OS3P

S MART , S AFE & S ECURE P LATFORM

BUILDING FLUID IOT SYSTEMS

Empowering MDE for IoT: Papyrus 4 IoT

Saadia Dhouib, PhD, CEA LIST
François Lefevre, CEA LIST
Shuai Li, PhD, CEA LIST
Angelo Corsaro, PhD, Prismtech
Fred Rivard, PhD, MICROEJ
Frédéric Rivière, PhD, MICROEJ

list
cea tech

MICROEJ

PRISMTECH

S3P Project

- Collaborative Research and Development (R&D) project that is building a Smart, Safe & Secure platform to provide software tools for the design of Internet of Things (IoT) systems.

- S3P is funded by the French Government program "Nouvelle France Industrielle". With a combined budget of 43 million euros

- S3P project consortium includes 14 industrial organizations* and 8 suppliers of technologies**.

- The CEA List Institute of the CEA Tech, with the support of both PrismTech and MicroEJ, has built the first demonstrator of the project using Papyrus.

*Airbus ; **Alstom** ; **Altran Connected Solutions** ; **AXA France** ; **Continental** ; **Eolane** ;**NXP Semiconductors** ; **Sagem** ; **Schneider Electric** ; **Sorin** ; **STMicroelectronics** ; **SurTec** ; **Thales**.

CEA Tech ; **Krono Safe ; **MicroEJ** ;**Prismtech** ; **Prove & Run** ; **Sysgo** ; **Telecom ParisTech** ; **TrustInSoft**.

About this S3P Collaboration

- **S3P tech. providers involved in this demonstrator**

  - CEA LIST / Saadia DHOUIB, PhD - François LE FEVRE - Benoit MAGGI - Shuai Li, PhD - Patrick TESSIER, PhD - Sara TUCCI, PhD
    (firstname.lastname@cea.fr)

  - PrismTech / Angelo CORSARO, PhD
    (angelo.corsaro@prismtech.com)

  - MicroEJ / Frédéric RIVIERE, Fred RIVARD, PhD
    (firstname.lastname@microej.com)

- **Three S3P technologies integrated into one IoT demo**

  - Papyrus
    - System Modeling/Supervision ➔
      UML (OMG standard),
      IoT-ML (will be standardized in the frame of Marte 2.0)
  - Vortex (DDS)
    - Data communication ➔ inter-devices communication
  - MiroEJ
    - Virtualization ➔ safe universal execution platform

- **Managing heterogeneous applications, development environments, devices and communication technologies**

    - **Models and design methods describing reliable interworking of heterogeneous systems (e.g. technical / economical / social / environmental systems)**

- **Identifying and monitoring critical system elements: detecting critical overall system states in due time**
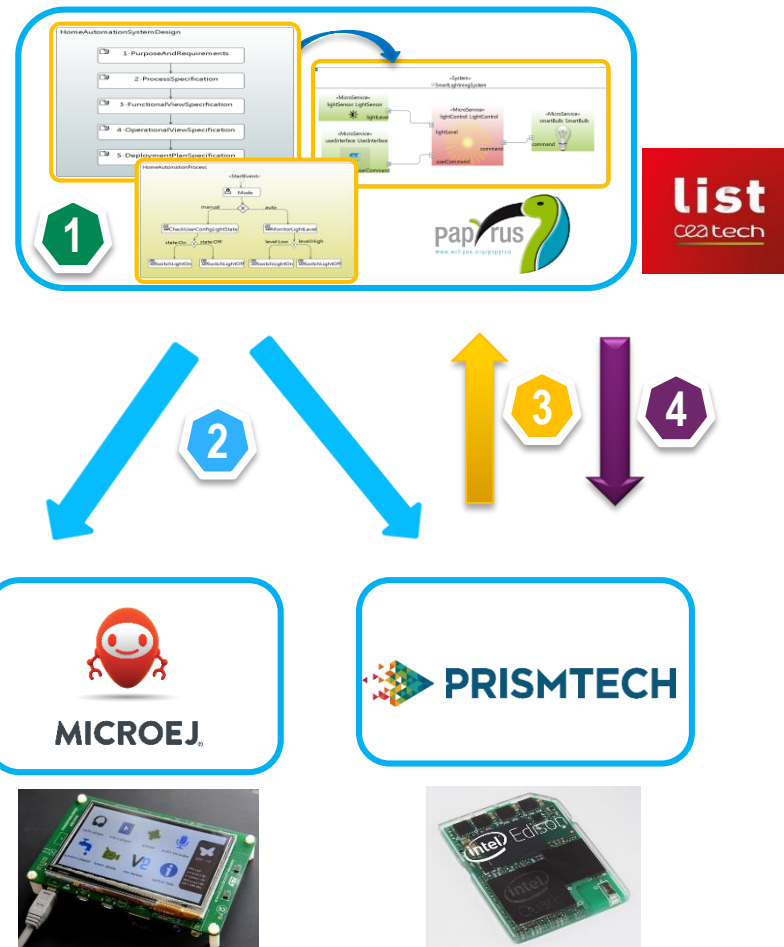
[*] *Internet of Things: From Research and Innovation to Market Deployment*. IERC-European Research Cluster on the Internet of Things, 2014

