Using OPC UA PA-DIM for field devices

Dr. Mika Karaila, Valmet mika.karaila@valmet.com

Agenda: Using PA-DIM for field devices

Prototyping & exploring

- FDI is based on OPC UA. Field device information model is based on PA-DIM.
  Basic concept of the FDI is first explained as an overview and give some background information.
  PA-DIM will provide standard information model for different field bus protocols with large scale of devices.
- UaExpert can be used to demonstrate & explore information model. Presentation will cover information model and short presentation of proto-type that illustrates possibilities of the model.
- FDI will be used to parametrize field devices and it can be also used for the condition monitoring.

- Related presentation by Frank Fengler, ABB:
  https://www.automaatioseura.fi/site/assets/files/2060/11-opc_day_helsinki_2019_opc_ua_for_process_automation_pa-dim_frank_fengler.pdf

Disclaimer: Examples are proto-types that are used to explore PA-DIM and it´s use & maturity.

OPC Foundation and information model: PA-DIM

Short summary generated by Enterprise AI

PA-DIM stands for Process Automation Device Information Model. It is a joint specification developed by several industry organizations to define a common and interoperable information model for process automation devices. PA-DIM enables protocol and vendor independent data exchange between intelligent field instruments and host systems, such as control or asset management systems. PA-DIM also supports control-in-the-field strategies, which allow field devices to communicate directly with each other and maintain safe operations during a host system failure. PA-DIM is based on OPC UA technology, which provides a standard platform for data communication and integration. PA-DIM covers various use cases, such as telemetric and asset management, and defines the parameters, names and semantic identifiers for different device types. PA-DIM aims to facilitate information-driven systems that can offer increased connectivity, real-time data and advanced analytics for process automation[1][2][3].

Learn more: 1. opcfoundation.org 2. fieldcommgroup.org 3. isa.org

FDI & OPC UA PA-DIM: Designed for understanding semantic identification

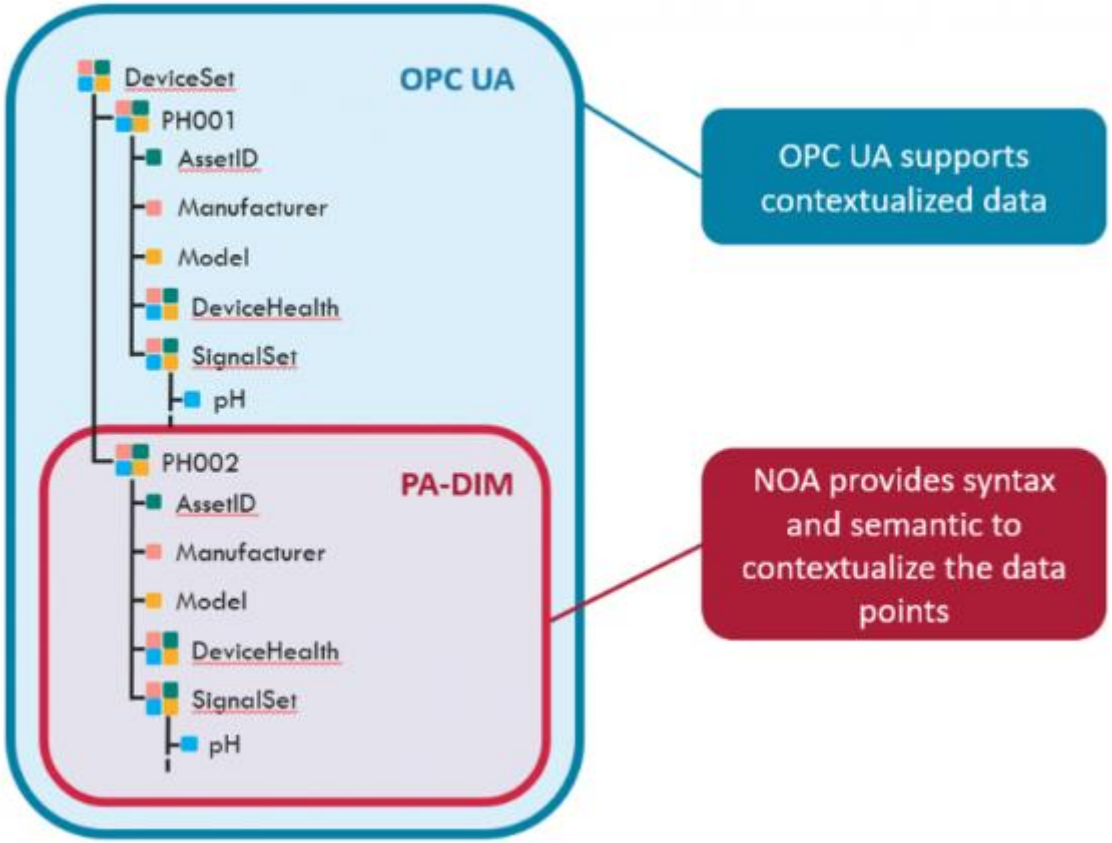
Map protocol specific parameters to standardized terms for parameters.

- **PA-DIM** is a manufacturer independent Information Model with a structured hierarchy for standardized data access for devices.
- Reuse **DI-model** (nodeset): A Device (Asset) has Set of Signals (Functions) e.g. Process Variables, which are based on OPC UA Part 100 Devices (**DI-model**). **PA-DIM** re-use IVendor-, ITag- Nameplate and IDeviceHealth- Interfaces from **DI**.
- In addition, **PA-DIM** defines IAdministration- and ISignalSet- Interface, so that it is reusable by other Information Models. All parameter are defined **in IEC 61987 CDD** with Semantic ID (**IRDI-model**), which ensures that parameter can be easily re-used between different software tools and protocols (by reference HasDictionaryEntry).
- Another main part is the SignalVariableType, which is an extension of OPC UA Part 8: Data Access an describes analog and discrete variables including its simulation. This variable can be used at any object also from other Information Models. The PA-DIM **SignalType** Object uses the SignalVariableType and adds the **SignalTag** and specific methods like **ZeroPointAdjustment** or **AutoAdjustPositioner**.

