

Object-Oriented Internet - Azure interoperability

Mariusz Postól¹, Piotr Szymczak², and Clemens Vasters³

^{1, 2}Institute of Information Technology, Lodz University of
Technology, Łódź, Poland

³Microsort

September 29, 2020

Abstract

Information and Communication Technology has provided society with a vast variety of distributed applications. By design, the deployment of this kind of application has to focus primarily on communication. This article addresses research results on the systematic approach to the design of the meaningful Machine to Machine (M2M) communication targeting distributed mobile applications in the context of new emerging disciplines, i.e. Industry 4.0 and Internet of Things. This paper contributes to the design of a new architecture of mobile IoT solutions designed atop of the M2M communication and composed as multi-vendor cyber-physicals systems. The described reusable library supporting this architecture designed using the reactive interoperability archetype proves that the concept enables a systematic approach to the development and deployment of software applications against mobile IoT solutions based on international standards. Dependency injection and adaptive programming engineering techniques have been engaged to develop a full-featured reference application program and make the proposed solution scalable and robust against deployment environment continuous modifications. The article presents an executive summary of the proof of concept and describes selected conceptual and experimental results achieved as an outcome of the open-source project Object-Oriented Internet targeting multi-vendor plug-and-produce interoperability scenario.

Keywords— Azure, Cloud Computing, Object-Oriented Internet, Industrial communication, Industry 4.0, Internet of Things, Machine to Machine Communication, OPC Unified Architecture

1 Document Header

1.1 Title proposals

- Object-Oriented Internet - reactive visualization of asynchronous data using AZURE

- Object-Oriented Internet - Azure interoperability

1.2 Abstract

1.3 Key words

Azure, Cloud Computing, Object-Oriented Internet, OPC Unified Architecture, Machine to Machine Communication, Internet of Things, OPC UA, RxNetworking

2 Introduction

- **Subject** - A basic matter of thought, discussion, investigation, development, etc. Describe the problem and the motivation for undertaking the effort to solve the problem.
- **Goal** What we are going to achieve - the result or achievement toward which effort is directed.
- **Scope** - What we must do to prove the goal have been achieved. Extent or range of development, view, outlook, application, operation, effectiveness, etc.
- **Related work** - Any information about available reusable deliverables related to this work.

3 Azure Main Technology Features

- **Selection of the service**
- **Metadata** - must be discussed in context of the design/run time stages.
 - **Device Template (DT)**
 - **Device Capability Model**
 - **Interface**
 - **Digital Twin Definition Language (DTDl)**
- **Simple, complex and structural data processing**
- **Connectivity**
- **How to implement** All about available libraries and tools

4 OOI Main Technology Features

- Machine To Machine communication based on the semantic data
- OOI PubSub Implementation Architecture
- Simple, complex and structural data processing

5 Azure to Sensors (A2S) connectivity deployment (field level connectivity)

The title must be revised

- **Architecture** - Domain model presenting relationship between the: Azure, PubSub Gateway, Device, Design and development tools
- **Connectivity** - Describe reactive nature of the Azure monitoring process data (telemetry) services.
- **Deployment phases**
 - Design
 - Gateway and devices registration
 - Authentication
 - Device/Service association
 - Device/Application association
 - Establishing session
 - * Device/Device Template (Device Capability Model) association
 - establishing a semantic-context
 - * Security management - establishing security-context
 - Interconnection - exchange of data
 - Maintenance

We have selected IoT Central because:

- provides process data visualization user interface
- allows to describe devices using metadata containing telemetry data types

6 Gateway implementation

- Architecture
- Protocol selection and mapping
- Configuration
- Testing

7 Conclusion

The OPC UA PubSub to Azure gateway (**AzureGateway**) implementation has been just published on GitHub as the open-source (MIT licensed) as a part of the more general concept of the Object-Oriented Internet reactive networking. It is proof of the concept that

1. OPC UA PubSub can be implemented as a powerful standalone package
 - no C/S dependency

2. Azure interoperability can be implemented as an out-of-band communication (MQTT, AMQP, HTTP) - no PubSub dependency
3. Process data functionality can be composable at run-time - no programming required