



Semantic Evaluation at Large Scale Tutorial

Evaluation Description Definition

Michael Schneider,
Jürgen Bock

FZI Research Center for Information Technology

Outline

- Evaluation Descriptions
 - Evaluation Descriptions in the SEALS Platform
 - Requirements
- The SEALS Evaluation Description Language (SEDL)
- Evaluation Description Bundles
- The Evaluation Descriptions Repository Service (EDRS)
 - Histories and Versions
 - Repository Access

EVALUATION DESCRIPTIONS

5/5/11

3

SEALS
Semantic Evaluation at Large Scale

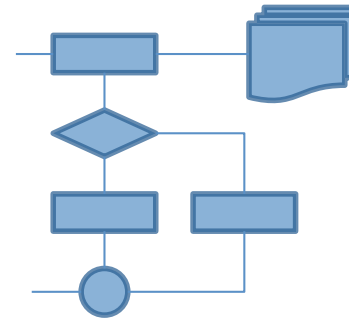
Evaluation Descriptions

Evaluation Description:

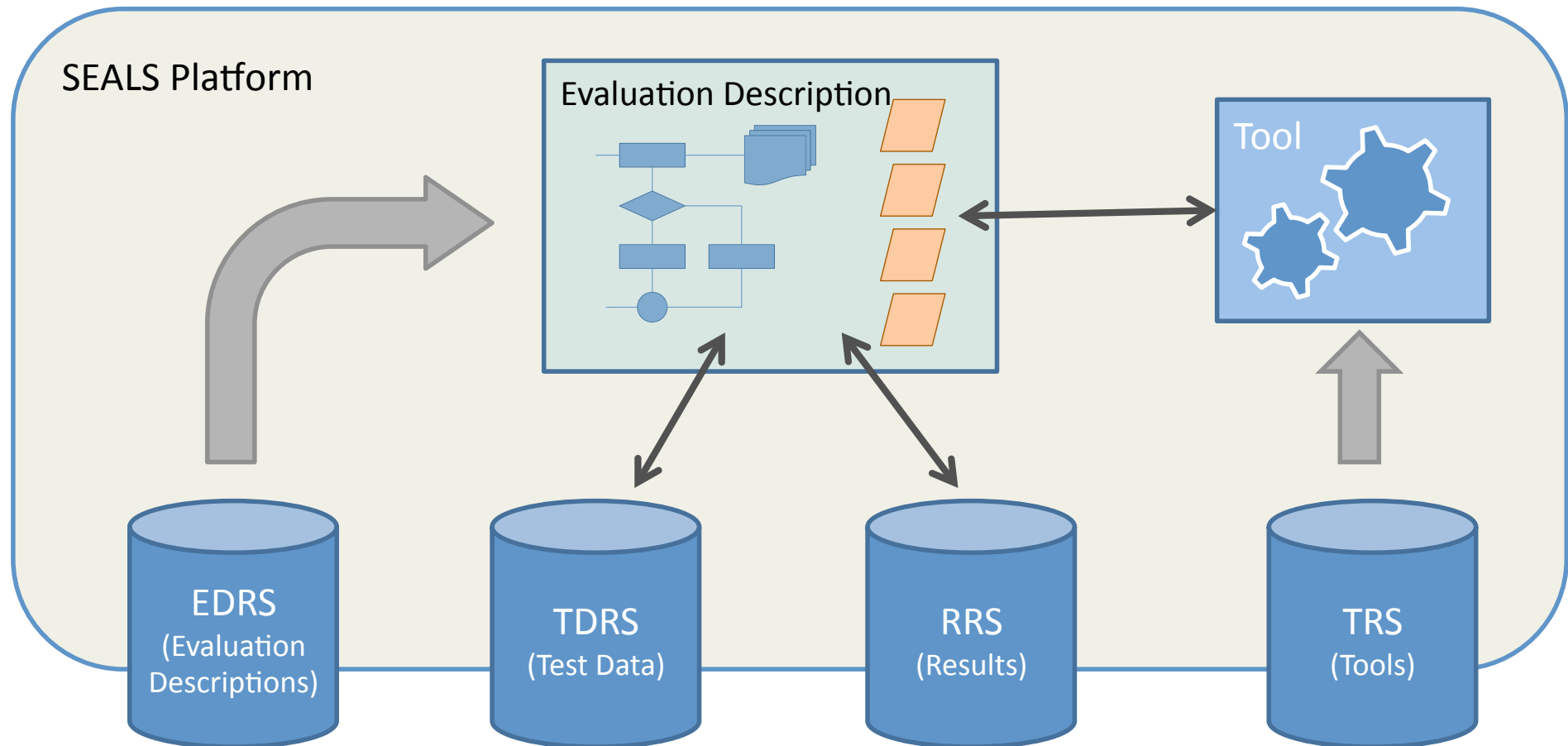
describes how an evaluation is to be conducted