Valmet

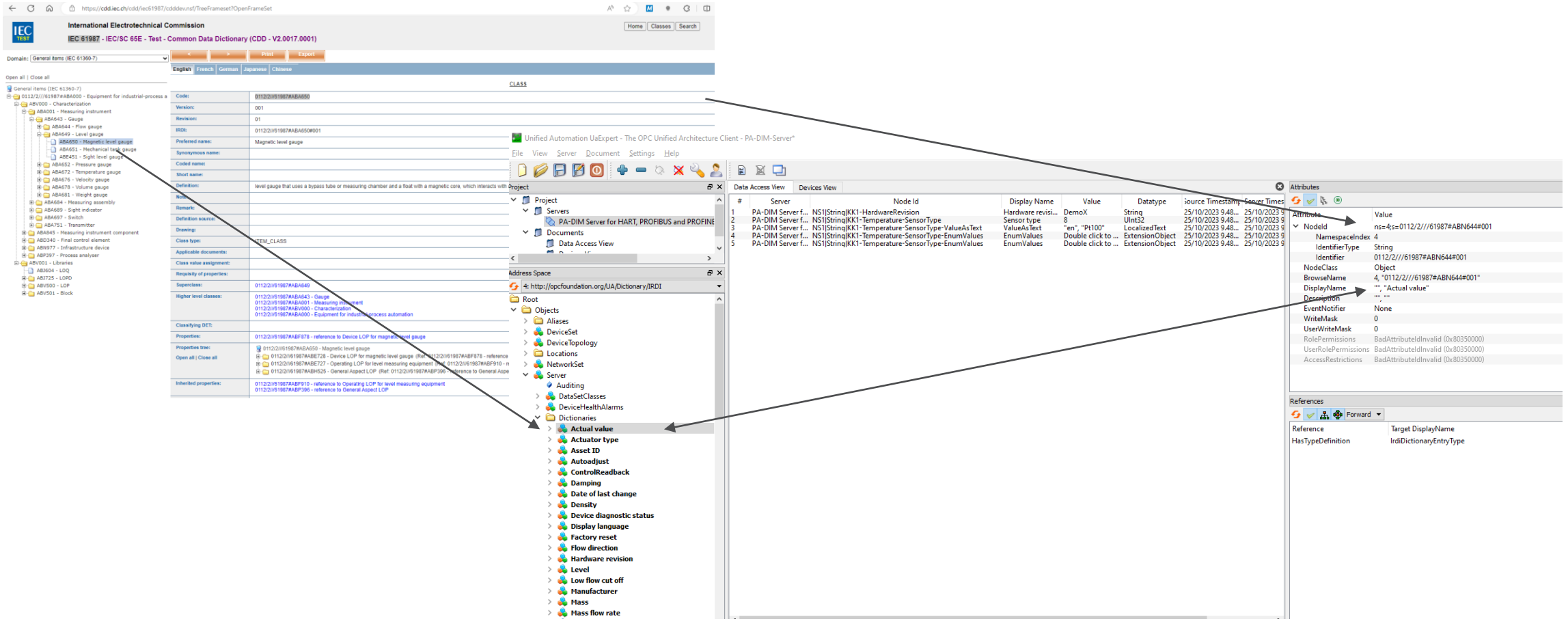
PA-DIM Simplified Data Access

- Manufacturer Independent
- Sorted
- Structured Hierarchy
- Prioritized
- Mapping can be understood by standard interface tools
- NOA = NAMUR Open Architecture
- IEC 61987 CDD = Common Data Dictionary



IEC 61987 CDD = Common Data Dictionary

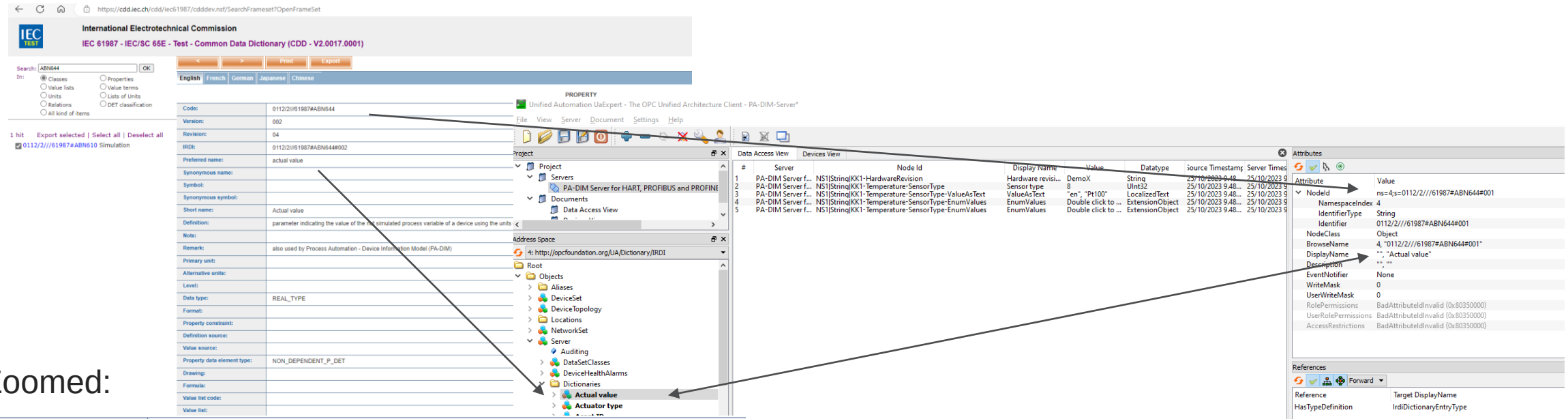
Code is found from the Server Dictionary, IRDI nodeset



https://cdd.iec.ch/cdd/iec61987/cdddev.nsf/TreeFrameset?OpenFrameSet

IEC 61987 CDD = Common Data Dictionary (detail view)

