Enhancement activities to oneM2M open sources

Prof. Song JaeSeung (oneM2M TP Vice Chair)
Sejong University
(jssong@sejong.ac.kr)

Contents

- Overview on Open Source Software Design and Implementation

- Enhancement activities
  - Adopting new database
  - Drag-and-Drop resource modelling
  - Add bindings
  - Interworking proxies
  - Dashboard

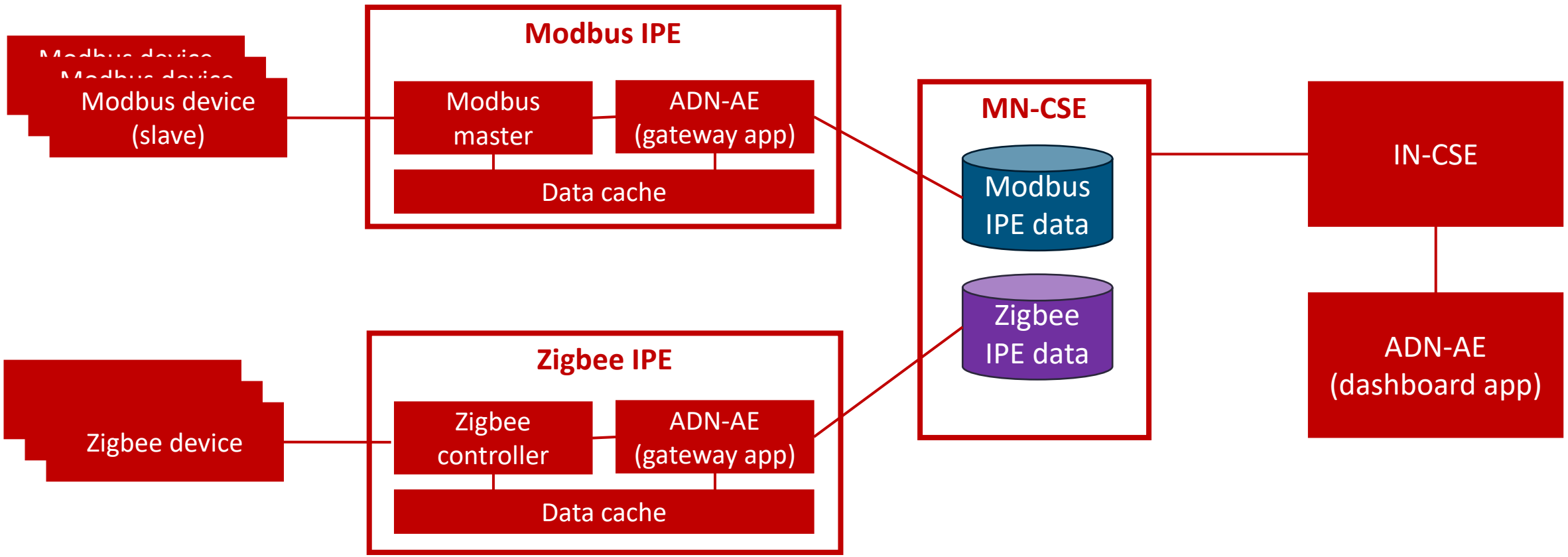
Introduction to Open Source SW Class

- Open source software design and implementation class
  - University: Sejong University
  - Objective: Introduce open source software projects and encourage students to contribute to open source communities (targeting oneM2M open source projects)
  - Duration: 01. Sep. 2023 ~ 15. Dec. 2023 (Open every year in autumn term)
  - Credits: 3 credits (3 hours in a week for 15 weeks)
  - Number of students: 30~40 students (5~8 teams)
- Curriculum
  - Introduction to IoT technologies (definition, history, network technologies, cloud platforms, etc.)
  - oneM2M introduction (architecture, bindings, APIs, main features, etc.)
  - Tutorials from open source communities: ACME (Andreas Kraft, Deutsche Telekom), OM2M (Sherzod, Synchtechno), Mobius (Hyeonseo Son, SJU), tinyIoT (Jieun Lee, SJU) → Installation, configuration, basic running commands, code analysis and review for platform and application
  - Git tutorials (history, overview, basic commands, etc.)
  - Final pitch for the team projects

Topics for OSSD team project

- 2018 ~ 2022
  - Development of IoT application or service using open source projects
  - Given basic sensor and development kits
  - Install oneM2M IN-CSE, develop device (Raspberry pi + sensors), develop web-app
- 2023
  - Adopting PostgreSQL database to ACME
  - **Drag-and-Drop resource modelling for tinyIoT and ACME**
  - Add CoAP bindings to tinyIoT
  - **Modbus Interworking proxy for OM2M**
  - Zibgee Interworking proxy for OM2M
  - Dashboards for Mobius
  - **Digital Twin Elevator**

