



Focus on Celtix ESB

Adrian Trenaman, Principal Consultant, IONA Technologies.

adrian.trenaman@iona.com

Introduction

- This presentation is about Celtix – an open source Java ESB
 - Hosted on ObjectWeb, with IONA Technologies as a major contributor.

- Roadmap:
 - What is Celtix, and what problems does it solve?
 - Celtix training, support and consultancy from IONA Professional Services
 - Celtix development roadmap
 - Summary

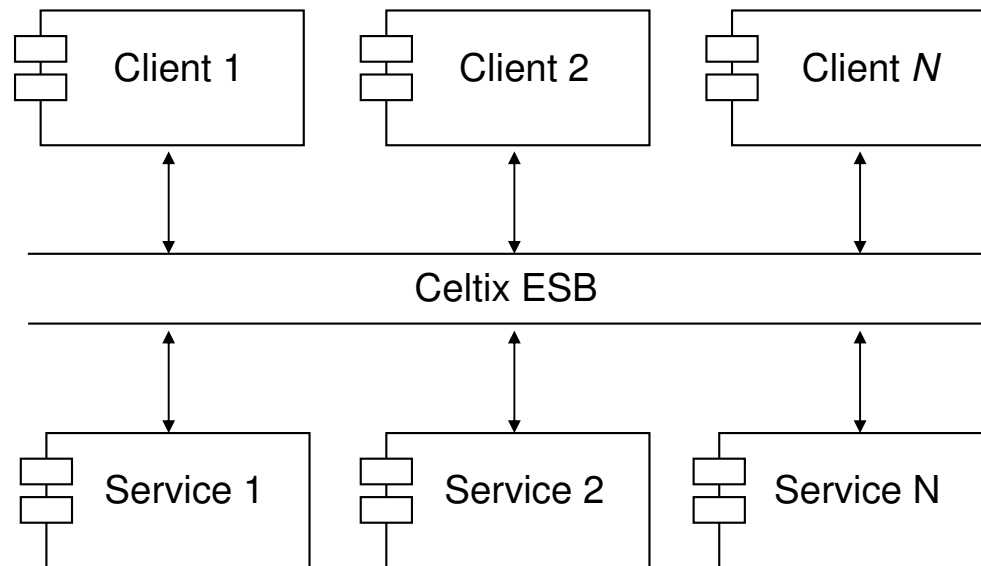
What is Celtix?

What is Celtix?

- Celtix is an open-source Java ESB
- Celtix is hosted by ObjectWeb, an open-source foundation that focuses on middleware technology.
 - IONA Technologies is a major contributor to the Celtix project.
- The name “Celtix” is pronounced “sell-ticks”.
 - The word “Celtic” (usually pronounced “kell-tick”) is typically used to refer to something originating from Brittany, Scotland, Wales or Ireland.
 - The “ix” follows the “Unix” and “Linux” naming tradition.
 - There are a number of sports teams that go by name “Celtics”, namely:
 - Boston Celtics – an American basketball team (also an American baseball team)
 - Glasgow Celtix – a Scottish football team
 - ... some people like the sporting reference, but it is not intentional.

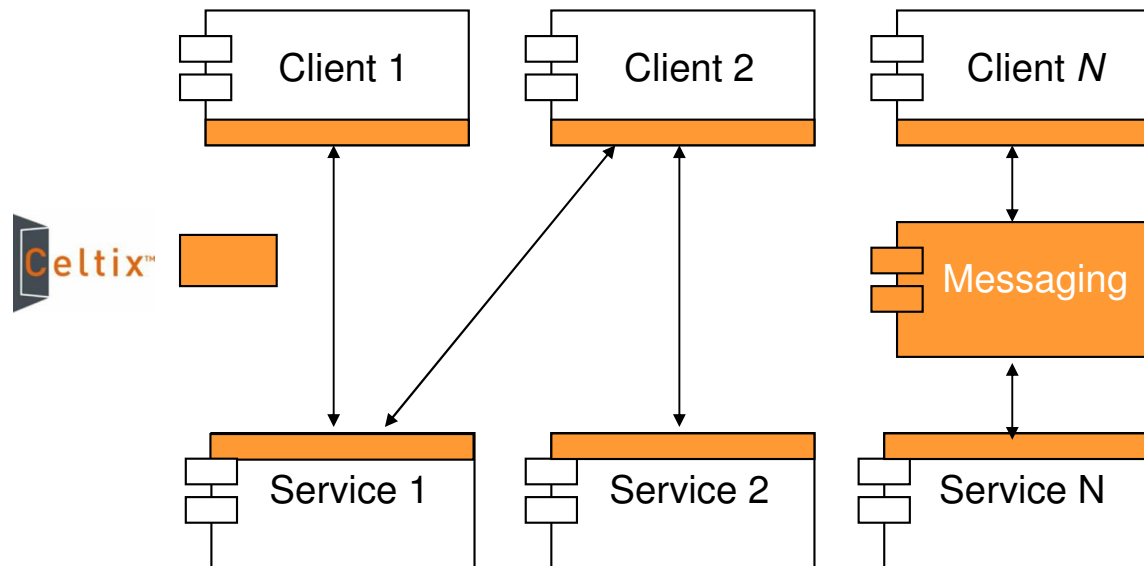
What is Celtix? (cont')

