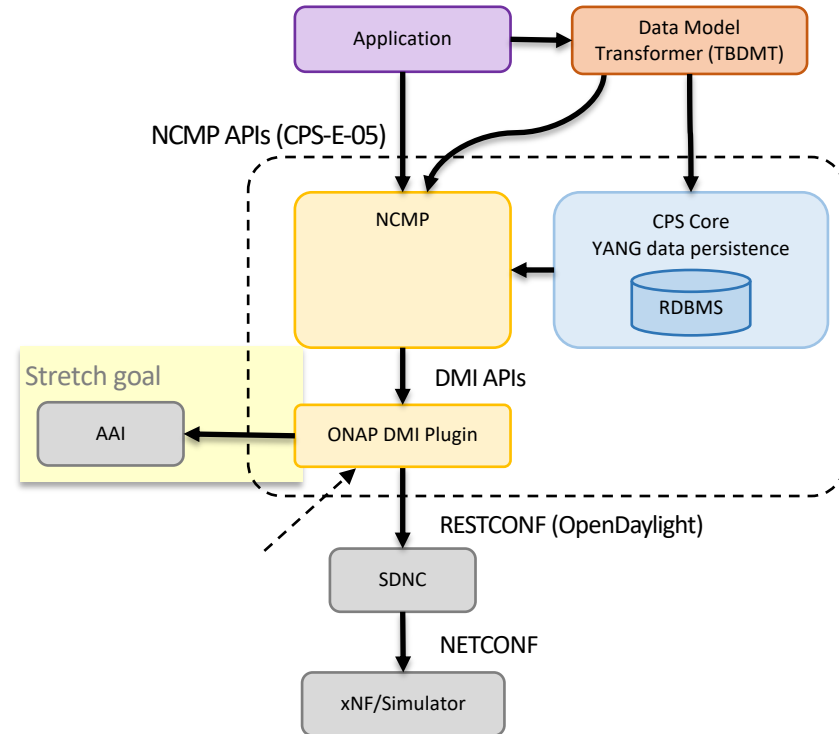# CPS Development Updates for Istanbul