- Access test data
- Invoke tool
- Collect results
- Interpret results
- Store results
- ...

orchestrated by an
evaluation workflow



Evaluation Descriptions in the SEALS Platform



Requirements for Evaluation Descriptions

An Evaluation Description must represent:

- **evaluation workflow**
(platform conformance criteria apply)
- **execution contract**
(test data category, category of results to be produces, ...)
- **custom services**
(service definitions, type definitions, implementations, ...)
- **metadata**
(author, creation time, description, ...)

Evaluation Workflow

- Workflow specified in WS-BPEL 2.0
(Web Services Business Process Execution Language 2.0)
- Platform compliance criteria
 - BPEL workflow must implement generic evaluation description WSDL
<http://www.development.seals-project.eu/resources/res/engine/evaluation/wSDL/{version}/evaluation.wSDL>
 - Pass-through of header information
(header part of BPEL input variable must be passed to header part of each invoked web service)
 - Asynchronous callback
(generic evaluation description WSDL specifies callback operation to be called when BPEL process terminates)

Execution Contract

- What test data, tools and results can a workflow consume?
- What results does it produce?
- Execution Contract specifies categories (in terms of ontology classes)
- Input arguments can be validated prior to execution
→ early check for compatibility

Custom Services

- Not provided by the *SEALS platform*
- Inherent to *evaluation description*
 - Result interpretation service
 - Special utility services
- Custom Services must be shipped with the evaluation description
 - Interface definition (WSDL)
 - Types (XSD)
 - Implementation (JAR)

Metadata

Evaluation Descriptions annotated with

- Name
- Author
- Creation time
- Description

Additionally: metadata managed by repository

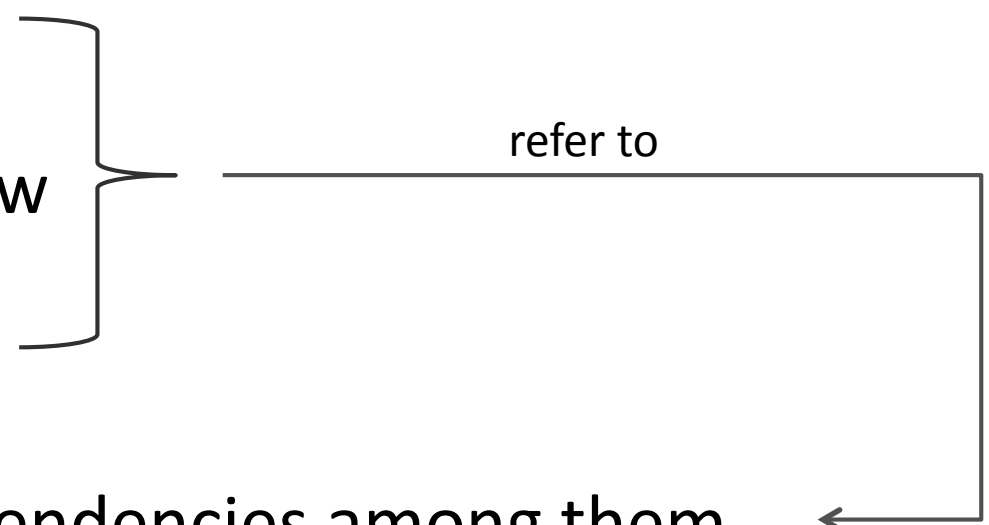
SEALS EVALUATION DESCRIPTION LANGUAGE (SEDL)