- **Model-Driven Engineering for IoT**

  - **Abstraction**
    - Models to specify and design complex IoT system: **Models@Design-Time**
    - Models reflecting the running system: **Models@Runtime**
    - **Microservices** as a unit of (de)composition, deployment and management
  - **Deployment**
    - Deployment of IoT systems models
    - Semi automatic code generation, usage of generic Langages (C, Java, asm, …), Design Patterns, …
  - **Models@Runtime**
    - Supervision of the running system using design time models
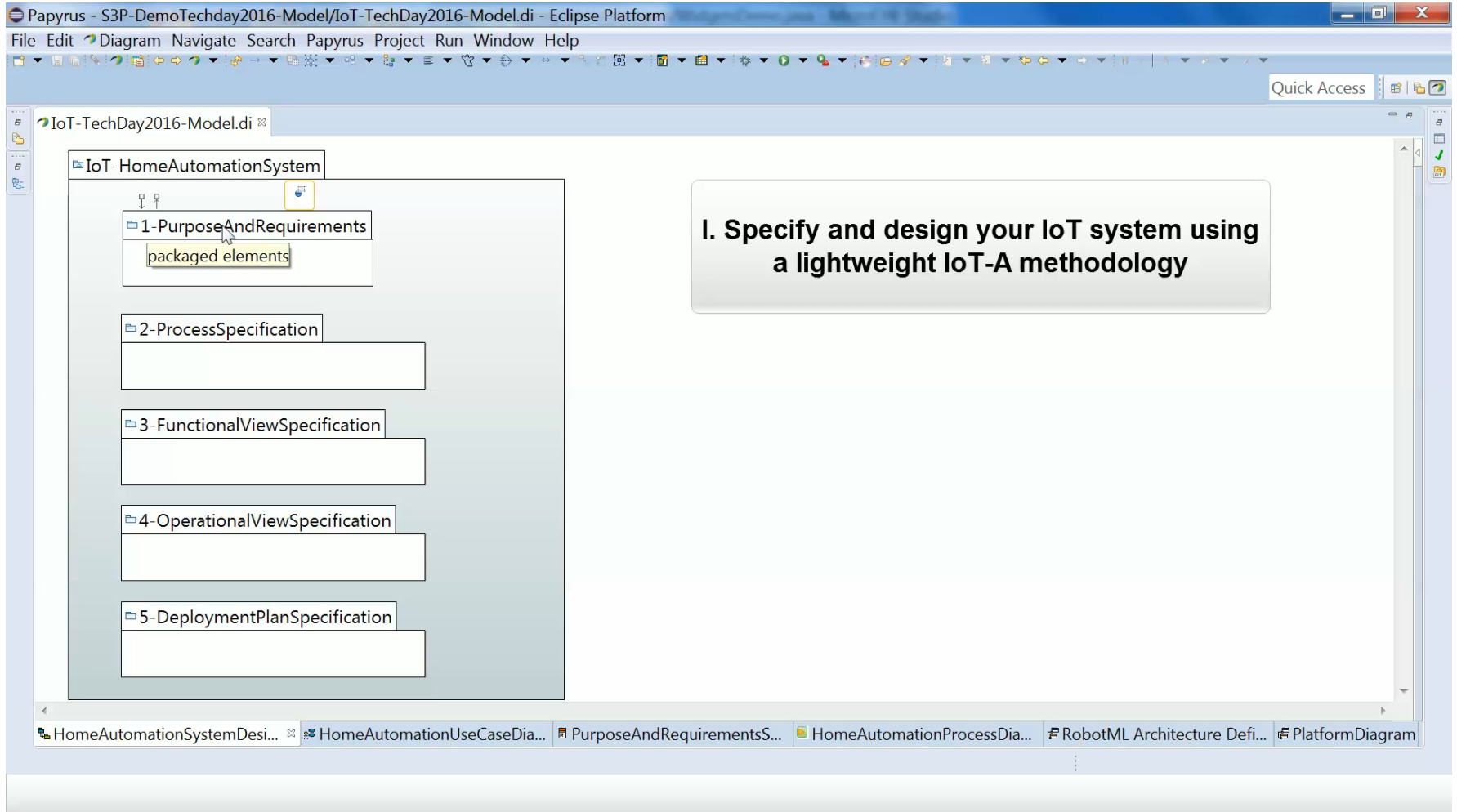
A Short Course on MDA Specifications. B. Selic, In: INFWEST Seminar on Model Driven Software Engineering, Pirkkala, Tampere, Finland (2006)

