to search and create issues or fix)
  - Add and improve device service docs (**Iain** to lead discussion about what is needed in next DS WG meeting)
  - Getting started guide – possibly replace HiveMQ with EMQ Xbroker (**Jim** to explore; create issues as necessary)
  - Take care of to-do in GUI with Snaps (<https://docs.edgexfoundry.org/2.1/getting-started/tools/Ch-GUI/#snaps>) (**Tony** & Canonical to work)

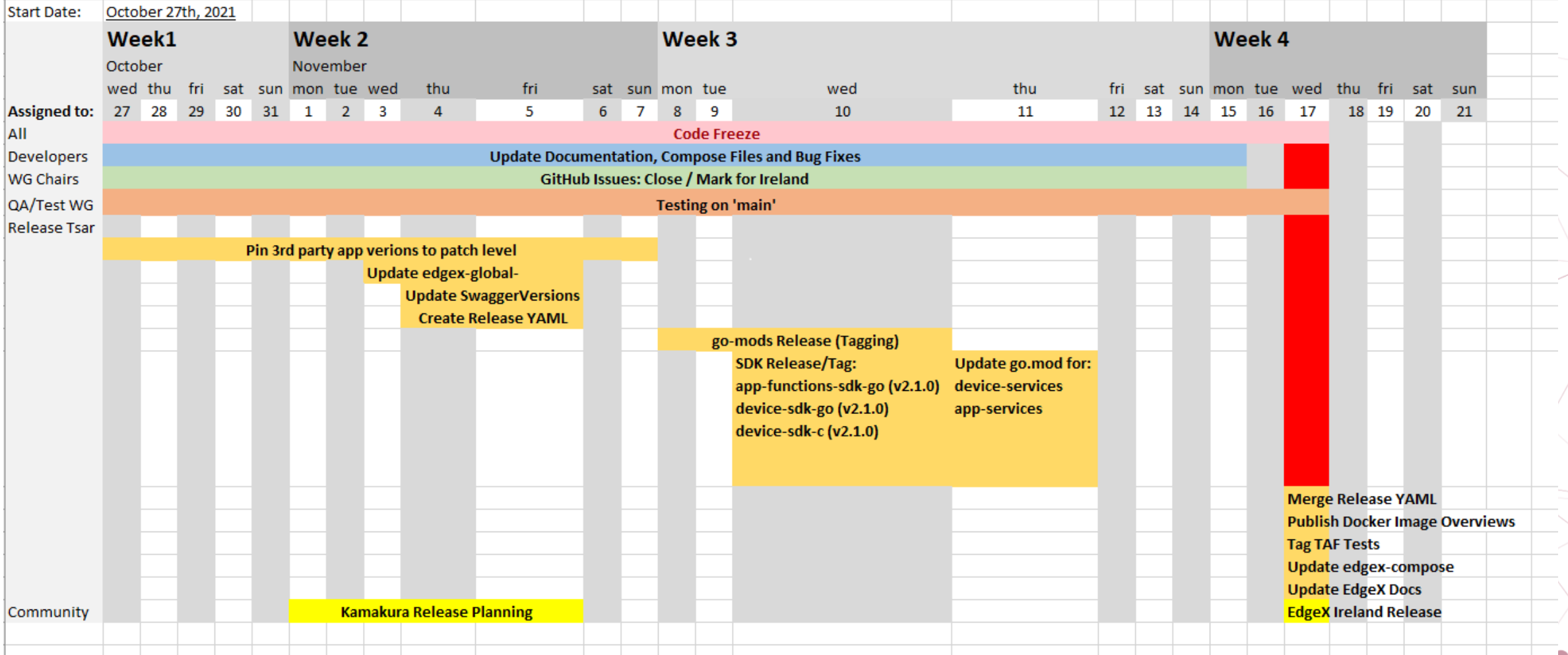
# Jakarta Manual Testing for LTS

Test	Owner/Actions
Test any configuration not already handled by new TAF tests	<ul style="list-style-type: none"> <li>Persistence on/off (running without core data and with persistence = false pass through) may be covered</li> <li>DS REST vs Message Bus (Redis) – test REST with Virtual DS</li> <li>Test CORS support – check headers there if enabled (both ways secure/non secure)</li> </ul> <p>Jim Action</p>
Test reboot of the system. Make sure EdgeX can come back up	<p>Containerized - Lenny Action</p> <p>Snap reboot test – Tony Action</p>
Test device services (with real hardware where possible)	<p>Test something that uses C SDK</p> <p>Bacnet update – simulator - Iain Action</p> <p>Modbus, SNMP, GPIO – real device – Jim Action</p> <p>CoAP, RFID, Grove – in process of port to V2; address when ported</p>
Working with service authors to identify minimal tests	<p>Start requiring contributions of DS to include this</p> <p>Kamakura task – add MD page to each DS (create issues – Jim Action)</p>
Test different combinations/permutations of app functions in app services	<p>MQTT and HTTP export are important ones – covered TAF</p> <p>Use sample config to drive this – Jim Action</p>
<del>Test on Windows Native</del>	Kamakura Arch topic
Test running with multiple services of same type (device service and app services)	<p>App service – done in TAF</p> <p>DS – should be done in all; not a blocker</p> <p>Generic test – 2 DS running simultaneous</p> <p>Jim Action – try MQTT, Modbus</p>

