POLITECNICO
MILANO 1863

POWER ENJOY

Inspection Document

Lorenzo Casalino - 877421

Tommaso Castagna

Document version 1.0

Contents

1 Functional Role

1.1 Class Under Inspection

The code inspection activity described in the following of this document takes as subject the **HumanResEvent.java** class, a java compilation unit belonging to the **Human Resource** application, part of the OFBIZ open-source project.

The namespace of reference for this class is the *org/apache/ofbiz/humanres*.

1.2 Functional Role

Before analyzing in detail the functional role covered by the java class assigned us for code inspection, a brief description of the context and explanation of the most relevant elements is provided.

1.2.1 OFBIz overview

The Open For Business project is an enterprise-oriented suite of applications developed to support most of the aspects that an enterprise application has to take care of.

The applications share the same underlying architecture, using common data, logic and process components.

A loosely coupled approach is used as base for the architecture, allowing an easy extension of the suite itself.

Each application is loosely coupled with the others, easing the updating and the extension.

1.2.2 Entities and Services

As stated by the official documentation:

- **Entities**: an entity is a relational data construct that contains any number of Fields and can be related to other entities. Basic entities correspond to actual database structures.
- **Services**: a service is a simple process that performs a specific operation.

1.2.3 Project's architecture

The architecture of the entire suite is composed by 4 sets of components:

- **Framework**

- **Applications**
- **Special Purpose**
- **Hot-deploy**

The sets of components, as well the contained components, are in a dependency relationship according to the dependency-arcs shown in the following diagram.

The dependency flow goes from top to bottom, either for component sets and for components.

This means that components and applications on the top are dependent on elements on the bottom of the same diagram. The viceversa should not be allowed.

The type of dependency may vary: foreign key dependency in the data model, application's service calling another application's service and so forth.

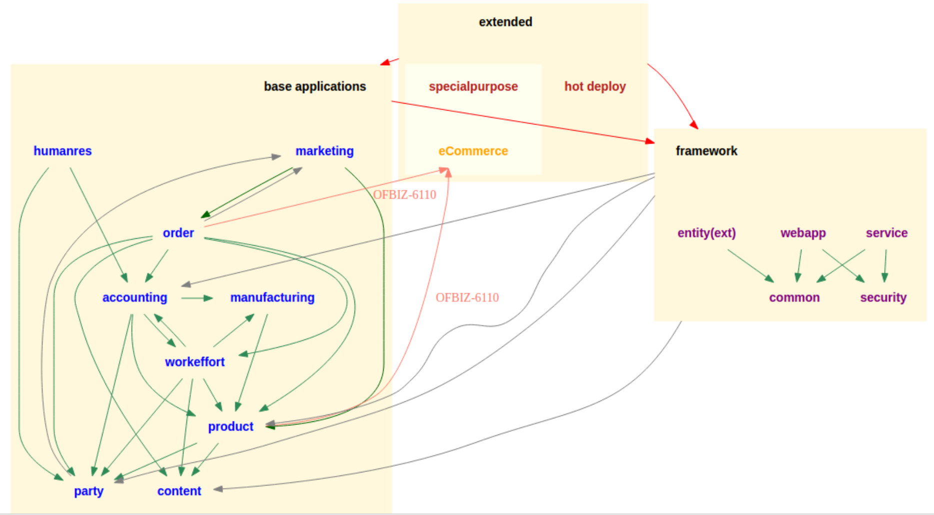

Figure 1: Components dependencies

As can be seen, there's a relation of dependency between the human resource application and the party application.

1.2.4 Parties

According to the *OFBIz project's overview*[1]:

Party can be either a Person, or a group of Parties. A Party Group could be a company, an organization within the company, a supplier, a customer, and so forth. Information that describes Parties or is directly related to Parties is contained in these entities.

According to the party's data model, each party, either a person, a group and so forth, is identified by a unique ID number.

1.2.5 Human Resource Entities

According to the *OFBIz project overview*:

The Human Resources entities are used to keep track of positions, responsibilities, skills, employment, termination, benefits, training, pay grades and payroll preferences, performance reviews, resumes and applications, and other Human Resources related information.

