Manufacturing Execution in combination with Autonomous Agents based on SAP and OPC UA

Ruediger Fritz, SAP

October 11th 2017
OPEN INTEGRATED FACTORY – GENERATION 2017
Smart Products
OPC UA as the standard for each machine unit
The System Layers
The Digital Plant in the Extended Supply Chain

**MONITOR**

**DESIGN**

**PLAN**

**RESPOND**

**PRODUCE**

**DELIVER**

**OPERATE**

---

**ERP S/4**
- Production plan
- Bill of materials (BoM)
- Variant management
- Production steps...

**MES (ME & MII)**
- Production details management
- Detailed and flexible
- Production step routing
- Shop floor controls for each step...

**PCo**
- Mapping of ME production details to PLC control parameters (recipe)
- Buffer recipe for fast access
- Set/Get parameters - Server/Client...

**PLC (Machine)**
- Sensor detects material carrier
- Requests control parameters from ME...

**Connectors**
- Serial Numbers
- Quality results per lot
- Order confirmation
- Inventory update, Equipment usage...

- Log parametric data
- Tolerance checks
- Return “conformance” or “non-conformance” decisions

- Mapping of measured values/results to Business Data Model

- Machine reports completion and requests next operation

*only excerpt of involved entities*
Automation Workflow Example

Loop M → PCo: “Mover with SFC 123 arrived at Pos Y4”
PCo → Loop B: “Move SFC 123B to Pos B4”
Loop B → PCo: “Mover with SFC 123B arrived at Pos B4”
PCo: (Get Lock for Rob B)
PCo → Rob B: “Do Job #3 – Handle SFC123”
Rob B → PCo: “Job #3 – Handle SDF123 done”
PCo → Loop B: “Release Mover” (move on)
PCo → Loop M: “Move SFC 123 to Pos 5”
Loop M → PCo: “Mover with SFC 123 arrived at Pos 5”
PCo → Cam: “Take Photo [Par: expected colour blue]”
Cam → PCo: “Camera result: [not blue, URL to .jpg …]”
Log Non-Conformance in ME-System

…
Real Life Example: Open Integrated Factory – Generation 2017
Machine Units seen as Service Providers and Service Consumers (SOA)

(1) Each machine unit is independent
» In a departure from convention, the units are not linked to each other by a single program inside a single PLC
» Each unit comes with its own controller

(2) Units are talking to each other on the basis of OPC UA
» From business perspective (production order details like routing and recipe/set-points) the units are orchestrated by SAP Plant Connectivity (Vertical Integration)
» From technical perspective, some machine units – here Camera and Robot - exchange information directly (Horizontal Integration)

(3) Units are OPC UA Client and OPC UA Server at the same time
» A server can offer tags, events and methods
» A client can consume/react on tag changes and events and call methods

(4) Units publish their capabilities
» A Service Oriented Architecture with regards to hardware is possible
SAP Plant Connectivity: OPC Client and OPC Server Communication Patterns in Context of Machine Integration

A Notification

Insights, Alarms etc.

Transaction → DB Record (Time Series)

(Guaranteed) Notification Delivery, Buffer mechanisms, Filter/Rule Framework, Customer specific code, ...

B Query

Dashboards, Applications

Destination (e.g. MII, ME, HANA, SAP Cloud ...)

Tag-Query

C Synchronous Service Calls

SAP PCo (OPC Client)

OPC UA Methods, Custom Orchestration*

Query-Interface (Read/Write)

Classical Gateway Task: Protocol conversion, Source/Destination mapping

Data Source/Machine (e.g. OPC UA-Server)

Services

*Project specific configuration/implementation

© 2017 SAP SE or an SAP affiliate company. All rights reserved.
Autonomous Agents based on OPC UA in Production
Challenges to Manufacturing Today…

Cyber-Security
- Increasing Interoperability = Increasing Vulnerability
- Manage Complexity

Modular Assembly
- Assembly Line replaced by Cellular Manufacturing
- New organizational structures require ad-hoc decisions
- Increasing Interoperability

AI / Machine Learning
- Insight to Automation (immediate action)
- Pattern Recognition
- Autonomous Systems, Edge Processing