- Use Celtix to build an enterprise service bus, connecting clients (consumers) to business services.



What is Celtix? (cont')

- Keep in mind that a bus is *not* a messaging hub.
 - Celtix supports both synchronous point-to-point and asynchronous messaging technologies
- The bus is *distributed* over the network.



What kind of problems does Celtix solve?

- Celtix allows you to develop service-oriented components
 - Build service-oriented applications
 - Wrap existing systems in a service-oriented fashion
 - Integrate existing systems using service-oriented technologies
- It does this by:
 - Employing WSDL for interface definition
 - Supporting different styles of communication (one-way, request-response, messaging and publish-subscribe)
 - Providing synchronous (SOAP/HTTP) and asynchronous (SOAP/JMS) messaging capabilities with hooks for providing additional bindings and transports
 - Allowing Celtix components to be deployed flexibly (in JBI containers, J2EE containers and servlet engines) or as standalone client/server.
- The Celtix approach is discussed in the next sections

Service definition using WSDL

Service definition with WSDL

- Celtix components expose service interfaces using WSDL.
- Using WSDL as an interface language offers a number of benefits:
 - Programming language neutral: uses XML schema for type definition
 - Separates the logical interface (what a server does) from the physical interface (how to communicate with the server)
 - Services can support multiple communication protocols
 - Separates the service implementation from the service description
 - Services can be implemented in any language, OS or hardware.
 - Universally supported by the industry

Communication Models

Communication Models

- Celtix can be used for different kinds of communication styles:
 - *One-way*: raise event, submit document
 - *Request-response*: request information, request service
 - *Messaging*: send/receive document
 - *Publishing*: publish-subscribe
- The ability to support all these styles using a choice of transports is a key strength of Celtix.
- The following slides discuss each communication model...
- ... and show how to implement this model using Celtix.

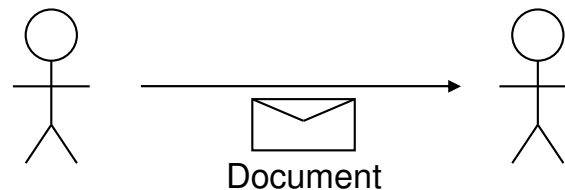
One-way: raise event

- The “raise event” pattern involves an event emitter generating asynchronous events.
- Events are modeled as one-way operations in WSDL.
- The events can be transmitted using either SOAP/JMS or SOAP/HTTP.

