SAP BusinessObjects

SAP BusinessObjects BI Customization Guide

- SAP BusinessObjects Business Intelligence Suite 4.1

2013-07-02

SAP

Copyright

© 2013 SAP AG or an SAP affiliate company. All rights reserved. No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice. Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors. National product specifications may vary. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty. SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries. Please see http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark for additional trademark information and notices.

2013-07-02

Contents

Chapter 1 Document History....7

Chapter 2 Getting Started....9

2.1 About this guide....9
2.2 Before you begin....9
2.2.1 Customizing SAP BusinessObjects Business Intelligence platform....10
2.2.2 Customizing SAP Crystal Reports....11

Chapter 3 Business Intelligence Platform Installer Customization....15

3.1 Introduction....15
3.2 Quick start for the Business Intelligence platform (Windows)....15
3.3 Quick start for the Business Intelligence platform (Unix or Linux)....16
3.4 To download the server installation program....17
3.5 Planning the customization process....18
3.5.1 Best practices....18
3.6 Creating the configuration file....19
3.6.1 Configuration file overview....20
3.6.2 Renaming the product....21
3.6.3 Customizing user input....28
3.6.4 Removing installation screens....29
3.6.5 Embedding a keycode....30
3.6.6 Removing features....30
3.6.7 Preventing prerequisite checks....31
3.6.8 Removing language packs....31
3.6.9 Preventing the WDeploy tool from running....32
3.6.10 Removing the default database....32
3.6.11 Changing resources....33
3.6.12 Removing items from the Collaterals folder....36
3.7 Running the tool....37
3.7.1 Command line parameters....38
3.8 Customizing update installation programs....40
3.8.1 Frequently asked questions about update installation programs....40

3.8.2 Quick start for update installation programs....41
3.8.3 How to customize update installation programs....42
3.9 IDs and codes for BI Platform customization....44
3.9.1 Feature IDs....44
3.9.2 Shortcut deployment unit IDs (Windows only)....47
3.9.3 String IDs....48
3.9.4 Language codes....49
3.9.5 Installation screen and property IDs....50

Chapter 4 Web Application Customization....61

4.1 Introduction....61
4.1.1 Key concepts....61
4.1.2 Testing your customizations....63
4.2 Quick start....63
4.3 Customizing BI launch pad....64
4.3.1 To customize the favicon image....65
4.3.2 To customize logos....65
4.3.3 Customizing other user interface elements....65
4.3.4 Working with BI workspaces and compound modules....72
4.3.5 To change the name of BI launch pad....73
4.4 Customizing OpenDocument....74
4.5 Customizing the Crystal Reports JavaScript viewer....75
4.5.1 Customizing the viewer....76

Chapter 5 SAP Crystal Reports 2011 Customization....79

5.1 Introduction....79
5.2 Quick start for Crystal Reports....79
5.3 To download the installation program....80
5.4 Planning the customization process....81
5.4.1 Best practices....81
5.5 Creating the configuration file....82
5.5.1 Configuration file overview....83
5.5.2 Renaming the product....84
5.5.3 Customizing default user input....91
5.5.4 Removing installation screens....92
5.5.5 Embedding a keycode....92
5.5.6 Removing features....93
5.5.7 Preventing prerequisite checks....94
5.5.8 Removing language packs....94
5.5.9 Changing resources....95

5.5.10 Removing items from the Collaterals folder....98
5.6 Customizing the report designer....99
5.6.1 Customizing the splash screen....99
5.6.2 Customizing the start page....99
5.6.3 Customizing menu strings....100
5.6.4 Deploying the OEM customization file....102
5.7 Running the tool....102
5.7.1 Command line parameters....103
5.8 Customizing update installation programs....105
5.8.1 Frequently asked questions about update installation programs....105
5.8.2 Quick start for update installation programs....106
5.8.3 How to customize update installation programs....107
5.9 IDs and codes for Crystal Reports customization....109
5.9.1 Feature IDs....109
5.9.2 Shortcut deployment unit IDs....113
5.9.3 String IDs....113
5.9.4 Language codes....114
5.9.5 Installation screen and property IDs....115

Appendix A More Information....119

Document History

The following table provides an overview of the enhancements made to this document.

Version	Date	Description
SAP BusinessObjects Business Intelligence Suite 4.1	May, 2013	First release of this document.

Getting Started

2.1 About this guide

The SAP BusinessObjects Business Intelligence Suite provides a set of tools and templates that allow you to customize the SAP BusinessObjects Business Intelligence platform and the SAP Crystal Reports Designer. This guide shows you how to use these tools and templates to create your desired customizations.

Depending on the needs of your customers, you can remove features and language packs to reduce the size of the installation program and the installed product. And if you want to differentiate your system and apply your own unique corporate branding, you can personalize the appearance of your products, including product name, logos, colors, and other elements of the user interface. Your customization can be as simple as a logo change, or as detailed as a complete re-skinning.

The best thing is that your customizations are supported throughout the life cycle of the products. It is easy to maintain your changes during future upgrades and updates.

This guide is meant for anyone customizing SAP BusinessObjects Business Intelligence Suite products. You won't need to read the entire document; the Before you begin section describes the relevant workflows for each major area of product customization and tells you where to find the information you need.

Guide conventions

The following variables are used throughout this guide.

Variable	Description
<INSTALLDIR>	The file path where the BI platform is installed. On a Windows machine, the default file path is `C:\Program Files (x86)\SAP BusinessObjects\`.

2.2 Before you begin

This guide covers the different types of customization for the different products in the SAP BusinessObjects Business Intelligence suite. You need to read only those sections that cover the products you are planning to customize.

2.2.1 Customizing SAP BusinessObjects Business Intelligence platform

You can customize many aspects of the Business Intelligence platform deployment:

- Customize the installation program.

  You can remove features, language packs, and resources to reduce the size of the installed product, rename the product, change images, hide unwanted installation screens, embed a keycode, and pre-populate user input.

