ECLIPSE AH – DITTO INTEGRATION

ECLIPSE ARROWHEAD BI-WEEKLY MEETING
JANUARY 18, 2022

SRT, EISLab @ LTU

Presented by
Abdullah Aziz
Olov Schelén

LULEÅ UNIVERSITY OF TECHNOLOGY

Industrial device virtualization by digital twin (DT) as a proxy (DTaaP)

- Energy efficiency
  - Devices can sleep (duty cycle) while DT services are available
- Availability & persistence
  - DT provides storage, physical device can seamlessly be replaced
- Contention control
  - Actuation of device may be offered to many, under strict control
- Security
  - Devices not capable of strong security talks only to their DT, while the DT registers services and provides strong security

LULEÅ UNIVERSITY OF TECHNOLOGY

Status & coming steps

Status at LTU

- A proof of concept (PoC) based on Eclipse Ditto has been developed
- Paper to be published at International Conference on Industrial Technology and Management (ICITM), Oxford, UK, Feb 2022.

Challenges

- The PoC is not yet integrated in Eclipse Arrowhead
- The PoC delivers just partly on objectives (prev page)

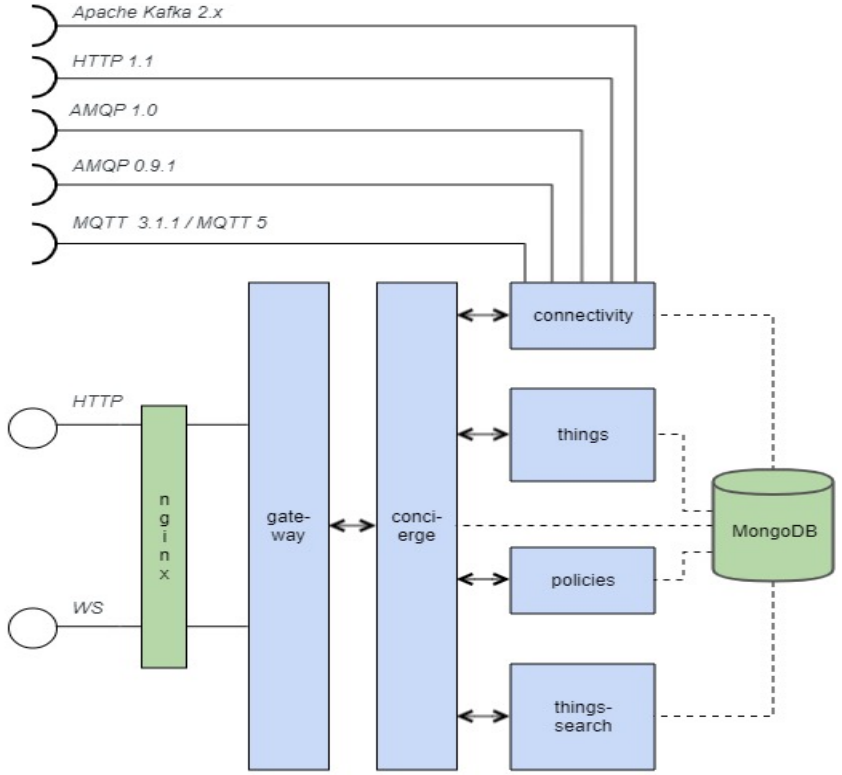
Proposal

- Integrate Ditto in Arrowhead (two options to be presented)
- Evolve the DT concept towards the objectives

LULEÅ UNIVERSITY OF TECHNOLOGY

Eclipse Ditto Architecture

- Ditto consists of multiple “microservices” as shown in the component view.
- A “microservice” in Ditto :
  - has its own data store
  - has an API
  - can be accessed by other services only via the defined API



Eclipse Ditto Architecture

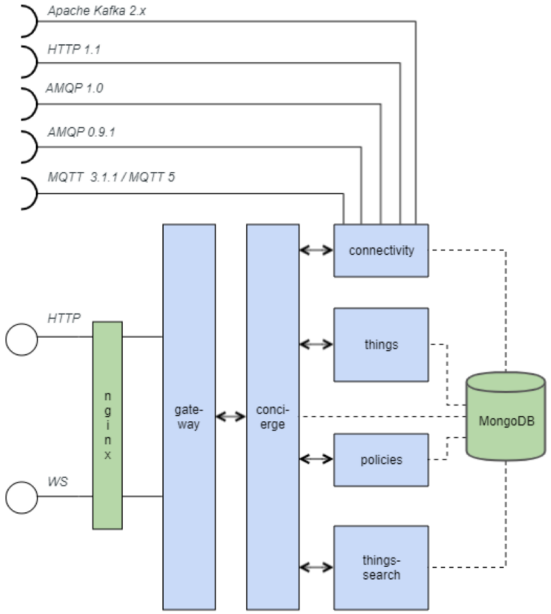
```
{
  "policy": {
    "policyId": "my.namespace:policy-a",
    "entries": {
      "owner" : {
        "subjects": {
          "nginx:ditto": {
            "type": "nginx basic auth user"
          }
        },
        "resources": {
          "thing:/": { "grant": ["READ","WRITE"], "revoke": [] },
          "policy:/": { "grant": ["READ","WRITE"], "revoke": [] },
          "message:/": { "grant": ["READ","WRITE"], "revoke": [] },
        }
      },
      "observer" : {
        "subjects": {
          "nginx:observer-client": {
            "type": "technical client"
          },
          "nginx:some-users": {
            "type": "a group of users"
          }
        },
        "resources": {
          "thing:/features/featureX": { "grant": ["READ"], "revoke": [] },
          "thing:/features/featureY": { "grant": ["READ"], "revoke": [] }
        }
      },
      "private": {
        "subjects": {
          "nginx:some-users": {
            "type": "a group of users"
          },
          "resources": {
            "thing:/features/featureY/properties/location/city": {
              "grant": [], "revoke": ["READ"]
            }
          }
        }
      }
    }
  }
}
```