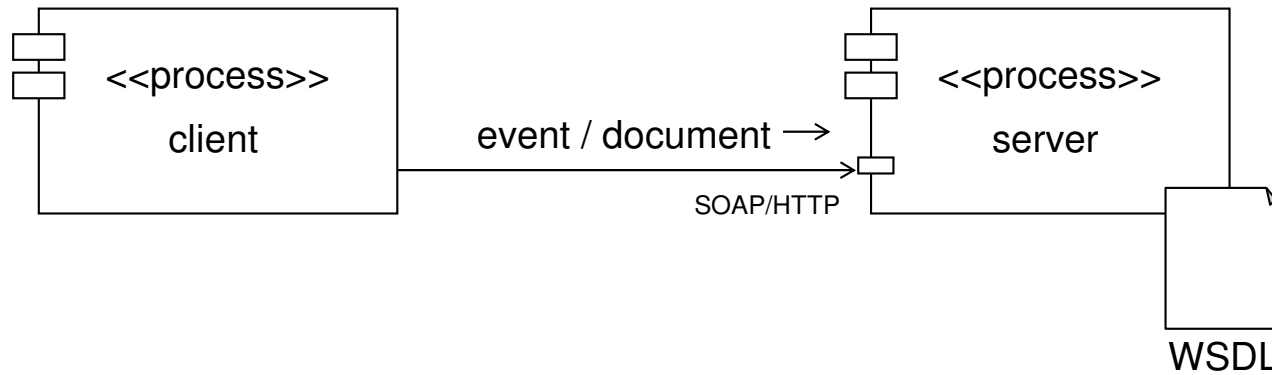


One-way: submit document

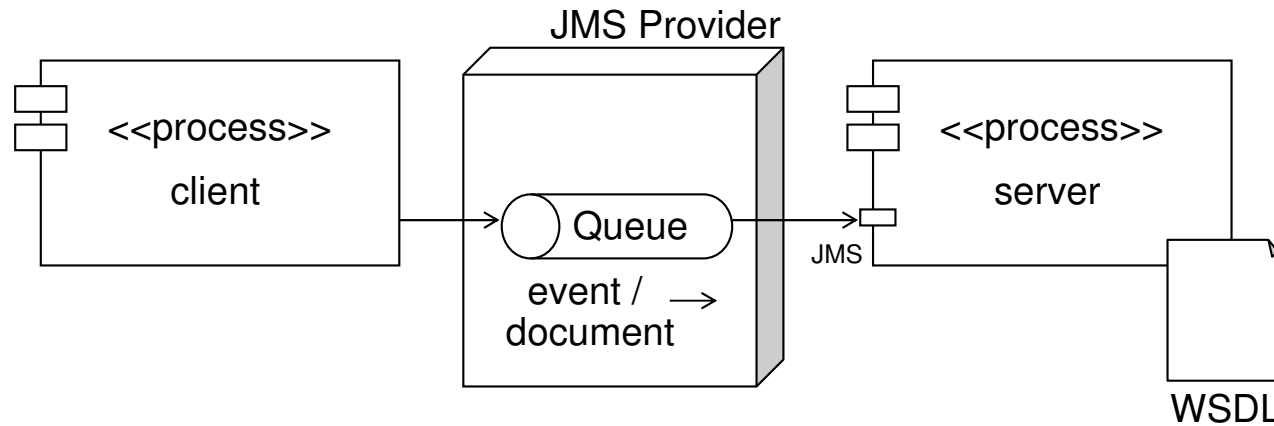
- The “submit document” pattern is logically similar to the “raise event” pattern.
- The pattern differs only in semantics: the document submitter passes a document to a document processor, and does not require a response.
- Documents are modeled in XSD and passed using one-way operations in WSDL.
- The documents can be transmitted using either SOAP/JMS or SOAP/HTTP.



Celtix one-way using SOAP/HTTP

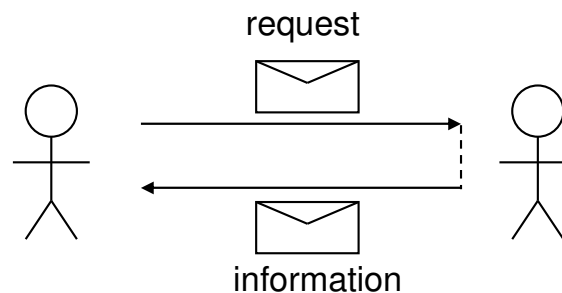


Celtix one-way using SOAP/JMS



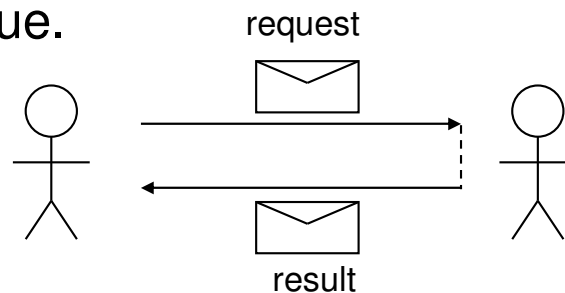
Request-response: request information

- A client requests information from a server.
 - The client sends a request message, for example `getCustomerInfo()`
 - The server sends a response message containing the information.
- The server responds in near time.
- This is modeled as a two-way operation in WSDL
- The request and response can be sent using SOAP/HTTP or SOAP/JMS queue.