  See the Introduction for “Business Intelligence Platform Installer Customization” in this document.

- Customize BI launch pad and OpenDocument web applications.

  You can change the titles and the URLs that are used to access web applications. You can change the appearance and branding of these applications using custom images and Cascading Style Sheets (CSS).

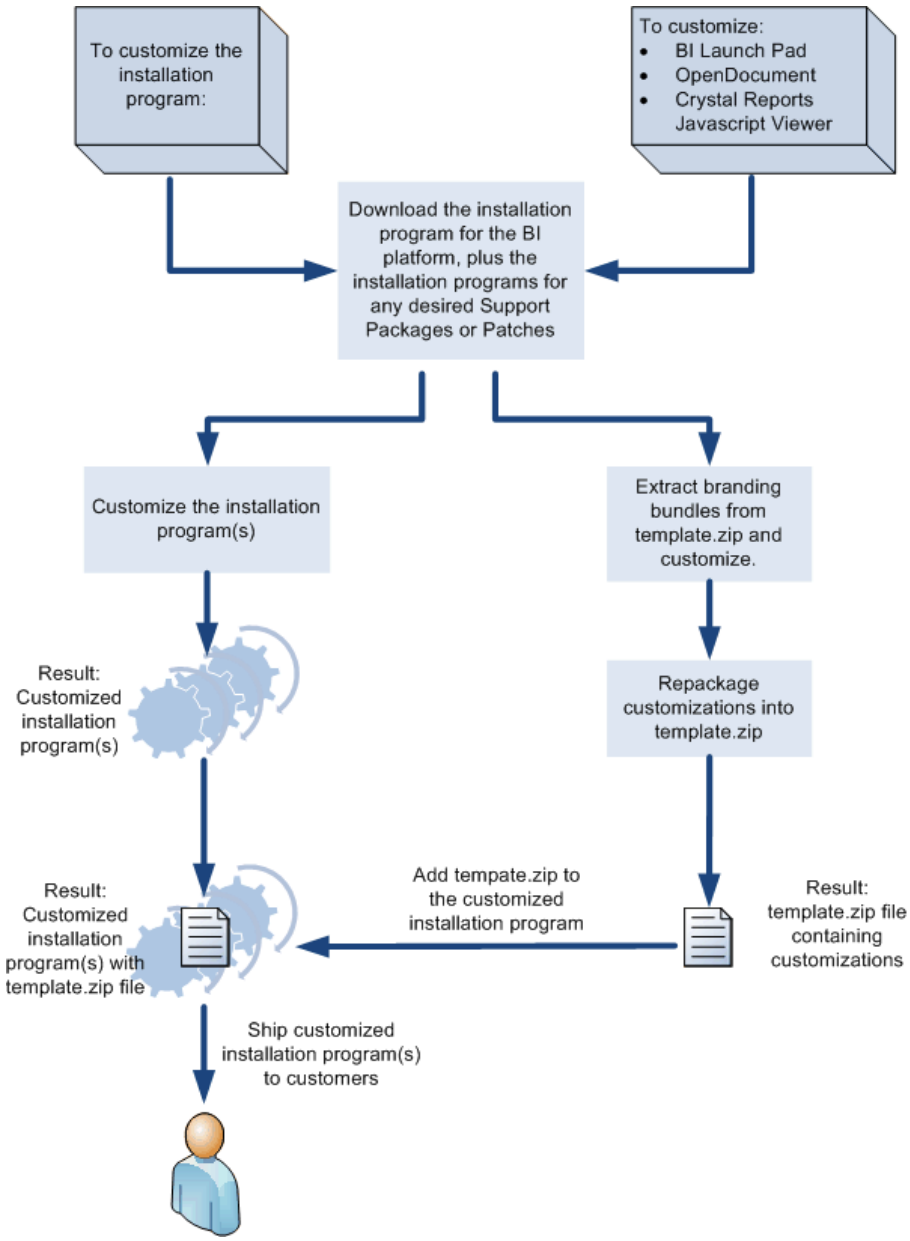
  See the Introduction for “Web Application Customization” in this document.

- Customize the Crystal Reports JavaScript API report viewer.

  You can change the logo and customize the visual style of the viewer using custom images and Cascading Style Sheets (CSS). You can add your own event and action listeners to the existing JavaScript API, or add your own external JavaScript files.

  See Customizing the Crystal Reports JavaScript viewer in this document.

You can customize the web applications, the installation program, or both. The following diagram illustrates the workflow where all types of customizations are performed:



