

Distributed Control Patterns using Device Profile for Web Services



J. Marco Mendes^{1,2}

Alexandre Rodrigues³

Paulo Leitão⁴

Armando W. Colombo²

Francisco Restivo¹



2008 Middleware for Web Services (MWS 2008) Workshop
12th IEEE International EDOC Conference (EDOC 2008)
Tuesday, 16 September 2008, Munich, Germany

Overview

- ❑ Introduction and Motivation
- ❑ Device Profile for Web Services
- ❑ DPWS for the communication of automation components
- ❑ Enhanced Interaction patterns
- ❑ Prototype application
- ❑ Conclusions and future work

Introduction and Motivation

- ❑ New manufacturing concepts:
 - Increasing **flexibility**, **modularity** and **re-configurability**
 - Different approaches for RMS: IEC 61499, MAS, **SoA**.
- ❑ **SoA in Industrial Automation/Manufacturing:**
 - Adoption of a unifying technology for all levels of the enterprise
 - Support life-cycle and reconfiguration processes
- ❑ Open questions
 - Control architecture
 - Usage/benefits of Device Profile for Web Services (DPWS)
 - Complex control/interaction patterns
 - Integration
- ❑ Goals:
 - **Generic communication stub/skeleton** for components (devices)
 - Enhanced **interaction patterns** towards real-life services

Motivation

Everything should be made as simple as possible, but not simpler

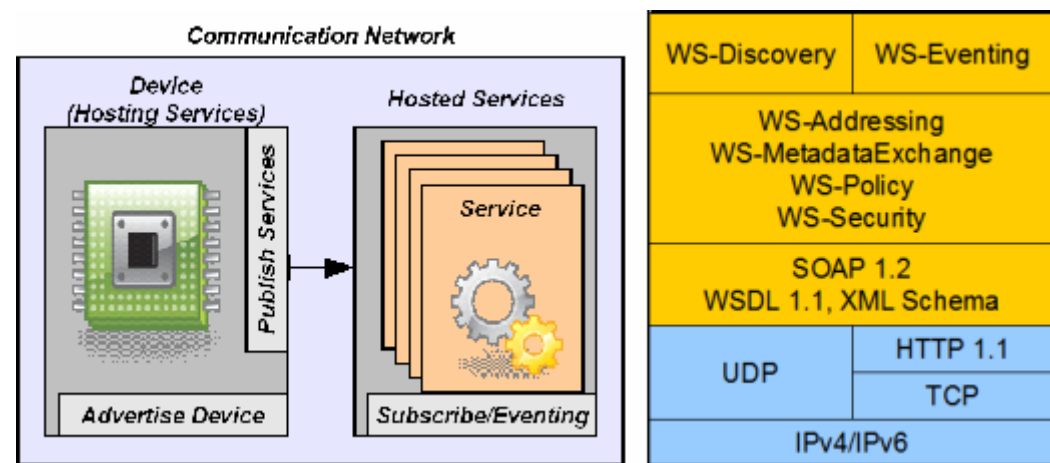
by Albert Einstein

DPWS (Device Profile for Web Services): Overview

- ❑ Extensions for using Web services in electronic devices
 - Taking in account their specific constraints
- ❑ Definition of web services for devices
 - Considering the peer-to-peer direct communication

❑ Main features:

- Description of WS
- Secure messaging
- Dynamic discovering
- Subscription and events



- ❑ **Actual status:** public consultation draft (February 2006)
 - Recently a Technical Committee of OASIS is working on standardization (Web Services Discovery and Web Services Device Profile)
- ❑ DPWS was used as a basis for the ITEA SIRENA project (2005)

DPWS: Advantages and disadvantages

- ☺ Reliance on Web services
- ☺ Programming language independence and generic mechanisms of abstraction
- ☺ Complete set of functionalities for device integration
- ☺ Reduces interdependencies between components (devices)
- ☺ Limited constraint functionalities (targeting small devices)

- ☹ Complex descriptions and verbosity of XML Web services
 - Embedded systems rarely have enough memory and processing power
- ☹ Limited set of features for complex manufacturing operations
 - Required to add other technologies over DPWS
- ☹ Based on older standards (such as WSDL 1.1)
- ☹ Acceptance from Industrial Automation and manufacturing partners (standardization)