Network CM Proxy & Temporal
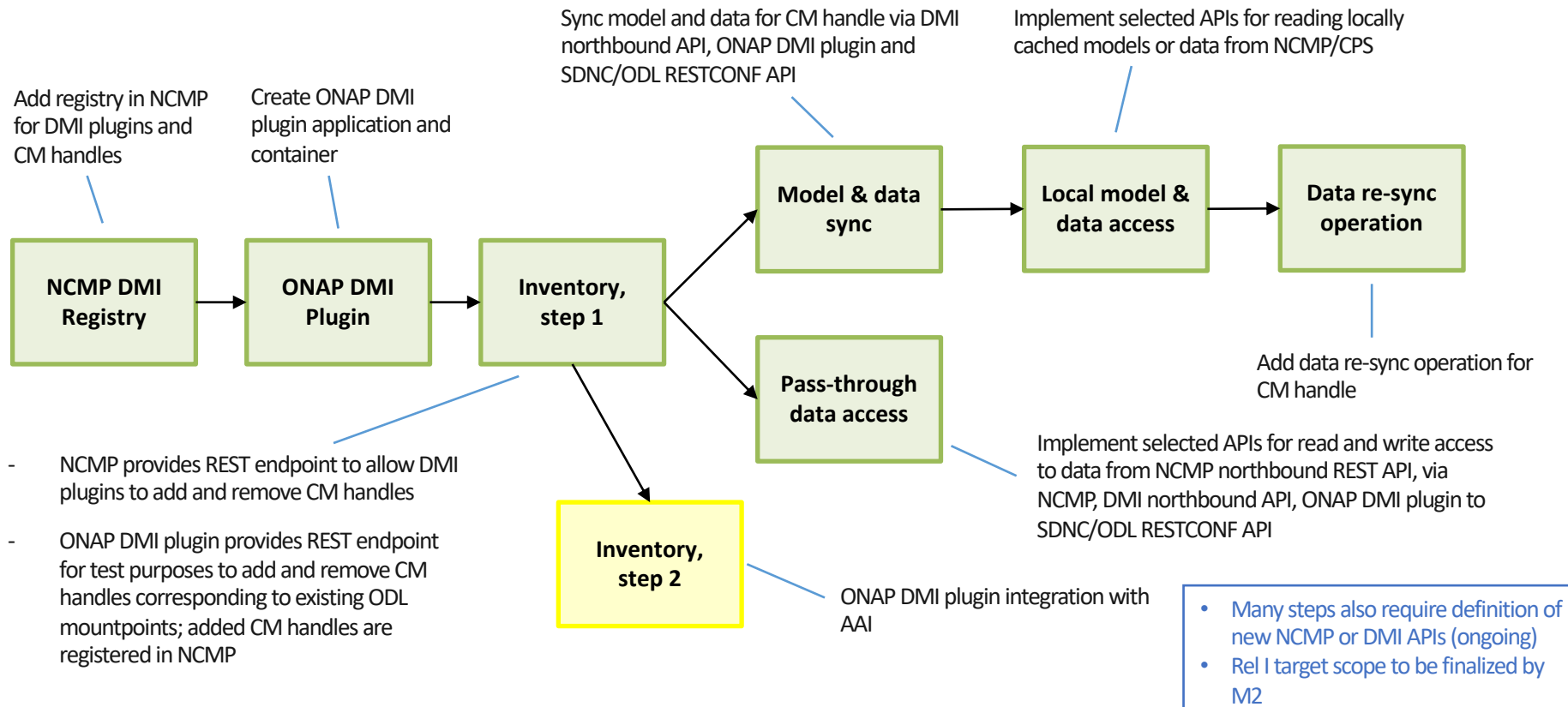
2021, June 8th