# Jakarta LTS Documentation

- Agreement by community about what is included in LTS
  - All edgex-go services (core data, metadata, command, support scheduler, notifications) – v2.1
  - App functions SDK v2.1
  - App service configurable v2.1
  - Device Service SDK Go v 2.1
  - Device Service SDK C v2.1
  - Device services v2.1s (virtual-go, modbus-go, snmp-go, rest-go, mqtt-go)
  - Not included in LTS
    - LLRP app service and device service
    - SMA (executor and agent) – with caveat
      - The system management services were deprecated with the Ireland. Should a significant issue (critical or high level issue) be discovered on the system management services and a fix to the issue has been created, the community will include the fix with any other LTS patch release. However, a fix to a system management service will not trigger an LTS patch release alone.
    - device-camera-go but we will have one for 1st quarter 2022 (either replacement or improved current)
    - device-gpio, device-grove-c, device-bacnet-c, device-uart-c , coap
    - GUI/CLI tools are dev tools – not included officially in LTS
      - Best effort to keep these compatible, but they are use at your own risk
- Place these details on the LTS wiki page - Jim action item to create page and reaction for TSC (Done✓)

# Jakarta Release LTS / 2.1.0



# Kamakura Release – General Themes/Objectives

- Version 2.2
  - Must still be backward compatible
  - Can add, but not remove deprecated items yet
  - Not LTS, but must consider we have LTS now on the books
- Major additions under consideration
  - Improved security secrets seeding and allowing for delayed service starts
  - Initial North to south message bus
  - Metrics collection
  - Dynamic device profiles (changes to device profiles)
  - Better (native) Windows support
  - Improve testing (including first real hardware testing)
  - V2 of the CLI

# Business and Outreach

- Web site
  - No major changes to the site (English or Chinese) for next release
  - Looking to move costing to LF/LF Edge (**Jim** action to work with LF)
    - Find out situation with funding/costs on Chinese site (**Jim** action) (**Done✓**)
  - Goal: get more stats from Chinese site, docs, Wiki
    - Try to understand what is driving traffic to any of these
- Developer Evangelism
  - EdgeX Ready – evangelism/marketing social media campaign – **Rodney/Jim** action
    - Participation and results from EdgeX Ready will determine what comes next
  - Developer Badging – kick off with Jakarta release
    - **Ernesto** to demo at next DevOps
    - Need marketing campaign at time of the release – **Jim** action
  - EdgeX Smart \* Challenge 2022
    - Need sponsors – **Jim** action
- Adopter series
  - Hold more ad hoc than try to do it monthly
  - Create separate Wiki page to highlight this series and the recordings from it – **Jim** action
- Events/Shows – on hold until spring 2022
- Regular blog/vlog – shooting for something every month – **Jim** action

# Cadence Check

- April & Oct/Nov remain target release months
  - Kamakura release – May 2022
  - Levski release – Oct 2022
  - Minnesota release – May 2023
  - **Napa** release – Oct 2023 (thank you Ernesto)
- Looking to hold next planning meeting (Levski planning) as face-to-face in April 2022
  - Revisit in Q1 2022 as companies emerge from Covid protocols
  - Looking for organization willing to host
- No changes in work group meeting times
  - However – the calendar meetings need to shift for standard time (because they are all in UTC). **Jim** action item.



