Celtix 1.0 Documentation Plan

Table of Contents

Rationale

Celtix is IONA's Open Source ESB offering. It is also going to provide the core for a redesigned Java Artix. This requires that the documentation be created to address several needs:

- the process must be open and visible to the community

- the tooling must be be freely available

- the resulting documentation must be easily incorporated into IONA's documentation processes

To address these concerns the Celtix documentation will be written using the newly standardaized OpenDocument format and use styles developed to be easily imported into the existing IONA Framemaker templates. To further enhance the flexibility of the Celtix documentation, the library will written in a highly modular fashion. While the library will retain the idea of a book, the core organizing unit for the Celtix documentation will be topics. This has the following benefits:

- topics can be better prioritized

- topics can be delivered faster than books

- topics allow the work to be easily distributed among the team

- topics promote reuse across multiple books

Referenced Documents

The following documents were referenced when creating this documentation plan:

- [Celtix Milestones](#)

- [Celtix Features](#)

- [Celix Development Plan](#)

Audience

User Types

We have identified four basic users for the Celtix documentation:

Architect: A high-level person in an IT department. This user will have a good understanding of both the business need to be implemented and distributed application technology. This user's primary concerns are 1) can this product meet my needs? 2) how do I use it to design my systems in a SOA? They are not going to be writing code with the product, but may be writing WSDL.

Application Developer: A Java developer with a good understanding of distributed application development. This user will at least be familiar with basic Web services concepts including XML, WSDL, SOAP, and HTTP. This user's primary concerns revolve around implementing functionality with the tool. They will most likely also be writing WSDL and will be deploying applications into test environments.

System Administrator: A person responsible for deploying and managing processes in a business environment. This user will not necessarily have a background in software development, but they will be very familiar with the operating systems with which they work. This user's primary concerns are 1) what

resources do I need to make available for this application to run? 2) how do I deploy services developed using this product? 3) how can I manage applications developed with this product.

Contributer: A Java developer who is interested in playing with the Celtix source code distribution and may wish to contribute to the project. This user will be an experienced Java developer and will be highly fluent in Web services technologies. This user's primary concerns will revolve around understanding the architecture of the core product, it extension hooks, and how to submit code to the project.

Targeted Audience

The initial release of documentation is going to focus on the application developer. This user has the broadest needs of the four users and is most likely going to be the most common user of Celtix. By focusing on the application developer's needs we can provide coverage for all of the major features and address the needs of 80% of the audience.

As time allows, the system administrator and architect users will be addressed. System administrators are most likely not going to be early adopters of Celtix, but will represent a significant portion of the user base once the Celtix begins to be adopted. Architects will likely be able to turn to IONA's marketing materials for the information they need that is not addressed in the application developer documentation.

The contributor is a low priority for the documentation team. Their needs are more appropriately addressed by the design documents written by the development team. In addition, the code should be commented well-enough to address this audience.

Staffing

The following writers are working on the Celtix 1.0 documentation:

Name	Contact Number
Fintan Bolton	+49-89-14347132
Eric Johnson	8536
Stephen McCarthy	2463
David Porter	2194

Delivery Formats

Development

The documentation is going to be written using the OpenDocument format. The documents must use the formats defined in the templates posted on the Celtix Web site. The source will be stored in the Celtix SVN repository.

The release notes will be maintained as a flat text file.

Web Site

The documentation will be published on the Celtix Web site in both HTML and PDF.

In addition to the PDF and HTML documentation, the Web site will have a link to the release notes which will be a flat text file. The Java doc for the APIs will also be published.

Distributions

Binary Distribution

The binary distribution will include PDF copies of the documentation and a copy of the flat text file holding the release notes. The Java doc for the APIs will also be included in the distribution.

Source Distribution

The source distribution will include the ODF source of the documentation as well as PDF copies of the documentation. In addition, the flat text file containing the release notes and the Java doc for the APIs will be included in the source distribution.

Dates

The product release milestones are as follows:

- **Documentation Project Starts**: March 27, 2006
- **Celtix RC**: March 31, 2006
- **Celtix 1.0, Doc MS1**: April 28, 2006
- **Doc MS2**: May 26, 2006
- **Doc MS3**: June 30, 2006
- **Doc 1.0**: July 28, 2006

Organization of Library

The Celtix library is going to be organized around the anticipated tasks that an end user will need to perform. These include:

- installing the product
- developing services
- deploying services

While each of these tasks could represent a book in a product library, the Celtix library will be made up of small chunks of documentation that resemble what would traditionally be chapters in a larger book. Breaking the library into chapter sized topics makes it easier to distribute the work load among the documentation team and makes it easier to reuse chunks of the documentation.

