ITEA Office
High Tech Campus 69 - 3
5656 AG Eindhoven
The Netherlands
T + 31 88 003 6136
E info@itea3.org
W www.itea3.org

ITEA 3 is a EUREKA strategic ICT cluster programme

Exploitable Results by Third Parties

ITEA*2* 12011 BaaS – Building as a Service

A Semantic Service Framework for Building Automation Systems

Project details

Project leader:	Franz-Josef Stewing
Email:	franz-josef.stewing@materna.de
Website:	http://baas-itea2.eu

Name: BaaS Reference Architecture		
Input(s):	Main feature(s)	Output(s):
▪ Requirements ▪ Communication technologies ▪ Semantic concepts ▪ Run-time platform	▪ Domain model ▪ Lifecycle view ▪ Functional view ▪ Information view ▪ Behavioral view	▪ Architecture for semantic service framework ▪ Semantic service platform

Unique Selling Proposition(s):	▪ Reference architecture for semantic service framework for building automation systems (as well as automation systems for other domains)
Integration constraint(s):	▪ Use of OSGI or other approach to micro-service deployment recommended ▪ Use of semantic Web language like OWL, RDF or json-ld recommended ▪ Use of model-based development approach recommended
Intended user(s):	▪ Software architects that want to create a (building) automation system architecture compliant with the BaaS reference architecture. ▪ Developers of building automation components, systems and products that want to implement a building automation platform following the BaaS approach.
Provider:	▪ BaaS Project
Contact point:	▪ BaaS Website, http://baas-itea2.eu/
Condition(s) for reuse:	▪ The BaaS Reference Architecture and its concepts may be freely used by any party for the specification of a concrete architecture, provided a reference to BaaS Deliverable D05/2 is given. ▪ The reference to BaaS deliverable D05/2 should read: N. Vicari (Ed.), “Deliverable D05 – BaaS Reference Architecture, Version 2”, ITEA2 Project 12011 “Building as a Service”, May 20, 2016.

Latest update: 20.10.2016

Name: Bug Fixes for Eclipse Californium (Cf) CoAP Framework		
Input(s):	Main feature(s)	Output(s):
▪ GET request with resource type (rt) or other search parameters specified	▪ Constrained Application Protocol (CoAP) ▪ CoRE Resource Directory	▪ Discovered CoAP resources in link format

Unique Selling Proposition(s):	▪ Several bug fixes in Californium implementation made CoRE resource discovery work correctly with an arbitrary number of search parameters e.g. at the CoRE Resource Directory
Integration constraint(s):	▪ One bug report has already been submitted but must still be integrated into the Californium CoAP framework ▪ Other bugs have been discovered and fixed but must still be submitted to the Eclipse Californium CoAP Framework project
Intended user(s):	▪ IoT developers using the CoAP communication protocol for CoRE resource discovery at CoAP endpoints or at the CoRE Resource Directory
Provider:	▪ Eclipse Californium CoAP Framework project
Contact point:	▪ https://github.com/eclipse/californium
Condition(s) for reuse:	▪ Eclipse Distribution License 1.0 (BSD) ▪ Eclipse Public License 1.0

Latest update: 20.10.2016

Name: jCoAP, a java implementation of CoAP		
Input(s):	Main feature(s)	Output(s):
▪ Java based software realizing some business logic	▪ Fast, lightweight and flexible CoAP implementation ▪ Independent modeling of Resources ▪ Update: Improved multicast ▪ Update: Extended documentation ▪ New: Multiple MIME types per Resource ▪ New: Individual rights management for resources ▪ New: real-time capable branch	▪ Software, enabled to act as a CoAP client, server or both

Unique Selling Proposition(s):	▪ Flexible and fast ▪ Existing real-time capable expansion
Integration constraint(s):	▪ Java 2 SE 1.6+ runtime environment
Intended user(s):	▪ IoT developers and researchers using the CoAP communication protocol for CoRE resource discovery at CoAP endpoints or at the CoRE Resource Directory
Provider:	▪ University of Rostock, IMD / WS4D ▪ https://gitlab.amd.e-technik.uni-rostock.de/ws4d/jcoap ▪ http://www.ws4d.org/ws4d-jcoap/
Contact point:	▪ Björn.Butzin – bjoern.butzin@uni-rostock.de ▪ Hannes.Raddatz – hannes.raddatz@uni-rostock.de
Condition(s) for reuse:	▪ Apache License, Version 2.0

