From Technologies to Solutions

Apache OFBiz Development

The Beginner's Tutorial

Using Services, Entities, and Widgets to build custom ERP and CRM systems

Jonathon Wong Rupert Howell

PACKT PUBLISHING

Table of Content

Preface

Apache Open For Business or OFBiz as it is more commonly known, is an open source framework designed to facilitate the building of Enterprise Resource Planning (ERP) software. ERP is a general name for any system which attempts to integrate all business processes and underlying data into one single system. Indeed the OFBiz framework not only facilitates the building of your own custom software, but also comes packaged with many tools you would expect from an ERP system, and much more. The extent to which you wish to use these applications is entirely up to you and the needs of your business. Some businesses choose to use some or all of these components virtually straight out of the box. Others may spend time and money customizing these components or building new ones to suit their own needs and their own unique business processes. Since OFBiz is licensed under the Apache License Version 2.0, organizations can use, customize, extend, modify, repackage, and even resell OFBiz completely free of charge.

OFBiz is aimed primarily at ecommerce businesses, giving easily customizable tools such as a full Warehouse Management System (WMS), an accounting system and full order and product management systems. It even has a full front end, customer facing website and shopping cart with tools and features comparable to multimillion dollar websites such as Amazon, not to mention its own set of self maintenance and administrative tools. Out of the box, OFBiz is a multi-currency system working just as well with British Pounds, Euros, or any other currency as it does with US Dollars. It is multilingual and is able to display text in different languages depending on where in the world the user or customer is looking. It is so versatile it is not even tied to one database, and fully supports most well known databases.

The main reason for its versatility and size has been its open source model. OFBiz is truly a collaborative effort with a small number of committers who have volunteered to develop and maintain a code base supplied by both themselves and a growing community. Although documentation on the tools is often thin on the ground (this is mainly because of the speed at which the project and components evolve), there are free and active mailing lists set up that will become an invaluable learning tool and source of information as you progress with OFBiz. The OFBiz project employs the use of the well known JIRA application (a bug and issue tracking and project management tool – which is using the OFBiz Entity Engine, a major part of the framework). This allows developers and users to tell the community about any bugs they find in the software or request new features that they would find handy but perhaps don't have the resources to develop for themselves. Who knows? Once you have read this book you may even want to have a go at developing an outstanding issue or fixing a bug for the project yourself!

A Brief History of OFBiz

OFBiz was created in May 2001 by David E Jones and Andy Zeneski after they discovered that the problems they had tried to solve independently were similar and that existing solutions at the time were not sufficient. They realized that these problems could be solved using an Open Source model.

During the course of the next 5 years the OFBiz community grew and as such the number of organizations adopting OFBiz as their ERP system rose, and in January 2006 the project was accepted by the Apache Incubator Project Management Committee (PMC).

In December 2006 the Board of Directors voted to make Apache OFBiz a top-level project. Today OFBiz remains a highly active, community driven, not for profit project enjoying a growing membership and a wider adoption.

About This Book

The examples and tutorials featured within this book aim to, by the end of our journey, come together to produce a working web-based application. Various approaches will be explored during this project. Different combinations of techniques will be tried and evaluated, so we can learn when best to use them.

We will find that OFBiz is more than just a framework or toolkit. A good portion of OFBiz is comprised of ready-made functions and structures that easily lend themselves to being building blocks for ERP software, the application components. Whilst it is possible to use only the OFBiz framework to build a complete ERP application, it will be at least several magnitudes faster to build on top of these ready-made application components. We will study some of these applications while doing our project, which will also lead to a better understanding of some tried and tested best practices.

What This Book Covers

Chapter 1, using a Windows machine, guides us through downloading and installing the necessary software we need to obtain and run the OFBiz framework. We are shown how to create an Eclipse project and once running we are shown a few of the components as we place an order on the applications customer facing website and fulfill the order using the back office's Order Manager component.

In *Chapter 2* we learn the structure of OFBiz. We are introduced to the concepts of the framework, applications, and hot-deploy directories. We perform our first customization on an existing OFBiz component and finally create the structure of our own bespoke application.

In *Chapter 3* we take a look at how the output to the screen is constructed using the screen widgets. We start by creating a simple screen in our learning component, showing a basic output. By the end of the chapter we have learned how to create complex screens, made up of different sections.

In *Chapter 4* we study form widgets. We learn how they are used within screen widgets and can save us development time and effort by quickly producing XHTML forms so we can input information to the application.

In *Chapter 5* we complete our investigation into the presentation layer of OFBiz by learning how to use Menu-Widgets to navigate around our component. We also take more of a look at how FreeMarker can help us display more complicated screens.

In *Chapter 6* we re-visit the Control, learning more about how OFBiz makes use of the Front Controller pattern to configure the flow through our component in just one place. We learn how OFBiz handles different types of requests and we are introduced to the concept of security. By the end of the chapter, we have added "log in" functionality to our bespoke application and have seen how easy it is to force a request to be "secure".

In *Chapter 7* we move on to the concepts of the Entity Engine and learn how OFBiz employs the use of the Delegation Pattern to give us easy access to methods to persist data. We learn how OFBiz creates the database structure, adding fields, tables, constraints and indexes from definitions in XML files. We see how, by using View Entities, we can perform joins across tables allowing us to create complex queries. We are also introduced to the Webtools administration component of OFBiz and discover how to access the raw data through these screens.