# Tracking project decisions

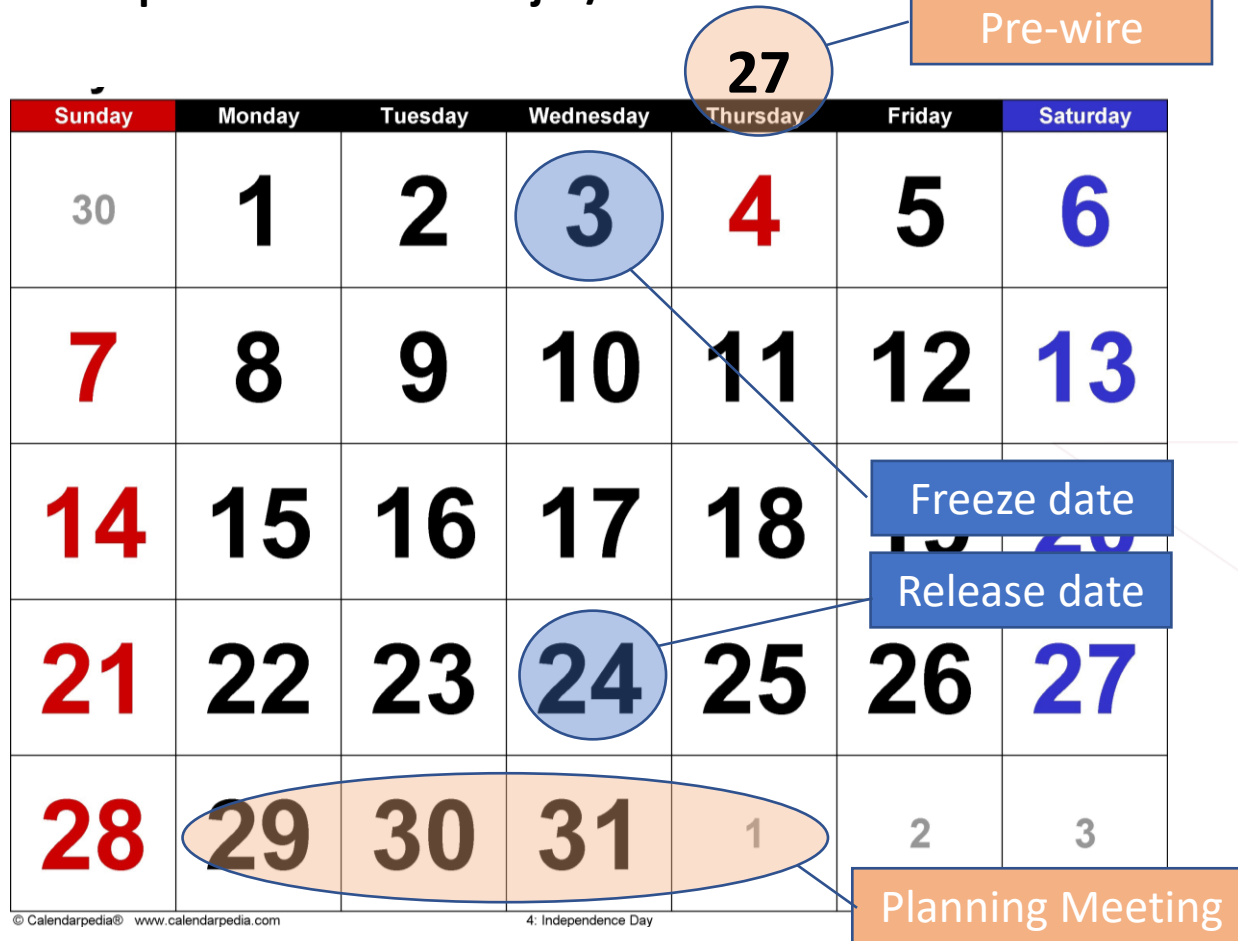
- Use permanent project board in Github
  - Use cards or issue (issue if code action/PR is needed)
  - All decisions covered (small to large)
    - Large and “significant architecture” changes also have an ADR
  - WG leads have action to update the board with any/all WG decisions
  - Column per release (but never take down or archive the columns)
  - Check to make sure all WG leads have access to add/make changes
  - **Jim** action to draft Wiki document on this approach for TSC review
    - Setup project board in Github

# Release Timing

- Move planning meeting to week following the release
- Generally attempt to select release date about 2 months in advance of the release
  - Typically early March and early September for spring/fall releases
  - Release date preferred to have be a Wednesday
  - Adjust per circumstances and TSC review
- Release schedule for minor release
  - Freeze date 2 weeks in advance of release date
  - Prewire, Thursday prior to freeze date
- Release schedule for major or LTS release
  - Freeze date 3 weeks in advance of release date
  - Prewire, Thursday prior to freeze date
- Planning meeting – generally the week following the release
  - Virtual meeting: Monday – Thursday with Friday for training/non-conference
  - In person: Tuesday-Thursday with Thursday afternoon for training/non-conference
- **Jim** to create Wiki page outlining the above

# Formalized Planning

Example Calendar for Major/LTS Release



Example Calendar for Minor Release



# General Action Items

- Research alternatives to big 3<sup>rd</sup> party enterprise components (Kong, Vault, Consul)
  - Tyk, Consul API Gateway, NginX, ...
- Check poor 3<sup>rd</sup> party dependencies for any updates/better support
  - **Jim** action
- What is required to trigger performance tests from edgex-taf?
  - Is this already being done?
  - **James** action
- Call help wanted on record and replay feature
- Have conversation with eKuiper team
  - Steer eKuiper to offer a separate credentials file (versus their general configuration file)
  - **Lenny** action
- **Jim** to update Wiki Backlog and Github issues with status of decisions per pre-wire and planning meeting