Latest update: 06.10.2016

Name: JMEDS Framework		
Input(s):	Main feature(s)	Output(s):
▪ Mostly technology-independent Java source-code for clients, devices and services	▪ The framework provides an abstraction layer to ease the implementation of clients and services utilizing different communication technologies.	▪ Client, device and service applications running different communication technologies

Unique Selling Proposition(s):	▪ The API of JMEDS follows the paradigm of a service-oriented architecture (SOA) build on devices. It is inspired by the Devices Profile for Web Services (DPWS) and applies XML Schema 1.0 to describe the data types that are provided by services. It has been proven that the structure of the APIs abstraction layer is common to technologies like UPnP, Bluetooth and BACnet. Plugins have been implemented successfully for these technologies. The JMEDS framework library is lightweight, modular und extendible, has no external dependencies (e.g., it does not need an application server) and can run on a Java SE Embedded 8 compact1 VM.
Integration constraint(s):	▪ Java SE Embedded/SE/EE 8
Intended user(s):	▪ Software developers in the internet of things (IoT) area
Provider:	▪ Materna GmbH Information & Communications
Contact point:	▪ https://sourceforge.net/projects/ws4d-javame/ ▪ http://www.ws4d.org/
Condition(s) for reuse:	▪ Eclipse Public License, version 1.0

Latest update: 17. October 2016

Name: Materna Management Tree (MMT)		
Input(s):	Main feature(s)	Output(s):
▪ Java data handler plugin implementations	▪ Homogenous integration facade for integrating heterogeneous sources of management data.	▪ Technology-independent view of the system's management state

Unique Selling Proposition(s):	▪ The MMT provides location-transparent access to the heterogeneous management data. These data are not stored within the tree itself; it rather offers a homogeneous and consistent interface for management data retrieval and manipulation. ▪ To ensure scalability within large distributed monitoring environments, the management tree itself can be distributed across the network. ▪ When the monitoring environment grows, e.g. when new management-relevant components are added or when disjoint monitoring environments are merged, the management tree can be rearranged or extended at runtime. ▪ In order to maintain location-transparent access of the management data, management tree instances can be nested.
Integration constraint(s):	▪ Java SE 7/8 ▪ OSGi (optional)
Intended user(s):	▪ Software engineers in the landscape of IoT and Cloud Computing
Provider:	▪ Materna GmbH Information & Communications
Contact point:	▪ Christoph Fiehe - christoph.fiehe@materna.de ▪ Ingo Lück - ingo.lueck@materna.de
Condition(s) for reuse:	▪ Licensing (OSSW licensing is under discussion)

Latest update: 19. October 2016

Name: BaaS- Authorization Adaptor		
Input(s):	Main feature(s)	Output(s):
▪ Companies, users, BaaS Services, user and BaaS Service Roles	▪ Configuration and managing of BaaS Platform services and developers. This software package allows managing inter-BaaS service authentication and authorization.	▪ Authentication and authorization software module

Unique Selling Proposition(s):	▪ Software package/Library of methods prepared to authenticate and authorization
Integration constraint(s):	▪ Keycloak (open source OAuth2.0 software tool) ▪ Java ▪ JSON ▪ REST Services libraries
Intended user(s):	▪ Programmers who need IDM and service IDM security ▪ Systems Integrators who need IDM and service IDM security
Provider:	▪ KoçSistem
Contact point:	▪ Erdem Ergen – erdem.ergen@kocsistem.com.tr
Condition(s) for reuse:	▪ Licensing

Latest update: 18 October 2016

Name: BaaS-Zigbee plug-in		
Input(s):	Main feature(s)	Output(s):
▪ BaaS Gateway Services Interfaces ▪ BaaS entities ▪ Zigbee Gateway specification ▪ Zigbee devices	▪ Connection between BaaS Platform and Zigbee devices. This software package allows the communication between the BaaS architecture implementation and the physical Zigbee devices	▪ BaaS-Zigbee plug-in

Unique Selling Proposition(s):	▪ Software package/Library of methods prepared to be adapted to new Zigbee scenarios
Integration constraint(s):	▪ Profile SmartEnergy v1.0, Zigbee Gateway specification ▪ Java 7 ▪ JSON ▪ REST Services libraries ▪ Requires pre-configuration of the Zigbee devices
Intended user(s):	▪ Programmers using the BaaS Architecture approach and implementing a BaaS Platform ▪ Building Automation Systems Integrators
Provider:	▪ everis Spain SLU
Contact point:	▪ Raúl Juanes Pascual – raul.juanes.pascual@everis.com
Condition(s) for reuse:	▪ Licensing