Code is found from Server Dictionary, IRDI nodeset



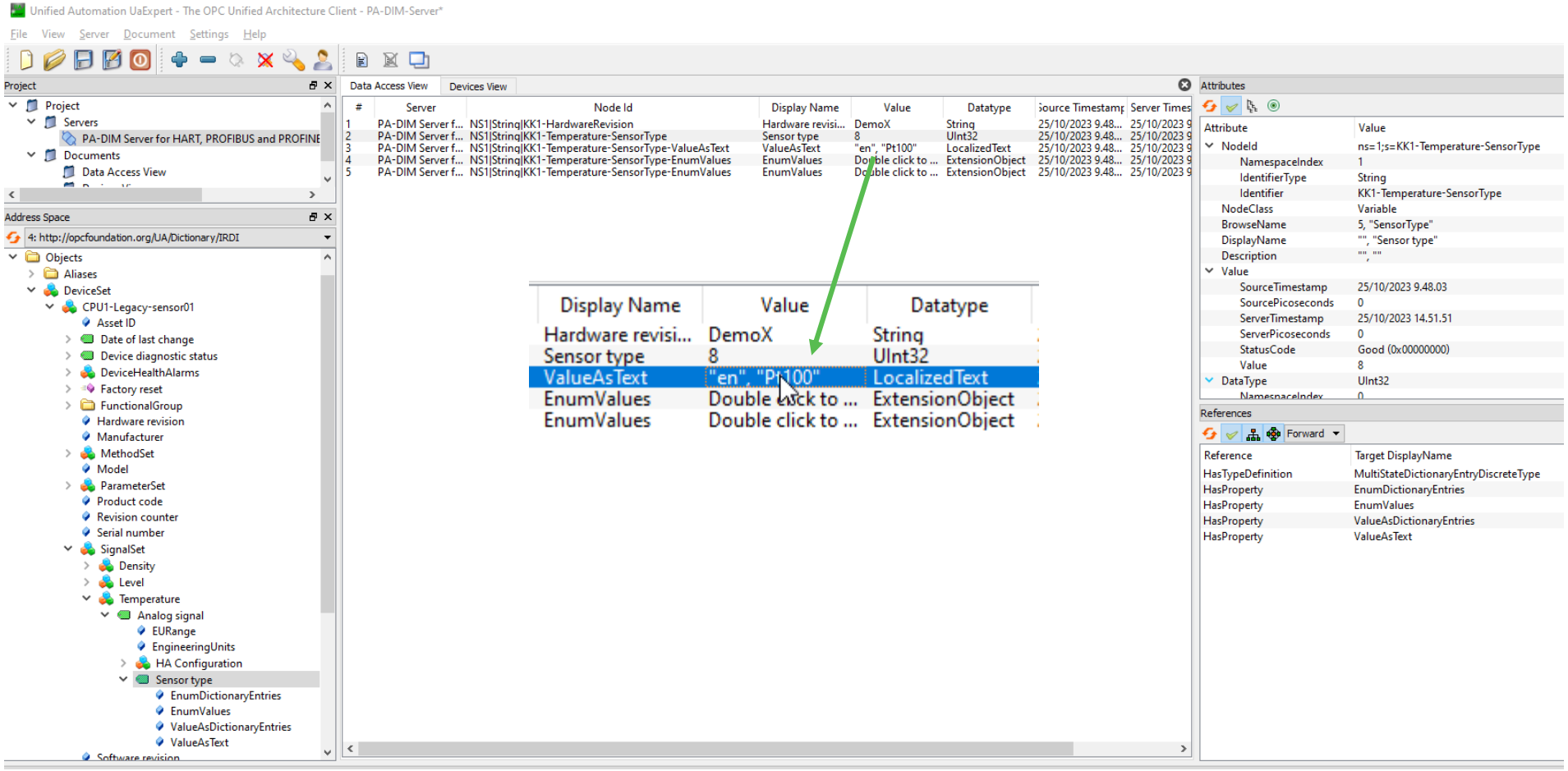
Zoomed:

Code:	0112/2///61987#ABN644
Version:	002
Revision:	04
IRDI:	0112/2///61987#ABN644#002
Preferred name:	actual value
Synonymous name:	
Symbol:	
Synonymous symbol:	
Short name:	Actual value
Definition:	parameter indicating the value of the not simulated process variable of a device using the units of measure of the process variable
Note:	
Remark:	also used by Process Automation - Device Information Model (PA-DIM)

