IIP-Ecosphere

Next Level Ecosphere for Intelligent Industrial Production

Further Platform Capabilities - What did we not touch?

Gefördert durch:

Bundesministerium für Wirtschaft und Klimaschutz

Contents

- Some further topics
  - Connectors to external data
  - Generic (platform-supplied) services
  - Containers
  - Deployment plans
  - Monitoring
  - Management UI

© IIP-Ecosphere Konsortium

External Connectors (1)

- Generic, usually platform-supplied “service”

- Two-sided “service” with I/O
- Realizes a “protocol”, e.g., OPC UA, MQTT, AAS
- Auto-integrated, customized into applications

© IIP-Ecosphere Konsortium

External Connectors (2)

```
OpcUaV1Connector myPlcOpcUaConn = {
    id = "PlcNextOpcConn",
    host = opcUaServer,
    port = 4840,
    samplingPeriod = 500,
    security = {…}
    input = {{type=refBy(PlcInput)}},
    output = {{type=refBy(PlcOutput)}},
    inInterface = refBy(PlcInput),
    outInterface = refBy(PlcOutput)
};
```

“Goodies”

- field mapping
- caching
- mocking
- OPC UA companion specs

© IIP-Ecosphere Konsortium

Generic Services

- Typically platform-supplied services, e.g., KODEX, RTSA, etc.
- Implementation handles arbitrary data
- Specialized for application by code generation
- Special care on
  - Configuration model
  - Integration / code generation
  - Implementation of the service (wrapper)
  - IPR / licensing

Call us!

© IIP-Ecosphere Konsortium

Containers

- Containerization of applications is beneficial
  - Required on some devices
  - Managing and removing system and Python dependencies
- If enabled, application instantiation creates containers
- Relies on service dependencies in configuration
- Supports: Docker, LXC