Agenda

- Network CM Proxy (NCMP)

- Temporal Service
    - Current and historical views on operational data
    - Architecture
    - Querying
    - REST API

- Questions

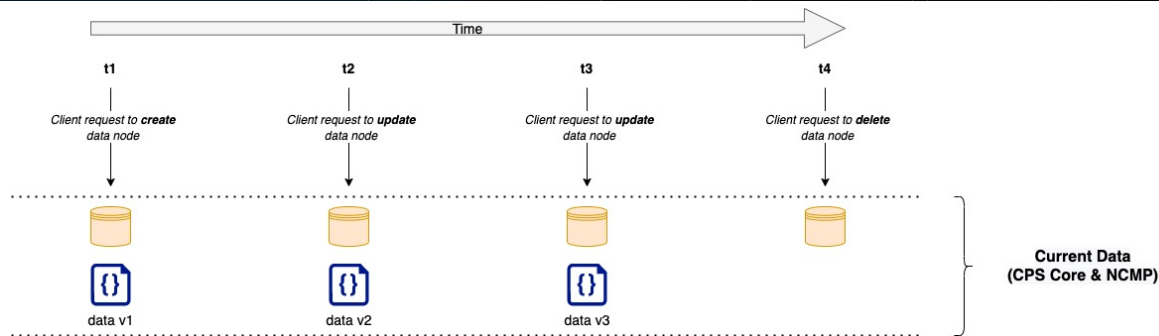# Network CM Proxy (NCMP) Overview

- NCMP provides access to network configuration data
    - Will also support local cache using CPS Core library
    - Manages data based on original xNF YANG modules