https://cdd.iec.ch/cdd/iec61987/cdddev.nsf/TreeFrameset?OpenFrameSet

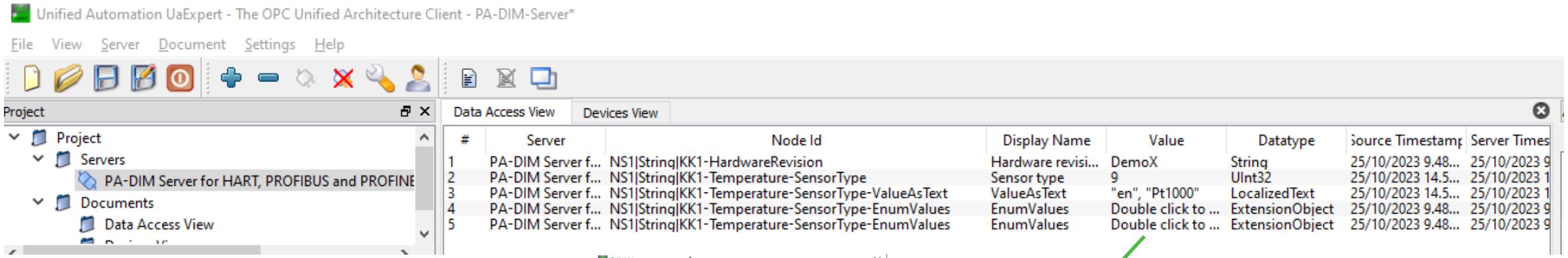
Example: Sensor type

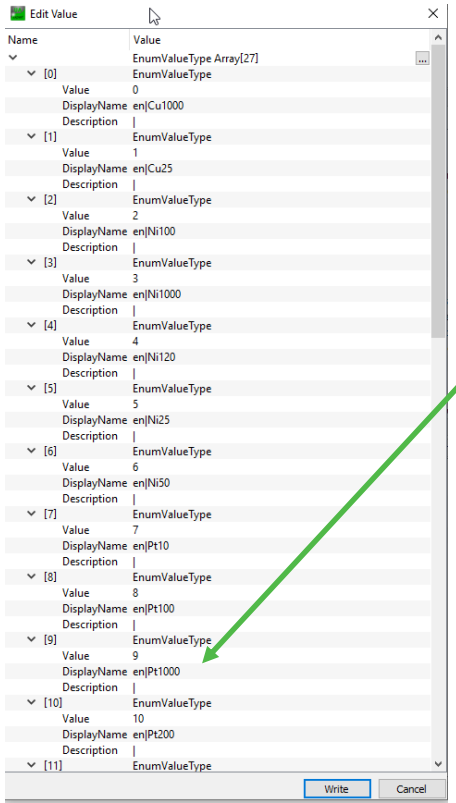
Device uses UInt32, PA-DIM can show value as text



Example: Sensor type

Change value from 8 to 9 will automatically update ValueAsText = Pt1000

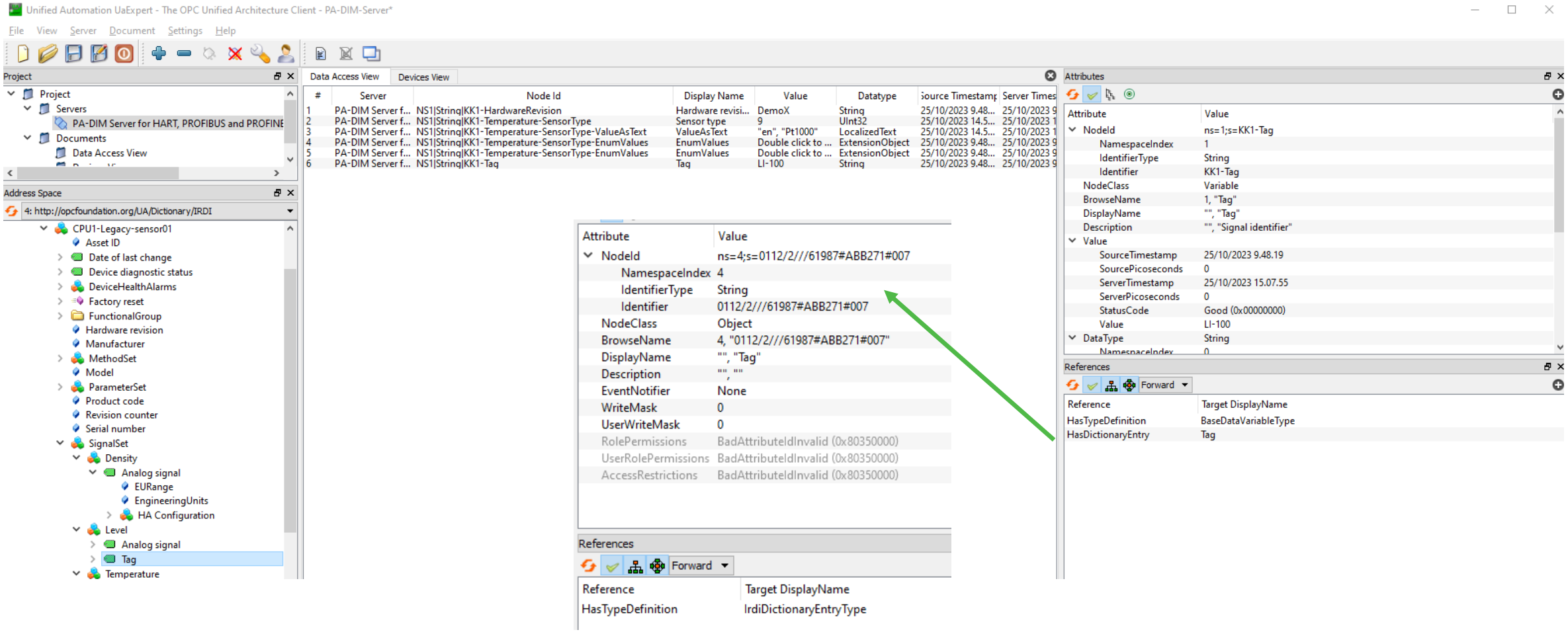




Enumeration values

Example: Level measurement, Tag: LI-100

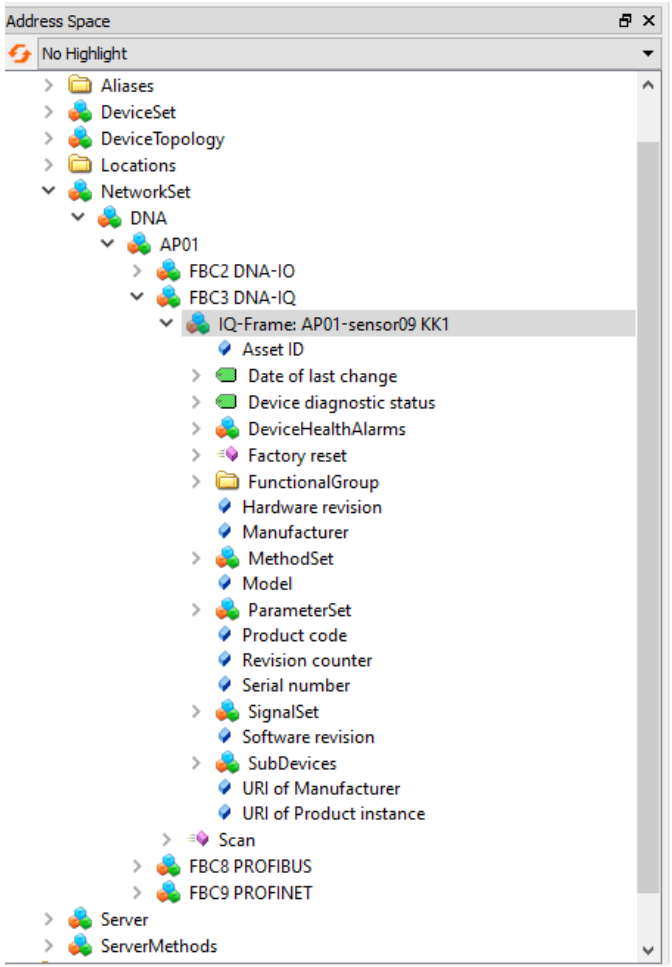
Navigate from the reference to Dictionary