Modbus / Zigbee Interworking



Drag&Drop Resource Modelling Tool

oneM2M Resource Modelling Tool

Local storage

Save in .json format

Save

Load

Main Page (User Interaction)

Main page

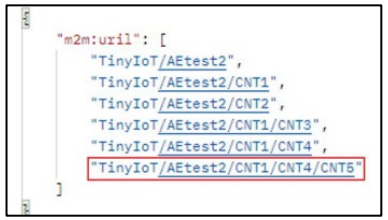
- Drag & Drop Resources
- Modelling oneM2M resources
- Hierarchical resource tree
- Resource validation
- Applicable to various platforms (ACME, Mobius, tinyIoT, etc.)

Resource & Attribute Validation check

Create request

HTTP or MQTT

Retrieve CSE Attributes



Creation Request by Hierarchical Addressing

oneM2M Platforms

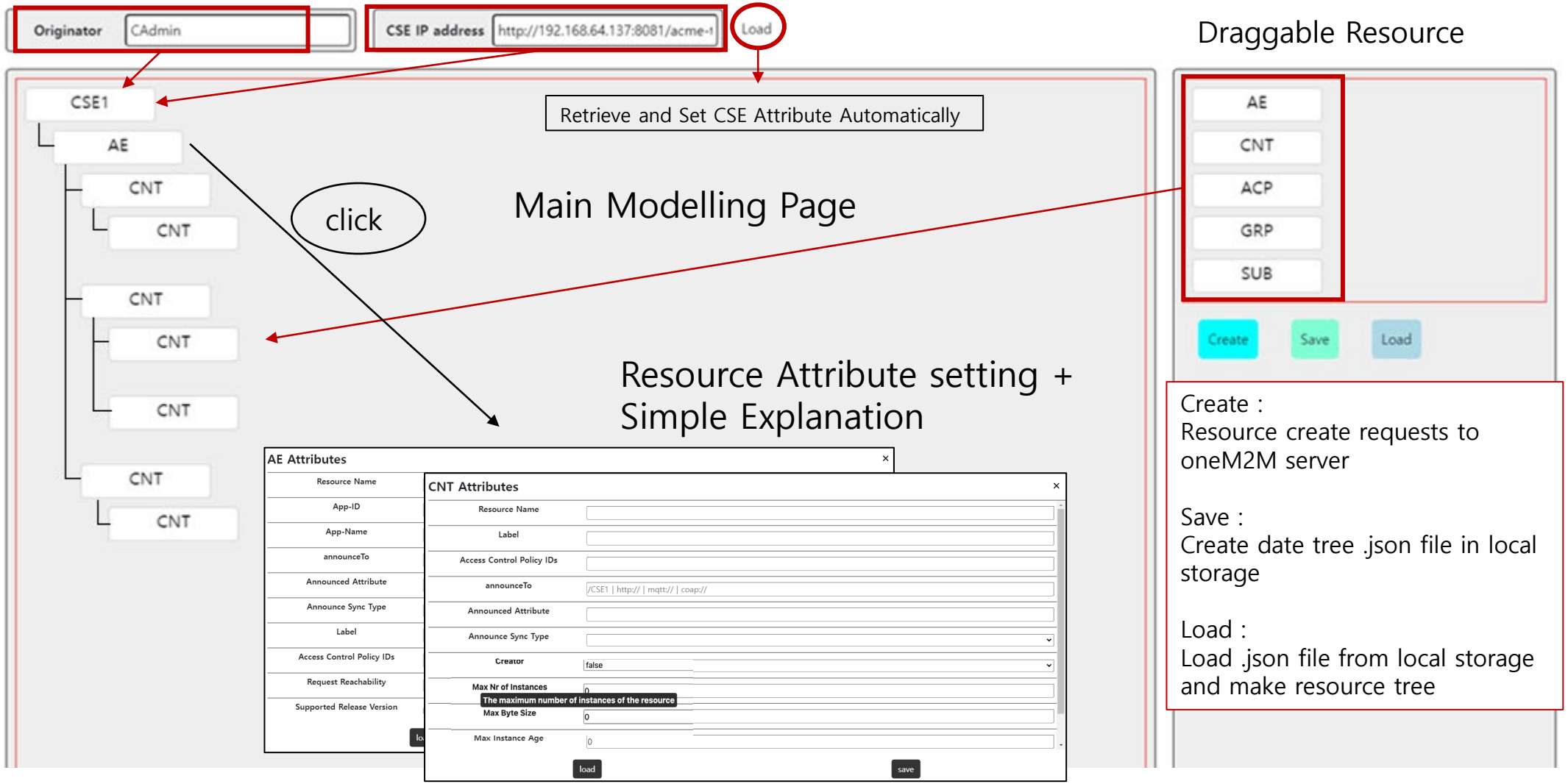
MOBIUS

ACME

OM2M Connecting things

TinyIoT