# Add to Monthly Architect's Meeting discussion list

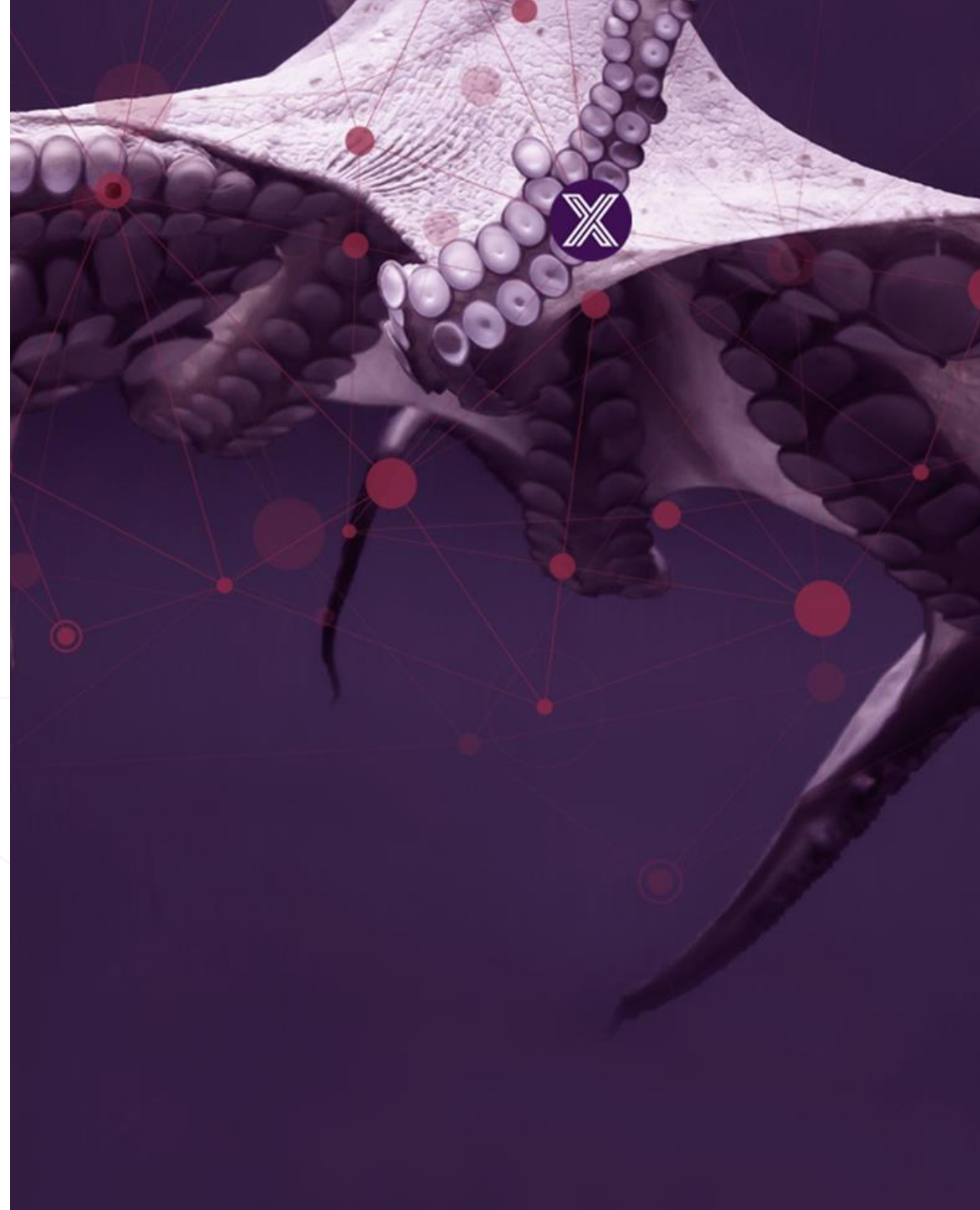
- *In addition to ADRs currently in scope*
- Support device/sensors that have a functional protocol / API
  - Discuss how to start and where work originates
- Global configuration
  - Should be covered and decided before next major release cycle
- Securing the EdgeX message bus when using MQTT implementation
  - How to generate credentials and tell the services about these
  - How to initialize the MQTT secrets for the internal message bus comms

# Items to add/move/return to the backlog

- Control plane events
  - Needs a new ADR but may not be covered in this release
  - This may change in regard to the Eaton requirements (TBD)
- Move Command API (and functionality) into metadata service
- Offer Windows artifacts (MSI, exe, ?)
- Global configuration
- EdgeX size concern (diet plan for 3<sup>rd</sup> party components to include Kong, Vault, Consul)
- SMA alternatives/replacement
- Improved and expanded support notifications
- Replace poorly supported 3<sup>rd</sup> party libs
- Prometheus integration
- Record and replay
- Handle both JSON request bodies as well as CBOR request bodies in commands to DS
  - For Go; already done in C
- DS Filtering
- CloudEvent as TargetType
- Secure MQTT message bus (for internal communications)
- Use existing instance of Vault (bring your own Vault)
- Integrate GoKart security scanning (specific to Go Lang)