5/5/11

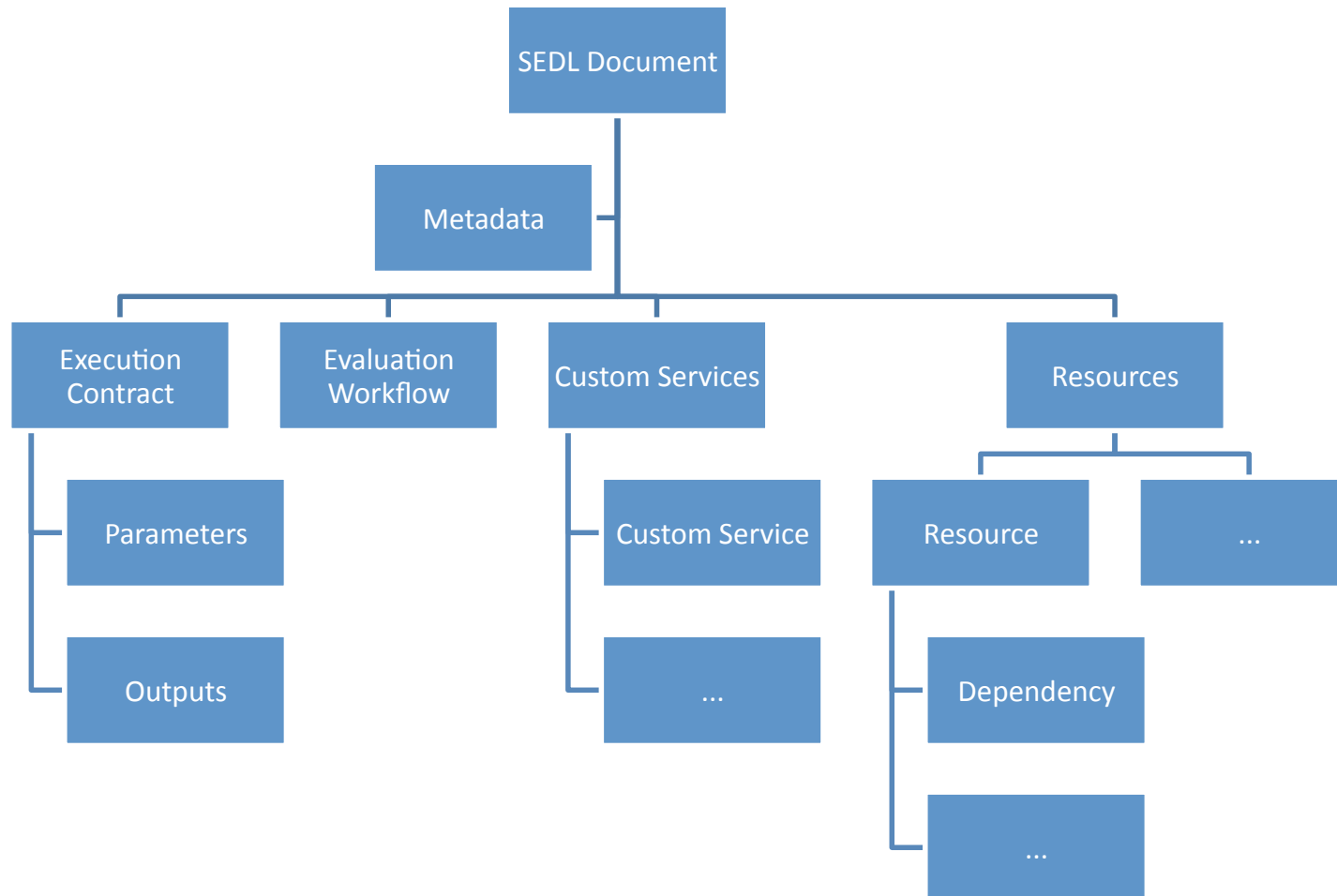
11

SEALS
Semantic Evaluation at Large Scale

SEALS Evaluation Description Language (SEDL)

