Endpoints

This is a set of notes aimed at describing how we can update the Apache Tuscany Java SCA runtime to create an Endpoint structure to represent a configure reference or service binding.

Motivating mail thread http://www.mail-archive.com/dev@tuscany.apache.org/msg04531.html

Motivation

1. The OSOA reference binding list was used to represent resolved/configured bindings which meant breaking the model to resolve references
2. OASIS have a different approach to matching references/service so factoring out endpoint references/enpoints from references/services will allow a switchable algorithm to be deployed
3. New policy model can use endpoint models as policy subjects
4. Support late reference resolution

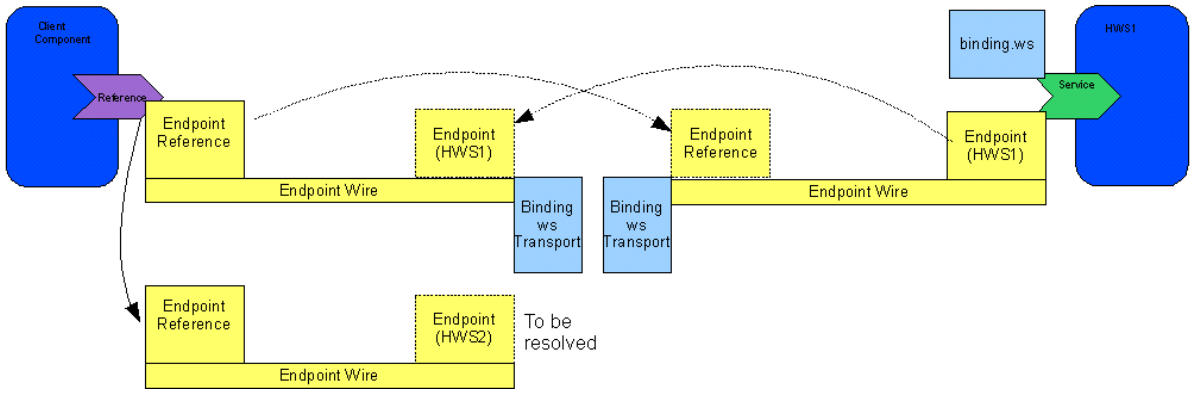
Scenarios

1. Static domain - where all composites are present before nodes are started
    - as now composites will automatically be configured with unwired bindings
2. Incremental deployment - where nodes are started with composites before the domain is complete
    - Endpoint logic will ask the domain for information required to resolve endpoint references with endpoints
3. Service Location Transparency - a future scenario where nodes can be moved
    - Endpoint/binding logic can again be used to ask the domain for new endpoint information
    - Could just let each binding handle it but maybe some advantage to having the enpoint reference handle it by stopping/reconfiguring /starting the binding
4. Dynamic Domain - in the future but some scenarios call for components to be reconfigured at runtime
    - Endpoint reference and endpoint concepts focus the effort of adding removing wires