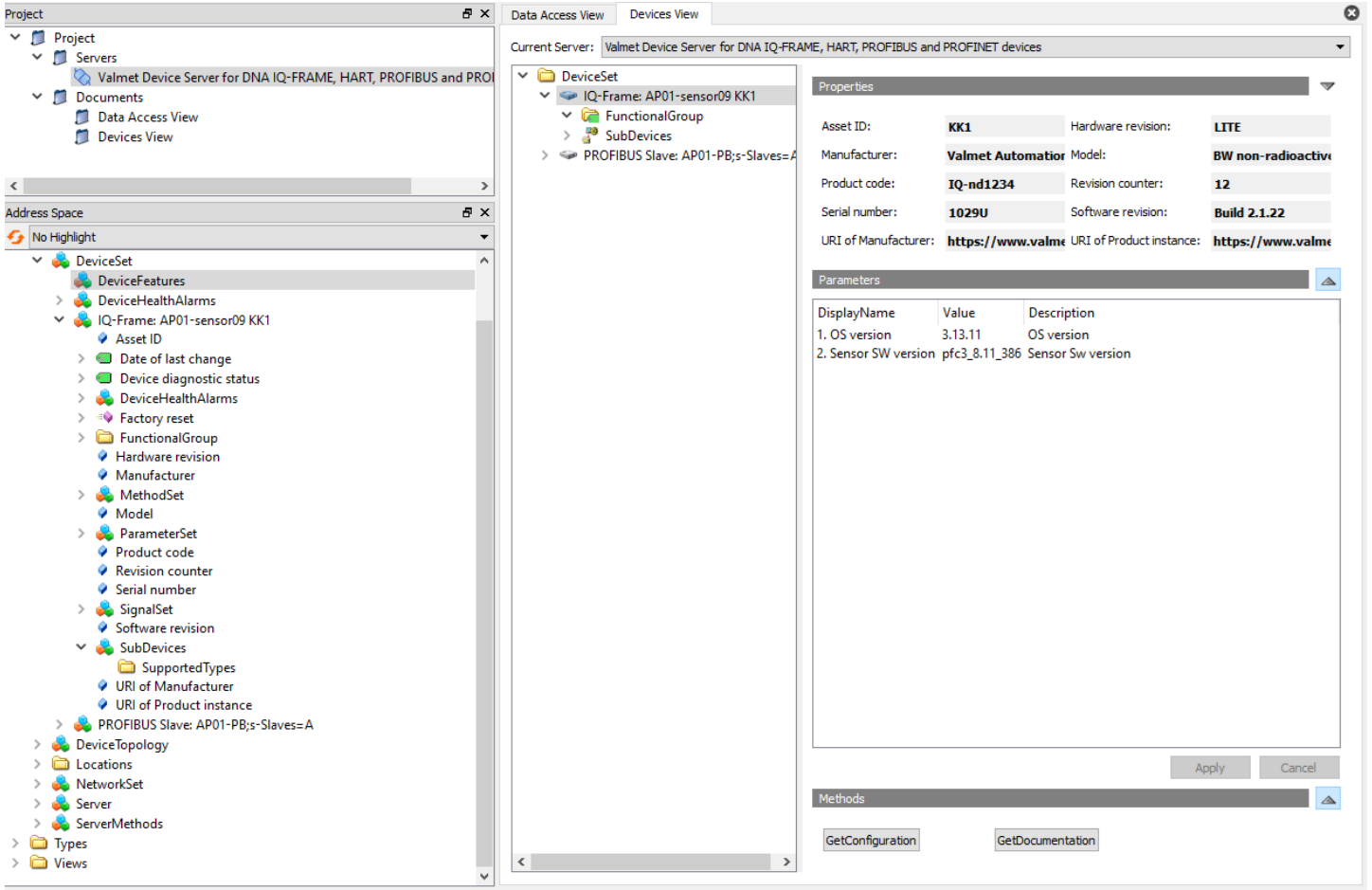


Valmet PA-DIM Demo server

Valmet Device server (OPCUA with PA-DIM)