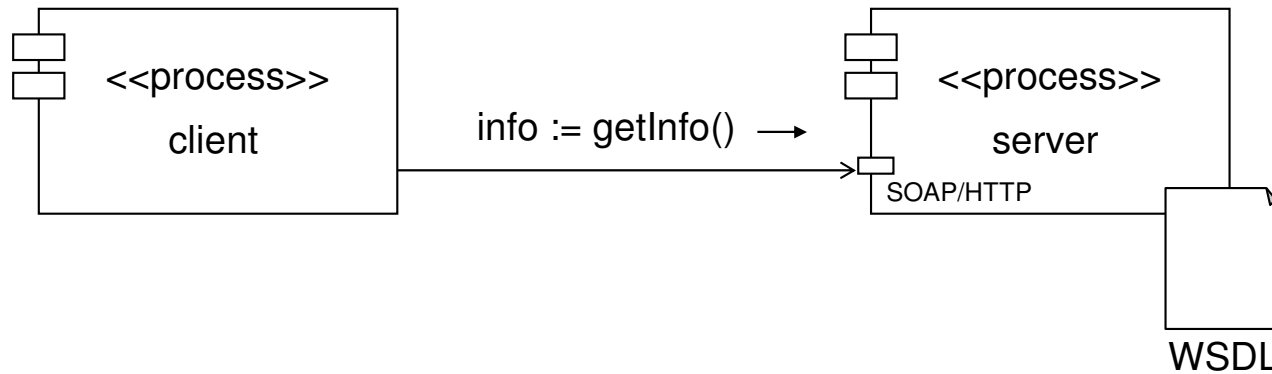


Request-response: request service

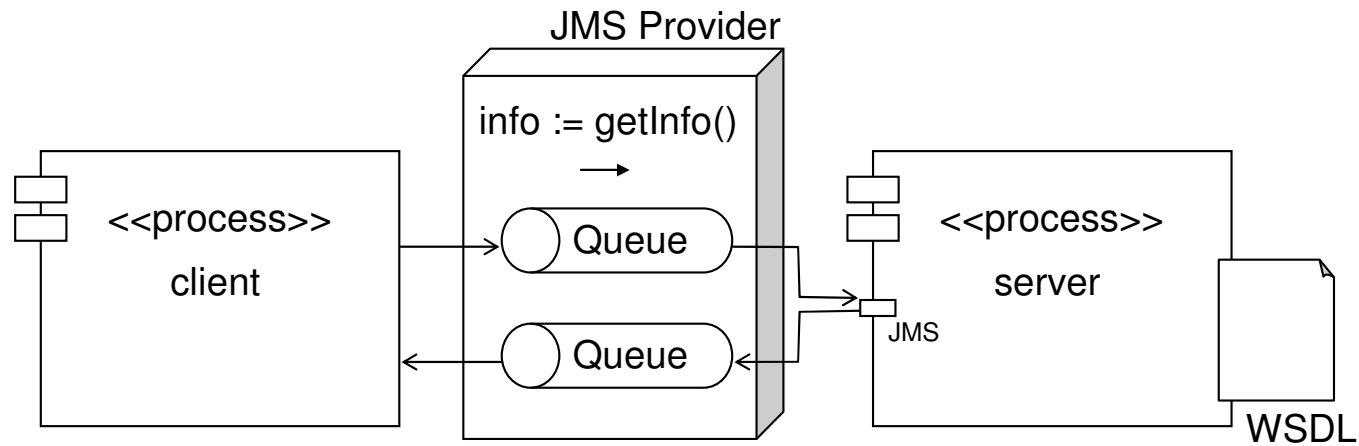
- A client requests that a server perform some task.
 - The client sends a request message, for example `transferFunds()`
 - The server sends a response message which may:
 - Contain the results of the task.
 - Indicate success or failure.
- The server responds in near time.
- This is modeled as a two-way operation in WSDL
- The request and response can be sent using SOAP/HTTP or SOAP/JMS queue.



