

OFBiz – an open source ERP system

A handout for the ERP system course at Aalborg University

Installation and first steps

OFBiz (formally known as opentaps) is a web-based application, which brings with its own database (Derby) and Apache Tomcat. It brings several technologies together, from scripting over database to high level programming technologies. It is developed as MVC framework (Model-View-Controller) and using the decorator pattern. You will find further details about the architecture [here](#).

To use OFBiz on your computer (in our example a Microsoft Windows system), you first need to **download the actual stable package and unpack it into a directory, e.g. C:\ofbiz**. You will find the download site on [sourceforge](#), where to download the zipped file opentaps-1.0.0-preview4.zip (the one we are using within our course). The unpacked directories are under Subversion source control. Also install Subversion for controlling changes on the OFBiz files (get it [here](#)).

You now have to decide whether to work with the OFBiz files via file explorer or – better – via Eclipse (OFBiz comes with an Eclipse project). If so, download and **install the [current Eclipse version](#)**. Of course, before you can start running and using OFBiz, you will need to **download and install the [Java SDK](#)**. Be sure, that the automatic build process within Eclipse is switched off (Project -> Build Automatically).

As mentioned before, OFBiz comes with embedded servers and databases, so running OFBiz is very simple. Just click on the startofbiz.bat file in the opentaps directory or use your command tool window and navigate to your opentaps directory and type in startofbiz.bat. It may be necessary to change this file, so that the *java* keyword is found. At least *java -jar ofbiz.jar* must be called. Better call *java -jar ofbiz.jar > runtime/logs/console.log*, so that console output is written into a log file (as in the starofbiz.bat file). The start process may last some minutes, because applications and add-ons are dynamically loaded into OFBiz and database tables, services, forms, ... are generated on the first build. When the build process is ready, you can visit the application at <http://localhost:8080> in your favorite web browser and you will – hopefully – see the application as in Figure 1.



Figure 1, OFBiz start screen on <http://localhost:8080>

By clicking of one of the squares, the according application is called. Point of Sales is grayed out and can't be called directly. Use `java -jar ofbiz.jar -pos` instead. Whenever you have to login, you can type **admin** and **ofbiz** as password.

By closing the console window you are closing the application as well.

Architecture of OFBiz

Directories and files

OFBiz is built as component-based application: each application (Sales & Marketing, Human Resource ...) is a component. Any application created by you can be added easily. Existing applications can be maintained any-time. Within the OFBiz directory (or opentabs directory) there are several files and directories (see Figure 2). First to the directories:

/application: here the core application components are stored (which you can see on the start screen in **Fehler! Verweisquelle konnte nicht gefunden werden.**). If you want to change an existing application, this is the place to go.

/framework: this directory holds the components of the OFBiz framework, such as entity engine, minilang, and service engine. You normally do not have to change anything within this directory. Within the framework directory, the base directory may be important. It contains configurations and XML Schema files (beside many others).

/hot-deploy: here new applications can be dropped in. The OFBiz framework loads these applications on startup (after loading the ones in the application and specialized directory). This is the place to go, if you create new applications. By installing the demo data, some directories are created here (such as `/dataimport`, `/financials`, `/mycompany` ...).

/resource: Resources for the OFBiz website (e. g. start screen seen in Figure 1).

/runtime: here directories and folders are contained, which are necessary for runtime (such as `data/derby`) or are created during runtime (such as `/logs/console.log`).

/specialpurpose: contains additional applications which are not within the OFBiz core applications.

In several of the above shortly described directories there is a file named **component-load.xml**. These files are parsed by the framework that detects which components have to be loaded in which order. In the hot-plugin directory the component-load.xml file may look like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<component-loader xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="http://www.ofbiz.org/dtds/component-loader.xsd">
  <load-component component-location="{ofbiz.home}/hot-deploy/myApp"/>
</component-loader>
```

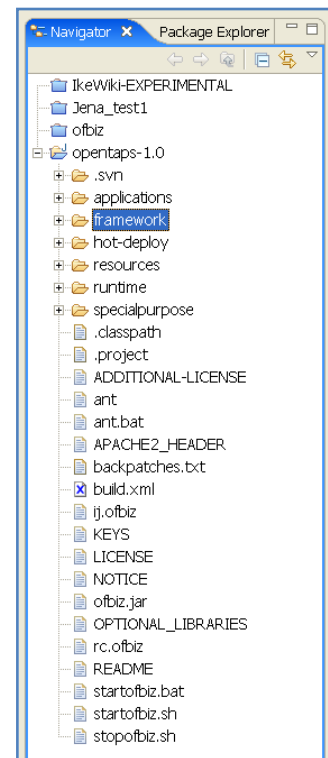


Figure 2, OFBiz directory

In every XML file there is the link to the XML Schema file. The biz.home} entry points to the base directory of the framework.

Now look at the application directory (see Figure 3Figure 2). Each application holds its data model definitions, business logic, and presentation tier (but they also may use the ones of other applications and therefore may depend on each other). Within an application directory (e. g. /product) there are several important sub-directories:

- /build, which contains the built Java code (*.class files) and used libraries (*.jar files)
- /data, which contains XML files, which fill the database tables
- /entitydef, which contains definitions for entities (also as XML files)
- /script, which contains XML files defining methods in the OFBiz minilang
- /servicesdef, which tells the framework where to find insert, update and getter methods (either in Java files [/src] or minilang XML files [/script])
- /src, which contains the Java code defining methods
- /webapp, which contains the web interface applications (which can be more than one for an application)



Figure 3, Folder structure of an application

The file build.xml tells the framework how to build an application and can be compared with a Java makefile.

Technologies

The OFBiz framework combines several technologies in order to use the best parts of each technology (e.g. Java programming for high level programming, script languages for fast development, HTML and XML files for defining the interface, etc.). But there are also technologies used which might be new to you and will now be described in short.

OFBiz divides its web interface into small pieces, such as sidebars, headers and footers, which can be shared by many OFBiz applications. For this the “decorator pattern” is used. OFBiz also divides its applications into actions (data retrieving from database, page navigation) and presentation (displaying the data).

Actions are usually written in beanshell scripts (*.bsh-files), a scripted version of Java. A description for Beanshell can be found [here](#). Presentation files are written as HTML files and – more often – as [Free-marker](#) template pages (*.ftl-files), which contain parts of HTML pages, such as a list, div-Elements, etc. and may contain variables prepared by a Java application.

References

- Websites used for creating this document:
- Open For Business (OFBiz) Project: <http://www.opensourcestrategies.com/ofbiz/index.php>
 - OFBiz Documentation Index: <http://docs.ofbiz.org/display/OFBADMIN/OFBiz+Documentation+Index>
 - Debugging OFBiz: http://www.opensourcestrategies.com/ofbiz/ofbiz_eclipse.php
 - Running and Debugging OFBiz in Eclipse: <http://docs.ofbiz.org/display/OFBIZ/Running+and+Debugging+OFBiz+in+Eclipse>