Challenges to Manufacturing Today…

» Modular Assembly
  » Assembly Line replaced by Cellular Manufacturing
  » New organizational structures require ad-hoc decisions
  » Increasing Interoperability

Modular Assembly

» Dynamic Routing → alternative operations, alternative resources
Ad-hoc decisions based on frequent machine-to-machine communication (status, set-up, availability …) linked to business data (order details, master data …)

Arbitrary sequence
Either / Or
All possible paths
AI / Machine Learning

» Autonomous Systems, Edge Processing
Low Cost Set-up for demonstrating a powerful concept of modern Manufacturing Execution
Resource A - D could represent e.g. 4 (identical) packing machines
A single box could represent e.g. an Autonomous Guided Vehicle (AGV) or just a kind of Handling Unit.
• each Box one RPi with OPC UA Client
• all Resources managed by another RPi3 with OPC UA Server
Technical Basis
SAP Plant Connectivity on RPi Linux

(1) Networking over Wireless LAN
(2) Software Development
   » .Net Mono Framework
   » OPC UA 1.0.3 libraries
   » Custom-developed .Net dll to provide RPi Device IO connectivity

April 24 – 28, 2017
Hannover, Germany
Negotiation can start – pressed Button initiates communication via WiFi
First step: SAP PCo on RPi as OPC UA Client to SAP PCo as OPC UA Server wrapping the SAP MES: “get_current_sfc” (means get the Production Order from MES which is in work right now)
Second step: SAP PCo on RPi as OPC UA Client to other SAP PCo as OPC UA Server managing the (four) resources: “get_free_resource”

Behind this method any sophisticated logic could run -
e.g. a machine learning based algorithm!
Logic on OPC UA Server RPi proposes and allocates the appropriate Resource (OPC UA Method response)
Validation by NFC
Validation by NFC, follow up communication/confirmation to SAP MES
Short comment about challenges of OPC UA …?
Thank you!
Thank you.

Contact information:

Rüdiger Fritz  
Director Product Management SAP Plant Connectivity  
Dietmar-Hopp-Allee 16, 69190 Walldorf  
0049 6227 740142  
ruediger.fritz@sap.com
Appendix
SAP Plant Connectivity: OPC Client and OPC Server Communication Patterns in Context of Machine Integration

**A** Notification

- Insights, Alarms etc.
- Transaction
- DB Record (Time Series)
- (Guaranteed) Notification Delivery, Buffer mechanisms, Filter/Rule Framework, Customer specific code, ...

**B** Query

- Dashboards, Applications
- Tag-Query
- Query-Interface (Read/Write)

**C** synchronous Service Calls

- **SAP PCo**
  - OPC UA Methods, Custom Orchestration*
  - Classical Gateway Task: Protocol conversion, Source/Destination mapping

- **Data Source/Machine**
  - (e.g. OPC UA-Server)
  - Device Tag (Data Point)
  - Services

**Foundation:**

- **OPC UA Methods**, Custom Orchestration*
- **OPC UA Client**

*Project specific configuration/implementation
SAP Plant Connectivity – more than just a simple Gateway for Connectivity

**Supported protocols:**
- **OPC UA, MQTT, OPC DA, OPC HDA, OPC Alarms, Citect, IP2I, Osisoft PI (2), Proficy Historian, File-Monitor (2), ODBC, DLE DB, Socket, Modbus**
- **SDK for proprietary, specific agents** (e.g. ifm Linerecorder, UDP, RFC1006, Euromap 6x, Atlas Copco Open Protocol, Kafka and multiple other project specific implementations)

**Devices, Logic Controllers, Historians**
Each production step documented and available for analytics.
Another Device integration
Torque To-be and as-is PDC
Plant Connectivity – An Example of Configuration
AI / Machine Learning with immediate action on automation level

» SAP PCo (OPC Client) → Cloud / Data Lake → Apply Model to Dynamic Edge Processing → SAP PCo to Automation

**Input (Edge):**
- Sensors
- Images
- Production Order
- Master Data
- ... and more

**Machine learning (Cloud):**
- Train model
- Prepare data
- Apply model
- Capture feedback

**Output (Edge):**
- Applied model
  - Run Machine Learning Algorithm at the Edge
  - Immediate action can be taken