Celtix request-response using SOAP/HTTP

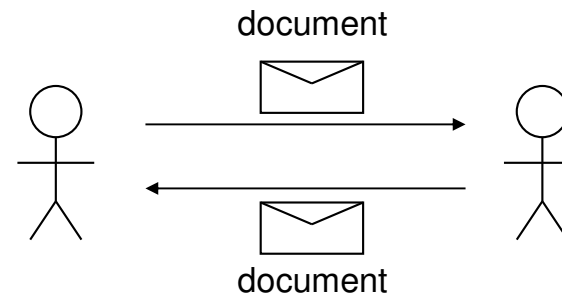


Celtix request-response using SOAP/JMS

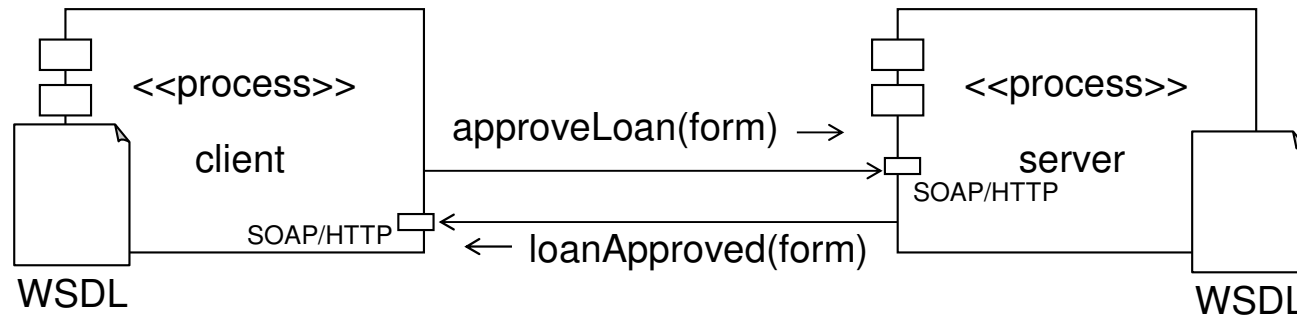


Messaging: send/receive document

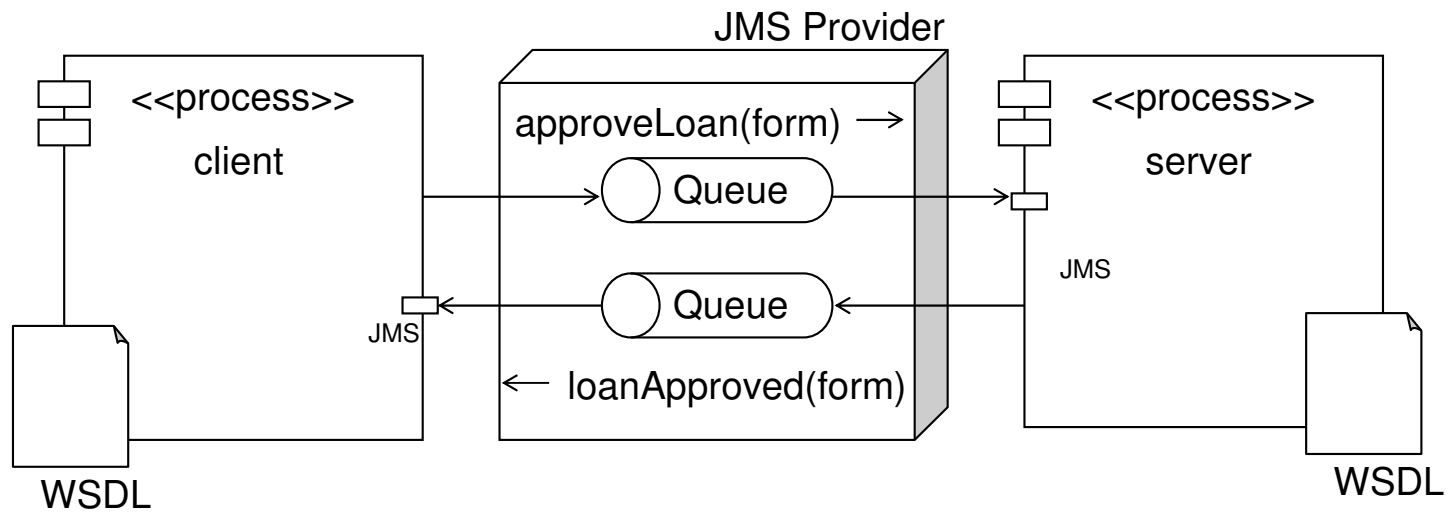
- Celtix can be used for document-oriented systems
 - Clients place documents in an “outbox” for delivery to other services
 - Clients receive documents from other services in an “inbox”.
- This is ideal for long-lived transactions.
- The outbox and inbox functionality is defined using WSDL one-ways.
- Messages can be sent using either SOAP/HTTP or SOAP/JMS-queue.
 - SOAP/JMS is a more natural choice.