DPWS: Old and current implementations and projects

□ Implementations

- SoA for Devices (SOA4D)
- WS for Devices (WS4D)
- Integrated in Windows Rally

□ Projects

- ITEA SIRENA project
- ITEA SODA project
- EU Research Project SOCRADES

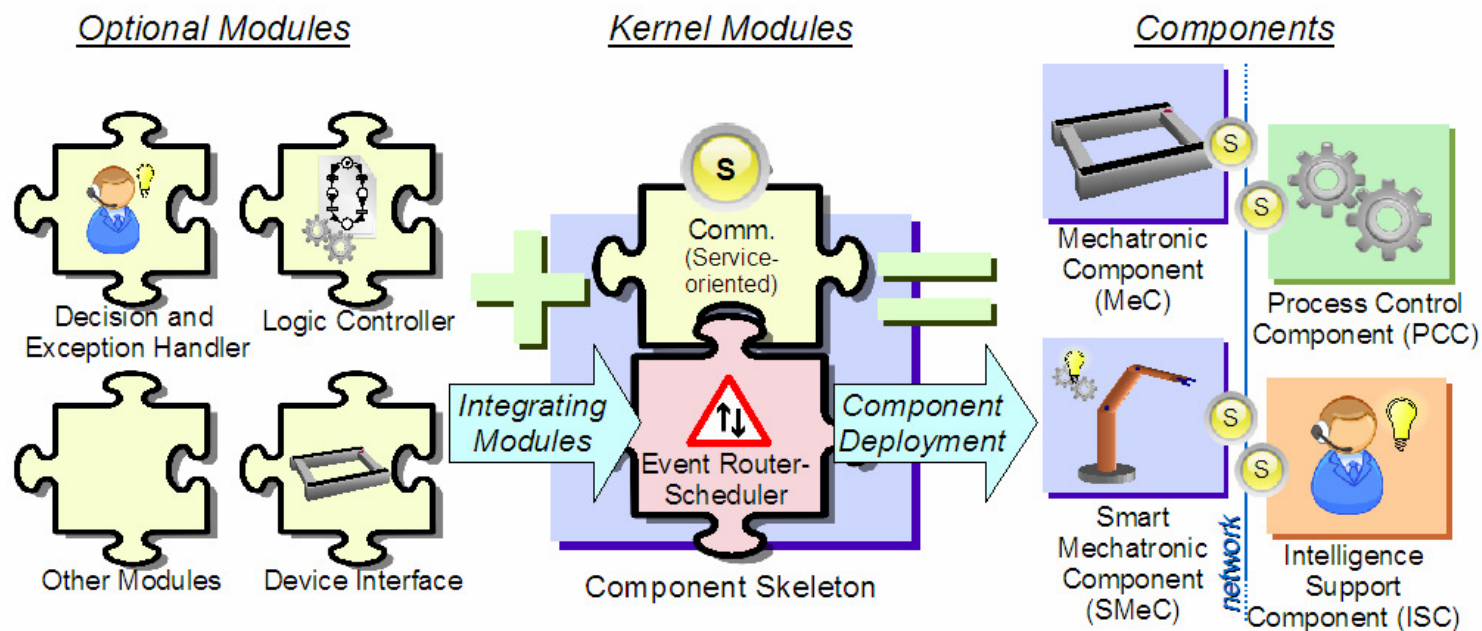
□ Applications:

- Integration of 2D-3D engineering tools with physical devices
- Methods for developing efficient diagnosis mechanisms in devices and systems
- Control mechanisms using DPWS
- ...

