

Eclipse Scout

Release Notes

Scout Team

Version 10.0

Table of Contents

About This Release	1
Service Releases	1
Obtaining the Latest Version	1
Widgets	3
GroupBox: New MenuBar Position TITLE	3
TabBox: New Tab Style	3
TileGrid Groups: Show Loading State per Group	4
RadioButton and CheckBox: Wrap Text	4
Button: Support for HTML and Binary Resources	5
ImageField: Support for File Upload	5
Mode Selector: Implemented alternative display style	5
Data Objects	6
Renamings	6
Dependencies	6

About This Release

The Eclipse Scout 10.0 version was released as part of the [Eclipse 2019-06 Simultaneous Release](#) on June, 2019.

The latest version of this release is: 10.0.0.009_Simrel_2019_06.

You can see the [detailed change log](#) on GitHub.

Coming from an older Scout version? Check out the [Migration Guide](#)!

Service Releases

After the final simultaneous Eclipse release, there are no more Eclipse *service releases*. Scout 10.0 will continue to be maintained for a while and a new build may be released from time to time. Beside bug fixes, these releases may even contain some minor features. See the [Simultaneous Release FAQ](#) for details.

The following enhancements were made after the initial 10.0 release.

Simrel 2019-06 (10.0) — Release Expected June, 2019



The here described functionality has not yet been released and is part of an upcoming release.

(Section intentionally left blank for possible future release)

Obtaining the Latest Version

Runtime (Scout RT)

Scout RT artifacts are distributed via Maven:

- [10.0.0.009_Simrel_2019_06](#) on *Maven Central*
- [10.0.0.009_Simrel_2019_06](#) on *mvnrepository.com*

Usage example in the parent POM of your Scout application:

```
<dependency>
  <groupId>org.eclipse.scout.rt</groupId>
  <artifactId>org.eclipse.scout.rt</artifactId>
  <version>10.0.0.009_Simrel_2019_06</version>
  <type>pom</type>
  <scope>import</scope>
</dependency>
```

Eclipse IDE Tooling (Scout SDK)

You can download the complete Eclipse IDE with Scout SDK included (EPP) here:

[Eclipse for Scout Developers](#)

To install the Scout SDK into your existing Eclipse IDE, use this update site:

http://download.eclipse.org/scout/releases/10.0/10.0.0/009_Simrel_2019_06/

Demo Applications

The demo applications for this version can be found on the [features/version/10.0.0.009_Simrel_2019_06](#) branch of our docs repository on GitHub.

If you just want to play around with them without looking at the source code, you can always use the deployed versions:

- <https://scout.bsi-software.com/contacts/>
- <https://scout.bsi-software.com/widgets/>
- <https://scout.bsi-software.com/jswidgets/>

Widgets

GroupBox: New MenuBar Position TITLE

In addition to the existing menu-bar position TOP and BOTTOM, the GroupBox now supports TITLE. The menu-bar is placed in the header of the group-box right of the title-label DIV. Scout Classic: see property `menuBarPosition` in `AbstractGroupBox` and constant `IGroupBox.MENU_BAR_POSITION_TITLE`. Scout JS: see property `menuBarPosition` in `GroupBox.js` and constant `scout.GroupBox.MenuBarPosition.TITLE`.

Group Box	Menu 1	Menu 2	Menu 3
String Field 1	String Field 3		
String Field 2	String Field 4		

TabBox: New Tab Style

The tab box has a new property `ITabBox.PROP_TAB_AREA_STYLE` with currently two possible tab styles.

- `ITabBox.TAB_AREA_STYLE_DEFAULT`: Default appearance
- `ITabBox.TAB_AREA_STYLE_SPREAD_EVEN`: This style will spread all the tabs over the available space. When using this style, all the menu will be pushed into the ellipse menu.

TAB_AREA_STYLE_SPREAD_EVEN

TAB_AREA_STYLE_DEFAULT

TileGrid Groups: Show Loading State per Group

Each Group in a TileGrid can now display a loading indicator in the group header individually. This is useful if each displayed group loads data from an individual data source. Scout Classic: call `AbstractGroup#setLoading(boolean)`, Scout JS: call `Group.js#setLoading(boolean)`. Note: it is still possible to set the loading state on the TileGrid, to indicate the whole grid (and not an individual group) is loading data.

✓ Search Results (2)



RadioButton and CheckBox: Wrap Text

Both widgets `RadioButton` and `CheckBox` (aka `BooleanField`), now support the `wrapText` property. This means a radio button or a check box can have a label with a long text on multiple lines. In order to see the wrapped text, the field must have a `gridH > 1` or set the `gridUseUiHeight` property to true.

- ☒ Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Button: Support for HTML and Binary Resources

The **Button** now supports the property **htmlEnabled** which allows to use HTML in the label part of the button. You can even reference binary resources in your HTML. Simply call the method **AbstractButton#setAttachments(Collection<? extends BinaryResource>)** and define a reference, say an image URL, in your label HTML:

```

```

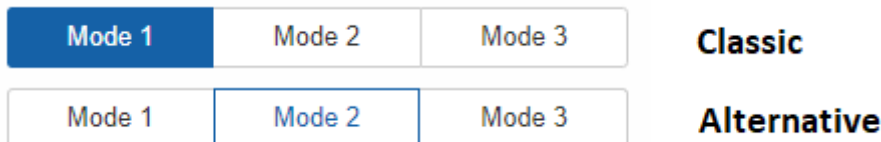


ImageField: Support for File Upload

The **ImageField** has a new property **uploadEnabled**. When set to true, the field opens the native file chooser and performs a file upload.

Mode Selector: Implemented alternative display style

The mode selector now has a different appearance for the two field styles **IFormField.FIELD_STYLE_CLASSIC** and **IFormField.FIELD_STYLE_ALTERNATIVE**.



Data Objects

The Scout data object support was moved from the Scout platform to the module `org.eclipse.scout.rt.dataobject`. The package imports of all data object related classes therefore changed: From `org.eclipse.scout.rt.platform.dataobject` to `org.eclipse.scout.rt.dataobject`

Renamings

`org.eclipse.scout.rt.client.ui.desktop.datachange.DoChangeEvent`
`org.eclipse.scout.rt.client.ui.desktop.datachange.ItemDataChangeEvent` →

Dependencies

All modules which use data objects were extended with a dependency to `org.eclipse.scout.rt.dataobject`

- `org.eclipse.scout.rt.rest`
- `org.eclipse.scout.rt.mom.api`



Do you want to improve this document? Have a look at the [sources](#) on GitHub.