1.2.6 Employee Position

Also abbreviated as *position*, it is an entity used to represent a work position inside the company. Quoting the *official documentation of the project*[2]:

In OFBiz a position is the authorization, typically from the budget of an internal organization, for the Company to engage one person to do a job. OFBiz handles positions in a flexible manner so you can think of a position as an authorization for a full-time equivalent (FTE).

This means that you can fulfill a position with a person in a number of different ways. You can fill a position with one full time person, change the assignment of a position from one person to another over time, or split a position across more then one person at the same a time.

As implemented a position can be fulfilled by ***either a person or organization****.*

The data model[3] representing a employee position is quite huge and complex, so only the essential part for our analysis are shown and described.

NB: *The whole data model diagram on the wiki is not up to date, therefore there could be some discrepancy between it and the documentation.*

[1] https://cwiki.apache.org/confluence/display/OFBIZ/Component+and+Component+Set+Dependencies

[2] https://cwiki.apache.org/confluence/display/OFBIZ/Employee+Position

[3] https://cwiki.apache.org/confluence/display/OFBIZ/Data+Model+Diagrams

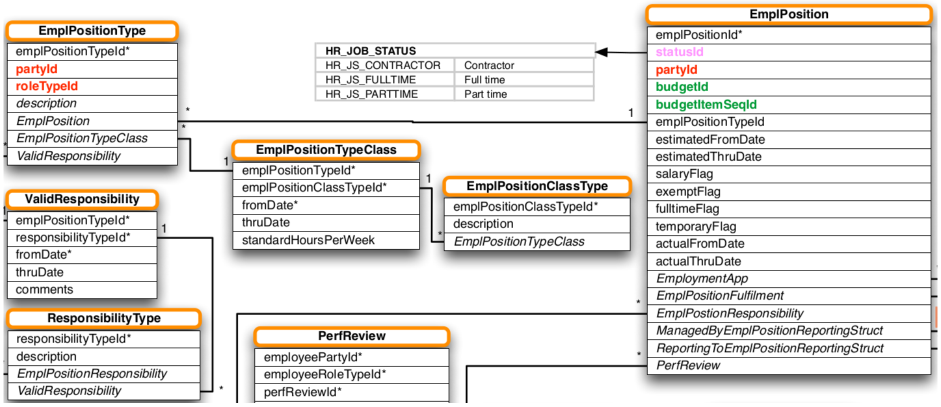

Figure 2: Employ Position - Employ Position Type relationship

- **EmplPosition**: entity used to represent an employee position inside the Human Resource Application.

  Each entity is characterized by several. The most relevant for our analysis are:

  - **emplPositionId:** an **unique** identifier used to distinguish a position from another one.
  - **statusId:** a string identifying the status of the position.

    Actually, the documentation and the data model of the emplPosition entity diverge about the content of this field.

    The former specifies that it is used to state if the position is **Active/Open**, **Inactive/Closed** or **Planned for**, while the data model diagram report is as a field specifying the type of position: **full time**, **part time** or **contractor**.

    As stated previously, the data model is not updated, therefore we take as reference what described by the documentation.
  - **partyId**: the ID of the **Internal Organization** authorized to fill the position.

- **EmplPositionType:** entity representing a possible type for a position. An example of position type is the Business Analyst or the System Administrator. Relevat fields are:

  - **emplPositionTypeId:** unique identifier used to discern between position types.
  - **partyId:**
  - **description:** a description of the type of position.

  - **EmplPosition:** identifier of the employee position having this specific type.

