IoT Systems Development Aspects with Executable OPM Models

Hanan Kohen, Dov Dori – Technion

Gordon – Jan 19-20 2021
GOAL:
Enhance model-based systems engineering (MBSE) to include all parts of an IoT system lifecycle.
Model-Based System Engineering
MBSE – Current state and future needs
MODEL-BASED SYSTEMS ENGINEERING (MBSE)

- SE covers the entire system lifecycle
  - Conceptual design
  - Prototyping
  - Testing
  - Manufacturing
  - **Using – Including Operating & Controlling in real time**
  - Maintaining
  - Retiring

- Conceptual modeling is the primary activity required for engineering systems to be understood, designed, and managed

- Using model-based approaches has numerous benefits
Hardware-Software Fusion: The Future of Systems Engineering

- What does SE need in order to remain relevant and drive the IoT systems vision forward?
  - We must use a language and a methodology that cater to modeling and architecting systems with the hardware-software fusion paradigm
Internet of Things (IoT) & IoRT

- Internet of Things (IoT) refers to the interconnection of everyday things, often accompanied by intelligence.
- A platform that interacts electronically, sharing specific information and data with the world around it.
- IoT applications are already being leveraged in diverse domains, such as
  - medical services, smart retail,
  - customer service, smart homes,
  - environmental monitoring, industrial internet.
- IoT is the enabler of IoRT – Internet of Robotic Things
  - Enables collaboration among robots.
What does MBSE need to be agile and IoT-ready?

- A conceptual modeling language and methodology
- Capable of modeling complex systems
- Include all of design and development aspects:
  - Design
  - Tradespace exploration
  - Optimization
  - Prototyping
- Treat hardware and software on equal footing
- Represent in the same and single kind of diagram the three system aspects:
  - Function
  - Structure
  - Behavior
- Include human in the loop
OPM & OPCODE

Web-based tool for modeling in OPM
Only two OPM Things: **Objects** and **Processes**

**Object:** A thing that exists or might exist physically or informatically.

**Process:** A thing that transforms or might transform one or more objects.

Charging changes Battery from depleted to charged.
\[ \text{OPM} = \text{OPD} + \text{OPL} \]

**OPL**
- Global Warming is physical and systemic.
- Global Warming changes Quality Of Living of Human Group from high to low.
- Global Warming changes Temperature of Earth from low to high.
- Global Warming requires Greenhouse Gas Set.
- Global Warming affects Atmosphere and Earth.
- Solar Heat initiates Global Warming, which consumes Solar Heat.
- Global Warming yields Escaped Heat.
- Industrial & Private Activity is physical and environmental.
- Human Group handles Industrial & Private Activity.
- Industrial & Private Activity yields Greenhouse Gas Set.

**OPD - Object Process Diagram**

**OPL - Object Process Language**
Optimal Light Power Consumption System, $s$, is physical.
User is physical.
Electrical Power Consumption Level of User can be low or high.
Led Bulb is physical.
Room Surroundings Light Intensity is physical and environmental.
OPCloud main features

- **Cloud-based Application**
  - Ability to work from anywhere, any time

- **Collaboration**
  - Simultaneous work of multiple distributed users, one editor at a time with built-in edit right transfer

- **Connectivity**
  - with to other systems (DOORS, ARAS, PTC...) using OSLC

- **Latest Web Dev. Technologies:**
  - Firebase, Angular, Rappid...

- **Correctness-by-Construction**
  - via context sensitivity for choice of links and other features

- **Backward Compatibility**
  - for OPM models prepared with OPCAT
Optimal Light Power Consumption System Example
Optimal Light Power Consumption System Example

An example of a system including:

- Conceptual modeling
- Quantitative modeling
- Tradespace exploration
- Optimization
- Connections to external software systems (MATLAB/SIMULINK)
- Connecting to real hardware (sensors, microcontroller)
Optimal Light Power Consumption
System Textual Overview

• Initially, CDS sensors of the system measures the light intensity of the room.
• Then the microcontroller calculates the needed power to supply to the LED according to an algorithm to have the suitable light intensity for the room.
• Then, the amount of electrical power is delivered to the LED.
Optimal Light Power Consumption: OPM system model, top-level diagram
Tradespace exploration: Finding the optimal sensor configuration
Simulating - Value validation

- \([V_{\text{min}}..V_{\text{max}}]\) or \((V_{\text{min}}..V_{\text{max}})\)
- We can combine range with default : \([V_{\text{min}}..\text{dflt}..V_{\text{max}}]\)
- Multi range: \([\text{min1}..\text{max1}] , [\text{min1}..\text{max1}]\)
- Textual value range: "Present", "$\text{Absent}"
- "hard validation" or "soft validation"
Optimization

- Searching for the best system configuration on multi-objective problems
- Using Design-Structure Matrix (DSM) based methods
- Applying graph database querying using Neo4J integrated into OPCloud
Incorporating Hardware-in-the-Loop (HIL)
Hardware-in-the-Loop (HIL) simulation

- Hardware-in-the-Loop (HIL) simulation is a method for developing and testing embedded systems.
- Entails embedding parts of the real hardware during the system development.
- Allows to thoroughly test the complex control device in a virtual environment.
- Provides advantages of:
  - Earlier testing in the development process
  - Reduction of testing costs
  - Increase of test coverage
  - Better test repeatability.
OPM and Hardware-in-the-Loop (HIL)

- Methodical Approach to Executable Integrative Modeling (MAXIM) is integrated into and extends ISO 19450:2015 OPM.

- MAXIM enables seamless quantitative computations, embedded within the qualitative conceptual model.

- Using MAXIM, we incorporate hardware, specifically sensors and actuators, connected to Arduino into OPCloud computation and simulation capabilities.
The System Operating System Idea

- OPCloud
- Node Server
- Arduino Uno
- Arduino-OPCloud Controlled Device
- Connector Set
- Sensor Set
- Programmable Chip
- Arduino IDE
- Device Control Engineer
- Arduino Program
OPCloud-HIL Architecture

- The OPCloud extension can connect to any external hardware or software system:
  - Arduino, MATLAB/Simulink...
Model Execution with HIL
Conclusion
Extending MBSE to Use, Operating and Controlling

- The current practice is for MBSE to cover primarily the conceptual phase: problem formulation, requirements engineering, design
- We are extending MBSE from a conceptual modeling approach to cover the objective of the system – value providing through use.
- We seamlessly combine high-level, abstract conceptual and computational modeling with low-level, near real-time hardware, such as sensors and an Arduino controller
- The same OPM modeling paradigm is used all the way from highly abstract functions to down-to-earth operations.
- Answering all design needs and aspects.
- This enables performing all major system lifecycle stages, from modeling and architecting, via testing and implementing, to using, operating and controlling the actual system.
- This is a paradigm shift in the role of MBSE and its centrality
Thanks for listening!

Visit our Lab site:
http://esml.iem.technion.ac.il/

Experience OPCloud, Cloud-based OPM modeling:
https://www.opcloud.tech/

Contact us:

Tel: +972-77-8872441

OPCloud@technion.ac.il

Enterprise Systems Modeling Laboratory 121 Bloomfield, Technion - Israel Institute of Technology Haifa, 3200003 Israel