- AAI is responsible for ONAP inventory
    - Inventory items supporting configuration will have corresponding *CM handles* in NCMP
    - May bypass AAI for testing
- NCMP accesses all network Data-Model-Inventory (*DMI*) information via plugin
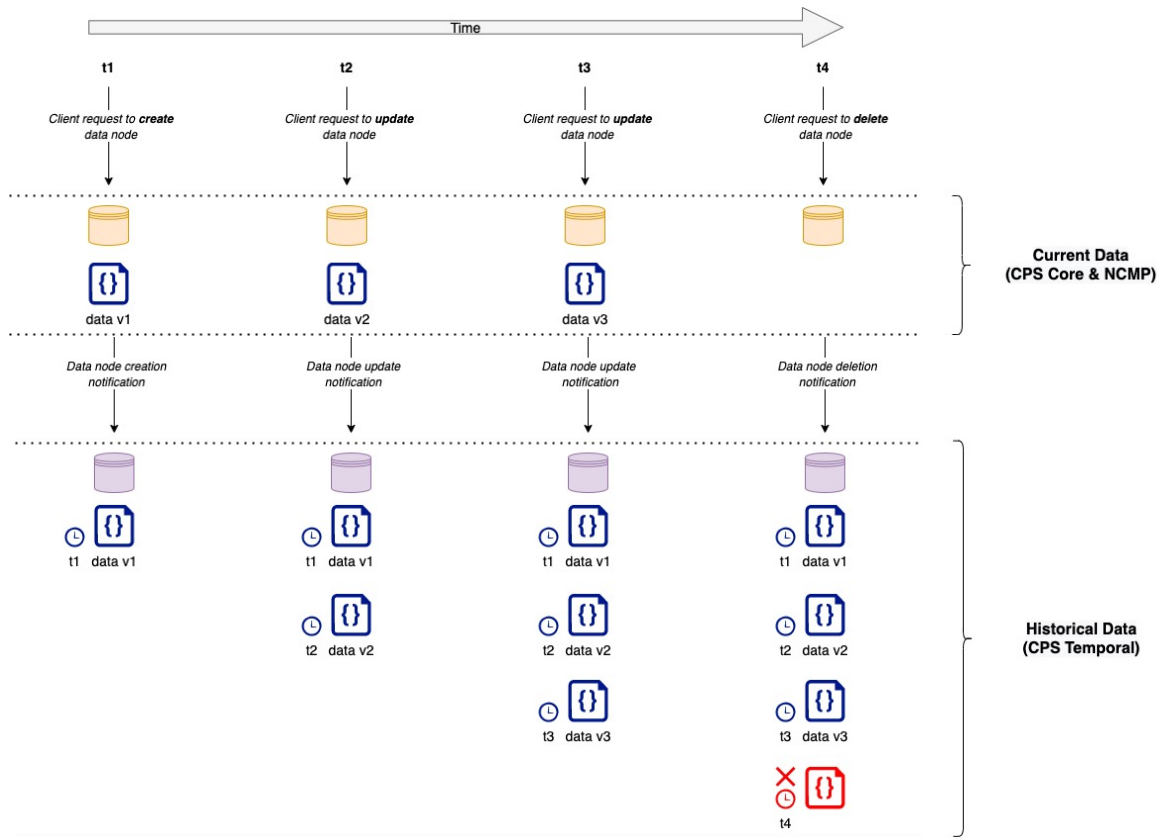   - A standard ONAP plugin is being developed
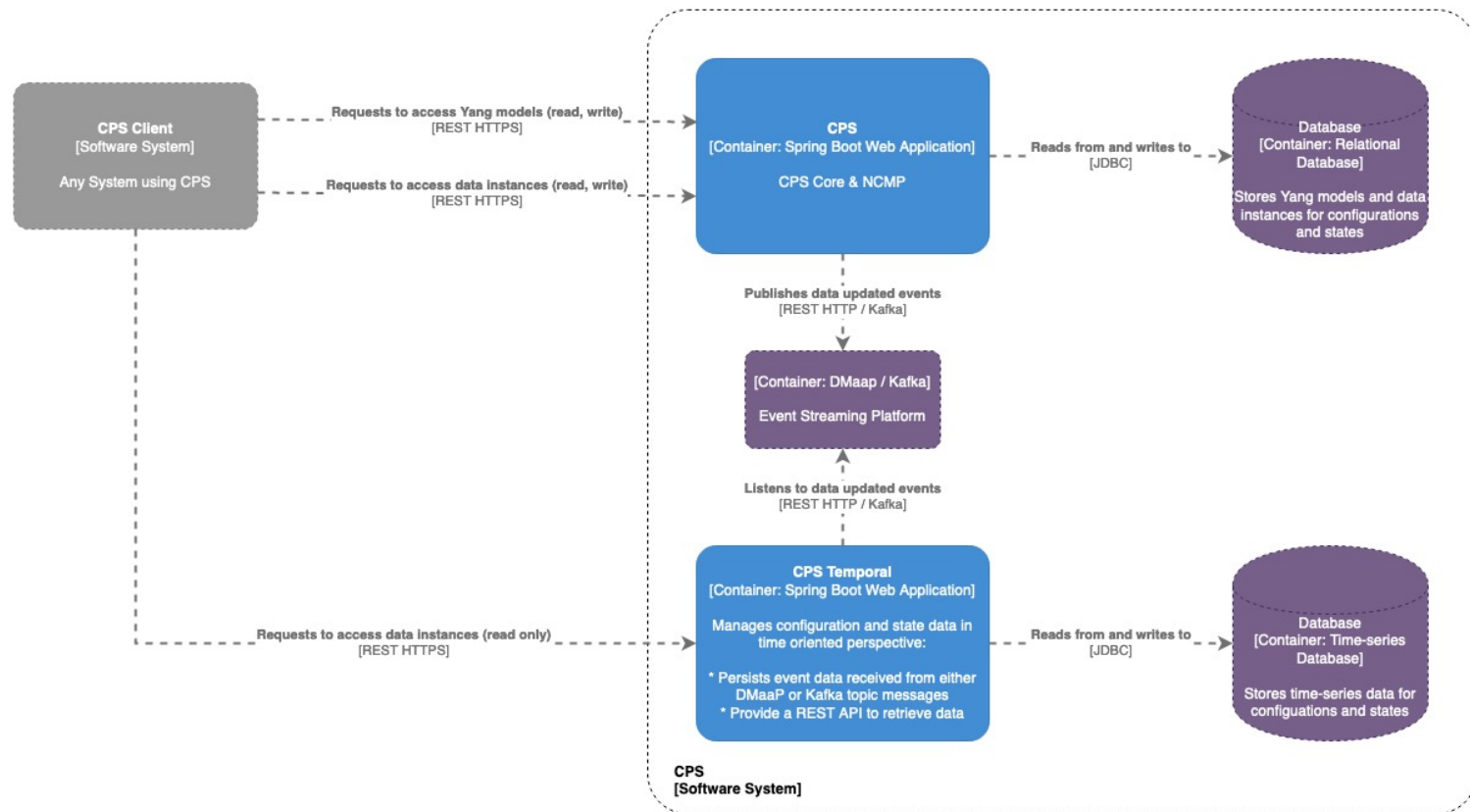
NCMP Roadmap – Rel I (preliminary)