- XML-based descriptor file format
 - Contents:
 - Execution Contract
 - Evaluation Workflow
 - Custom Services
 - Metadata
 - Resources and dependencies among them
- 
- ```
graph LR; A[Execution Contract] --- B[]; B --- C[Evaluation Workflow]; C --- D[Custom Services]; D --- E[Metadata]; E --- F[Resources and dependencies among them]; F -- "refer to" --- G[]; G --- H[Resources and dependencies among them];
```

# SEDL Overview



# SEDL Content Details: Execution Contract

- Parameter:
  - Name (refers to name of BPEL input argument)
  - Type (refers to ontology class)
  - Resource reference (refers to ontology resource, specifying the type class)
- Output:
  - same as above

# SEDL Content Details: Evaluation Workflow

- Evaluation Workflow:
  - Process name  
(refers to name of the BPEL process)
  - Resource reference  
(refers to BPEL resource specifying the process)

# SEDL Content Details:

## Custom Services

- Custom Service:
  - Documentation (textual description of the service)
  - WSDL
    - Port type (refers to port type for this service provided by the WSDL)
    - Resource reference (refers to WSDL resource specifying this port type)
  - Implementation
    - Class name (refers to the name of the implementation Java class)
    - Resource reference (refers to binary resource specifying this class)

# SEDL Content Details: Resources

- Resource
  - Location  
(pathname relative to the SEDL document)
  - Dependency
    - Either with reference to another resource ...
    - ... or with child resource elements

# EVALUATION DESCRIPTION BUNDLE

5/5/11

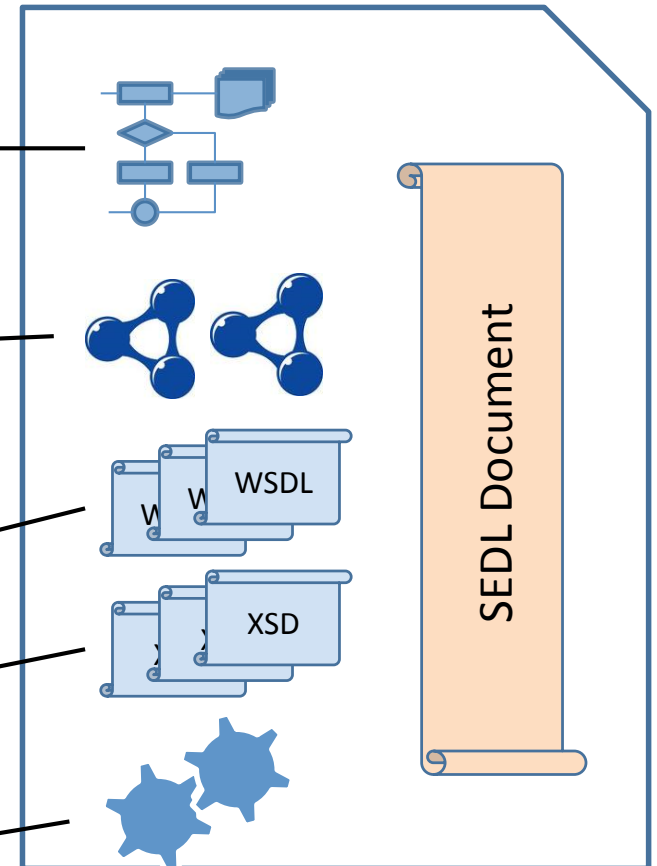
18

**SEALS**  
Semantic Evaluation at Large Scale

# Bundle Contents

Everything is a resource

- Evaluation BPEL workflow
- Execution Contract requires
  - ontologies  
(to define category classes)
- Custom Services require
  - WSDL definitions
  - XSD type definitions
  - Implementation JAR binaries