DPWS for the communication of automation components: General architecture

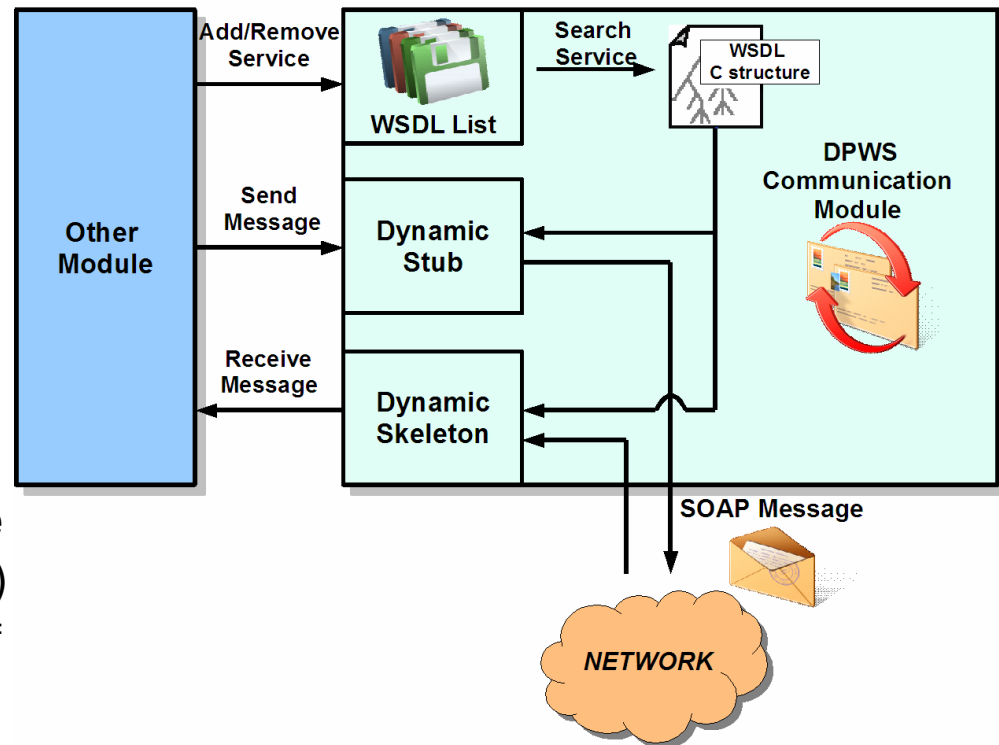
Architecture and environment:

- Service-oriented components with different roles
- Developing/deploying components
 - Component framework with several functional modules



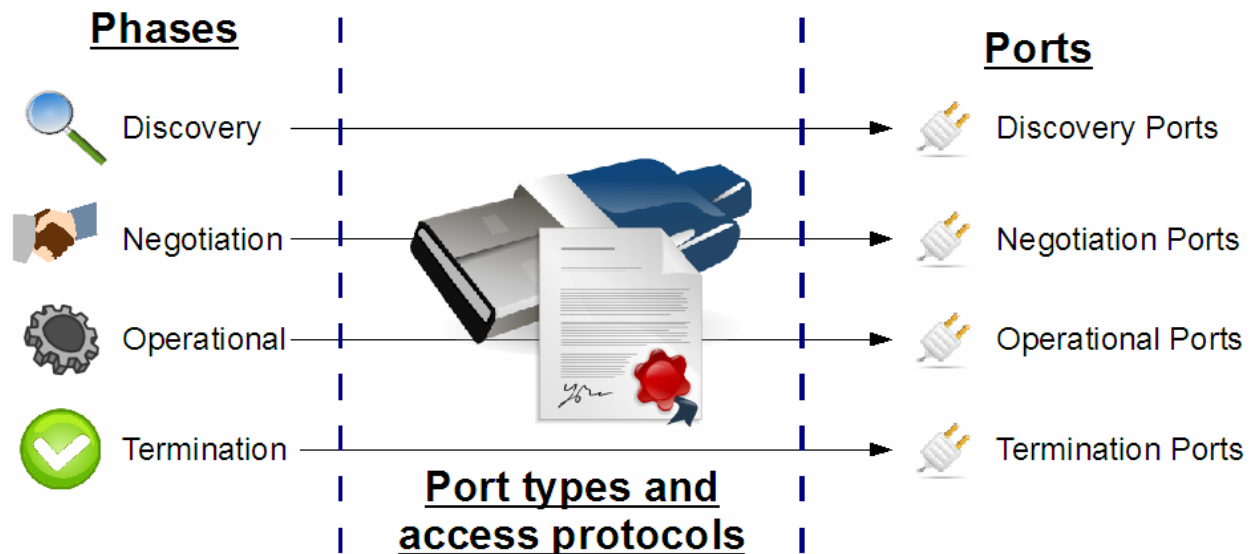
DPWS for the communication of automation components: Communication module