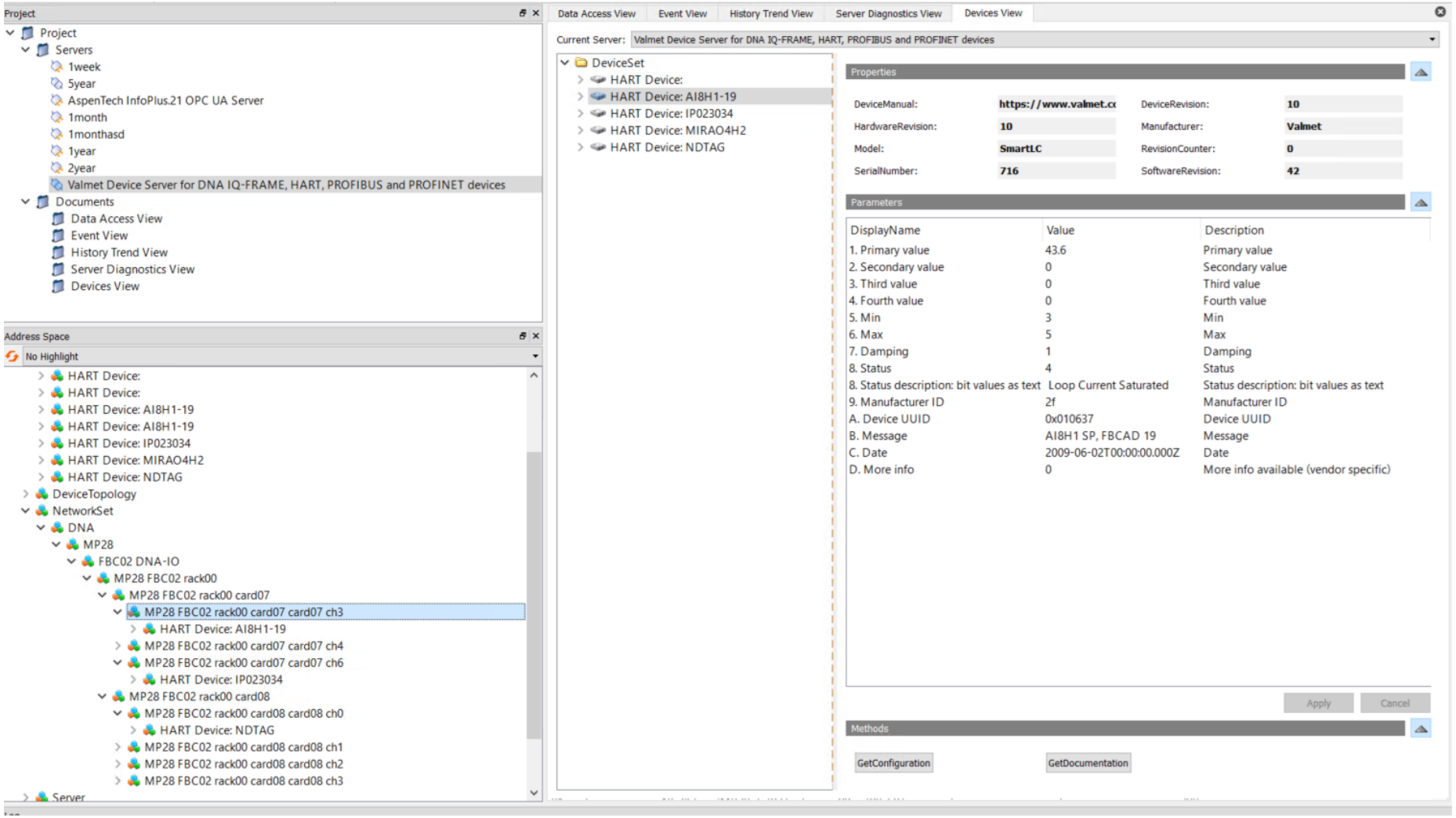
DNA network topology & Device information





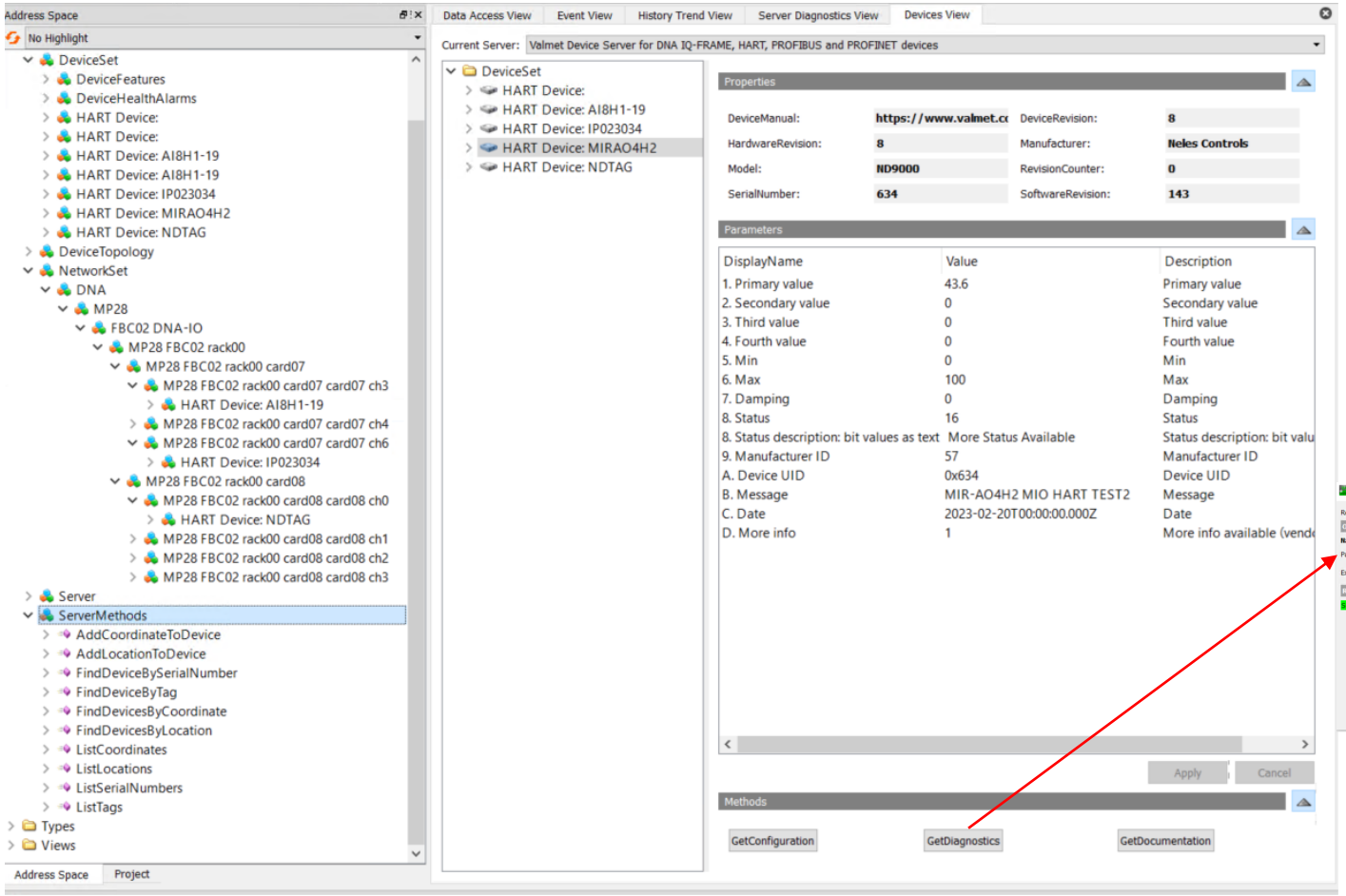
Valmet Device server: Example HART device