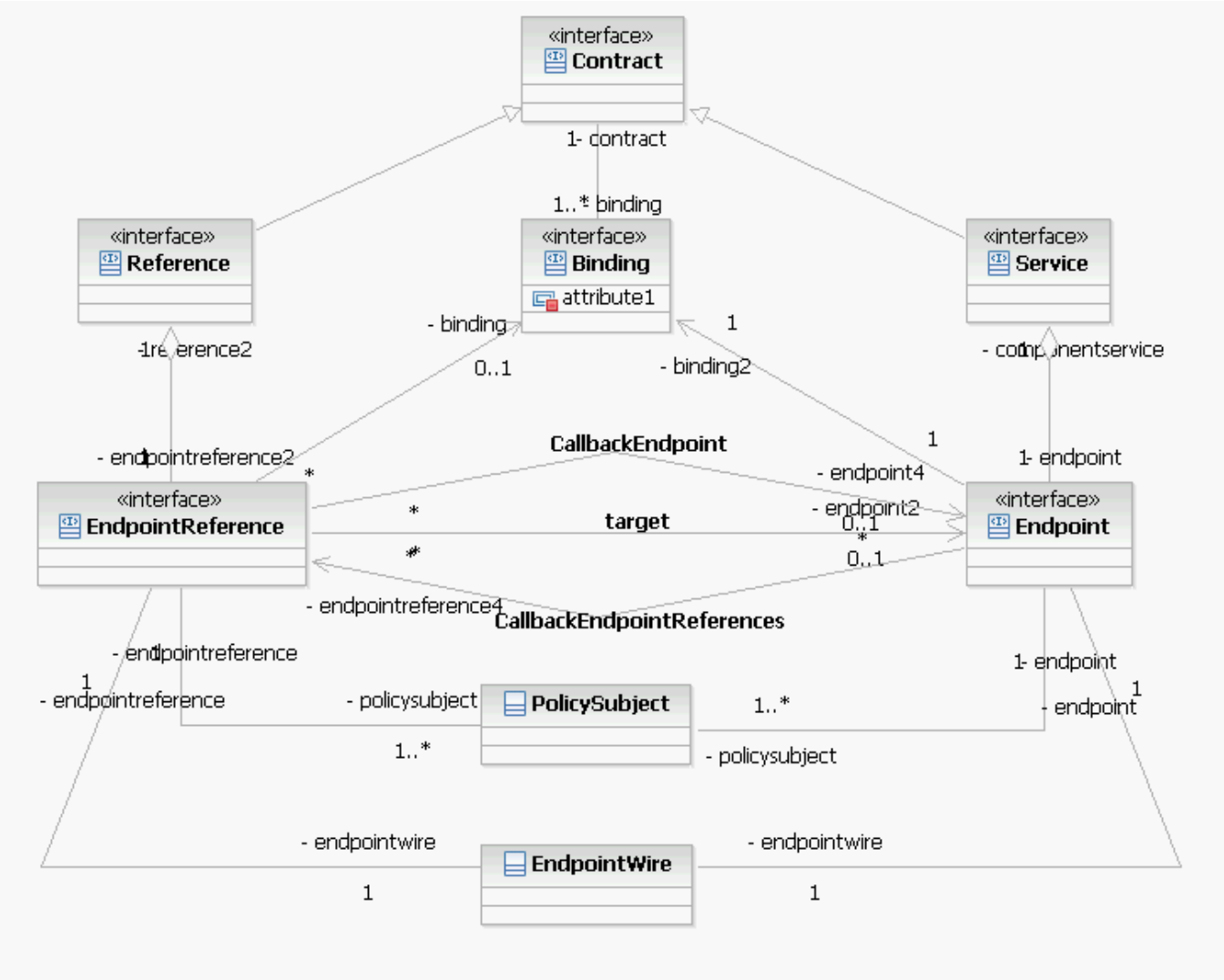
Overview

```
<component name="ClientComponent">
  <implementation.java class="helloworld.HWClientImpl"/>
  <reference name="helloWorldService" target="HWS1 HWS2">
  </reference>
</component>
```