- Features of the DPWS communication module
 - Uses and extends the SOA4D DPWS toolkit
 - Generic stub and skeleton for managing services: Automatic serialization of SOAP messages into C structures according to the WSDL file (and vice-versa)
 - Avoid the static generation of stub/skeletons by the DPWS toolkit
 - Creation of new services classes, port types and operations are automatically done just by adding the WSDL C structure to the module
 - Parsing of events directly to the application through callbacks
 - Integration into the component framework



Enhanced Interaction patterns

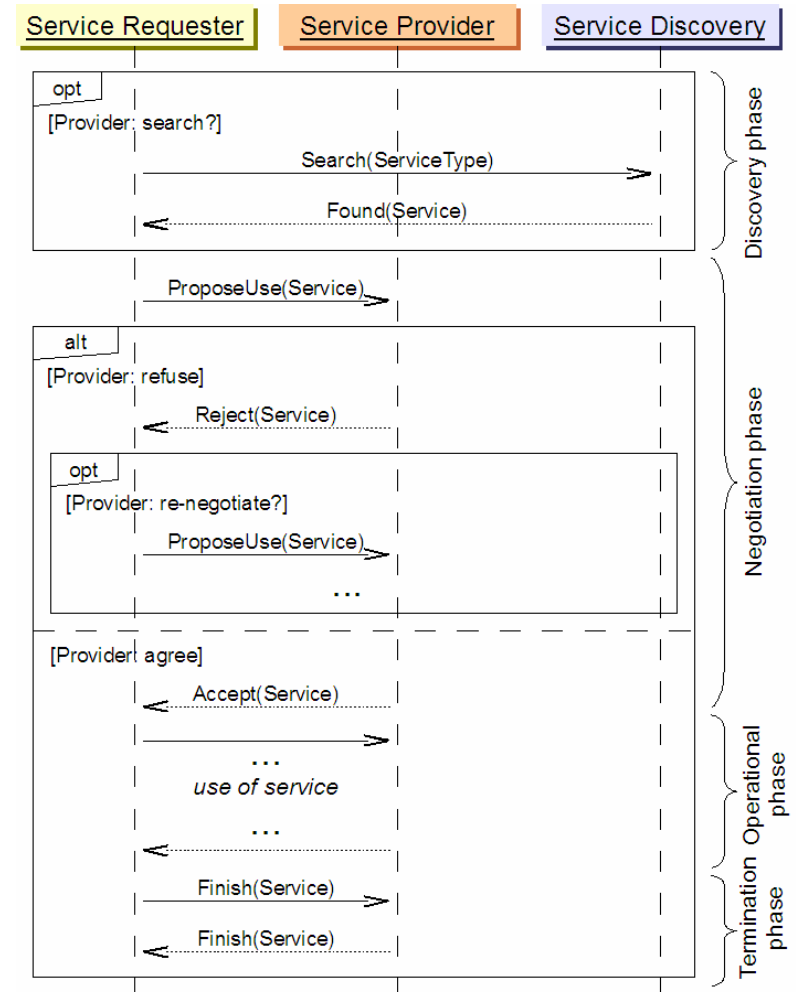
- ❑ Shortcomings of DPWS in terms of advanced interaction patterns
- ❑ Different phases involved in the interaction of the service requester and provider



- ❑ Implemented by the DPWS Communication module
- ❑ Logical ports of the operational phase can be directly related to the physical ports of devices (e.g. conveyors)