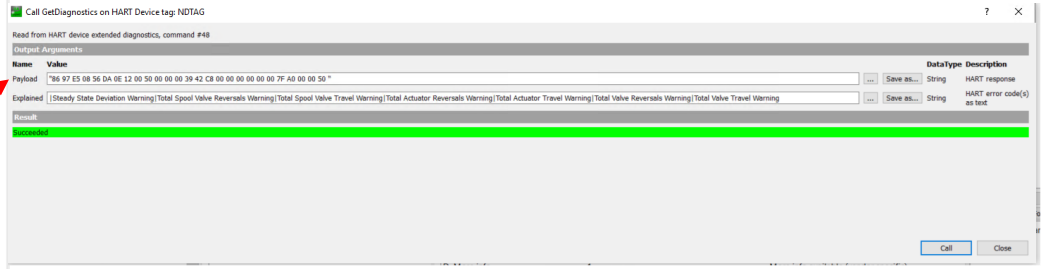
Methods: own legacy in this case, aggregates PA-DIM data from the DNA-OPCUA-Server



Valmet Device server: Extended functionality

Uses HART command 48 to get vendor specific status information



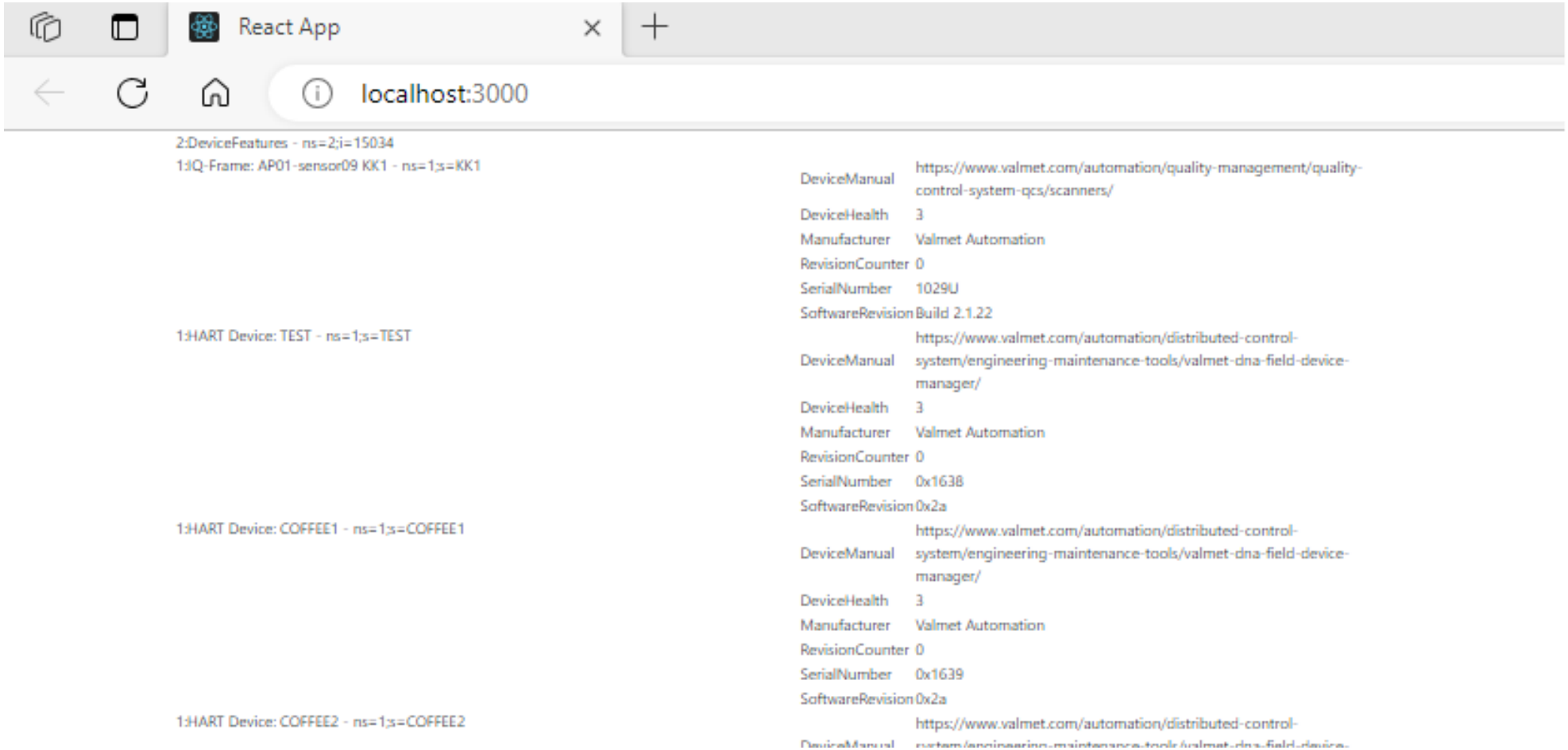


Uses device description to find out text for the error(s)

