Lesson 12
Eclipse IoT Stack for End-to-End Solutions

Eclipse IoT stack

- Eclipse Pi4J, Eclipse Koneki, Eclipse Mihini, Eclipse Krikkit provide Physical cum Data-link and Adaptation layers software.
- Stack also includes Gateway, Network, Transport and Application-Support layers

Implementations and frameworks at Eclipse IoT stack

- Eclipse Wakkamma: LWM2M Clients with LWM2M Server implementations in C
- Eclipse Californium: CoAP Clients. secure DTLS and CoAP server
- An implementation in Java using a Californium-based sandbox server which can register the CoAP clients
- The server (CoAP://iot.eclipse.org:5683) interacts with CoAP clients

Implementations and frameworks at Eclipse IoT stack

- Eclipse Lehshan LWM2M Clients. Secure DTLS and LWM2M sandbox server
- Java implementation of LWM2M for device management in Java and include the DTLS for IoT security
- Sandbox server can register and interacts the LWM2M clients
- The server (coap:// iot.eclipse.org:5684) interacts with the client Web UI and REST API.

Implementations and frameworks at Eclipse IoT stack

- Eclipse Moquette ‘C’ MQTT clients at devices
- An implementation in ‘C’ of the publish/subscribe protocol MQTT using TCP
- A sandbox server (tcp:// m2m.eclipse.org:1883) at cloud/web interacts with MQTT clients at an Application running at a computer/tablet/ mobile phone.

Implementations and frameworks at Eclipse IoT stack

- Eclise Paho MQTT clients at devices and Applications in pub/sub mode using a MQTT Broker
- A Java implementation of the MQTT client and Moquette which uses a Java MQTT broker (m2m.eclipse.org/paho)
- Paho also uses the JavaScript, Lua, Python, dot Net, dot net compact, dot net micro, Windows Phone, Android

Implementations and frameworks at Eclipse IoT stack

- Eclipse OM2M: M2M TEST APIs network, gateway and sandbox server interactions
- An implementation of the ETSI M2M standard, providing a horizontal Service Capability Layer (SCL) that can be deployed in an M2M network, gateway, or device
- M3DA sandbox server (http://iot.eclipse.org:44900) interacts with REST API (http:// iot.eclipse.org/m3da)

Implementations and frameworks at Eclipse IoT stack

- Eclipse Ponte M2M CoAP and MQTT Gateways and MQTTMQTT Broker and MQTTCoAP Broker
- A framework based on Java and OSGi services for IoT and M2M Gateways
- Ponte gateway includes bridge between the M2M/IoT protocols, for example, between MQTT and CoAP protocols using devices to the Web

Implementations and frameworks at Eclipse Jetty

- Eclipse Jetty WebSockets bi-directional communication
- A set of Java methods and annotations for WebSocket objects creation and sessions

Implementations and frameworks at Eclipse Kura

- Eclise Kura Gateways. Services, cloud connectivity, management of device, network configuration and Application
- A set of OSGi based or building IoT and M2M gateways.

Implementations and frameworks at Eclipse Kura

Kura Services

- (i) includes device abstraction
- CAN bus for automotive embedded devices
- Configuration management
- Integrated device cloud functions,

Implementations and frameworks at Eclipse Kura

- (ii) provides application portability
- Modularity
- Application management
- Provisions for built-in OSGi services for IoT apps

Summary

We learnt

- Eclipse IoT Stack also includes Gateway, Network, Transport and Application-Support layers JVM or Eclipse Concierge (a lightweight implementation of OSGi runtime).
- The stack enables the usages of lightweight M2M (OMA M2M standard)

Summary

We learnt

- Stack includes MQTT (OASIS IoT standard), CoAP (IETF IoT standard) and standard network protocols for the IoT Gateway Services for remote management and the applications management.

Summary

We learnt

- Gateway, network, transport, Application support layer software
- Moquette, Eclipse Paho, Eclipse Wakkamma, Eclipse Californium, Eclipse Lehshan, Eclipse OM2M, Eclipse Ponte and Eclipse Kura

End of Lesson 12 on
Eclipse IoT Stack for End-to-End Solutions