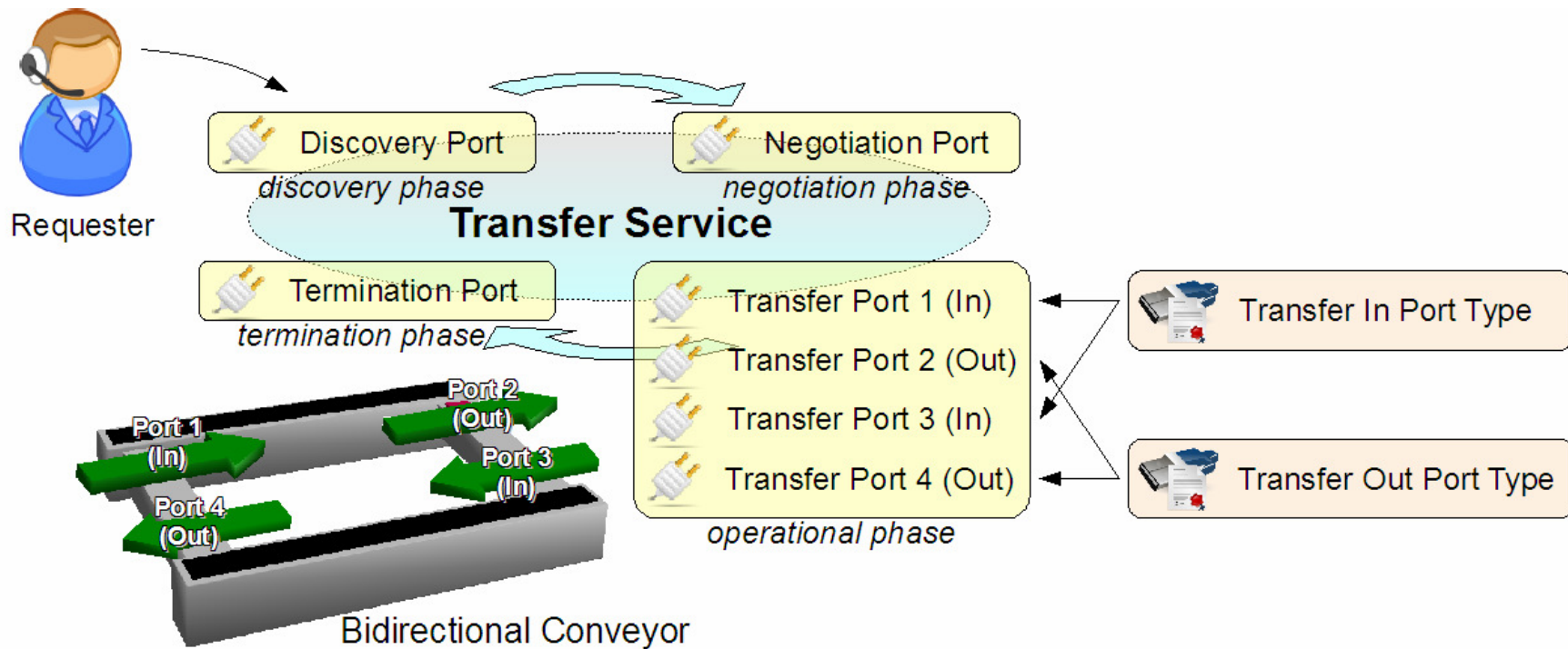
Enhanced Interaction patterns: Sequence diagram

- Sequence diagram of the process to interact with services using the four phases



Prototype application

- Bidirectional conveyor with four different ports



Prototype application (2)

- ❑ Simulator for the conveyor
 - Component with modules:
 - Event Router-Scheduler
 - DPWS Communication
 - Virtual Device Interface (simulates a conveyor)
 - User Interface



Conclusions and future work

- ❑ Specification and development of a DPWS Communication module:
 - Ease of development, integration, deployment, maintenance and (lifecycle) management of devices and services
 - Device can be changed with minimal reconfiguration and without the need of re-deploy components
 - Dynamic approach against the used toolkit
- ❑ Formalization for different phases for the usage of services
 - Port type/protocol to create port instances for the different phases
 - Association of services and its ports to physical ports of devices
- ❑ Enhancements towards flexibility and re-configurability
- ❑ Future work
 - Stabilize the implementation and improve the interaction concepts
 - Deployment in real devices and proof their applicability in industrial automation systems

Thank you!

J. Marco Mendes
marco.mendes@fe.up.pt

Alexandre Rodrigues
aar15616@fct.unl.pt

Paulo Leitão
pleitao@ipb.pt

Armando W. Colombo
*armando.colombo@
de.schneider-electric.com*

Francisco Restivo
fjr@fe.up.pt