Celtix messaging using SOAP/HTTP

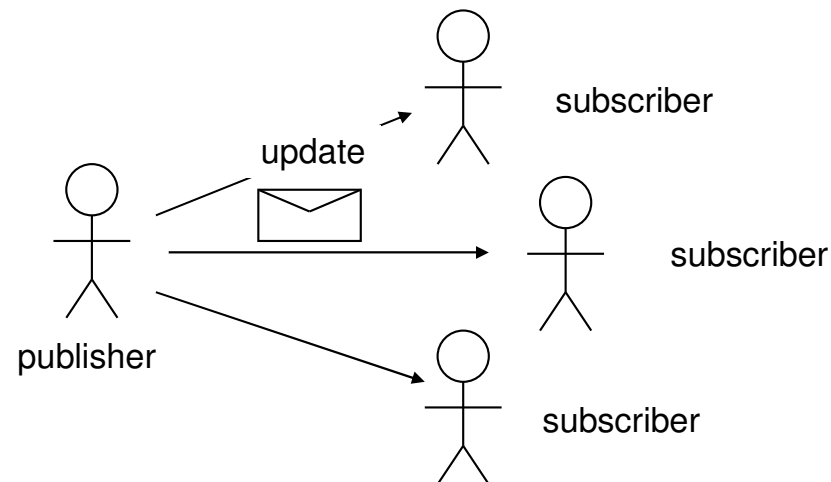


Celtix messaging using SOAP/JMS

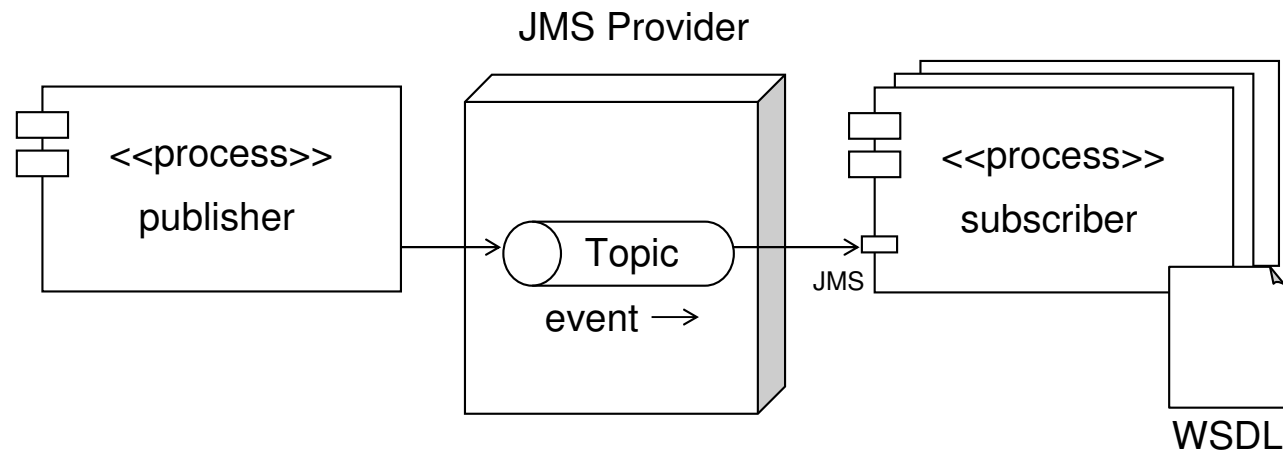


Publishing: publish-subscribe

- In the publish-subscribe pattern, a service pushes events out to a (potentially large) number of consumers.
 - Classic example: “Stock Quote Update”
- The event is modeled as a one-way operation
- The message is sent using SOAP to a JMS-topic.
 - Event consumers “subscribe” to the topic.