2.2.2 Customizing SAP Crystal Reports

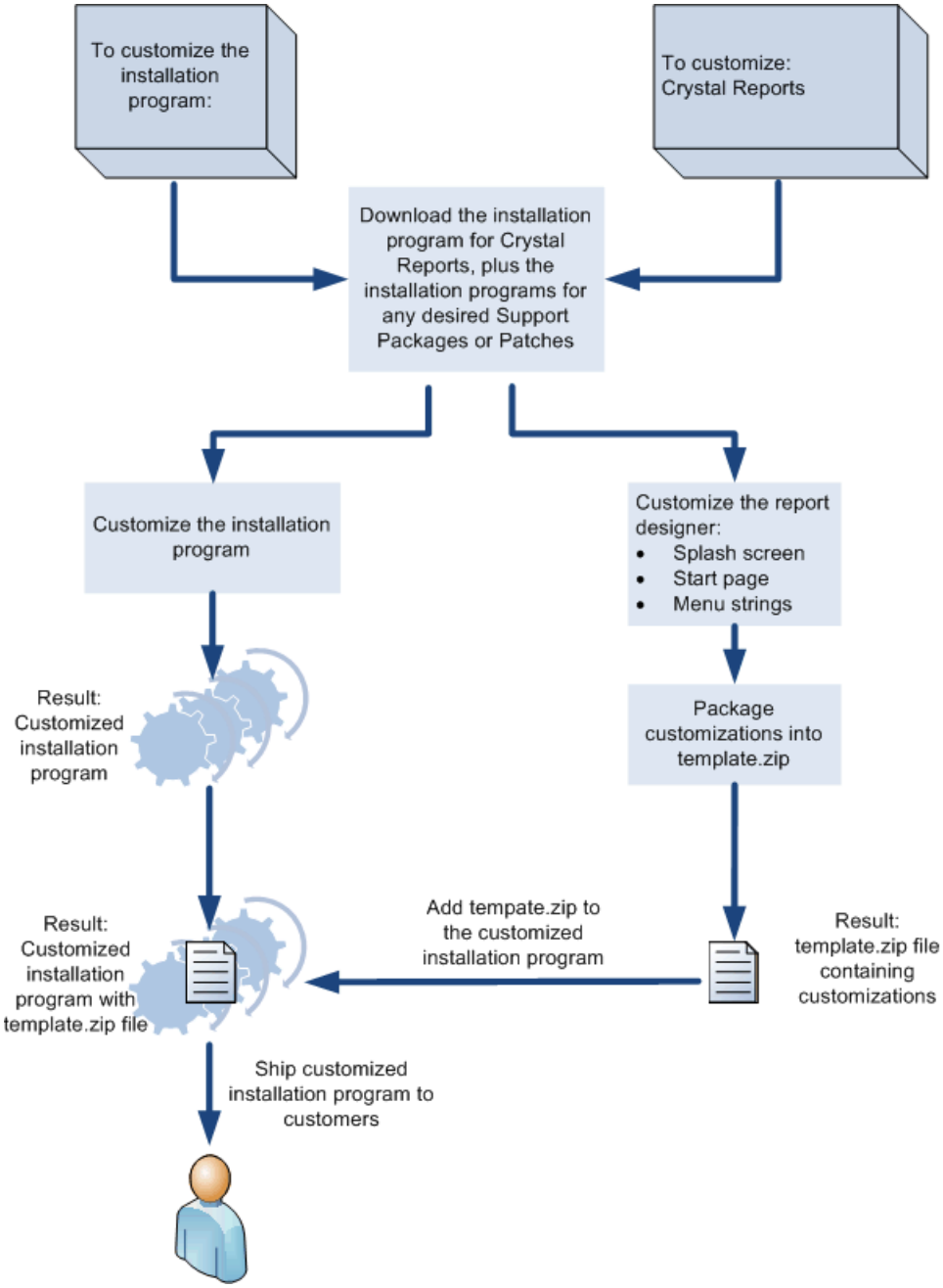
There are many customizations you can perform to enhance and personalize the design and customer experience for your SAP Crystal Reports users:

- Install and run the SAP BusinessObjects customization tool. For details, see Quick start for Crystal Reports.
- If you want to customize the SAP Crystal Reports installation program, you can change its appearance, hide unwanted screens from users in the wizard, and remove unused features to reduce the installed product size on client machines.

  See the Introduction for “SAP Crystal Reports 2011 Customization” in this document.
- If you want to customize the report designer, you can change the default splash screen or start page. You can also customize the product name, menus, and other assets of the report designer.

  See Customizing the report designer in this document.

The following diagram illustrates the workflow where all types of customizations are performed:


To customize the installation program:
To customize: Crystal Reports
Download the installation program for Crystal Reports, plus the installation programs for any desired Support Packages or Patches
Customize the installation program
Customize the report designer:
• Splash screen
• Start page
• Menu strings
Result: Customized installation program
Package customizations into template.zip
Result: Customized installation program with template.zip file
Add tempate.zip to the customized installation program
Result: template.zip file containing customizations
Ship customized installation program to customers

Business Intelligence Platform Installer Customization

3.1 Introduction

The SAP BusinessObjects Business Intelligence platform can be repackaged and sold by partners. You can customize the installed product and the installation program in order to target a specific customer base, or to resell it as part of your own product. The SAP BusinessObjects customization tool customizes the SAP BusinessObjects Business Intelligence platform and its installation program with changes such as the following:

- Reducing the product size
- Renaming the product
- Changing default properties in the installation program
- Hiding screens in the installation program

To make customizations, you write a configuration file to specify the changes then run the SAP BusinessObjects customization tool to create a customized installation program. Customers can use this installation program to install a customized version of the product.

The customization tool is available for Windows and Unix. It can be used to customize a full installation program, a Support Package installation program, and a Patch installation program.

Note:
This tool does not perform customizations on the SAP BusinessObjects Business Intelligence platform Client Tools.

3.2 Quick start for the Business Intelligence platform (Windows)

This section shows you how to run the customization tool to create a customized installation program for the SAP BusinessObjects Business Intelligence platform (BI Platform). It uses the sample configuration file that is provided with this tool. When you are finished this tutorial, you can run your customized installation package and install a customized version of the BI Platform.

The customizations include changing the default installation type, removing features, hard-coding the product keycode, changing the default installation folder, renaming the product, and changing the Windows **Start** menu shortcut for the Central Configuration Manager feature. They are described in more detail in the configuration file.

1. Set up the customization tool.
   a. Create a working folder on your development machine, for example `C:\SAPCustomTool\packages`.
   b. Copy the contents of the BI Platform installation package to `C:\SAPCustomTool\packages`.

      The installation package contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries. See To download the server installation program for instructions.
   c. (Optional). Add your keycode to the sample configuration file.

      In an XML editor, open the file `C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool\example_customization_win_boe.xml` and replace the phrase `PutYourKeyCodehere` with your BI Platform keycode. If you do not enter your keycode into the configuration file, you can use the Central Management Console to enter it after installing the customized BI Platform.
   d. Create the folder `C:\SAPCustomTool\output`.