Drag&Drop Resource Modelling Tool



Create :
Resource create requests to oneM2M server

Save :
Create date tree .json file in local storage

Load :
Load .json file from local storage and make resource tree

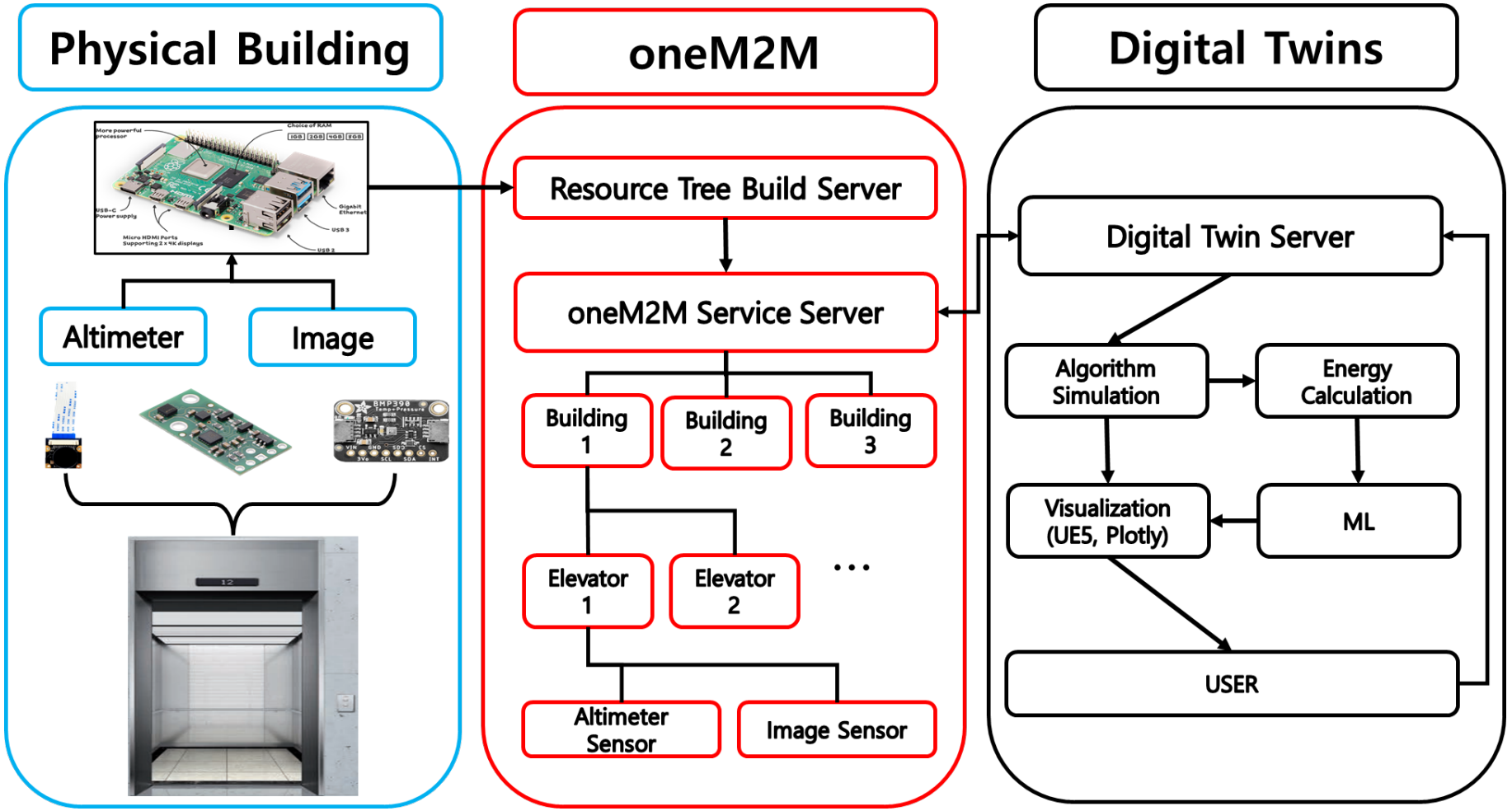
Digital Twin Application

- Development of a smart elevator system for energy saving

- Collect user data
  - Who pressed button?
  - Which level a button pressed?
  - Which level the passenger moved?

- Final expected dataset
  - Statistic of passengers
  - Actual elevator movement with time stamp
  - Energy usage (based on ISO standardized algorithm)

Digital Twin Application



Thank you!