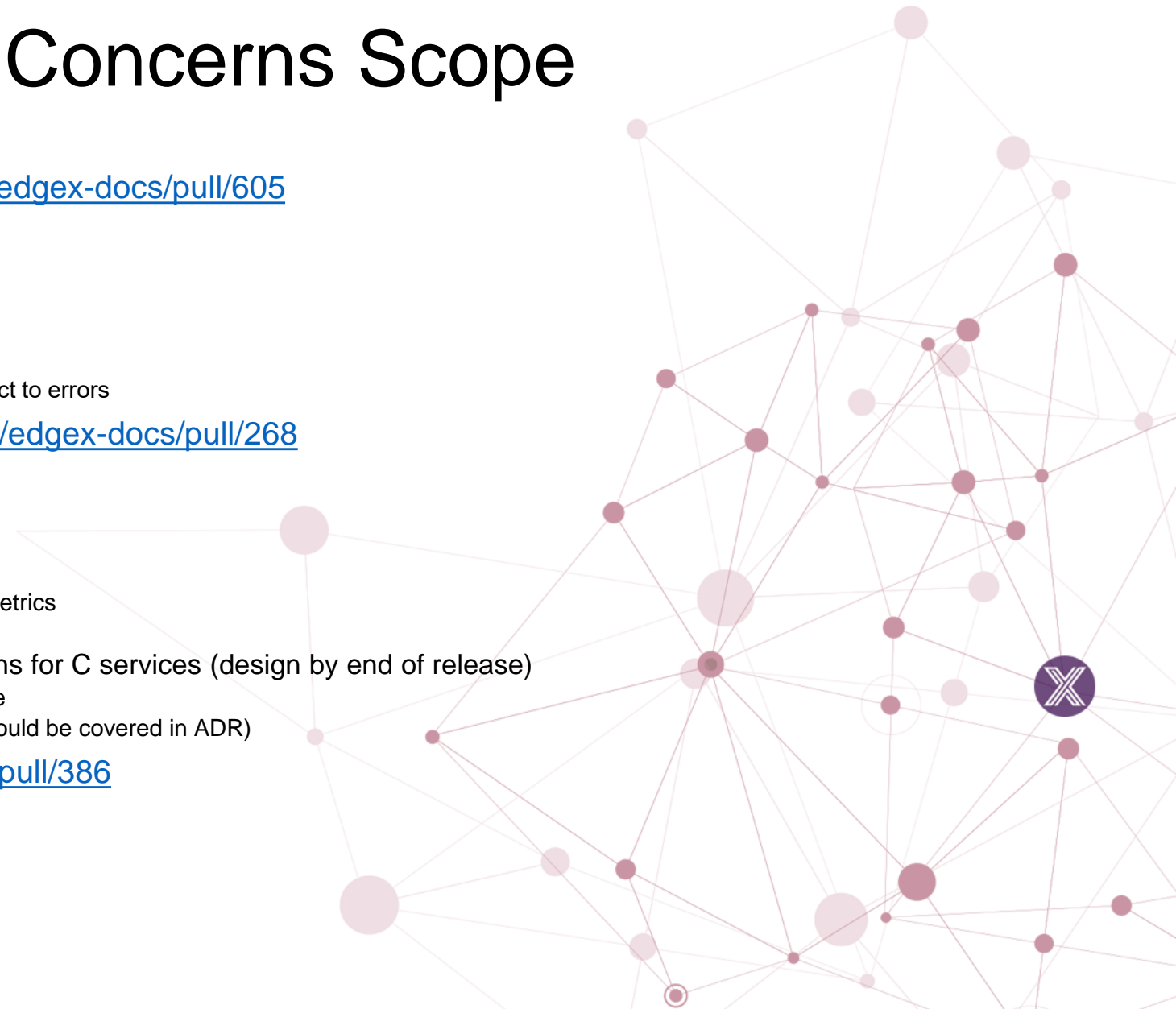
EDGE X FOUNDRY™

# Kamakura Scope



# General/Cross Cutting Concerns Scope

- Dynamic Profiles - <https://github.com/edgexfoundry/edgex-docs/pull/605>
  - Owner: architects/global to start
  - T-shirt size - medium
  - Finalize and approve ADR by end of year
  - “Make law”/enforce in the code during the release
    - Validation to occur in metadata
    - CLI, GUI, etc. won't have to validate, but will have to react to errors
- Metrics Collection - <https://github.com/edgexfoundry/edgex-docs/pull/268>
  - Owner: architects/global to start
  - T-shirt size – large
  - Finalize and approve ADR by end of year
    - ADR needs to look at lib/package for Go and C
    - ADR needs to cover configuration to turn on/off use of metrics
    - Does not include control plane events
  - Implement in Go services this release; research options for C services (design by end of release)
    - go-metrics already a dependency incorporate in Go code
    - May need feature flag for compile time incorporation (should be covered in ADR)
- UoM - <https://github.com/edgexfoundry/edgex-docs/pull/386>
  - Owner: shared Core/Device Service task
  - T-shirt size – small for ADR; impl TBD
  - Complete and approve ADR by end of January 2022
  - Implementation a stretch goal but TBD after ADR
    - Validation not covered in ADR or impl