# **EVALUATION DESCRIPTIONS REPOSITORY SERVICE (EDRS)**

5/5/11

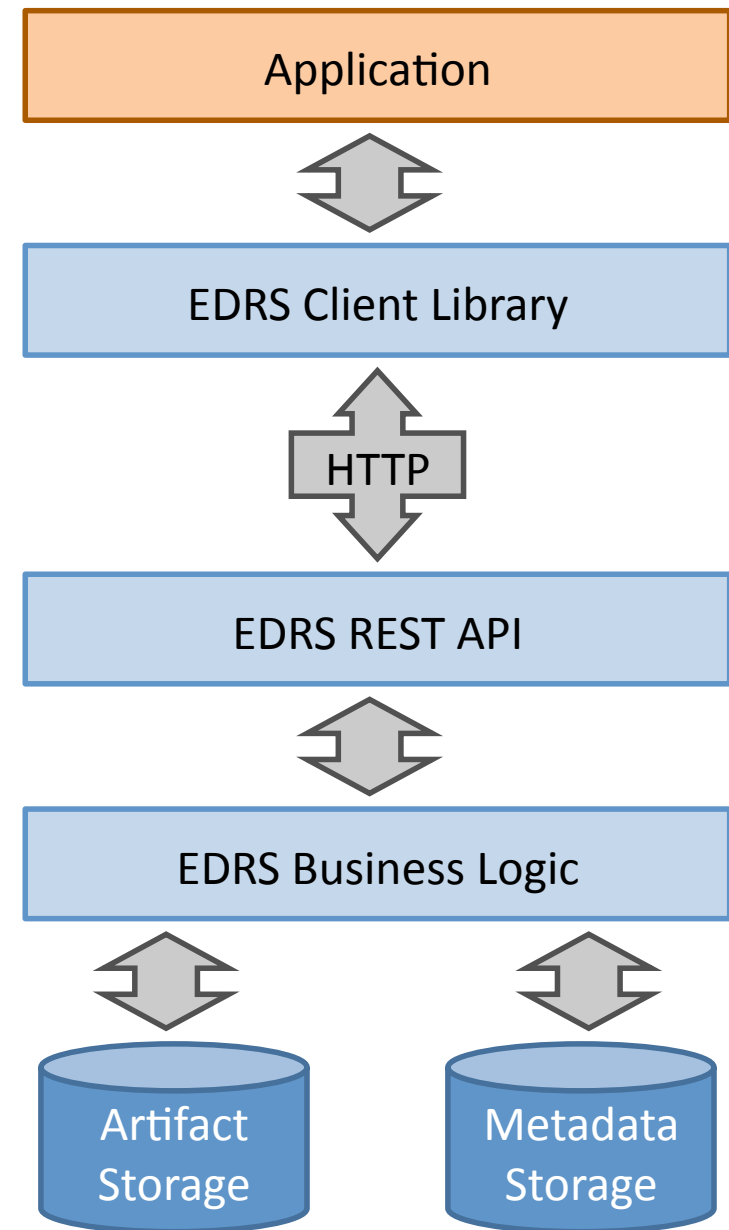
20

**SEALS**  
Semantic Evaluation at Large Scale

# Evaluation Description

## Repository Service: Service Stack

- Client-side Java API
- REST API
- Server-side Java API



# Histories and Versions

- EDRS stores evaluation description histories
- Several versions per history
- Evaluation Description artifact (ZIP bundle) is a version



current version



published version



unpublished version

# Repository Access (REST API)

## HTTP Resources

- **History collection resource** `http://[prefix]/eds`
- **History resource**  
`http://[prefix]/eds/[history]`
- **Version resource**  
`http://[prefix]/eds/[history]/[version]`
- **Publication resource**  
`http://[prefix]/eds/[history]/[version]/pub`

# Repository Access (REST API)

## HTTP Operations

- HTTP GET  
Retrieval of metadata / artifacts
- HTTP POST  
Storage of histories / version artifacts,  
publication of versions
- HTTP PUT  
Update of metadata
- HTTP DELETE  
Removal of histories / version artifacts,  
unpublication of versions

Thanks for your attention!