      This folder must be empty.
   e. Run the following command from the command prompt: `cd C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool`

      The folder `CustomizationTool` contains the executable `customizationtool.exe` and the sample configuration file `example_customization_win_boe.xml`.
2. Run the following command from the command prompt:

   ```
   customizationtool.exe xml=example_customization_win_boe.xml packageDir=C:\SAPCustomTool\packages outputDir=C:\SAPCustomTool\output logDetail=error > C:\oemlog.log
   ```

   Verify that the customized installation program was created at `C:\SAPCustomTool\output`. Ensure no errors were reported in the log file `oemlog.log`.

   Note:
   The customization tool may take several minutes to complete. You can check its progress by viewing the log file.
3. Use `C:\SAPCustomTool\output\setup.exe` to run the customized SAP BusinessObjects Business Intelligence platform installation program.

The BI Platform is installed with the customizations described in the configuration file.

3.3 Quick start for the Business Intelligence platform (Unix or Linux)

This section shows you how to run the customization tool to create a customized installation program for the SAP BusinessObjects Business Intelligence platform (BI Platform). It uses the sample configuration file that is provided with this tool. When you are finished this tutorial, you can run your customized installation package and install a customized version of the BI Platform.

The customizations include changing the default installation type, removing features, hard-coding the product keycode, changing the default installation folder, and renaming the product. They are described in more detail in the configuration file.

1. Set up the customization tool.
   a. Create a working folder on your development machine, for example `/usr/jdoe/bip/package`.
   b. Copy the contents of the BI Platform installation package to `/usr/jdoe/bip/package`.

      The installation package contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries. See To download the server installation program for instructions.
   c. (Optional). Add your keycode to the sample configuration file.

      In an XML editor, open the file `/usr/jdoe/bip/package/Collaterals/Tools/CustomizationTool/example_customization_linux_boe.xml` and replace the phrase `PutYourKeyCodehere` with your BI Platform keycode. If you do not enter your keycode into the configuration file, you can use the Central Management Console to enter it after installing the customized BI Platform.
   d. Create the folder `/usr/jdoe/bip/output`. This folder must be empty.
   e. Change to the folder `/usr/jdoe/bip/package/Collaterals/Tools/CustomizationTool`.

      This folder contains the executable `customizationtool.sh` and the sample configuration file `example_customization_linux_boe.xml`.
2. Run the following command from the command prompt:

   ```
   ./customizationtool.sh xml=example_customization_linux_boe.xml packageDir=/usr/jdoe/bip/package outputDir=/usr/jdoe/bip/output logDetail=error &> custombip.log
   ```

   The customizations that you see in the installation program and in the installed product are described in the configuration file `/usr/jdoe/bip/package/Collaterals/Tools/CustomizationTool/example_customization_linux_boe.xml`.

   Verify that the customized installation program was created at `/usr/jdoe/bip/output`. Ensure no errors were reported in the log file `custombip.log`.

   Note:
   The customization tool may take several minutes to complete. You can check its progress by viewing the log file.
3. From the commmand prompt, use `/usr/jdoe/bip/output/setup.sh` to run the customized BI Platform installation program.

The BI Platform is installed with the customizations described in the configuration file.

3.4 To download the server installation program

1. Go to https://service.sap.com/support > **Software Downloads**.
2. On the "Find your software" tab, under the "A–Z Index", click **Installations and Upgrades**.

3. Select **B** > **SBOP BI platform (former SBOP Enterprise)** > **SBOP BI PLATFORM (ENTERPRISE)** > **SBOP BI PLATFORM 4.1**.
4. Select **Installation and Upgrade** and then select your platform.
5. Select all of the packages titled "SBOP BI PLATFORM <version> SERVER" plus any additional add-on products you require, then follow the instructions on the website to download and extract the packages.

Note:
The software may take a long time to download, and you may need to contact the system administrator to ensure your company's firewall will not terminate the download process.

Support Packages and Patches are installation programs that contain updates to BI platform software. You can download them from https://service.sap.com/support > **Software Downloads**. On the "Find your software" tab, under the "A–Z Index", click **Support Packages and Patches**. For more information on installing Support Packages and Patches, see the SAP BusinessObjects BI Suite Update Guides.

3.5 Planning the customization process

To use the SAP BusinessObjects customization tool:

1. Download the installation program. See To download the server installation program.
2. Decide what customizations are required. See Creating the configuration file.
3. Write the configuration file to specify the customizations.
4. Run the customization tool to create a customized installation program.
5. Run the customized installation program to install a customized version of SAP BusinessObjects Business Intelligence platform.

3.5.1 Best practices

This section provides recommendations for creating a customized installation program.

Validate the configuration file

You may want to validate the configuration file before running the tool. Use the `validate` command-line parameter.

Reduce product size

Customers prefer a smaller installation program and a smaller installed product. To keep the product as small as possible:

- Remove any language packs that are not required.
- Remove any features that are not required.

- Remove any items from the `Collaterals` folder that are not required.
- Remove the default database if it is not required.

Apply customized names consistently

The product name and version number appear in several places in the installation program and in the installed product. Ensure you verify customizations in the following locations:

- Product name, product version, and product major version
- Windows **Start** menu entry and all feature shortcuts
- Windows "Add Remove Program" utility
- Default installation folder

Consider name change in all languages

It is good practice to consider how the customized name appears in all supported languages.

Modify patch installation programs to be consistent with the main installation program

You must apply the same customizations to Support Packages and Patches as you applied to the main release. If you release a customized main installation program, then try to release a Support Package or Patch installation program with different customizations, you might see unpredictable results which might not be repairable using standard rollback procedures.

Test rollback, modify, and repair installations for Support Packages and Patches

Rollback, modify, and repair are supported for customized Support Packages and Patches, provided they have been customized in a manner consistent with the main installation package. It is recommended to test these scenarios.