Add registry in NCMP for DMI plugins and CM handles

Create ONAP DMI plugin application and container

Sync model and data for CM handle via DMI northbound API, ONAP DMI plugin and SDNC/ODL RESTCONF API

Implement selected APIs for reading locally cached models or data from NCMP/CPS

NCMP DMI Registry → **ONAP DMI Plugin** → **Inventory, step 1** → **Model & data sync** → **Local model & data access** → **Data re-sync operation**

→ **Pass-through data access**

Inventory, step 2

Add data re-sync operation for CM handle

- NCMP provides REST endpoint to allow DMI plugins to add and remove CM handles

- ONAP DMI plugin provides REST endpoint for test purposes to add and remove CM handles corresponding to existing ODL mountpoints; added CM handles are registered in NCMP

Implement selected APIs for read and write access to data from NCMP northbound REST API, via NCMP, DMI northbound API, ONAP DMI plugin to SDNC/ODL RESTCONF API

ONAP DMI plugin integration with AAI

- Many steps also require definition of new NCMP or DMI APIs (ongoing)
- Rel I target scope to be finalized by M2

Temporal for Historical Data

Time

t1 — Client request to **create** data node — data v1

t2 — Client request to **update** data node — data v2

t3 — Client request to **update** data node — data v3

t4 — Client request to **delete** data node

Current Data (CPS Core & NCMP)

Temporal for Historical Data

Core &Temporal Architecture

Data Notification Filters

- Multiple levels for notification publication filtering from Core to Temporal:
  - System global (turn all notifications off)
  - Dataspace
  - Anchor
  - *Yang specific path level (?)* [*]

[*] Not included Istanbul scope

- Date and time
  - Data after specified timestamp
  - *Last X records* [*]
- Data types
  - Dataspace & anchor
  - Dataspace & schema set
  - *Dataspace & multiple anchors* [*]

- Data payload
  - Simple condition on json content value
- Pagination and sorting

[*] Not included Istanbul scope

Temporal REST API (by dataspace & anchor)

GET `/v1/dataspaces/{dataspace-name}/anchors/{anchor-name}/history` Get anchor data by name

dataspace-name * required

string

(path)

Dataspace Name

```
dataspace-name - Dataspace Name
```

anchor-name * required

string

(path)

Anchor Name

```
anchor-name - Anchor Name
```

after

string

(query)

Fetch data after
Format - 'yyyy-MM-ddTHH:mm:ss.SSSZ'

```
2021-03-21T00:00:00.000000-0:00
```

simplePayloadFilter

string

(query)

Payload filter

```
simplePayloadFilter - Payload filter
```

pointInTime

string

(query)

Consider data modified before
Format - 'yyyy-MM-ddTHH:mm:ss.SSSZ'

```
2021-03-21T00:00:00.000000-0:00
```

pageNumber

integer

(query)

Page number

Default value : 0

```
0
```

pageLimit

integer

(query)

The numbers of items to return

Default value : 1000

```
1000
```

sort

string

(query)

Sort by timestamp in 'asc' or 'desc' order. Supported values:
timestamp:desc
timestamp:asc

Default value : timestamp:desc

```
timestamp:desc
```

Temporal REST API (by dataspace & schema set)

LFN NETWORKING
LFN Developer & Testing Forum

GET `/v1/dataspaces/{dataspace-name}/anchors/history`   Get anchors data based on filter criteria

dataspace-name * required

Dataspace Name

string

```
dataspace-name - Dataspace Name
```

(path)

schemaset-name * required

Schema-set name

string

```
schemaset-name - Schema-set name
```

(query)

after

Fetch data after
Format - 'yyyy-MM-ddTHH:mm:ss.SSSZ'

string

(query)

```
2021-03-21T00:00:00.000000-0:00
```

simplePayloadFilter

Payload filter

string

(query)

```
simplePayloadFilter - Payload filter
```

pointInTime

string

(query)

Consider data modified before
Format - 'yyyy-MM-ddTHH:mm:ss.SSSZ'

```
2021-03-21T00:00:00.000000-0:00
```

pageNumber

integer

(query)

Page number

Default value : 0

```
0
```

pageLimit

integer

(query)

The numbers of items to return

Default value : 1000

```
1000
```

sort

string

(query)

Sort by timestamp in 'asc' or 'desc' order. Supported values:
timestamp:desc
timestamp:asc

Default value : timestamp:desc

```
timestamp:desc
```

Temporal REST API (response sample)

```
{
  "nextRecordsLink": "/v1/dataspace/my-dataspace/anchors/history?pageLimit=20&pageNumber=2",
  "previousRecordsLink": "/v1/dataspace/my-dataspace/anchors/history?pageLimit=20&pageNumber=0",
  "records": [
    {
      "timestamp": "2021-03-21T00:00:00.000000-0:00",
      "dataspace": "my-dataspace",
      "schemaSet": "my-schema-set",
      "anchor": "my-anchor",
      "data": {
        "status": "UP"
      }
    }
  ]
}
```

Questions & More …

Visit [Configuration Persistence Service Project](https://wiki.onap.org/display/DW/Configuration+Persistence+Service+Project) wiki page

https://wiki.onap.org/display/DW/Configuration+Persistence+Service+Project