TUHH
Technische Universität Hamburg-Harburg

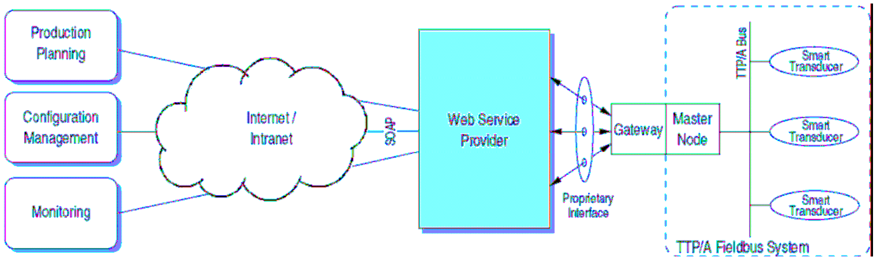
Application specific vs. standard Web service interfaces for the vertical integration of fieldbus systems

Marcus Venzke, Christoph Weyer, Volker Turau
Telematics Department,
Hamburg University of Technology

WISES 2005, 20.5.2005

Objective

- System integration: flexibility, speed, cost
- Vertical integration of automation systems
- Suitable technology: Web services



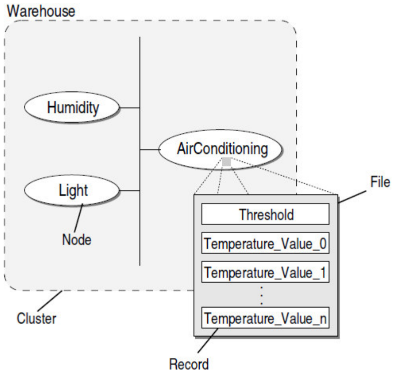
- Suitable abstractions for Web service interface?
- We introduce and compare 2 approaches

Outline

- TTP/A fieldbus systems
- Standard Web service interface OPC XML-DA
  - Mapping to TTP/A fieldbus systems
- Generation of application specific interfaces
  - Mapping to TTP/A fieldbus systems
- Comparison

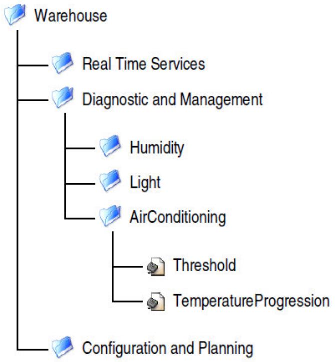
TTP/A Fieldbus Systems

- Clusters = Nodes connected by real-time bus
- 3 Interfaces:
  - Real-time service (RS)
  - Diagnostic and management (DM)
  - Configuration and planning (CP)
- Interface File System (IFS)
  - Four-level hierarchy
  - 32-bit records
  - Read, Write, Execute
- Meta-data (XML)
  - Node description
  - Cluster description



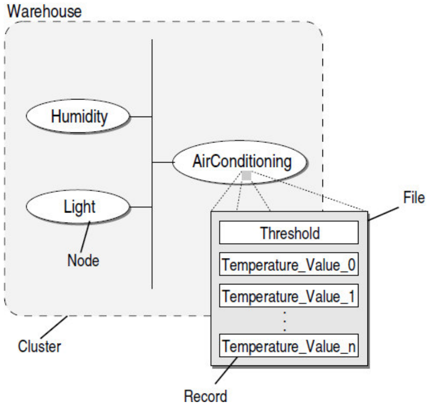
OPC XML-DA

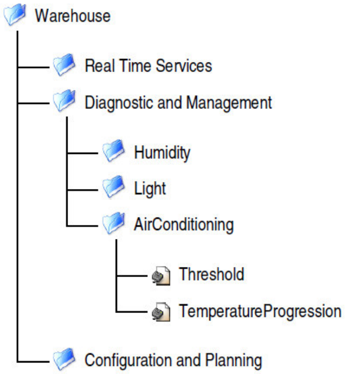
- Standard Web service interface
  - Reading & writing data to plant floor automation systems
  - By OPC Foundation (industry consortium, > 300 members)
- Model: Hierarchy of named items
  - One value per item
  - Properties describing items
- Operations:
  - Read & Write
  - Subscription (read periodically)
  - Browse (explore hierarchy)
  - GetProperties (read meta data)