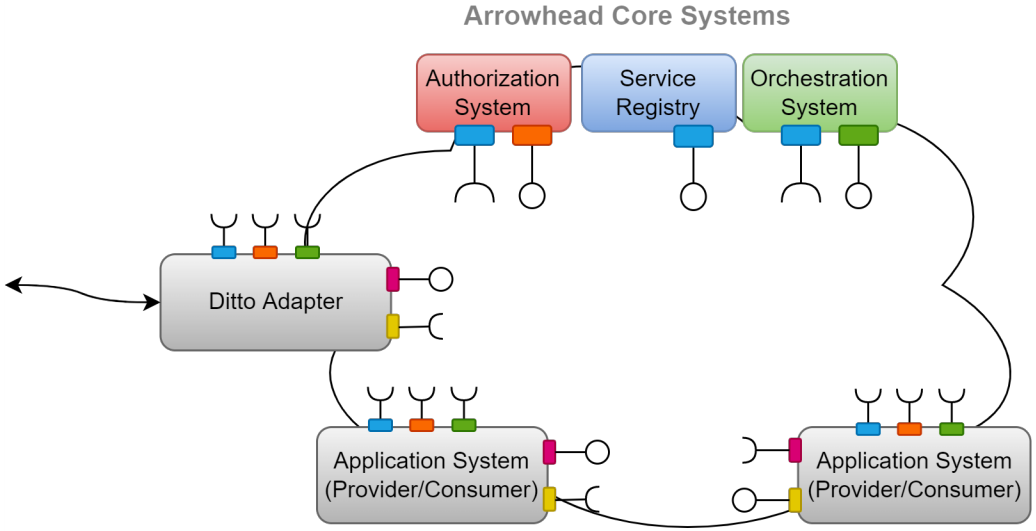
Policy

```
{
  "thingId": "my.namespace:thing-123",
  "policyId": "my.namespace:policy-a",
  "attributes": {"key": "value"},
  "features": {
      "featureX": {
          "properties": {
              "key": "value"
          }
      },
      "featureY": {
          "properties": {
              "location": {
                  "city": "Berlin",
                  "country": "Germany"
              }
          }
      },
      "featureZ": {
          "properties": {
              "key": "value"
          }
      }
  }
}
```

Thing

Option # 1

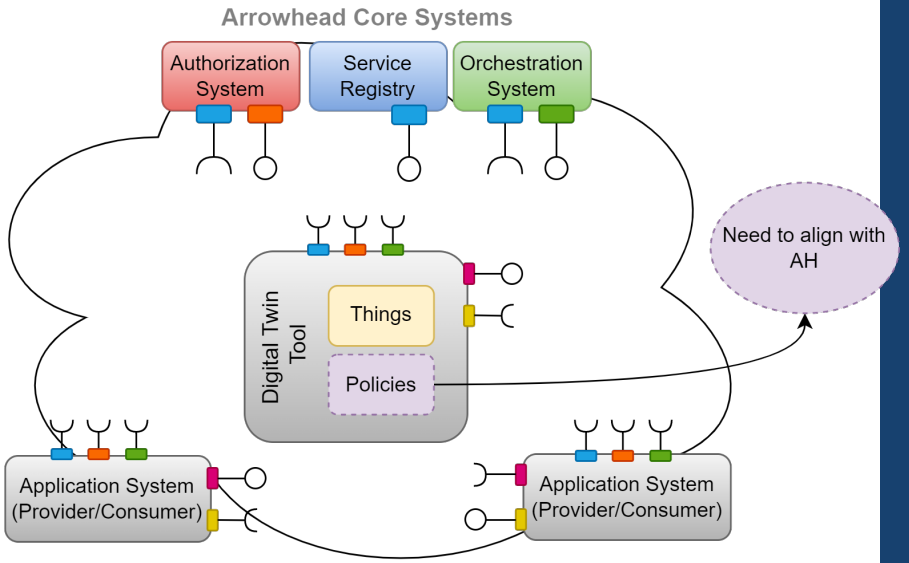




- Pros
  - Seems easy and quick integration.
  - Good for prototype
- Cons
  - Adapter need to expose all the services within Arrowhead against the Thing and Policies model.
  - Policies can be tricky, because of access rights based on users.
  - Seems to get more complex as DTs increase.

Option # 2

- Pros
  - AH compliant digital twin tool with
    - AH security features
    - AH connectivity mechanism
    - AH interoperability
    - All AH supporting tools can be integrated
  - Seems more stable and scalable.
- Cons
  - Needs to introduce and develop Ditto like Things components & align the Policies with AH.
  - Seems lengthy work.



Resources from LTU

- Abdullah Aziz (PhD student)
- Nicklas Nyström (software developer)
- Ulf Bodin and Olov Schelén (supervisors)

LULEÅ UNIVERSITY OF TECHNOLOGY