Not trivial: Optimal layering for maximum reuse and minimum transfer

© IIP-Ecosphere Konsortium

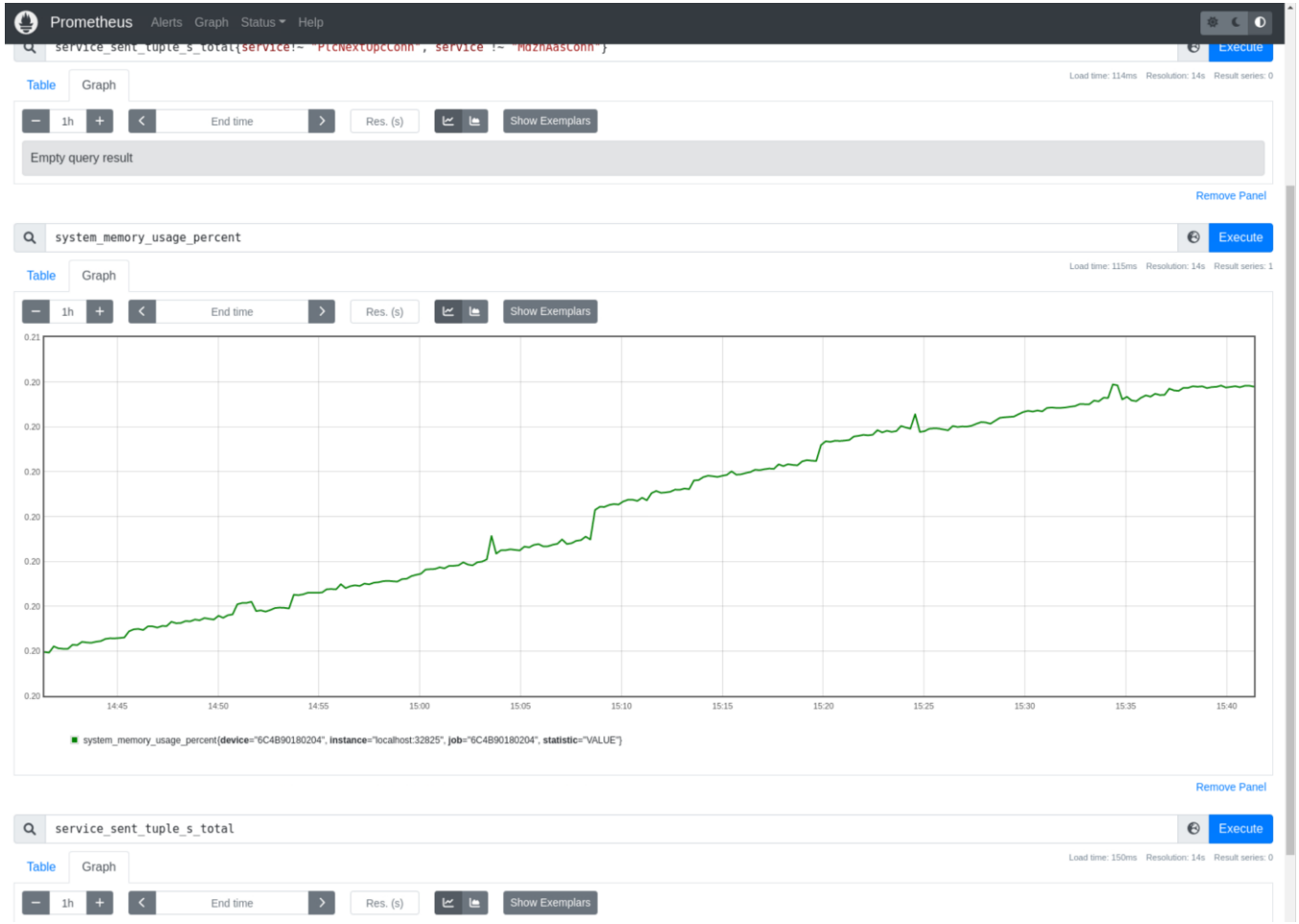
Deployment Plan

- Deploying an application can be a complex task
- Definition in configuration or Yaml file

```
DeploymentPlan plan1 = {
  name = "Separate sources/preprocessing and processing/sink.",
  id = "plan1",
  ver = "0.0.2",
  description = "Deployment on two resources",
  application = refBy(myApp),
  assignments={
      NamedServiceAssignment{resource="a6C4B90180204",
          services={refBy(mySource), refBy(myAnon)}
      }, NamedServiceAssignment{resource="A8741D0D23ED",
          services={refBy(myKi), refBy(myAasSink)}
      }
   },
   ensembles={{leader=refBy(mySource), member=refBy(myAnon)}}
};
```

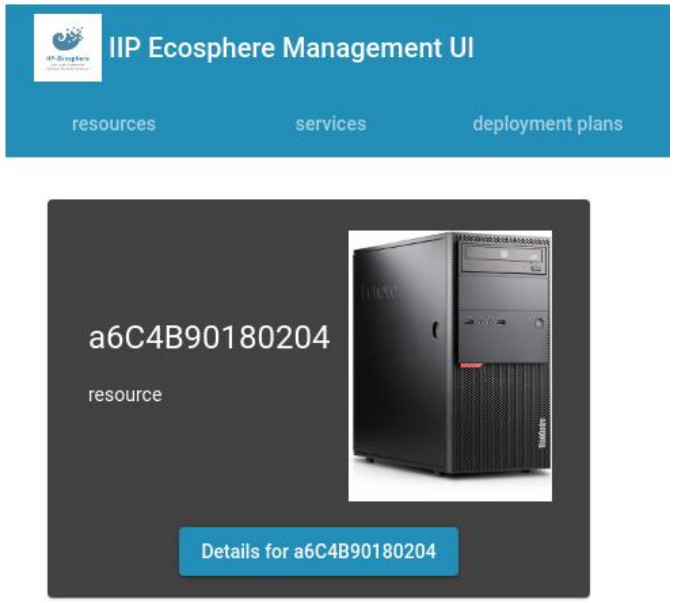
Monitoring

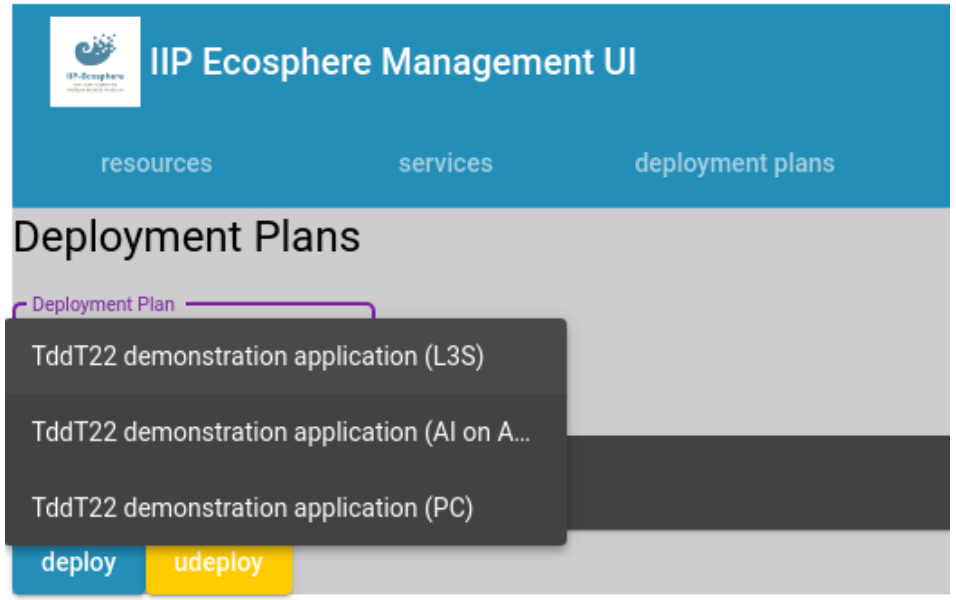
- Runtime monitoring of resource meters provided by platform components
- Service meters generated into code
- Binds against Prometheus
- Can cause alerts

