Simulating OPC UA Information Models

Jouni Aro & Lauri Saikko, Prosys OPC
13.11.2018
OPC Day Finland 2018
The Industrial Internet / IoT

Prosys OPC UA software products offer multiplatform capabilities, making them ideal building blocks of any networked system

- OPC UA connectivity
- Sensor data
- Smart devices
- Big data
- Analytics
- Optimization
- Automation
Prosys OPC UA Multiplatform Products

- **OPC UA Client**: Pro Beta available
- **OPC UA Client for Android**: Pro in development
- **OPC UA Java SDK**: FREE
- **OPC UA Historian**: FREE
- **OPC UA Modbus Server**: FREE
- **OPC UA Simulation Server**: Pro in development
Professional Services

- QuickStart
- Training & Workshops
- System Integration
- Design & Development
- Product Certification
- Consulting
OPC UA Workshop

• One, two or three days of training
• Covers the basics of OPC UA for experts and software developers
OPC UA Address Space Model

• Server Address Space consists of Nodes
  • Types
  • Objects

• Nodes are connected with References

• Types are abstract presentations of Objects
  • Define a structure
  • Objects follow the same structure
OPC UA Information Models

• Presentations of Semantic Models (=Types)
  • Use the concepts of the OPC UA Address Space Model

• OPC UA Standard Model
  • Base model

• Companion Models
  • Domain specific models: PLCopen, ISA-95, PackML, etc.

• Custom Models
  • Vendor specific models
Prosys OPC UA Simulation Server

• OPC UA Server with Graphical UI
• Developed using Prosys OPC UA Java SDK
• Test tool for offline client development
• Simulation of simple data
Objective

• Realistic reflection of a real production server
  • Support Information Models (Types)
  • Customized Address Space Structure (Objects)
  • Simulate Data in Objects
Objective: Simulate Any OPC UA Server

Real OPC UA Server

OPC UA Client

Simulation Server

Real Data

Export

Objects.xml

Import

Types.xml

Infomation Models

Simulation Data
Configuring simulation signals

• Configuring Types
  • Base configuration for all Objects of specific types
  • Simulating instances using TypeDefinition configurations

• Configuring Objects
  • Overriding type configurations
Advanced configuration of signals

- Type configurations can be utilized recursively
- More options for type-based simulation
- Configuration of the most specific TypeDefinition of an instance dominates
Simulation Server in practice
Simulation Server in practice
Conclusions

• Simulating a real production server in a test environment
  • Custom Information Models
  • Complex simulated data
  • Flexible configuration of simulation signals

• Future development
  • More accurate simulation models
  • Simulation not restricted to Node values
Questions?