ONEM2M SERVICE LAYER PLATFORM

Roland Hechwartner (Deutsche Telekom)
oneM2M TP Vice Chair
Roland.hechwartner@t-mobile.at
oneM2M www.oneM2M.org

© 2015 oneM2M
The Partnership Project

Over 200 member organizations in oneM2M
Purpose & Deliverables

Purpose
To specify and promote an
M2M Common Service Layer

Deliverables
Technical Reports and Technical Specifications
M2M Common Service Layer in a nutshell

• It is a software layer
• It sits between M2M applications and communication HW/SW that provides data transport
• It normally rides on top of IP
• It provides functions that M2M applications across different industry segments commonly need. Those functions are exposed to Applications via IT-friendly APIs.
• It allows for distributed intelligence (device, gateway, cloud apps)
Standardization approach

Use cases
- Automotive
- Home
- Energy
- E-Health

Requirements
- Security & privacy
- Device Management
- Data exchange
- Interworking

Architecture APIs and protocols
- IP communications
- Restful webservices APIs
- Reuse of existing protocols
- Semantics framework (future)

Test and Interop
- Reference points
- Device certification
- Open source

© 2015 oneM2M
oneM2M Architecture approach

**Pipe (vertical):**
- 1 Application, 1 NW,
- 1 (or few) type of Device
- Point to point communications

**Horizontal (based on common Layer):**
- Applications share common service and network infrastructure
- Multipoint communications

![Diagram showing oneM2M Architecture approach]
Common Service Functions

- Registration
- Discovery
- Security
- Group Management
- Data Management & Repository
- Subscription & Notification
- Device Management
- Application & Service Management
- Communication Management
- Network Service Exposure
- Location
- Service Charging & Accounting
Technical Specifications

- Requirements (TS-0002, WI-0001)
- Functional Architecture (TS-0001, WI-0002)
- Common Terminology (TS-0011, WI-0003)
- Service Layer Core Protocols (TS-0004, WI-0009)
- HTTP Protocol Binding (TS-0009, WI-0013)
- CoAP Protocol Binding (TS-0008, WI-0012)
- Management Enablnt - OMA (TS-0005, WI-0010)
- Management Enablnt - BBF (TS-0006, WI-0010)
- MQTT Protocol Binding (TS-0010, WI-0014)
- Security Solutions (TS-0003, WI-0007)
- Service Components (TS-0007, WI-0011)

ftp://ftp.onem2m.org/Work Programme/

Release 1
Design principles

- IP-based, but interworks with specific IP and non IP technologies in the M2M Area networks
- RESTful resource oriented APIs, resources are representations of devices, applications, things and related descriptions, etc.
- Distributed intelligence (device, gateway, edge, cloud)
- Reuse of existing device management frameworks
- Reuse of existing data exchange protocols
- Reuse of existing security
- Reuse of underlying network capabilities such as location, triggering, etc.
- Resource access control policies allows many to many communications framework
- Future proof – ready to add semantics support
- No mandated implementation (Database choice, intelligence location, etc.)
Candidate features for oneM2M next steps

Wide-scale deployment enhancements:
- Industrial Domain Enablement
- Home Domain Enablement
- API versioning
- Plug and play scenarios
- Any lessons learnt from prototypes and deployment

Interworking
- AllJoyn
- Enhance 3GPP interworking

Big data enablement
- Semantics support and use cases
- Ontology, query, reasoning

Testing and interoperability
- Test specifications
- (external) certification

Security
- E2e security
- Group authentication
- Role based security

Application enablements
- App APIs
- App development guide and SDK
- Service profiling
Vielen Dank!

Q&A
# oneM2M Release 1
## Technical Specifications

<table>
<thead>
<tr>
<th>oneM2M</th>
<th>ETSI</th>
</tr>
</thead>
<tbody>
<tr>
<td>TS 0001 v1.6.1 Functional Architecture</td>
<td>TS 118 101</td>
</tr>
<tr>
<td>TS 0002 v1.0.1 Requirements</td>
<td>TS 118 102</td>
</tr>
<tr>
<td>TS 0003 v1.0.1 Security Solutions</td>
<td>TS 118 103</td>
</tr>
<tr>
<td>TS 0004 v1.0.1 Service Layer Core Protocol Specification</td>
<td>TS 118 104</td>
</tr>
<tr>
<td>TS 0005 v1.0.1 Management Enablement (OMA)</td>
<td>TS 118 105</td>
</tr>
<tr>
<td>TS 0006 v1.0.1 Management Enablement (BBF)</td>
<td>TS 118 106</td>
</tr>
<tr>
<td>TS 0008 v1.0.1 CoAP Protocol Binding</td>
<td>TS 118 108</td>
</tr>
<tr>
<td>TS 0009 v1.0.1 HTTP Protocol Binding</td>
<td>TS 118 109</td>
</tr>
<tr>
<td>TS 0010 v1.0.1 MQTT Protocol Binding</td>
<td>TS 118 110</td>
</tr>
<tr>
<td>TS 0011 v1.2.1 Common Terminology</td>
<td>TS 118 111</td>
</tr>
</tbody>
</table>

19-Jan-15 © 2015 oneM2M
Architecture

**Application Entity**
Provides application logic for the end-to-end M2M solutions

**Network Services Entity**
Provides services to the CSEs besides the pure data transport

**Node**
Logical equivalent of a physical (or possibly virtualized, especially on the server side) device
Architecture

**Reference Point**  
One or more interfaces - Mca, Mcn, Mcc and Mcc’ (between 2 service providers)

**Common Services Entity**  
Provides the set of "service functions" that are common to the M2M environments

**Application Entity**  
Provides application logic for the end-to-end M2M solutions

**Network Services Entity**  
Provides services to the CSEs besides the pure data transport

**Node**  
Logical equivalent of a physical (or possibly virtualized, especially on the server side) device