Bringing Clouds Down to Earth: Modeling Arrowhead Deployments via Eclipse Vorto

Géza Kulcsár (IncQuery Labs)
Sven Erik Jeroschewski (Bosch.IO)
Kevin Olotu (Bosch.IO)
Johannes Kristan (Bosch.IO)

Eclipse SAM IoT 2020
Security | AI | Modelling

MADE AVAILABLE UNDER THE ECLIPSE PUBLIC LICENSE 2.0 (EPL-2.0)
How to maintain overview on a cloud?

The modeler

The ever-changing IoT cloud architecture

Source: https://www.br.de/nachrichten/kultur/instagram-caspar-david-friedrich-natur-bilder-aesthetik-kunst,RrnPw2Q
The need for bringing clouds down to earth

1. A high-level engineering design is up in the clouds, while its realization should be down here on earth
   • One of the main challenges of IoT design is an adequate representation of System-of-Systems (SoS)
2. Adding device models to the design representation allows for representing real (i.e., down-to-earth) connections between devices
   • To be more precise, those are ideal connections between digital twins, the other central concept of industrial IoT
Example: An “Arrowhead local cloud”
System models: the SysML language

- SysML is a wide-spread language and standard for systems modeling
  - Currently established version: 1.6, the release of version 2, constituting a major update, is planned in 2021
- Originally conceived for designing single, monolithic systems
  - Emerging support for distributed and dynamic scenarios
- Challenge: will SysML accommodate System of Systems (SoS)?
- SoSysML exploits one of the most powerful features, profiles for extending the language in a well-founded way
Eclipse Arrowhead ❤️ Vorto

- Eclipse Arrowhead (aka the Arrowhead Framework) is a software platform for operating dynamic industrial SoS installations, but without the “earthly details”

- As a recent addition, SoSysML has been proposed as a design-time extension for designing digital twin models

- Observation: Eclipse Vorto is an established device digital twin modeling approach
Eclipse Vorto - Tooling

Open Source Project for Semantic Modelling of Digital Twins and their Capabilities

Vorto consists of

- Vortolang: Domain Specific Language to describe Digital Twins
- Repository: Create, manage and distribute models (https://vorto.eclipse.org)
- Plugins: Transform Vorto models into something else (source code, request template, other representations)
- Telemetry Payload Mapping: Map the data sent by a device using a mapping specification based on a Vorto model
Eclipse Vorto - Vortolang

Information Model: describes a digital twin and its capabilities

Function Block: describes the capabilities that are implemented by the digital twin

- Properties of a Function Block are:
  - Status: describes properties, read-only
  - Configuration: describes properties, read-write
  - Event: describes events emitted by the device
  - Operation: describes a function that can be invoked on the device

Data type: describes complex data types or enumerations that can be assigned to Function Block properties

Mapping: describes platform- / implementation-specific information
Integration: Eclipse ❤️ MagicDraw!

System model authoring and integration

Vorto: Device modeling

Underlying model handling engine
Summary and future work

System modeling

Device modeling

Eclipse Arrowhead: a comprehensive industrial IoT integration platform
Thank you for your attention!

The research has received funding from the EU ECSEL JU under the H2020 Framework Programme, JU grant nr. 826452 (Arrowhead Tools project, https://www.arrowhead.eu) and from the partners' national funding authorities.

Project no. 2019-2.1.3-NEMZ\_ECSEL-2019-00003 has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the 2019-2.1.3-NEMZ\_ECSEL funding scheme.

Project no. 16ESE0367 has been implemented with the support from the Federal Ministry of Education and Research of Germany.