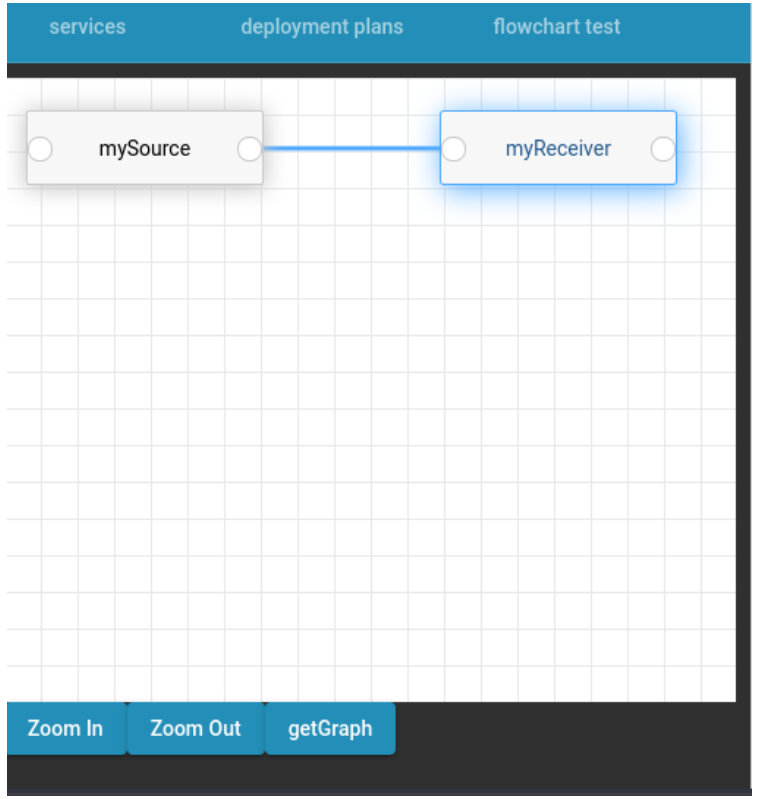


© IIP-Ecosphere Konsortium

Platform Management UI



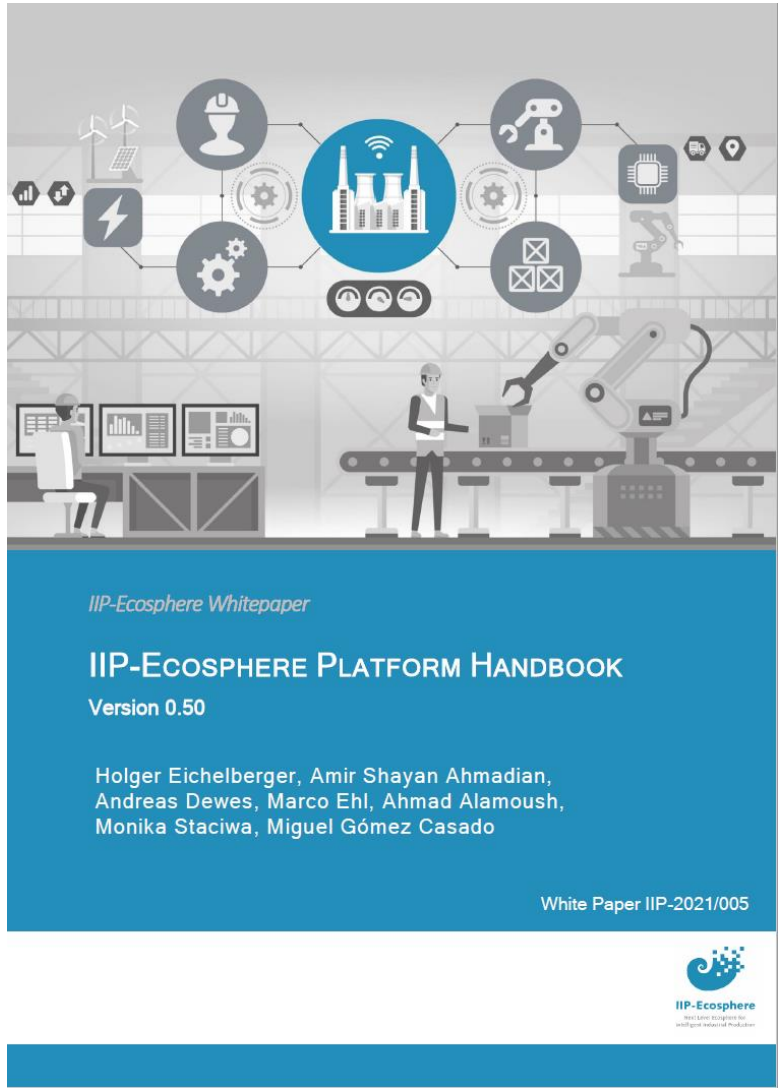




© IIP-Ecosphere Konsortium

More information

- IIP-Ecosphere platform handbook
- Github (https://github.com/iip-ecosphere/platform)
- IIP-Ecosphere (https://www.iip-ecosphere.de/)



© IIP-Ecosphere Konsortium

Contact

Dr. Holger Eichelberger

eichelberger@sse.uni-hildesheim.de

https://www.iip-ecosphere.eu

@de_iipecosphere