Platform Overview
Contents

• From Vision to Platform
  • From Activities to Requirements
  • Platform Design Process
  • Creating the Platform Architecture
  • Platform Architecture

• Kinds of Services within the Platform
• Service Meshes and Applications
• Distributed deployment and execution of Services
• Component Interaction
• Code Generation in the Platform

• Platform Documentation
From Vision to Platform

AAS = Asset Administration Shell / Verwaltungsschale
IDS = International Data Spaces
Following the Industrial Internet Reference Architecture (IIIRA) we:

1. Created a Usage view, capturing the Entities, Roles and Activities we need and wanted to enable in the platform.

2. Based on the elicited Activities and on the elicited technical requirements we derived the requirements of the Functional view of the platform.
1. **Start with an open-minded pre-survey**, e.g., surveys on IIoT platforms *

2. **Create a joint vision**: Identify further (research-) relevant topics.

3. **Stabilize the vision by detailed surveys**, i.e., assure the gaps through focused surveys *

4. **Create a usage and a functional view**:
   - Use joint vision as scope
   - Elicit the requirements in two complementing views.
   - Compare views and assess differences.

[Link](https://www.thinkmind.org/index.php?view=article&articleid=softeng_2022_1_20_90004)
Creating the Platform Architecture

- We identified 18 entities and 19 roles within the platform.
- Based on these we were able to elicit and describe 43 edge-related technical activities and 27 AI-related activities.
- These activities formed the base for 141 top-level requirements and the 181 sub-requirements.
Platform Architecture
Kinds of Services in the Platform

Platform-provided Service:

• Generic, parametrized services
• Can be applied to various setting

Application-specific Services (often AI-Services):

• Designed for use with a specific application

Hybrid Services:

• Generic Services that use plug-Ins or add-ons to perform application-specific tasks
Services within the platform can be chained, e.g. orchestrated, into Service Meshes, forming an Applications as follows:

The basic approach is to chain generic (parametrized) platform-services to get data from a source, link this to further processing services, for example an application-specific AI-service and pass the output to a data sink via another generic (parametrized) platform-service.
Abstracted view of an Application based on a Service Mesh

The platform git-hub repository provides several different examples of applications.
Distributed Execution of Services

Services in the platform can be deployed and executed in a distributed manner both locally (D1) and remote (D2). Their coordination is ensured by the central platform AAS server.
The IIP-Ecosphere platform generates much of the necessary "glue" code, such as configurations, connectors, service- and application-descriptions and AAS descriptions of components automatically.

The automated code generation is employing the Integrated Variability Modeling Language (IVML).

Code generation also performs a consistency and dependency constraint check, only allowing valid configurations.
IIP-Ecosphere website: https://www.iip-ecosphere.de/

Platform handbook:

Requirements documentation:

Usage View documentation:

Link to the platform git-hub: https://github.com/iip-ecosphere/platform
Contact

Dr. Christian Sauer

sauer@sse.uni-hildesheim.de

https://www.iip-ecosphere.eu

@de_iipecosphere