






















































## Core Working Group Meeting Notes (13-Jun-2019)

Attendees:

Participants (20)

Q Type to filter...

 Trevor Conn (Host, me)	 
 Akram Ahmad (Dell)	 
 Alex Courouble	 
 Ankur Gargi	 
 Anthony Bonafide	 
 Brandon Forster	 
 Daniel Harms	 
 Dave Rensberger	 
 Iain Anderson	
 Jim Wang	 
 Jim White	 
 Lenny Goodell (Intel)	 
 Michael Estrin	 
 Rebekah	 
 Rodney Hess	 
 Steve	<a href="#">Unmute</a> <a href="#">More</a>
 Tingyu zeng	 
 Tom Fleming	
	 

### Old Business

- Fuji deliverables
  - Dockerfiles and Snap all upgraded to Go 1.12

- Event/Reading timestamps in nanoseconds
  - Origin timestamp on the Event is now nanoseconds
  - We discussed last week changes to the API that would be necessary which could affect versioning if we also change Created and Modified timestamps
  - Decision from Device Service WG on Monday was that we would NOT change the above timestamps. Only Origin will have the nanosecond level of fidelity.
- Rules Engine
- Plug in alternatives
  - Check in with Lenny RE: Hashicorp plugins
    - *Will bookmark this for next week*
  - Provided examples are simple, actual usage a bit more complicated
  - Choice of RPC / GRPC

## New Business

- Value Descriptor mgmt.
  - Decided in Device Services WG on Monday that the creation of Value Descriptors will move out of the Device SDK and into the core services.
  - Proposing to drive the mgmt. of value descriptors from the device profile in a similar fashion to how commands are being reworked.
  - Additional discussion points here
    - <https://github.com/edgexfoundry/edgex-go/issues/1071>
  - *If we were to entertain the notion of a value descriptor's type information being folded into the reading, what fields do we want on the reading*
    - *Type, MediaType, FloatEncoding (could be split into separate type values)*
  - *What about the Rules Engine? It only validates the name, but does not utilize the type information currently.*
  - *Sequencing of development*
    - *Contracts → Core Services → SDK*
  - *Scope of potential V2*
    - *Core-data/event*
      - */reading (for queries)*
      - */value-descriptor (not included for V2)*
    - *Contracts → reading includes type information*
    - *SDK*
      - *Do not create VDs for V2*
      - *Add type information to readings sent to core-data*
    - *Rules Engine*
      - *Mods if using V2 to not rely on Value Descriptor (audit)*
    - *Application Services*
      - *Filter by Value Descriptor equivalent functionality in V2 since Name will still be on the Reading (audit)*

- *Rules / enrichment / actuation capabilities, how does this support?*
    - *Do we put this off until after Fuji?*
  - Device Provisioning
    - Will refer to notes I took during Monday's Device Service call for this.
    - *Clarified – Provision Watchers are not used today. We are trying to achieve parity with something that was present in the Java services*
    - *Discussion of Blacklist*
      - *Use case describes rogue device on a factory floor moving from place to place. Now all Provision Watchers in the plant have to be modified to block the device.*
      - *Provision Watchers are per Device Service so all watchers need to be updated to exclude this moving device.*
    - *Implement Provision Watcher as currently described, but extend with properties supporting exclusion patterns.*
    - *From Jim's summary email*
  - Keep the Provision Watcher schema with wild cards as is (or was with Java). We may have to do some work on the way the scan lists are specified to provide cleaner/better ways of specifying scan ranges/scopes, but we can work that as we go forward. Also remove the full device profile from the Provision Watcher (just have the name)
  - Use the internal scheduler (replacing use of the support scheduler) to trigger scans
  - As Tony suggested, not have a blacklist/whitelist, but use changes to the Provision Watcher scan list to exclude/include. If you un-provision a device and don't want it to be picked back up in the next scan, include this in a Provision Watcher update before removing the device
    - *Iain and Steve Oss to provide recommendation on how the device exclusion filter should be defined.*
    - *Mostly work on Device Service SDK side, Core team to support where necessary*
  - Go.sum checkins
    - I think the opinion toward committing this is positive
    - If so, based on DevOps call, we might need to add "go mod update" to the Makefile to ensure latest deps are always pulled in – locally and build pipeline.