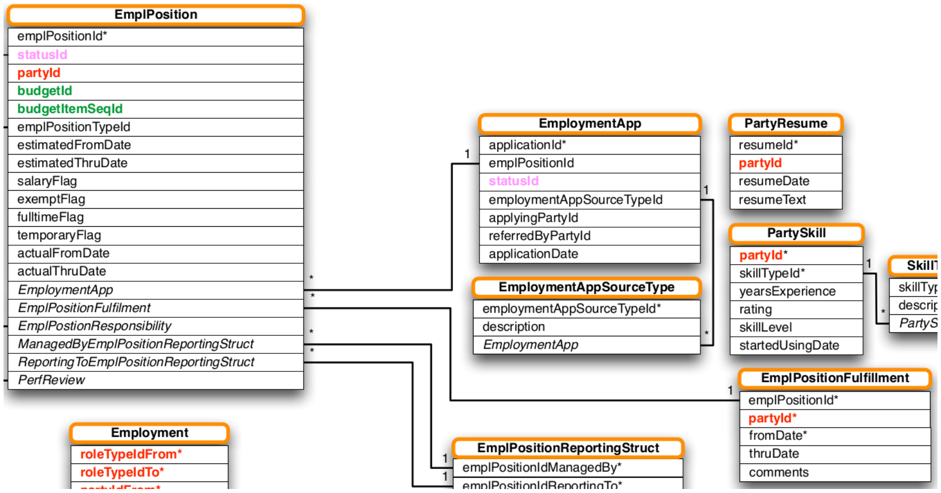

Figure 3: Employee Position - Employee Position Fulfillment relationship

- **EmplPositionFulfillment**: entity used to represent the party or parties that fulfill a specific position. Relevant informations of this entity are:
  - **emplPositionId:** the ID of the position fulfilled by the party.
  - **PartyId:** the ID of the party fulfilling the position.
  - **fromDate:** the date from which the position is fulfilled.

1.2.7 Internal Organization

Quoting the *human resource glossary*[4]:

Internal organization is the name of a relationship between a party group and your company.

This relationship is used to filter party groups as being part of your company to distinguish them from other groups which are external.

For example your marketing department is an internal organization while a suppliers sales department is not.

[4] https://cwiki.apache.org/confluence/display/OFBIZ/Human+Resources+Glossary

1.2.8 Human resource application:

The human resource application comes with a predefined set of functionalities that can be used to perform HR tasks or to provide a support for the creation of more complex HR applications.[5]

According to the documentation of the human Resources application[6]:

The Main window is the entry point into the Human Resources Application and displays the Company tree view for navigating to the main menu items.

There are three node types in the tree, each identified by a different icon. The top of the tree represents your Company, the highest level in the organization. The Company and departments under the Company can have sub departments or positions. Under positions are the people who fulfill the position.

Furthermore, as stated by the documentation, from the main screen of the application is possible:

- Navigate the company hierarchy, viewing departements, position and people.
- Add or remove a department
- Add a person
- Quickly open the profile of any item in the tree
- If an item is a position, you can add a person to fulfill the position.

The HumanResEvents class :

Despite the misleading namename given to this class, the main purpose of this class is to support the building of the company tree, This class is supposed to support all the functionalities that the humanres application backbone has to provide. This has been already listed in the previous paragraph.

Up to these release, the class presents 4 methods: one public and the rest private. The private methods are used as helper functions by the public one.

In the following the methods composing the class are described in details.

getChildHRCategoryTree: Only public method exposed by the HumanResEvents class. It's functional role is to manage the request of visualization of a given node of the tree describing the enterprise hierarchy.

[5] https://cwiki.apache.org/confluence/display/OFBIZ/Human+Resources+Guide

[6] https://cwiki.apache.org/confluence/display/OFBIZ/Human+Resources+-+Main+Window

The method retrieves the informations regarding the child requested. This, according to the party definition, may be an employee or the company or a department (partyGroup).

If the former is the case, the informations of the employee selected are returned. If the latter is the case, the departments that are child of the department selected or of the company, in the case the root of the tree has been selected, are returned, along with the informations about the employees that work in the selected department or company

getCurrentEmployeeDetails: Helper method used to retrieve the informations of an employee that currently is covering a specific position. The position is identified by the "partyID" parameter. First retrieve the presence of a position matching the ID passed as parameter. Then, retrieves the list of employment fulfillments regarding that position. Indeed, more than an employee can cover that position at the same time. The filter returns only the current fulfillment. The query count is used, but the interval is [0, 1].

For each fulfillment, the employee name and surname is retrieved. Maybe the fulfillment is covered by a company or organization. The informations retrieved are the Name and Surnamen of the employee.