Related Topics

- Command line parameters

3.6 Creating the configuration file

The following section describes the customizations you can make to the installation program by editing the configuration file:

- Renaming the product
  - Customizing the product name and version number
  - Customizing the Windows **Start** menu shortcuts
  - Customizing the Windows "Add Remove Program" utility
  - Customizing the installation folder
- Customizing user input
- Removing installation screens
- Embedding a keycode

- Removing features
- Preventing prerequisite checks
- Removing language packs
- Preventing the WDeploy tool from running
- Removing the default database
- Changing resources
  - Changing the images in the installation program
  - Changing the license agreement
- Removing items from the Collaterals folder

3.6.1 Configuration file overview

The SAP BusinessObjects customization tool uses information in the configuration file to perform the customizations. The configuration file is an XML document, and you use XML elements to describe your customizations. The sample configuration file is contained in this folder in the installation program:

Platform	Location of sample configuration file
Windows	`Collaterals\Tools\CustomizationTool\example_customization_win_boe.xml`
Unix or Linux	`Collaterals/Tools/CustomizationTool/example_customization_linux_boe.xml`

The file must have this format:

```
<oem name="<Any name>">
  <cloneProduct sourceId="product.businessobjects64-4.0-core-32">
  ...
  ...
  </cloneProduct>
</oem>
```

The configuration file for the full installation program can have any name, for example, `oem.xml`.

The configuration file for the Support Package installation program is described in the section How to customize update installation programs.

Note:
The configuration file must be written in correct XML syntax. Use an XML editor to create and edit the file, and verify the format is correct before running the tool.

Example:

This example file specifies these customizations:

- Change the product long name to Custom Company Server for all languages.
- Change the product short name to Custom CS for all languages.
- Remove the installation screen titled "Choose Installation Type" and set the installation type to Custom.
- Specify that the only language packs included in the installation package are English, French, German, Italian, and Chinese.

```
<oem name="CustomCompanyServer">
  <cloneProduct sourceId="product.businessobjects64-4.0-core-32">

    <replaceString id="product.boe64_name" value="Custom Company Server" lang="all"/>
    <replaceString id="product.boe64_shortname" value="Custom CS" lang="all"/>

    <replaceProperty id="InstallType" defaultValue="custom"/>
    <removeDialog id="ChooseInstallType.dialog"/>

    <languageIncludeList value="en;fr;de;it;zh_CN"/>

   </cloneProduct>
</oem>
```

3.6.2 Renaming the product

You can rename the product in the following ways:

- Customize the product name and version number.
- Customize the Windows "Add or Remove Programs" entry. (Windows only)
- Customize the "Start" menu entry for feature shortcuts. (Windows only)
- Customize the default installation folder.

The following sections explains these steps.

3.6.2.1 Customizing the product name and version number

You can customize the product name and version number. Use the `replaceString` element with the desired string ID:

```
<replaceString id="<string id>" value="<new value>" lang="<language list>"/>
```

There are four strings that represent the product name and version number: the product long name, the product short name, the product version number, and the product major version number. The full product name is composed of the product long name and the version number. The product short name and product major version are used in the Windows shortcut menu.

Table 3-2: Product name and version number

String description	String ID	Default value
Product long name	`product.boe64_name`	SAP BusinessObjects BI platform
Product short name	`product.boe64_shortname`	BI platform server
Product version	`product_version`	4.1
Product major version	`product_majorversion`	4

Note:
You should customize the product version and product major version together. For example, if you change product version to 1.0 you should also customize product major version to 1. Otherwise the version number in the menus will not match the version number in the product.

You can specify a new name for each language.

Example:

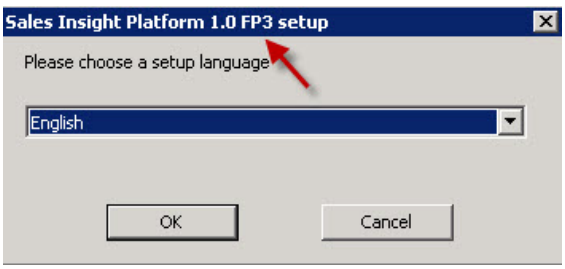
Change the product long name to Sales Insight Platform and the product short name to Sales Platform for English. Change the product long name to Sales Insight Platform (French) and the product short name to Sales Platform (French) for French. Change the product version to 1.0 and the product major version to 1 for both French and English. The product name and version number in languages other than English and French will remain as the default value.

```
<replaceString id="product.boe64_name" value="Sales Insight Platform" lang="en"/>
<replaceString id="product.boe64_shortname" value="Sales Platform" lang="en"/>

<replaceString id="product.boe64_name" value="Sales Insight Platform (French)" lang="fr"/>
<replaceString id="product.boe64_shortname" value="Sales Platform (French)" lang="fr"/>

<replaceString id="product_version" value="1.0" lang="en;fr"/>
<replaceString id="product_majorversion" value="1" lang="en;fr"/>
```

The customization appears below. Notice the version number “FP3” is not removed:



To remove instances of “FP3” from the installation program

When you run the installation program, you may see instances of “FP3” in the product name. To remove “FP3”, modify the lines in the following files:

File name	Original line	Modified line
dunit\product.businesobjects64-4.0-core-32\setup.ui.framework\uitext\BusinessObjects64\product.lang_<language code>.uitext.xml	<string id="product name_patch" value=" FP3"/>	<string id="product name_patch" value=""/>
dunit\product.businesobjects64-4.0-core-32\setup.ui.framework\uitext\framework\setup.ui.framework.lang_<language code>.uitext.xml	<string id="product_patch" value="FP3"/>	<string id="product_patch" value=""/>
Same as above	<string id="product_patch_prespace" value=" FP3"/>	<string id="product_patch_prespace" value=""/>

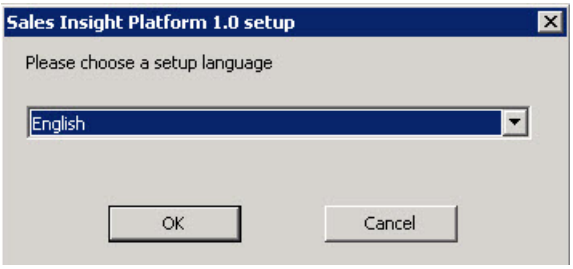
You must modify one file for every language that the installation program supports. For a list of language codes, see Language codes. When you run the customization tool and then run the installation program, all instances of “FP3” will be removed. This process will be simplified in a future release.