Short live demo before final slides

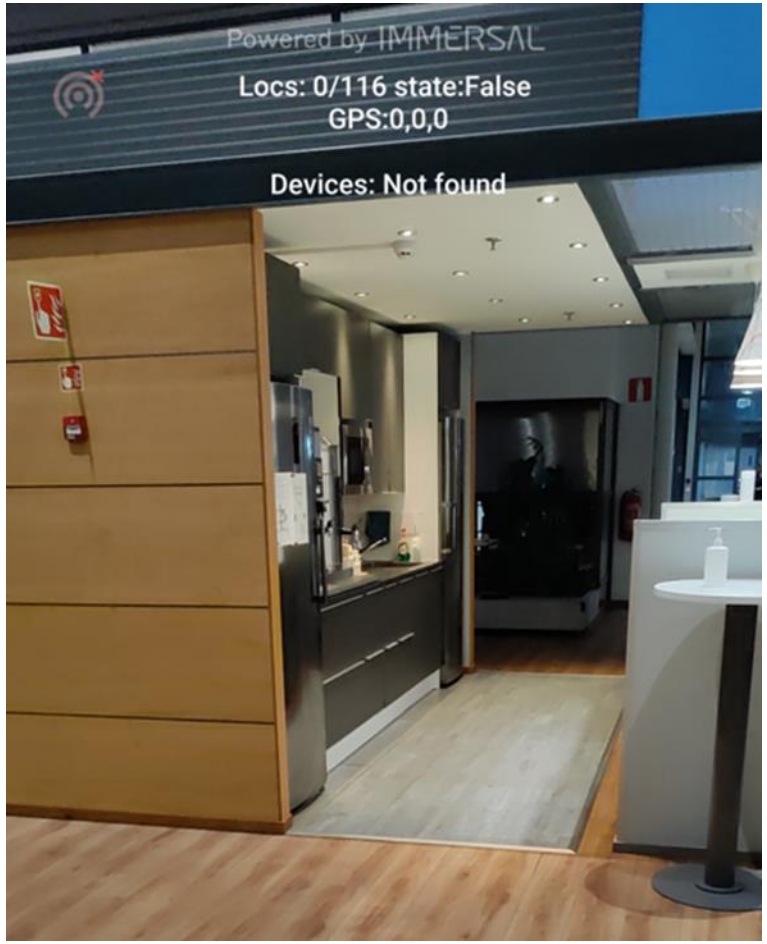
Condition monitoring: Web server (implemented as React application)

Disclaimer: proto-type – not production ready (proxy component can be licensed from Sterfive)



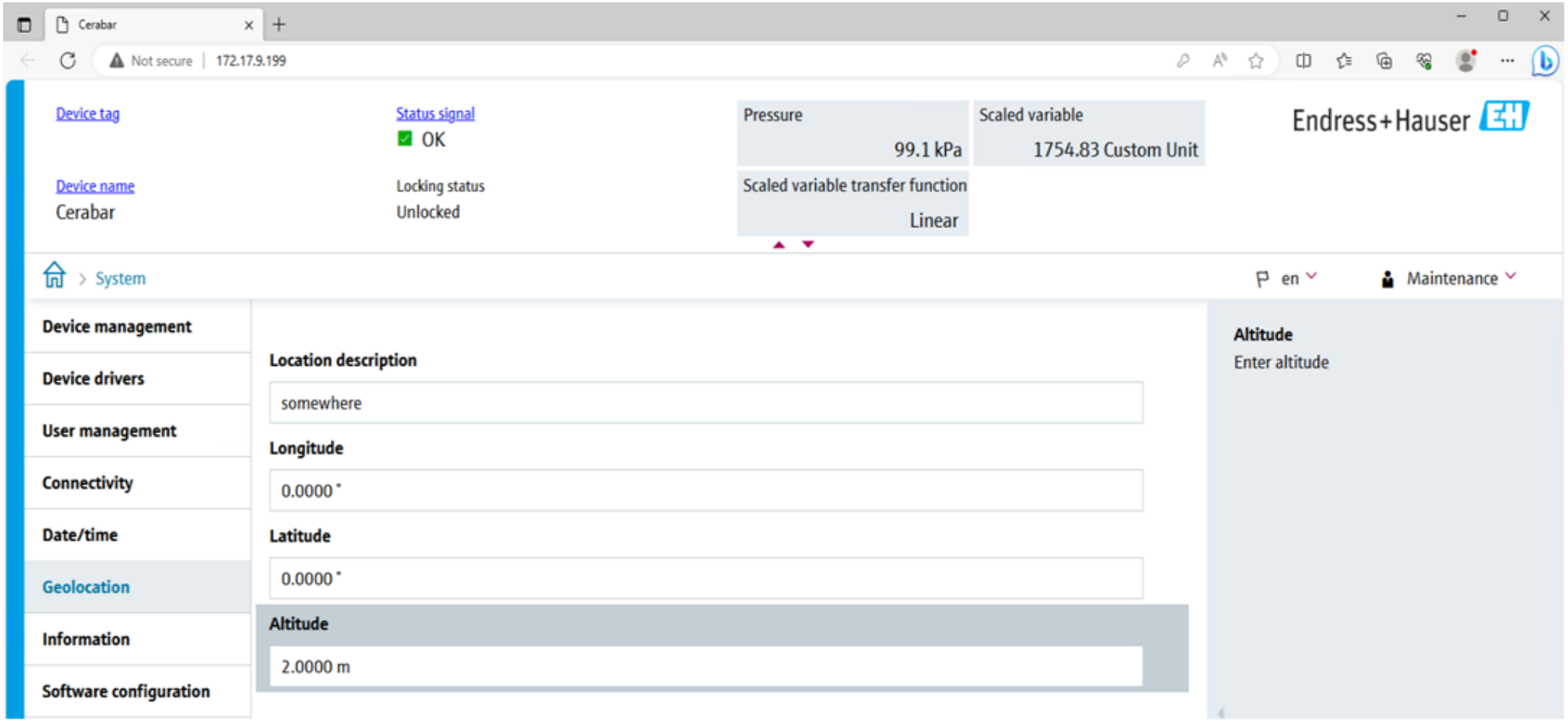
Immersal: Visual Positioning System (VPS) with PA-DIM Server

Location aware mobile phone AR application that will get live-data from PA-DIM server



Extra information for use case

E&H Ethernet APL device contains fields for the Geolocation



Questions & comments

- Special thanks to Etienne Rossignon etienne.rossignon@sterfive.com
  - Implementing node-opcua and support for extending OPC UA into the real-life use
  - GITHUB: https://github.com/node-opcua/node-opcua
  - Enterprise support for development recommended
  - React components need separate license, not available as open-source code

- Valmet Device server build with TypeScript
  - Proto-type without Valmet related bindings/integration:
    Source code could come available with MIT License (not yet final & decided)
  - Can be compiled to executable with pkg (protects your own source code implementation)


Valmet
FORWARD