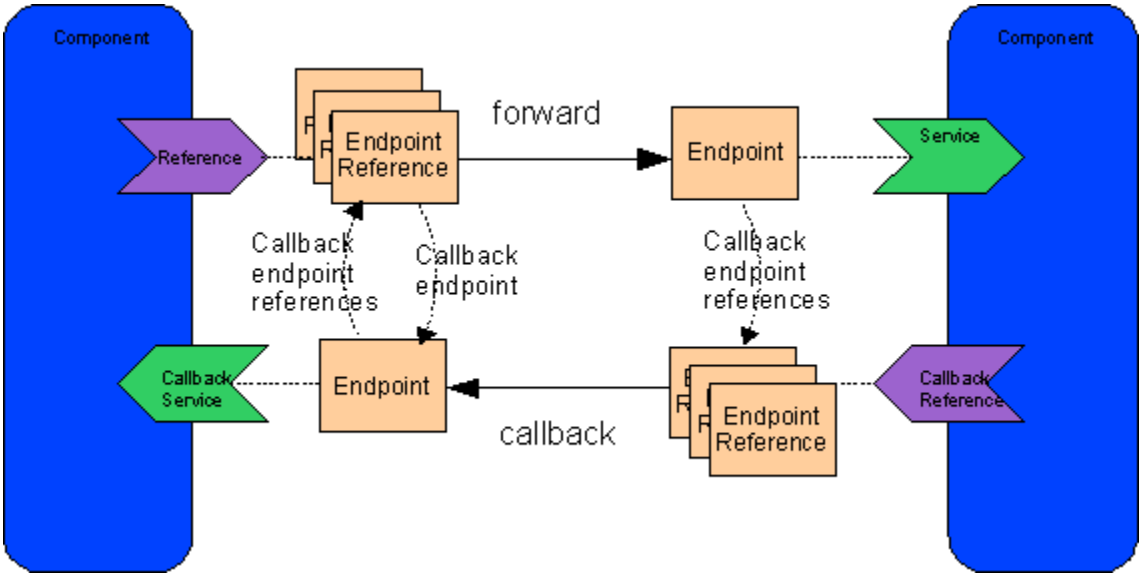
```
<component name="HWS1">
  <implementation.java class="helloworld.HWServiceImpl"/>
  <serivce name="helloWorldService" >
    <binding.ws/>
  </service>
</component>
```