Example:

To remove “FP3” from the English installation program, modify the following files:

- `product.lang_en.uitext.xml`
- `setup.ui.framework.lang_en.uitext.xml`

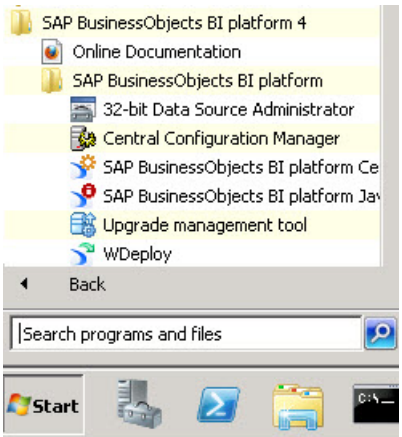
The customization appears below:



3.6.2.2 Customizing the Windows Start menu shortcuts (Windows only)

The Windows **Start** menu contains shortcuts for features such as the Central Management Console and BI launch pad. You can customize the name, location, and tooltip for each shortcut. Any shortcut that you do not customize will be grouped under the default **Start** menu, "SAP BusinessObjects BI platform 4".

The default **Start** menu in English installations looks like this:



Use the `shortcut` element to customize the location, shortcut name, and tooltip for each feature:

```
<shortcut duSourceId="<shortcut deployment unit ID>">
  <arg id="linkFullPath" value="<full path to shortcut link>" lang="<language list>"/>
  <arg id="description" value="<tooltip string>" lang="<language list>"/>
</shortcut>
```

Attribute	Value
duSourceId	The shortcut deployment unit ID that you want to modify. Typical values in clude: • product.businessobjects64.shortcut.ccm-4.0-core Central Configuration Manager • product.businessobjects64.shortcut.infoview-4.0-core BI launch pad • product.businessobjects64.shortcut.cmc-4.0-core Central Management Console For a complete list of sourceId values, see Shortcut deployment unit IDs (Windows only).
linkFullPath	The full path to the shortcut link. Be sure to add .lnk to shortcut link or the link will not be created. You can put the shortcut link on the **Start** menu or you can put it on the desktop. The SAP BusinessObjects customization tool will create the links correctly. You can specify one link for each language. For a list of language codes, see Language codes.
description	The tooltip string to display when the user hovers the mouse over the shortcut. You can specify one tooltip for each language.

Note:

You can customize the link, but not the tooltip, for the following shortcuts:

- BI Launchpad (formerly InfoView)
- Online documentation
- WACs stored in InfoView
- Web Application Container Server

This will be resolved in a future release.

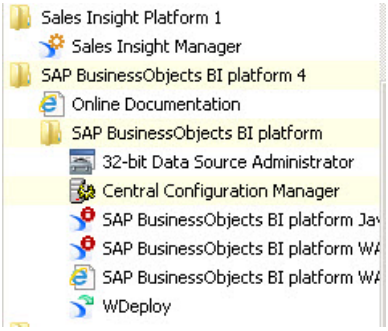
Example:

This example customizes the name of the Central Management Console shortcut to Sales Insight Manager for English and Sales Insight Manager (French) for French, and places the shortcuts under the **Start** menu entry called Sales Insight Platform 1. It also customizes the tooltip to Launch Sales Manager for English and Launch Sales Manager (French) for French. The shortcut name and tooltip will remain unchanged for all other languages.

```
<shortcut duSourceId="product.businessobjects64.shortcut.cmc-4.0-core">
   <arg id="linkFullPath" value="[programmenufolder]\Sales Insight Platform 1\Sales Insight Manager.lnk"
lang="en"/>
   <arg id="linkFullPath" value="[programmenufolder]\Sales Insight Platform 1 (French)\Sales Insight Manager
```

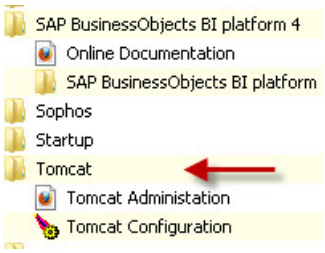
```
 (French).lnk" lang="fr"/>
   <arg id="description" value="Launch Sales Manager" lang="en"/>
   <arg id="description" value="Launch Sales Manager (French)" lang="fr" />
</shortcut>
```

The customization appears below:



Modifying the tomcat shortcut

There are two links for the Tomcat shortcut: "Tomcat Administration" and "Tomcat Configuration", shown below:



You must take extra steps to customize this shortcut. Use this `shortcut` element to customize the "Tomcat Administration" link. Note the `pathToTarget` element.

```
<shortcut duSourceId="product.businessobjects64.shortcut.tomcat-4.0-core" pathToTarget="http://localhost:[Tom
catConnectionPort]/manager/html">
  <arg id="linkFullPath" value="<full path to shortcut link>" lang="<language list>"/>
  <arg id="description" value="<tooltip string>" lang="<language list>"/>
</shortcut>
```

Use this `shortcut` element to customize the "Tomcat Configuration" link. Note the `pathToTarget` element.

```
<shortcut duSourceId="product.businessobjects64.shortcut.tomcat-4.0-core" pathToTarget="[INSTALLDIR]tom
cat\bin\tomcat7w.exe">
  <arg id="linkFullPath" value="<full path to shortcut link>" lang="<language list>"/>
  <arg id="description" value="<tooltip string>" lang="<language list>"/>
</shortcut>
```

Example:

This example customizes the name of the "Tomcat Administration" shortcut to tomcat(english and french) shortcut1 for English and French installations, and tomcat (German) shortcut1 for German installations. It places the shortcuts in the **Start** menu entry called Company Programs . It customizes

the tooltip to tomcat(english and french) shortcut1 for English and French, tomcat (all others) shortcut1 for all other languages.

```
<shortcut duSourceId="product.businessobjects64.shortcut.tomcat-4.0-core" pathToTarget="http://localhost:[Tom
catConnectionPort]/manager/html">
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat(english and french) shortcut1.lnk"
 lang="en;fr"/>
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat (German) shortcut1.lnk"
lang="de"/>
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat (all others) shortcut1.lnk"
lang="it;zh_cn"/>
  <arg id="description" value="tomcat(english and french) shortcut1" lang="en;fr"/>
  <arg id="description" value="tomcat (German) shortcut1" lang="de" />
  <arg id="description" value="tomcat (all others) shortcut1" lang="it;zh_cn" />
</shortcut>

<shortcut duSourceId="product.businessobjects64.shortcut.tomcat-4.0-core" pathToTarget="[INSTALLDIR]tom
cat\bin\tomcat7w.exe">
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat(english and french) shortcut2.lnk"
 lang="en;fr"/>
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat (German) shortcut2.lnk"
lang="de"/>
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\tomcat (all others) shortcut2.lnk"
lang="it;zh_cn"/>
 <arg id="description" value="tomcat(english and french) shortcut2" lang="en;fr"/>
  <arg id="description" value="tomcat (German) shortcut2" lang="de" />
  <arg id="description" value="tomcat (all others) shortcut2" lang="it;zh_cn" />
</shortcut>
```

3.6.2.3 Customizing the Windows Add Remove Program utility (Windows only)

You can customize the display name, the publisher, and the icon in the Windows "Add Remove Program" (ARP) utility. You cannot customize the version number. Use the following element:

```
<arp duSourceId="product.businessobjects64.arp-4.0-core">
  <arg id="publisher" value="<publisher name>"/>
  <arg id="display_name" value="<product name>"  lang="<language list>"/>
  <arg id="display_icon" value="<full path to icon>"/>
</arp>
```

Icons displayed in the Windows "Add Remove Program" utility are typically 16x16. Refer to Windows documentation for complete information on creating the icon.

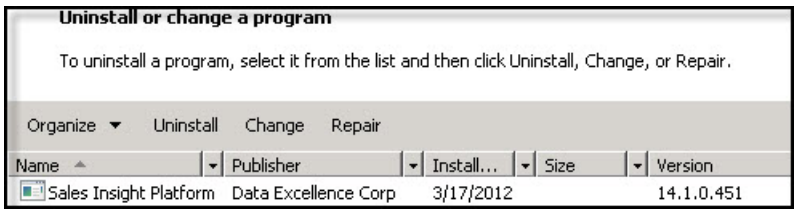
Example:

Change the product name in the Windows ARP utility to Sales Insight Platform. This change will only affect English installations. Change the publisher to Data Excellence Corp. Replace the display icon with the icon located at `C:\SAPCustomTool\DEC_logo.ico`.

Note:
To use this example you must put an icon called `DEC_logo.ico` in the location `C:\SAPCustomTool`.

```
<arp duSourceId="product.businessobjects64.arp-4.0-core">
  <arg id="publisher" value="Data Excellence Corp"/>
  <arg id="display_name" value="Sales Insight Platform" lang="en"/>
  <arg id="display_icon" value="C:\SAPCustomTool\DEC_logo.ico"/>
</arp>
```

The customization appears below:



3.6.2.4 Customizing the installation folder

You can customize the default installation folder. Use the `replaceProperty` element with `id="InstallDir"`:

```
<replaceProperty id="InstallDir" defaultValue="<default installation folder>"/>
```

Use this element for both Windows and Unix installations.

Example:

Change the default installation folder to `C:\Program Files (x86)\SalesDataInsight`.

```
<replaceProperty id="InstallDir" defaultValue="C:\Program Files (x86)\SalesDataInsight"/>
```

3.6.3 Customizing user input

You can customize the default value of the user input that is collected by the installation program. Use the `replaceProperty` element with `id="<property id>"` and the new default value:

```
<replaceProperty id="<property id>" defaultValue="<value to use as default value>"/>
```

For a list of property IDs, see Installation screen and property IDs.

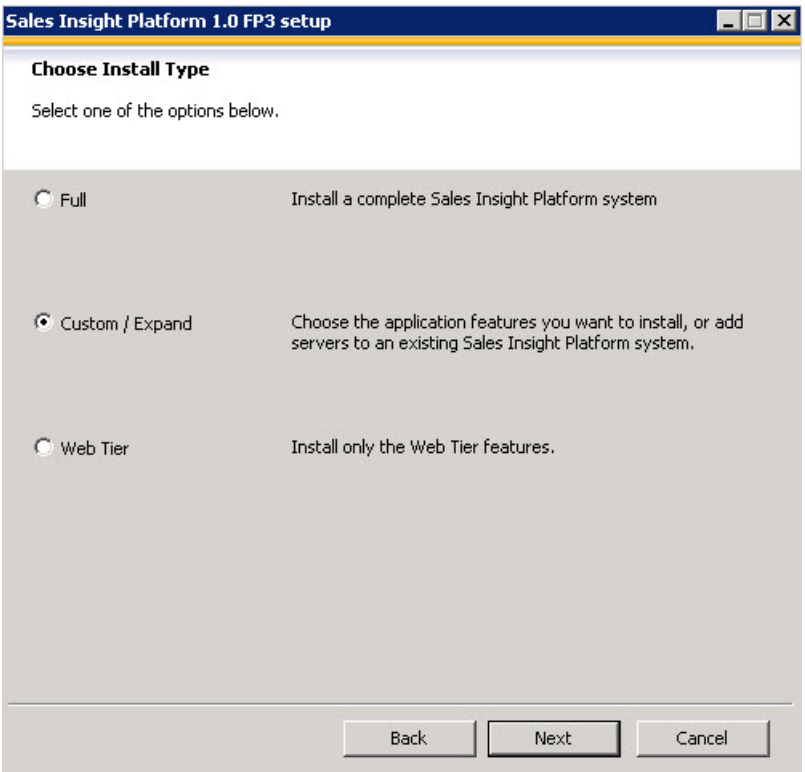
The Windows installation program collects user input using dialog boxes, radio buttons, and other user interface elements. The Unix and Linux installation program collects user input using the console entry. Both installation programs are customized in the same way.