Latest update: 03 October 2016

Name: Smart Booking Management Suite		
Input(s):	Main feature(s)	Output(s):
▪ Interfaces of the services (basic and value-added) ▪ Buildings' information (maps) ▪ Users' roles ▪ Location of the devices ▪ BaaS Gateway Services' Interfaces	▪ Set of software applications to manage bookings, rooms, floors and buildings facilities taking into account the users' feedback. A 3D building viewer is integrated for easy and intuitive maintenance. It is possible to configure policies in energy saving into the applications. It supports the control of lights and measurements of temperature, luminance and presence.	▪ Smart Booking Room application ▪ Maintenance application ▪ Feedback application ▪ 3D Building Maintenance Application (integrated or separately)

Unique Selling Proposition(s):	▪ All-in-one suite for managing private or public facilities. Users are involved in the booking process by taking into account their previous feedback through the mobile app developed for that purpose. In addition, a maintenance application is available to provide entire control/vision/action to the Facility Manager on the bookings and devices (sensors/actuators) by way of a 3D building viewer to be bought separately.
Integration constraint(s):	▪ Java 7 ▪ Zigbee SmartEnergy v1.0, Zigbee Gateway ▪ REST Service libraries, JSON ▪ Oracle Database. Express Edition 11g ▪ Hibernate, HTML5, Ajax, Tomcat Server, Spring MVC, Bootstrap ▪ GIS Web and OGC standards ▪ BaaS Gateway Services Interfaces
Intended user(s):	▪ Building Automation Systems Integrators ▪ Private or Public sector clients/stakeholders ▪ Facility Managers ▪ TRL 7 – Software developers (apps) to provide enhancements and new features to the prototype. Collaboration with end users is required.
Provider:	▪ everis Spain SLU and Prodevelop
Contact point:	▪ Raúl Juanes Pascual – raul.juanes.pascual@everis.com (Smart Booking Room, Maintenance and Feedback applications) ▪ Miguel Montesinos - mmontesinos@prodevelop.es (3D Building Maintenance Application)
Condition(s) for reuse:	▪ Licensing

Latest update: 03 October 2016

Name: Heating Anomaly Detection Algorithm		
Input(s):	Main feature(s)	Output(s):
▪ heating installation process data	▪ Algorithm for anomaly detection in heating installation.	▪ Hints to anomalies in heating installations

Unique Selling Proposition(s):	▪ This algorithm for detecting anomalies in heating installations highlights the advantages of “semantic plugin” of algorithmic services in existing building control installations.
Integration constraint(s):	▪ Java 1.7 ▪ eventually JFreeChart (http://www.jfree.org/jfreechart/)
Intended user(s):	▪ Application developers and maintenance engineers working in the realm of building automation software.
Provider:	▪ Kieback & Peter GmbH & Co KG Berlin, http://www.kieback-peter.de/ Code available on https://github.com/meyer-kieback-peter/had_ex1
Contact point:	▪ Ullrich Meyer – meyer-mey@kieback-peter.de
Condition(s) for reuse:	▪ „Creative Commons Namensnennung-Weitergabe unter gleichen Bedingungen Deutschland“ / „Attribution-ShareAlike 3.0 Unported“, Version 3.0 („CC-by-sa 3.0/de“)
	Latest update: 19.10.2016

Name: Virtualized BaaS Gateway		
Input(s):	Main feature(s)	Output(s):
▪ MODBUS	▪ Virtualization platform based on ARM Cortex A7 CPU and KVM capable to split the MODBUS to more virtual instances	▪ Set of virtual service gateways
Unique Selling Proposition(s):	▪ Software package and device driver modification capable to assign devices connected to a common MODBUS bus to distinguished virtual machines	
Integration constraint(s):	▪ CPU based on ARM Cortex A7 or A15 ▪ KVM ▪ Linux	
Intended user(s):	▪ Programmers using the BaaS Architecture approach and implementing a BaaS Platform ▪ Building Automation Systems Integrators	
Provider:	▪ MDS Computer + X-COM BASE	
Contact point:	▪ Karel Slavicek – karel@ics.muni.cz	
Condition(s) for reuse:	▪ Licensing	
	Latest update: 23 October 2016	