Jan Jürjens, Kurt Schneider (Hrsg.): Software Engineering 2017,
Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2017 75

An Execution System for Self-healing Workflows in Cyber-physical Systems

Ronny Seiger[1], Steffen Huber[2] and Thomas Schlegel[3]

Abstract:

Within the Internet of Things software controlled sensors, actuators and smart objects enable a close coupling of the cyber and physical worlds. Introducing workflows into these cyber-physical systems (CPS) promises advantages regarding automation, resource utilization and flexibility of control systems. In this work, we present PROtEUS–an integrated system for process execution in CPS. PROtEUS integrates components for event processing, data routing, dynamic service invocation and human interaction. It is the basis for executing self-healing and model-based workflows that assure cyber-physical consistency by applying the MAPE-K feedback loop.

Keywords: Process Execution, Workflow System, Self-healing, Context, Internet of Things

Motivation and Approach

An increased level of automation in CPS and smart spaces (e. g., smart homes, smart offices or smart factories) leads to a better utilization of resources, saving of energy, improved efficiency and quality as well as to an increase of comfort. Workflows are a suitable means for achieving a higher level of automation in CPS. However, the application of BPM technologies in cyber-physical environments is still in its infancy and facing new challenges. With this work, we aim at addressing parts of these challenges in the BPM context including the processing of complex sensor data, the integration and dynamic selection of actuators and services, the inclusion of humans and cyber-physical synchronization.

The *PROtEUS* process execution system for CPS [SHS16] features the following software components to facilitate the introduction of workflows into CPS (cf. Fig. 1):

- Process Engine: The Petri net based process engine is the core component of the system. It executes process instances based on the meta-model described in [Se15].
- Process Manager: The process manager component is responsible for the management and control of the process execution via client applications.
- CEP Engine: The complex event processing engine is able to gather and process event data from a large number of sensors via sensor specific data adapters. Special types of events in a workflow define patterns within the stream of sensor data that trigger the activation of process steps in the workflow.

[1] Technische Universität Dresden, Software Technology Group, D-01062 Dresden, ronny.seiger@tu-dresden.de
[2] Technische Universität Dresden, SEUS Group, D-01062 Dresden, steffen.huber@tu-dresden.de
[3] Hochschule Karlsruhe, IUMS Group, D-76133 Karlsruhe, thomas.schlegel@hs-karlsruhe.de

- Local Service Platform: The integrated OSGi platform allows for the local deployment of services to access virtual or physical devices.
- Service Invoker: The service invoker component is used for calling remote or local Web services from within a workflow via standard protocols, e. g., REST or SOAP.
- Human Task Handler: The human task handler enables the process based interaction with users via human tasks on interactive clients to provide input or react to errors.
- Semantic Access Layer: This component contains a description of all devices, their capabilities and context in a knowledge base as well as connectors to the corresponding services. This knowledge base can be queried from a process to find and invoke a suitable resource for executing certain context-dependant tasks at runtime [Hu16].
- MAPE-K Feedback Service: This service uses the PROtEUS components to detect and repair inconsistencies between the assumed states of the process execution and the actual physical states. The MAPE-K loop is triggered in parallel to process activities to correlate the effects of the execution with changes in sensor data–**M**onitor this data; **A**nalyze it for expected changes; **P**lan a compensation in case of deviations; **E**xecute these compensation actions. This way *self-healing* and ensuring of *cyber-physical consistency* can be achieved as described in [Se16] and [SHS16].

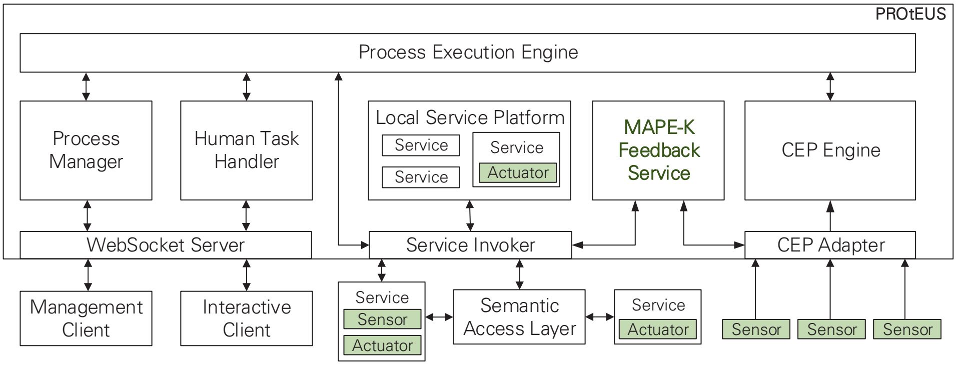

Fig. 1: Architecture of the PROtEUS process execution system [SHS16]

References

[Hu16] Huber, Steffen; Seiger, Ronny; Kuehnert, Andre; Theodorou, Vasileios; Schlegel, Thomas: Goal-Based Semantic Queries for Dynamic Processes in the Internet of Things. International Journal of Semantic Computing, 10(02):269–293, 2016.

[Se15] Seiger, Ronny; Keller, Christine; Niebling, Florian; Schlegel, Thomas: Modelling complex and flexible processes for smart cyber-physical environments. Journal of Computational Science, 10:137–148, 2015.

[Se16] Seiger, Ronny; Huber, Steffen; Heisig, Peter; Assmann, Uwe: Enabling Self-adaptive Workflows for Cyber-physical Systems. In: International Workshop on Business Process Modeling, Development and Support. Springer, pp. 3–17, 2016.

[SHS16] Seiger, Ronny; Huber, Steffen; Schlegel, Thomas: Toward an execution system for self-healing workflows in cyber-physical systems. Software & Systems Modeling, 2016.