# BACKUP

5/5/11

26

**SEALS**  
Semantic Evaluation at Large Scale

# SEDL XML Structure

```
<sedl:EvaluationDescription name="..."
 creator="..."
 created="...">
 <sedl:Documentation>
 <!-- documentation of the evaluation description -->
 </sedl:Documentation>
 <sedl:ExecutionContract>
 <sedl:Parameters>
 <sedl:Parameter name="..."
 type="..."
 resourceRef="..."/>
 <!-- ... -->
 </sedl:Parameters>
 <sedl:Outputs>
 <sedl:Output name="..."
 type="..."
 resourceRef="..."/>
 <!-- ... -->
 </sedl:Outputs>
 </sedl:ExecutionContract>
```

```
<sedl:CustomServices>
 <sedl:CustomService name="...">
 <sedl:Documentation>
 <!-- documentation of the custom service -->
 </sedl:Documentation>
 <sedl:WSDL portType="..."
 resourceRef="..."/>
 <sedl:Implementation className="..."
 resourceRef="..."/>
 </sedl:CustomService>
 <!-- ... -->
</sedl:CustomServices>
<sedl:EvaluationWorkflow processName="..."
 resourceRef="..."/>
<sedl:Resources>
 <sedl:Resource xsi:type="..."
 name="..."
 location="...">
 <sedl:Dependency resourceRef="..."/>
 <sedl:Dependency>
 <sedl:Resource .../>
 </sedl:Dependency>
 <!-- ... -->
 </sedl:Resource>
 <!-- ... -->
</sedl:Resources>
</sedl:EvaluationDescription>
```

# BPEL Input Variable

Example of a  
SOAP Body for  
an evaluation  
invocation

```
<soapenv:Envelope xmlns:soapenv=http://schemas.xmlsoap.org/soap/envelope/
 xmlns:n1=http://www.seals-project.eu/resources/res/engine/evaluation/wSDL/v1
 xmlns:n2=http://www.seals-project.eu/resources/res/common/types/xsd/v1
<soapenv:Header>
 ...
</soapenv:Header>
<soapenv:Body>
 <n1:initiate>
 <!--1 or more repetitions:-->
 <n1:argument>
 <n2:name>conformanceTestSuite</n2:name>
 <n2:value>Ontology Engineering Tools Test Suite</n2:value>
 </n1:argument>
 <n1:argument>
 <n2:name>conformanceTestSuiteVersion</n2:name>
 <n2:value>v1.0</n2:value>
 </n1:argument>
 <n1:argument>
 <n2:name>tool</n2:name>
 <n2:value>ProtegeOWL</n2:value>
 </n1:argument>
 <n1:argument>
 <n2:name>toolVersion</n2:name>
 <n2:value>ProtegeOWLVersion1</n2:value>
 </n1:argument>
 </n1:initiate>
</soapenv:Body>
</soapenv:Envelope>
```

# BPEL pass-through of header information

Example of a BPEL assign activity to copy the input header part to the header part of a request variable

```
...
<bpel:assign>
 ...
 <bpel:copy>
 <bpel:from part="header" variable="input"/>
 <bpel:to part="header" variable="GetDataItemRequest"/>
 </bpel:copy>
 ...
</bpel:assign>
...
```

# BPEL asynchronous callback

Example of the asynchronous BPEL callback invocation to notify the caller about termination of the workflow

```
...
<bpel:invoke name="callbackClient"
 partnerLink="client"
 operation="onResult"
 portType="tns:evaluationCallback"
 inputVariable="output"/>
...
```

# Evaluation Description Bundle Structure

Evaluation Descriptions are bundled as ZIP containers, containing the evaluation description document in SEDL format and all resources

## ZIP\_Container

- `./evaluation.sedl.xml`

- `./{any_directory_1}/{resource_1}`

- `./{any_directory_2}/{resource_2}`