Upgrading to Servoy 8.x.x

In This Chapter

- Upgrading Installations
  - Upgrade Servoy 7.x Installations to Servoy 8.x
  - Upgrading Servoy Application Server
  - Upgrading Servoy Developer
- Upgrading Existing Solutions
  - Behavior Changes
    - Changes That Might Break Existing Code
  - Unsupported features and workarounds
    - DBTreeView

Upgrading Installations

Upgrade Servoy 7.x Installations to Servoy 8.x

For smart and web clients 7 or 8 is for the most part the same, 8 could have a few fixes more (and will be fixed faster/more, not everything will be backported)

For using the NGClient see: NGClient Compatibility

Upgrading Servoy Application Server

Installations running an older version of Servoy can't be upgraded, thus a new installation is required. See Installing the Application Server for more information.

Upgrading Servoy Developer

Mac users need to do a clean installation because of the underlying change that eclipse made for OSX.

Existing Window users Servoy 7.x installations can be upgraded to Servoy 8.x. In order to do so, open the Available Update Sites preferences and edit the URL of the **Servoy update site** entry:

- Open **Window** > **Preferences** > **Install/Update** > **Available Software Sites**
- Find the entry named **Servoy update site** and select the row
- Click **Edit**
- Change the Location field from `https://www.servoy.com/developer/70x_updates` to `https://www.servoy.com/developer/8xx_updates`
- Close all dialogs
- Go to **Help** > **Check for updates** to trigger the update
- After the update has completed, restart Servoy Developer
- when Servoy Developer is launched again, it will prompt for an update of the underlying Servoy Application Server.

Servoy Developer installations of older Servoy version cannot be upgraded, so a new installation with the Servoy 7 installer is required. See Installing Developer for more information.

Upgrading Existing Solutions

Upgrading existing Solutions to Servoy 8 is as easy as opening Servoy Developer 8 and getting the Solutions into the workspace or opening an existing workspace in Servoy 8. Note that once solutions are edited using Servoy 8, they cannot be used in earlier versions of Servoy anymore.

Behavior Changes

This paragraph describes the behavior changes that are introduced in Servoy 8.0. There are three categories of behavior changes:

1. Changes that might break existing code
2. Real behavior changes
3. Changes that affect the display of the UI

Changes That Might Break Existing Code

- As of Servoy 8 databaseManager.getDataSetByQuery(QBSelect query, int max_returned_rows) will take existing table filters into account.
  A new call databaseManager.getDataSetByQuery(QBSelect query, boolean useTableFilters (default true), int max_returned_rows) has been added that can be used to ignore table filter in the query call.

Note that databaseManager.getDataSetByQuery(String server_name, String sql_query, Object[] arguments, int max_returned_rows) that uses the query as a string will not use table filters.

Unsupported features and workarounds

DBTreeView

Bean components are not supported in the new NG webclient.
In order to ease the porting of existing applications to the new NG client, Servoy 8 does include a DBTreeView web component that tries to replicate the behavior of the old bean. In order to use that, you should replace the old bean from you forms with this new web component. The main difference between them, from a developer point-of-view is the missing of Binding scriptable. In the new bean, the binding is set directly on the web component. Here is an example :

In Servoy 7:

```
var binding = elements.myDBTreeView.createBinding(dataSource);
binding.setTextDataprovider(myTextDataprovider);
```

In Servoy 8:

```
elements.setTextDataprovider(dataSource, myTextDataprovider);
```

Here is a list of other changes :

- "setRowHeight" is no more available in Servoy 8, you should instead set the row height using CSS in the solution style sheet, like this:

```
[name='bean_name'] .dbtreeview span.fancytree-node {
 min-height: 40px;
}
```

NOTE: Not all the api functions of the old DBTreeView are implemented in the new we component, yet.