# General/Cross Cutting Concerns Scope

- North-south message bus
  - Owner: architects/global to start
  - T-shirt size – small for ADR; impl large or x-large
  - Complete ADR in this release cycle
  - Stretch goal:
    - Implement MQTT subscription in command service
    - Implement DS'es with subscription capability (via Go and C SDKs) to start receiving commands from command service via message bus
- Kubernetes support
  - Owner: PM/TSC chair task
  - T-shirt size – small for research and doc; medium for impl
  - Get feedback from K8s experts
    - What doesn't work? What is insufficient with current examples and documentation
  - Create a page in the EdgeX docs
    - EdgeX is not CNCF but edge native. Cover why EdgeX is not HA, how to scale EdgeX (out not up), why EdgeX doesn't work in a cluster, limitations of EdgeX in K8s (like devices/sensors impact).
  - Based on expert feedback, create improved edgex-example or possibly even 1st level artifacts (like docker-compose and snaps)
- Zero touch provisioning
  - Owner: PM/TSC/Canonical chair task
  - T-shirt size – small for research
  - Research project – what do adopters want/need? What does zero touch mean to them

# General/Cross Cutting Concerns Scope

- (Native) Windows support
  - Owner: PM task
  - T-shirt size medium
  - Research / test Go compile direct to not include/compile ZMQ
    - Use zmq linux files special names that don't include them
    - Additional work pending success and review of this approach
- Support device/sensors that have a functional protocol / API
  - Owner: DS and Core WG task
  - T-shirt size – small for ADR; impl out of scope for this release
    - Impacts Command and device services
  - Example: ONVIF – camera has a SOAP based web service; function call that takes multiple parameters
  - Supporting REST vs SOAP or GRPC calls
    - Modeling we have is not up to the task to support these more complex function calls
  - Research and discuss this feature prior to ADR development in **Monthly Architects Meeting**
    - Discuss how to start and where work originates
  - Create ADR in this release if architect's concur
    - Avoid inventing something from scratch if we can
    - Potentially some research prior to ADR
    - Discuss in architecture group about how to start and where work originates