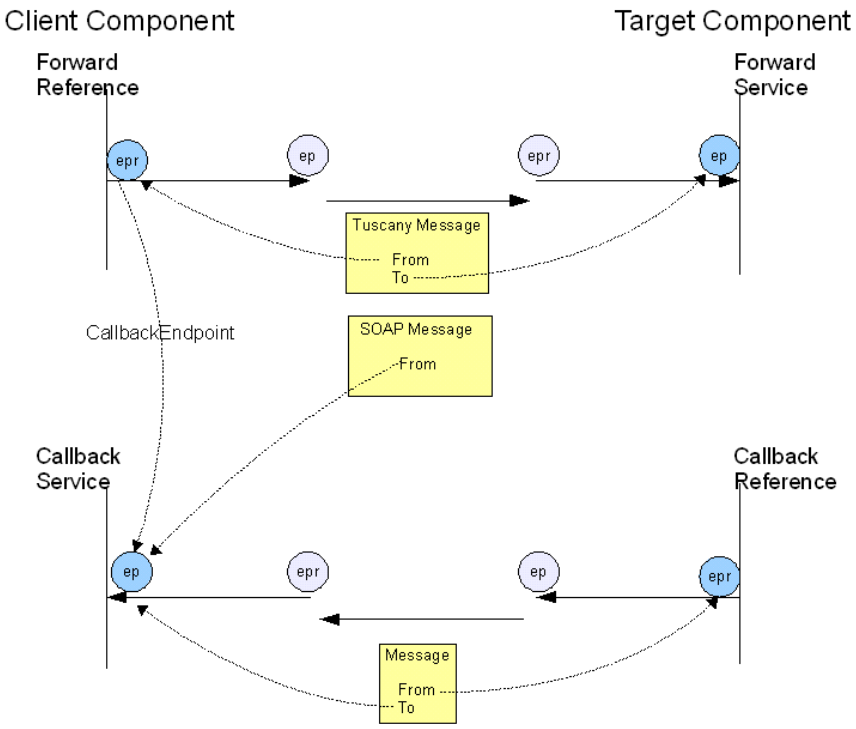


Model

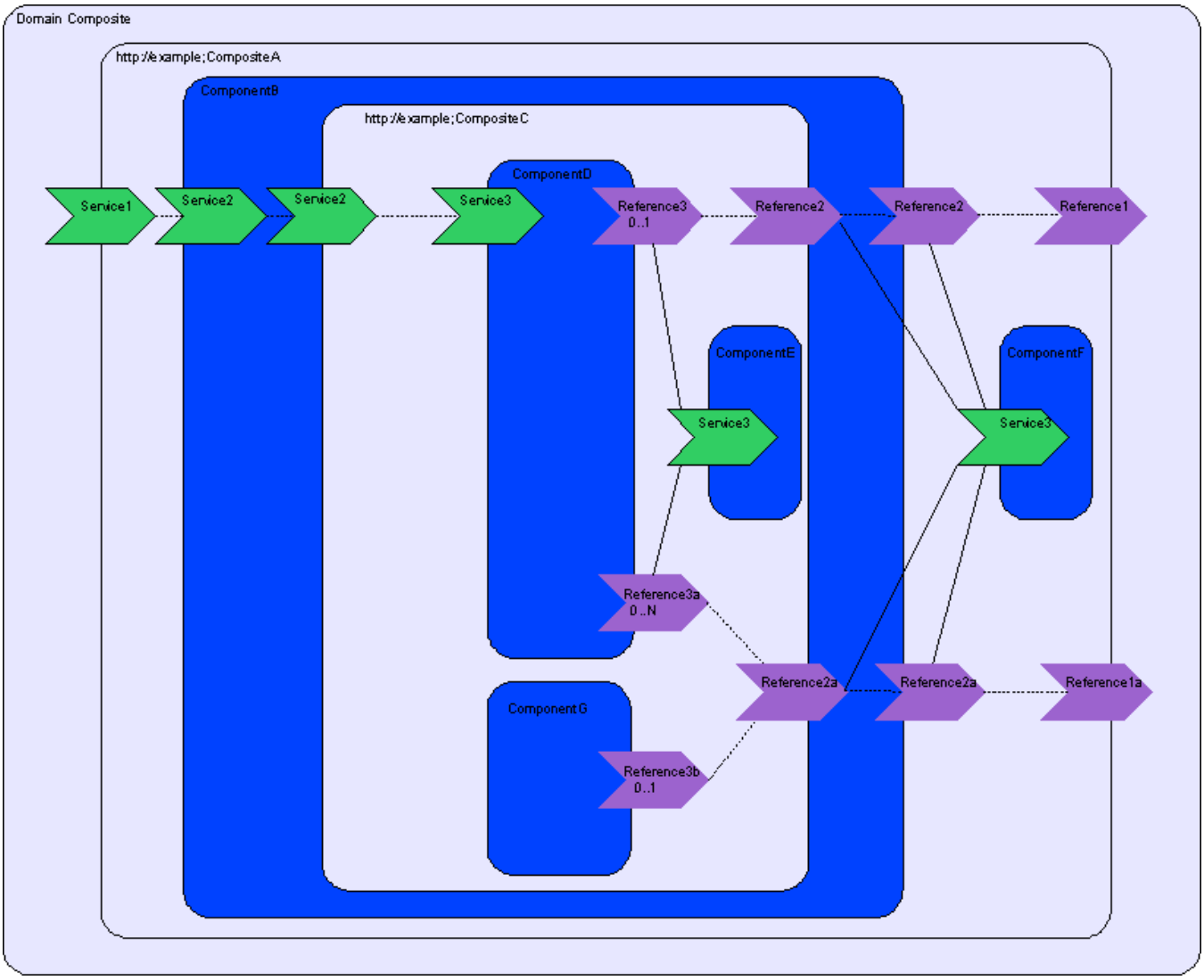

«interface»
Contract
1- contract
1..* binding
«interface»
Reference
«interface»
Binding
attribute1
«interface»
Service
- binding
1
0..1
- binding2
- endpointreference2
CallbackEndpoint
1
- endpoint4
1- endpoint
«interface»
EndpointReference
target
- endpoint2
0..1
«interface»
Endpoint
0..1
- endpointreference4
CallbackEndpointReferences
- endpointreference
1- endpoint
- endpointreference
- policysubject
PolicySubject
1..*
1..*
- policysubject
- endpoint
- endpointwire
- endpointwire
EndpointWire
1
1