Models@run.time - Foundations, Applications, and Roadmaps, Nelly Bencomo, Robert B. France, Betty H. C. Cheng, Uwe Aßmann, Lecture Notes in Computer Science, Springer 2014

- **In this demo, you will see how the integration of Papyrus4IoT-MicroEJ-Vortex enables you to…**

1. Specify and design your IoT system using a first lightweight system methodology

2. Deploy your IoT application on anything, from an embedded device to a cloud-based VM

3. Use Models@Runtime to *monitor* system states

4. Use Models@Runtime to *modify* the system's behavior at runtime in response to changes within the system

A first lightweight IoT methodology

Embedded videos are also available at: https://www.youtube.com/watch?v=7JqAPRH0bfU

Embedded videos are also available at: https://www.youtube.com/watch?v=7JqAPRH0bfU
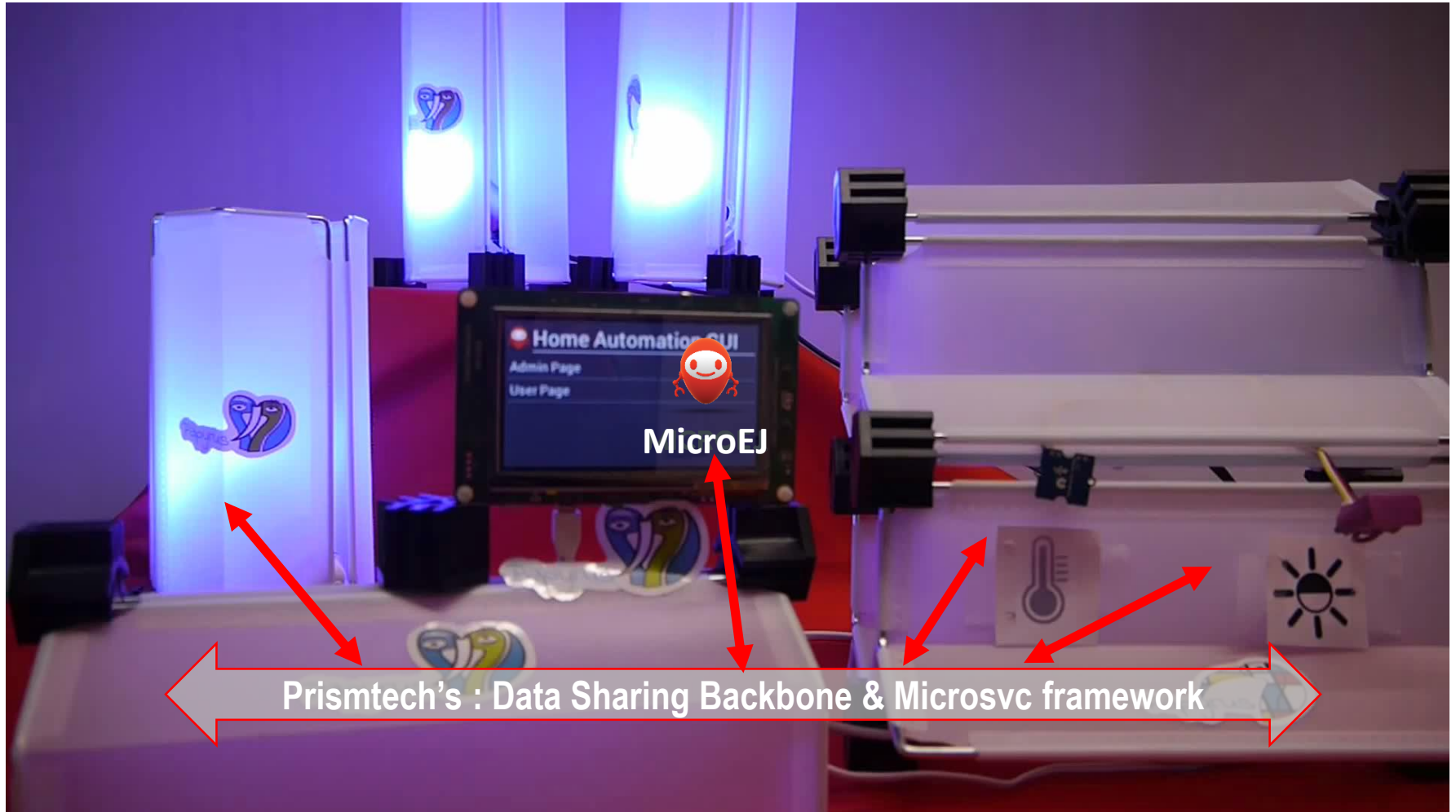
- **Once the system is deployed, you can use the same design models to supervise the running system**

  - **Use Models@Runtime to *monitor* system states**

  - **Use Models@Runtime to *modify* the system's behavior at runtime in response to changes within the system**

Use Models@Runtime to supervise the IoT running system

Embedded videos are also available at: https://www.youtube.com/watch?v=7JqAPRH0bfU