In *Chapter 8* we are led through a series of examples designed to showcase data lookup and persistence techniques. We learn how to use the GenericDelegator's methods to lookup and manipulate the underlying data. We discover how using the Entity Engine Cache can massively improve performance by cutting down the number of database queries and learn how complex queries can be created on the fly by using Dynamic View Entity. Finally we learn how to use the EntityListIterator to efficiently paginate through large record sets.

In *Chapter 9* we take a closer look at Java events by learning a number of techniques vital to programming the flow of the application. We also take a look at how we can assign users permissions and how these permissions are checked within the Java methods.

In *Chapter 10* we next see another type of event and a very important one—the services. We learn about the advantages of the Service Engine and how it works, learning how to define and write services in Java. We learn the difference between invoking these services synchronously and asynchronously and how services can be scheduled using the Job Scheduler. Finally we learn how to trigger these services using ECAs (Event Condition Actions).

In *Chapter 11* we move on to study complex permissions, learning how to assign users granular permissions, and how simply these permissions can be checked from our services. We learn by example how to restrict users from viewing or inputting data depending on access rights whilst building up our bespoke application.

In *Chapter 12* we learn about the OFBiz Mini-Language. We see how we can write simple services and events in Minilang, and we learn when it should be ideally used. We see its versatility and see how widespread its concepts are used throughout the framework.

In *Chapter 13* we come towards the end of learning about the framework, we see how easy it is to change the look and feel of the component and study the structure of the existing screens. The chapter moves on to some more advanced FreeMarker techniques that are commonly used throughout all of the components. Finally using the production of a PDF as an example we see how to output different formats.

In *Chapter 14* we learn some real world developing techniques, including how to debug through the different parts and languages found within the framework. We see how to connect a remote debugger to the application and step through the Java code line by line using the Eclipse IDE. We next learn the concepts behind getting the latest bug fixes and features and merging these into our project using Windows tools enabling us to successfully work with the latest and greatest version on OFBiz. Finally we see learn how to run OFBiz behind the Apache HTTP Server allowing us to create a scalable architecture.

What You Need for This Book

Full download and installation instructions for all software and applications required to obtain and run OFBiz can be found in Chapter 1. These included:

- The Java Development Kit
- TortoiseSVN
- Eclipse IDE

At the time of writing, the latest stable version of OFBiz is 4.0. We will be working with that version, and not with the trunk version. In OFBiz speak (version control speak, rather), such a stable version is termed a "release branch"; the single trunk that forges ahead 24/7 is simply called the "trunk". Each stable branch can still move forward, due to the application of bug fixes. Hence, even release branches do go through revisions.

The trunk is the "latest, greatest and riskiest" state of OFBiz, where new features are added frequently, possibly along with new bugs. Such frequent addition of features do not allow for adequate testing. Before the dust can settle after a new addition by one contributor, more additions may come in from other contributors around the world. Scary as that sounds, this rapid and globally collaborative environment gives the OFBiz framework the fantastic advantage of being improved at a breakneck pace.

On the other hand, release branches like OFBiz 4.0 have had a "feature-freeze", and are invested with testing efforts focused on removing bugs and enhancing stability rather than adding new and untested features. Release branches are ideal for deployment in production environments (for real business use) where stability and ease of maintenance and support are vital.

Working with the trunk presents a moving target, a challenge that is best taken when we have some decent competency with the OFBiz framework as well as familiarity with recent OFBiz trends and changes.

This book may refer to existing code and also to line numbers within files. Bug fixes may therefore have an effect on these line numbers. Since they are only bug fixes, the line numbers should not drastically alter and a simple "Find" in the file will point you in the right direction.

The official policy on the frequency of releases (creation of new release branches) is published once per year, as defined in the Apache OFBiz Release Plan - General Release Policies `http://docs.ofbiz.org/display/OFBADMIN/Release+Plan#ReleasePlan-GeneralReleasePolicies`.

Who is This Book For

This book is intended for people wishing to understand and begin to customize the OFBiz framework. A basic level of Java is required and some understanding of Object Oriented Design (OOD) concepts would be beneficial. A basic grasp of Simple Query Language (SQL), although not strictly necessary, is certainly helpful to quickly understand some of the concepts in OFBiz.

Some fundamental skill with XML syntax is required, but only as far as is required to create well-formed XML documents.

The book will offer more hand-holding to MS Windows users. Users of other Operating Systems, such as Linux, should naturally be more adept at grasping the concepts laid out using Windows-specific examples and translating those examples into their respective OS's dialects.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text are shown as follows: " The `<include-form>` element includes a form named `FirstForm` residing in a file `LearningForms.xml`."

A block of code will be set as follows:

```
<request-map uri="OneForm">
<response name="success" type="view" value="OneFormScreen"/>
</request-map>
```

When we wish to draw your attention to a particular part of a code block, the relevant lines or items will be made bold:

```
 server="default-server"
 location="webapp/partymgr"
base-permission="OFBTOOLS,PARTYMGR"
 mount-point="/partymgr"/>
```

New terms and **important words** are introduced in a bold-type font. Words that you see on the screen, in menus or dialog boxes for example, appear in our text like this: " We see two fields **First Name** and **Last Name**, and one **Submit** button.".

Warnings or important notes appear in a box like this.

Tips and tricks appear like this.

Reader Feedback

Feedback from our readers is always welcome. Let us know what you think about this book, what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply drop an email to `feedback@packtpub.com`, making sure to mention the book title in the subject of your message.

If there is a book that you need and would like to see us publish, please send us a note in the **SUGGEST A TITLE** form on `www.packtpub.com` or email `suggest@packtpub.com`.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on `www.packtpub.com/authors`.

Customer Support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.