Callback


Component
Component
Reference
Endpoint Reference
forward
Endpoint
Service
Callback endpoint references
Callback endpoint
Callback endpoint references
Callback Service
Endpoint
callback
Endpoint Reference
Callback Reference


Client Component
Target Component
Forward Reference
Forward Service
epr
ep
ep
epr
Tuscany Message
From
To
CallbackEndpoint
SOAP Message
From
Callback Service
Callback Reference
ep
epr
ep
epr
Message
From
To

Promotion



Endpoint and EnpointReference creation algorithm

	EndpointReference	Endpoint
Location	Leaftmost component reference	leaftmost component service
Multiplicity	One for each binding on a leaf reference + promoted references	One for each binding on a service
Interface	From the reference If none then from the implementation	From the service If none then from the implementation
Binding	If target then from service Else from reference	Binding on service If none then binding from promoted service If none then binding.sca
Intents	Sum of intents on references at all levels	Sum of intents on services at all levels
Policy Sets	Use attached policy sets If none then use policy sets attached to impl reference	Use attached policy sets If none then use policy sets attached to impl service

Results expected in itest/builder

Scenario	CAS1	CBS2	CCS2	CDS3	Endpoints at CB	Endpoints at CD
1 Binding	binding.ws				Ignored in SCA v1.1?	Ignored in SCA v1.1?
2 Binding		binding.ws			CD/S3/binding.sca	
3 Binding			binding.ws		CD/S3/binding.sca	
4 Binding				binding.ws	CD/S3/binding.sca	

Detailed Model

The EndpointReference and Endpoint have a number of fields and a status which indicates what they are connected to and how they will be processed. Both structures are externalizable. In the Endpoint case this allows a distributed registry of endpoints to be maintained. In the EndpointReference case this allows ServiceReferences to be serialized between remote components as called for in the SCA specifications.

TBD

Old Material

Scenario	CAR1	CBR2	CCR2	CDR3	EndpointReferences at CD
5 Binding	binding.ws uri=				Ignored in SCA v1.1?
6 Binding		target=CFS3			CD/R3/target=CFS3
7 Binding			target=CFS3		CD/R3/target=CFS3
8 Binding				target=CES3	CD/R3/target=CES3

Scenario	CAR1a	CBR2a	CCR2a	CDR3a	EndpointReferences at CD & CG
9 Binding				target=CES3	CD/R3a/target=CES3 CB/R3b/target=CES3
10 Binding		target=CFS3		target=CES3	CD/R3a/target=CES3 CD/R3a/target=CFS3 CB/R3b /target=CFS3
11 Binding		target=CFS3		target=CES3	TBD as 10
12 Binding		target=CFS3		target=CES3	TBD as 10
13 Binding				target=CES3	TBD as 9

Creation and Activation

Event	Action
Create/Start Node	Read Model Resolve model Build model Activate Runtime Start Runtime
Message through reference	(if not resolved) EndpointReference resolution Build Reference Chains
Service Not Available	TBD - possible future event
Add Service	TBD - Possible future event
Remove Service	TBD - Possible future event

Action	Description
Read Model	read composite file into in-memory model
Resolve Model	resolve composite against other artifacts in the composite and domain
Build model	create the appropriate relationships between parts of the model bases on SCA validation rules, Inludes: • Endpoint creation • EndpointReference creation
Activate Runtime	Create the runtime infrastructure. Includes: • EndpointReference resolution • EndpointReference wire creation • Endpoint wire creation
Start Runtime	Start all runtime providers
EndpointReference resolution	Locate the services that endpoint references refer to
Build Reference Chains	use the wire configuration to build the runtime interceptor chains

EndpointReference States

State	EndpointReference	Endpoint
Wired - service specified but unresolved	unresolved	unresolved
Wired - service resolved - binding not matched	resolved	unresolved
Wired - service resolved - binding matched	resolved	resolved
Unwired	resolved	resolved