Mapping OPC to IFS – Hierarchy

- RS, DM, CP as nodes in hierarchy
- From IFS hierarchy: Cluster and nodes
- From meta data: Names and logical variables





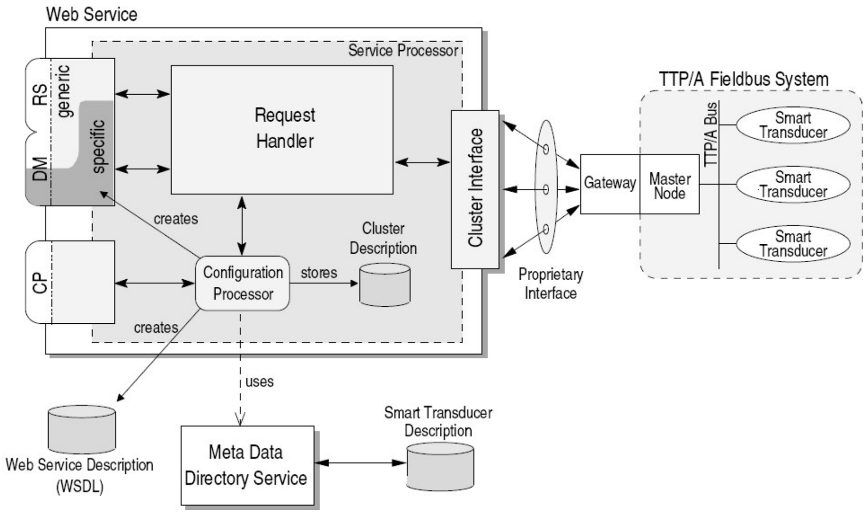
Mapping OPC to IFS – Operations

- Read / Write: Mapped to many IFS calls
  - Logical variables mapped to multiple records
  - Coarse granularity: Many variables with one call
- Execute not existing in OPC
  - Mapped to Write
  - Additional logical variable
- Subscription via Read
- Browse & GetProperties
  - Return meta data from cluster and node description

Application specific interfaces (1)

- 3 Web service interfaces: RS, DM, CP
- RS + DM data access similar to OPC XML-DA
- Node specific operations in DM
  - Generated from meta data
- CP: Operation Configure
  - Parameter: Cluster description (Node description implicit)
  - Configures and adds operations to DM

Application specific interfaces (2)



Comparison (1)

OPC XML-DA	Application specific interfaces
• One interface	• Interfaces RS / DM / CP
• For many applications	• For individual systems
• System characteristics expressed by hierarchy, items, naming	• System characteristics expressed by specific operations
• Standard clients: Implement interface (e.g. SIMATIC WinCC, SAP)	• Standard clients: Require generic Web service support

Comparison (2)

> Logical operation

> E.g. maintenance operation of robot in high bay warehouse
  – Navigate robot to position
  – Parameters: x, y
  – Result: status code

Photo from: http://www.schott-music.com /unternehmen/SchottWeltweit/mainz/

OPC XML-DA	Application specific interfaces
• 1. Write parameters 2. Execute via Write 3. Read status code	• Call operation in DM
• Workflow: 3 activities	• Workflow: one activity

Comparison (3)

OPC XML-DA	Application specific interfaces
• Configuration: • Write many variables • Logic in client	• Configuration: • One call via CP • Logic in server
• Type checking at run time (types determined dynamically)	• Type checking at compile time (types determined from WSDL)

Conclusion

OPC XML-DA	**Application specific interfaces**
One interface for accessing many automation systems	Support programmers view on automation system
Suitable for standard clients	Suitable for specifically implemented clients and workflow descriptions

Outlook: Combining both

- OPC XML-DA: RS + DM data access
- Standard-Interface for CP
- Application specific: DM operations

Questions or comments?