Because the documentation is primarily going to be accessed on the Web, the user can still be presented with a structure that resembles a traditional library.

Installing Celtix

This task contains two documents:

Installing Celtix covers choosing the appropriate package to download, the prerequisites for installing Celtix, and instructions for installing Celtix from either the binary distribution or source distribution.

Getting Started with Celtix is a tutorial that walks the new user through getting the HelloWorld demo running from both the binary distribution and the source distribution.

Developing Services

This task includes writing WSDL documents to define your services, implementing the services, and implementing consumers for the services. The following documents are part of this task:

Developing a Service with Celtix discusses how to develop a service using Celtix. This document assumes that the WSDL for the service is provided. It covers the basics ranging from generating code, service implementation, and mainline development.

Developing a Consumer with Celtix discusses how to develop a service consumer using Celtix. This document assumes that the WSDL for the service to be consumed is provided. It covers the basics including code generation and consumer implementation.

Creating Routes with Celtix discusses how to develop routing rules for use by Celtix's routing service.

Writing Handlers discusses how to develop Handlers for use in Celtix applications. There are four types of Handlers that can be used in Celtix applications. This document covers all of them. This may turn into four smaller documents: One for each Handler type.

Writing Asynchronous Services discusses using the JAXWS Async APIs to develop asynchronous Web services.

Implementing a Transport for Celtix discusses how to write custom transports for applications that use Celtix. This document is targeted at more advanced users such as the hobbyist.

Implementing a Binding for Celtix discusses how to write a custom binding for applications that use Celtix. This document is targeted at more advanced users such as the hobbyist.

Deploying Celtix Applications

This task includes:

- configuring a service
- using the non-HTTP transports
- deploying services in a Celtix environment
- deploying Celtix based services in non-Celtix environments

 The following documents are part of this task:

Celtix Configuration Guide discusses the basics of using the Spring framework to configure a Celtix application. This book will also discuss the relationships between WSDL, programmatic configuration, and configuration files.

Managing Celtix Services discusses using the JMX interface to manage deployed Celtix services.

Deploying Celtix Applications in an SCA Container discusses how to deploy Celtix applications in an SCA framework.

Deploying Celix Applications in a J2EE Environment discusses how to deploy Celtix applications into J2EE appservers. It will also discuss how to use Celtix to integrate with applications hosted in a J2EE appserver.

Deploying Celix Applications in a JBI Container discusses how to deploy Celtix applications into a JBI framework.

Deploying Celtix Applications in a Servlet Container discusses how to deploy Celtix applications into a servlet container.

Using the Celtix JMS Transport discusses how to configure and deploy Celtix applications that use the JMS transport.

Using Reliable Messaging in Celtix discusses how to configure Celtix application to use the WS-RM implementation shipped with Celtix.

Securing Celtix Services discusses how to add security to a Celtix environment.

Extending Celtix Configuration discusses how to extend the Celtix configuration susbsytem to include user defined policies.

Reference Material

Reference material is a nice addition to a documentation set. Some of this material will be produced by the engineering team.

Java API Reference will be generated Javadoc provided by the engineering team.

Command Line Tool Reference will contain a list of all the command line tools and their flags.

Celtix Architecture Guide will outline the high-level architecture of the Celtix runtime. It will be primarily addressed to system architects and users who are interested in having a better understanding of how the parts of the Celtix runtime work together. It is not intended for developers looking to contribute to the Celtix project or to understand the low-level details of how the runtime is designed.

Scope of Work

Need a breakdown of how much work each document will require.

Work Breakdown

The deliverables are detailed below. All status will be tracked on the Celtix ObjectWeb Task Tab at https://forge.objectweb.org/pm/task.php?group_id=192&group_project_id=182&func=browse.

Doc MS1

- **Installation Guide**(SMC)
- **Getting Started with Celtix**(DP)
- **Developing a Service with Celtix**(EJ)

- **Developing a Consumer with Celtix**(FB)
- **Celtix Configuration Guide**(SMC)

Doc MS2

- **Writing Handlers**
- **Creating Routes with Celtix**
- **Using the JMS Transport**
- **Deploying Celtix in a Servlet Container**

Doc MS3

- **Writing Asynchronous Services**
- **Securing Celtix Services**
- **Using Reliable Messaging in Celtix**
- **Deploying Celtix Applications in a J2EE Environment**

Docs 1.0

- **Deploying Celtix Applications in an SCA Container**
- **Deploying Celtix Applications in a JBI Container**
- **Command Line Tool Reference**

Post 1.0

- **Implementing a Celtix Transport**
- **Implementing a Celtix Binding**
- **Extending Celtix Configuration**
- **Tutorials**
- **FAQs**