Service Orchestration with OPC UA in a Graphical Control Language

Thörin, Alfred; Hagsund, Johan; Johnsson, Charlotta

Published in: [Host publication title missing]

DOI: 10.1109/ETFA.2014.7005351

2014

Citation for published version (APA):
Service Orchestration with OPC UA in a Graphical Control Language

Alfred Theorin, Johan Hagsund, and Charlotta Johnsson
Department of Automatic Control, Lund University, Lund, Sweden

Background
Increasing market demands require highly flexible automation systems. A successful approach to compose flexible software is Service Oriented Architecture (SOA). In this work a convenient way to do SOA service orchestration with OPC Unified Architecture (OPC UA) is presented.

OPC Unified Architecture
OPC UA offers robust and secure communication for distributed systems. It is backward compatible feature-wise with the classic OPC standards and is platform and language independent. It also offers SOA features and more powerful modeling capabilities with types and instances.

The Grafchart Language
Grafchart is a graphical programming language based on the PLC standard language SFC. The basic building blocks are steps (application states) with actions (what to do when) and transitions (how and when to change application state).

OPC UA Integration in JGrafchart
An OPC UA Object is bound to a node in an OPC UA server. The OPC UA variables and methods are then used like native variables and methods.

Use Case
The OPC UA support was tried out on a physical device in an Industry 4.0 setting. The device consists of a punching machine (1) and a conveyor belt (2). A microcontroller (3) was used to expose the device as variables in an OPC UA server.

Conclusions
The new integrated OPC UA support in JGrafchart is a convenient way to do SOA service orchestration in the automation domain. Both DPWS and OPC UA are required to cope with industrial requirements on device level and now JGrafchart supports both. With an OPC classic adapter, JGrafchart can now also interact with most industrial automation systems.

Acknowledgments
Financial support from the VINNOVA-FFI project LISA is gratefully acknowledged. The authors are members of the LCCC Linnaeus Center and the eLLIIT Excellence Center at Lund University.

Download: http://www.control.lth.se/Research/tools/grafchart.html