# General/Cross Cutting Concerns Scope

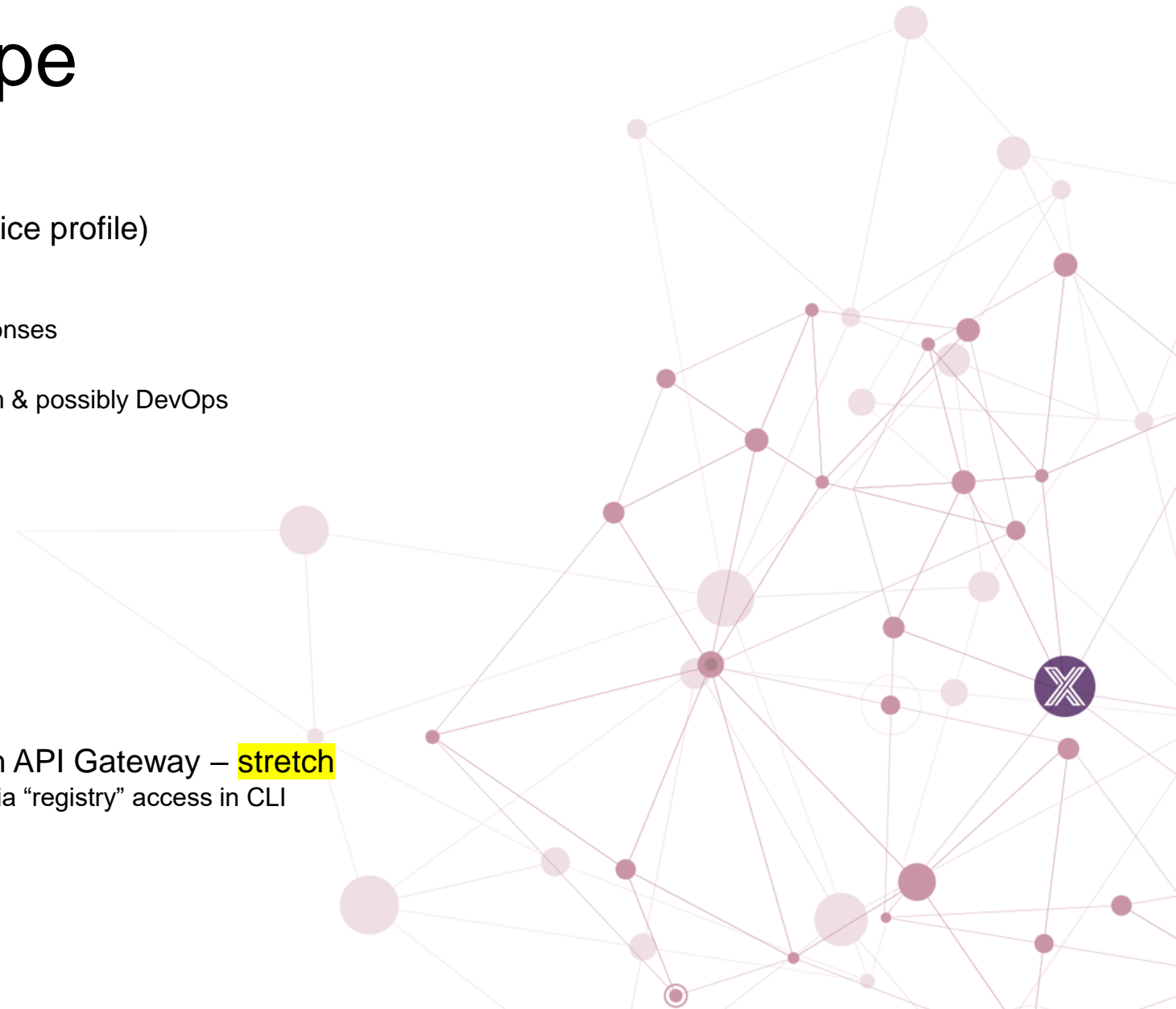
- Upgrade Go (1.17)
  - Owner: smoke test: Core/James, apply across services: DevOps/Ernesto
  - T-shirt size medium
  - Do in December
- Add "make lint" target and add to "make test" target (only for GO)
  - Owner: Jim/Lenny
  - T-shirt size: medium; roll out as we can through the release period
  - <https://github.com/edgexfoundry/go-mod-secrets/issues/129>
  - Needs to be done in mods and DS SDKs and DSs, LLRP AS
  - Based on what the lint tool finds, could add to scope of work to all services

# Core Scope

- Always send CBOR option for performance
  - T-shirt size: medium
  - Test first and apply if it makes sense
    - Run functional tests at the same time to make sure we are not breaking something
  - What would “significant” savings look like: >100 milliseconds
- Update lastConnected should not trigger device service callback
  - T-shirt size: small – tech debt we need to clean up
- Refactor the pkg.Encode func name
  - T-shirt size: small - tech debt; just need better naming
- Research move to typing in JSON (Impl **stretch**)
  - T-shirt size: small to research; impl TBD
  - Protocol properties; not just internal because of provision watcher
  - Tech debt to use interfaces vs string:string in properties
  - Complexity - using the string match on provision watcher
- ADR for addressing support of specifying parent/child relationship
  - Owner: Eaton
  - T-shirt size: small
  - Use of labels vs use of custom attribute in device object
- Add new tests to address any new features
  - T-shirt size: TBD
- Test with Core Data not in place
  - T-shirt size: ?
- Test send of sensor data from DS via REST (vs message bus)
  - T-shirt size: ?
- Test that data successfully comes back from eKuiper
  - T-shirt size: ?
  - Coming with eKuiper 1.4
- Define what other stress testing we need
  - T-shirt size: small
  - Implementation may be in scope
- Test Redis use of memory
  - T-shirt size: ?
  - Test Redis in-memory database and how much can we persist
  - What’s the breaking point?
- Real hardware testing
  - Owner: **Jim**
  - T-shirt size: small for pilot
  - Research setting up accessible hardware in LF Edge Lab
  - Understand CI/CD implications
  - May be manual testing to start

# Core GUI and CLI Scope

- GUI
  - Fix to no allow banned activity (ex: modify device profile)
    - T-shirt size: small
    - To be handled after we address dynamic profiles
    - Don't validate in the GUI, just react to API call responses
  - Add testing
    - T-shirt size: large – impacts GUI team, testing team & possibly DevOps
    - What tools to use?
    - How much is automated versus manual
- CLI
  - Complete V2 implementation and update docs
    - T-shirt size: ?
  - Support app and device services - stretch
    - T-shirt size: ?
  - Scrub issues list - stretch
    - T-shirt size: ?
  - Support registry and proxy setup going through API Gateway – stretch
    - T-shirt size: medium; concern on what is covered via “registry” access in CLI
    - Research first
    - Likely to complete only one of these at best