Celtix publish-subscribe using SOAP/JMS

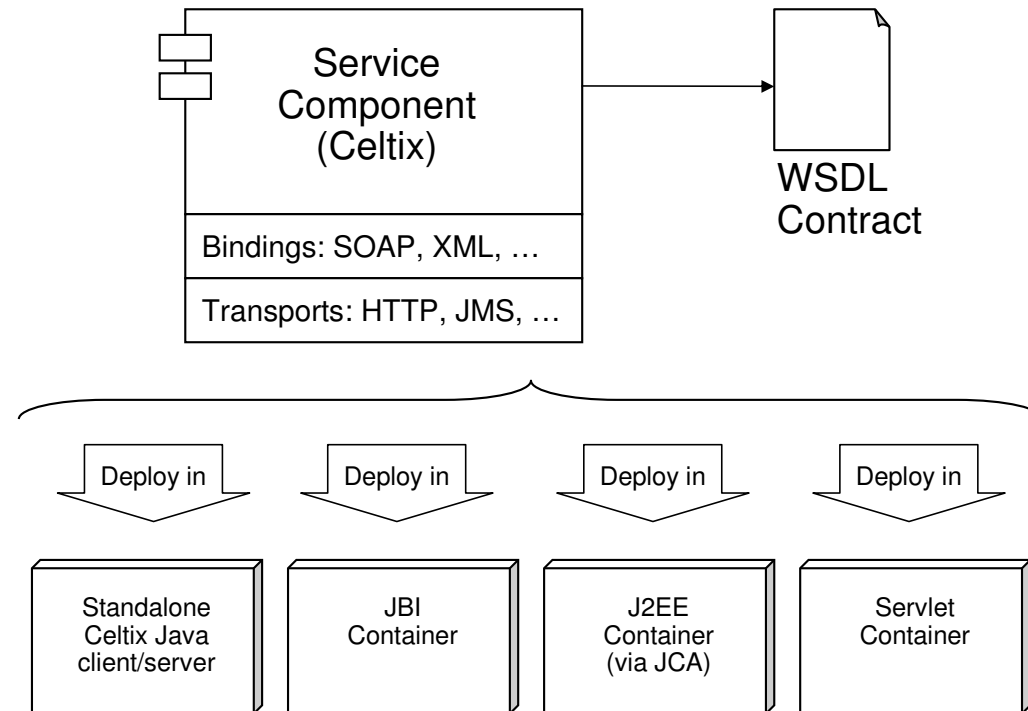


Celtix deployment models

Celtix deployment models

- Celtix can be used to write stand-alone Java clients and servers.
- Celtix can also be deployed in a number of different container technologies:
 - In a servlet engine like Tomcat
 - In a J2EE container (using bi-directional JCA)
 - In a JBI (Java Business Integration) container
- This flexible deployment approach allows you to deploy your Celtix service-oriented components anywhere.

Flexible Deployment



Celtix training, support and consultancy

Celtix training, support and consultancy

- IONA Professional Services provides training, support and consultancy services for Celtix.

- Training:
 - Web Services Foundations (one-day course)
 - Celtix for Developers (four-day course)
 - Advanced Celtix for Developers (three-day course)
 - Celtix for Architects (half-day course)

- Support, consultancy:
 - See <http://www.iona.com/info/services/oss> for details

Celtix delivery roadmap

Celtix delivery roadmap

- The Celtix project plan is available on the Celtix web-site <http://celtix.objectweb.org>
- There are 5 milestones in the project plan.
 - Current milestone is M3 (December 16th 2005)
 - Next milestone is M4 (January 31st 2006)
 - M5 is due in the beginning of Q2 2006.
- The current milestone (M4) provides a fully functional ESB messaging solution.
 - JAX-WS 2.0, JAX-B 2.0, SOAP 1.1, WSDL, HTTP 1.1 (using Jetty), JMS (using ActiveMQ), WS-Addressing, servlet deployment,...
- M5 will see advanced features complete:
 - JMX, JCA 1.5, JBI, JAX-RPC 1.1, routing, transformation, etc.

Summary

Focus on Celtix - Summary

- Celtix: the open-source Java ESB
 - Interface definition using WSDL and XSD
 - Supports both synchronous and asynchronous communication
 - SOAP/HTTP (using Jetty)
 - SOAP/JMS (using ActiveMQ or any other JMS provider)
 - Can be extended for other payloads and transports.
 - Support for major WS standards (WS-Addressing, WS-Reliable Messaging, ...)
- Hosted on ObjectWeb, backed by IONA Technologies
- Get involved – become a contributor!

<http://celtix.objectweb.org>