# Component and Component Set Dependencies



## Component Sets

There are a small number of component sets current in OFBiz (from lowest-level to highest-level):

1. framework
2. applications
3. plugins

Dependencies between components in these sets must ONLY go up the list. For example components in plugins can depend on components in applications, but components in applications can not depend on components in plugins.

**OFBIZ-3500** - Getting issue details... STATUS addresses those issues.

## Components Within Sets

### *framework*

The framework is the underlying data structures and utilities that supply the essential common features used by the base applications and the extended components. These higher level components will have many dependencies on the framework components. There should be no dependencies in the framework on other parts of OFBiz but a few have crept in. They are shown as the gray arrows in the diagram.
There should be a full discussion on the mailing list before any new dependencies are added. Actually, they should be avoided and even an effort to remove the existing one would be welcome. Framework-only distribution is the first step in this direction...

An overview of the components in the framework stack can be found here:

Framework stack

### *base applications*

The base applications components depend on each other as shown in the diagram. The reason these dependencies are needed is because the components are organized according to high-level business concepts and in the real world business processes tend to jump between many of them. In technical terms, this results in foreign keys from an entity in one component to an entity in another, and to service calls from services in one component to services in another, and other similar effects.

While components can depend on one another in the base applications set they should follow the established priority of dependencies. In the diagram below you see existing dependencies and the component dependency priority going from the top of the page down (in other words components higher on the page depends on components lower on the page).

Whenever some data structure or feature could go in multiple components or could depend on elements in multiple components, the design should ensure that lower level components don't depend on something in the higher level ones.

The gray arrows in the diagram represent dependencies that violate the rule and are subject of ongoing discussions about how to fix them.

Graphviz output could not be displayed here.

An overview of the base applications can be found here:

Base Application stack

### Plugins

Components within the plugins set should not depend on each other. If there is a need for components in this set to depend on something in another set, that something should be put into the most appropriate component in the applications set and both plugins components can depend on the applications component. We have currently a dependency of the applications components from, at least, the ecommerce component with regards to the demo data, see **OFBIZ-6110** - Getting issue details... STATUS for details.

An overview of the plugins applications can be found here:

User-defined components that are not part of OFBiz, as plugins to OFBiz. The same guidelines are recommended here as well.

Plugins stack

## Other information

The Framework-only distribution page is related and about "features that should be available in the framework-only OFBiz distribution".