# Device Service Scope

- C SDK work outstanding
  - T-shirt size: medium to large
  - Complete securing consul with access tokens
  - Complete securing the message bus
  - Secret provider for all
- BACNet DS (in C)
  - T-shirt size: small to medium
  - “Not as far to go as we thought”
- CORs support in C SDK / C device services
  - T-shirt size: medium (due to no library support)
  - Lower priority for this WG – therefore a **stretch** goal
- Research / ADR on use of cmake options to optimize dependencies
  - Owner: Eaton
  - T-shirt size: small
  - CBOR, MQTT – conditional compiles to minimize size of DS executable; so as to not include stuff they are not using
- Update/Upgrade to use Yaml v3 library
  - T-shirt size: small
  - Tech debt, needed for code hygiene
- Implement ProtocolProperties validation mechanism
  - T-shirt size: medium
  - Crawl first – callback defined in Metadata; way for the device service to set it
  - Endpoint on DS SDK to be passed device information
  - Metadata to start calling it
  - Involves both Go and C SDKs and services
- Assist in implement ADR 0019 - Delayed start secret store tokens
  - See Security Scope
- Approve/Accept new DS
  - CoAP, GPIO, UART, Grove

# Application Services Scope

- Research how to provide your own custom function that gets called in/from the pipeline
  - T-shirt size: small
  - Via nano message vs MQTT vs ?
  - Protocol of choice to be determined
  - Compare this to just another AS
- Research CloudEvent library
  - T-shirt size: small to research
  - If still big, push onto backlog
  - If smaller, explore AS export with CloudEvents (TBD size)

# Security Scope

- Implement ADR 0019 - Delayed start secret store tokens (SPIFFE ADR)
  - T-shirt size: large; cross cutting concern that will take most of the cycle to implement
  - <https://github.com/edgexfoundry/edgex-go/issues/3773>
  - Need to get the ADR done in early Dec
    - Need GRPC clarification (requirement in C) in SDK
    - Bryon – research whether GRPC is needed for this solution (for impact on C side)
      - See <https://grpc-c.lixiangyun.top>
  - Important for enabling Kubernetes
  - Built to be opt-in (in Kamakura – only DS and AS take advantage of it)
  - Probably a common library everyone uses
  - Another C SDK dev need
- Update docs and edgex-examples
  - T-shirt size: medium; some work to determine how to do this in secure mode
  - Update SSH Tunneling for remote Device Service how to for V2
    - <https://github.com/edgexfoundry/edgex-docs/issues/515>
  - Update Swarm for remote Device Service how to for V2
    - <https://github.com/edgexfoundry/edgex-docs/issues/516>
- Doc how to use SPIFFE as example of how to do distributed services in edgex-examples
  - T-shirt size: medium – **stretch** for Kamakura



# Snap Scope

- “Under the hood work”
  - Updating the basis for core 22
  - Set env vars via Snap CLI (today we map between env names to snap lowercase names)
  - Research – separate eKuiper into separate snap (eKuiper not actively maintaining their snap)
- Add github action to build full snap when changes made to all snaps
  - T-shirt size: medium
  - Process outside verification pipeline
  - Branch protections?
  - Need it for all the snaps not just edgex-go
    - Device services (work in progress)
    - LLRP, config AS
  - Parallel workflow
  - Review by DevOps (to make sure CI pipeline isn't broken by it)
- EdgeX GUI will be made a separate snap
- Expose Consul UI on the host when EdgeX is operating in secure mode (snap)
  - T-shirt size: small – **Stretch goal for Kamakura**
  - <https://github.com/edgexfoundry/edgex-go/issues/3744>
  - Add a config option in consul; snap option

# DevOps Scope

- Pipeline Optimizations:
  - Tech Debt: Fix multiple nodes being spun up during edgeXBuildGoApp and edgeXBuildCApp
    - T-shirt size: small
  - Spike: docker image multi-arch builds with buildx (spin up less nodes, speed up builds)
    - T-shirt size: small for the research; implementation TBD (see below)
    - Use of Docker buildx so we wouldn't need to spin up ARM image to build docker arm images
    - Explored in the past; revisit approach (separate images)
    - Implementation: depending on spike docker image multi-arch builds with buildx
  - Spike: rewrite git-semver in python
    - T-shirt – medium size (impl TBD)
    - Implementation: depending on spike, rewrite git-semver in python
    - Punt on Semantic release work for now
- Assist in go lang linting/scanning
  - See security scope
- Documentation
  - T-shirt size: medium
  - Continued documentation updates (DevOps online docs)
  - Find it through README of DevOps repository

# Docs and Examples Scope

- Examples updates/improvements
  - Cloud export – more explicit for Azure, AWS, Google
    - Updated to already work with Ireland
  - Update cloud templates
  - RP4
  - More Kubernetes support/examples
- Docs
  - Device service restructuring and adding undocumented features
  - Documenting the API (like discovery)