Example:

On the installation screen called "Choose Install Type", the default install type is "Full". This example changes the default install type to "Custom/Expand".

```
<replaceProperty id="InstallType" defaultValue="custom"/>
```

The customization appears below:



3.6.4 Removing installation screens

You can remove installation screens from the installation program. Use the `removeDialog` element with the installation screen ID:

```
<removeDialog id="<installation screen ID>"/>
```

For a list of installation screen IDs, see Installation screen and property IDs.

Example:

This example shows how to remove the installation screen titled "Select Java Web Application Server".

```
<removeDialog id="ChooseWebAppServer.dialog"/>
```

3.6.5 Embedding a keycode

You can embed a keycode in the installation program so the customer does not need to enter one. This task involves:

- Providing a default value for the keycode
- Removing the installation screen in which the user enters a keycode

Example:

Use the `replaceProperty` element with `id="ProductKey"` to provide a default keycode. Keycodes must have the format `XXXXX-XXXXXXX-XXXXXXX-XXXXXXX-XX`.

Use the `removeDialog` element with `id="EnterProductKey.dialog"` to remove the installation screen for the license key.

```
<replaceProperty id="ProductKey" defaultValue="XXXXX-XXXXXXX-XXXXXXX-XXXXXXX-XX"/>
<removeDialog id="EnterProductKey.dialog"/>
```

Related Topics

- Installation screen and property IDs
- Customizing user input
- Removing installation screens

3.6.6 Removing features

SAP BusinessObjects Business Intelligence platform is composed of many optional features. You can remove a feature from the installation program. Use the `removeFeature` element with `id="<feature id>"`:

```
<removeFeature id="<Feature ID>"/>
```

For a list of feature IDs, see Feature IDs.

When you specify that a feature will be removed, the SAP BusinessObjects customization tool removes all executables, installation screens, and other files that belong to that feature. Removing unnecessary features is a good way to reduce the size of the customized product.

Example:

Remove the Crystal Reports feature. This removes all Crystal Reports servers, files, and resources.

```
<removeFeature id="CrystalReportsServers"/>
```

Related Topics

- Feature IDs

3.6.7 Preventing prerequisite checks

Prerequisites are conditions that must exist on the host machine in order for the installation program to succeed. The installation program verifies the existence of these prerequisites before starting, and displays the results in the "Prerequisite check" screen. Removing the "Prerequisite check" screen prevents prerequisite checks from being performed. Use the `removeDialog` element with `id="CheckPreRequisites.dialog"`.

Note:
It is recommended that you remove this installation screen only if you are performing the prerequisite checks by some other means. If the prerequisites are not met, the installation program will fail.

Example:

This example removes the "Prerequisite check" screen and prevents prerequisite checks from being performed.

```
<removeDialog id="CheckPreRequisites.dialog"/>
```

3.6.8 Removing language packs

The installation program allows the user to select which language packs to install. A language pack contains translated versions of all the strings that are used by the installed product. By default, all possible language packs are included in the installation program. You can specify which language packs to include. Use the `languageIncludeList` element with a list of language codes:

```
<languageIncludeList value="<list of language codes>"/>
```

For a list of language codes, see Language codes.

Note:
Language packs can be large. The installation program will be smaller if fewer language packs are included.

Example:

Include English, French, and German language packs in the installation program. The user can select from this list during installation.

```
<languageIncludeList value="en;fr;de"/>
```

3.6.9 Preventing the WDeploy tool from running

If the user installs a web application server other than the default one, the WDeploy tool will run when the installation is finished. On Windows platforms, WDeploy is a GUI tool while on Unix and Linux platforms, it is a script.

You can turn this feature off. Use with the `replaceProperty` element with `defaultValue="0"`

Example:

```
<replaceProperty id="LaunchWDeploy" defaultValue="0"/>
```

3.6.10 Removing the default database

The default database is included with the installation program, and customers can choose to use it as the system database. The default database is Sybase SQL Anywhere.

If the default database is not required you can remove it and force customers to choose another. Removing the default database is a good way to reduce the size of the installation program.

To remove the default database

Use the `<removeFeature>` element with `id="PlatformServers.IntegratedDB.SQLAnywhere"`. You may also want to remove the installation screen titled "Select Default or Existing Database" and set the user input property to **Use an existing database**.

Example:

This example removes the default database. It also removes the installation screen titled "Select Default or Existing Database", and sets the user input property to **Use an existing database**.

```
<removeFeature id="PlatformServers.IntegratedDB.SQLAnywhere"/>
<removeDialog id="SelectDataSource.dialog"/>
<replaceProperty id="SelectIntegratedDatabase" defaultValue="0"/>
```

3.6.11 Changing resources

The installation program stores image and text files as resources in this folder:

`\dunit\product.businessobjects64-4.0-core-32\setup.ui.framework\resources`

You can customize the resources in this folder. Resources that are commonly customized include:

- Images in the installation program
- License agreement in the installation program

To customize a resource:

1. Create a custom resources folder, for example (on Windows)`C:\SAPCustomTool\MyResources`. The file can have any name, but will be visible to customers. Use the same folder for all resources that you customize.
2. Create a new resource with the same name and filepath as the original resource, and place it into the custom resources folder. See the related topics section for specific examples.
3. Add the `<resources>` element to the configuration file to specify the location of the custom resources folder, for example:

   ```
   <resources cleanTarget="no" sourcePath="C:\SAPCustomTool\MyResources"/>
   ```

cleanTarget attribute

If you set `cleanTarget='yes'`, the customization tool will delete the original `resources` folder and use only those resources included in the custom resources folder. This option is not recommended.

Related Topics

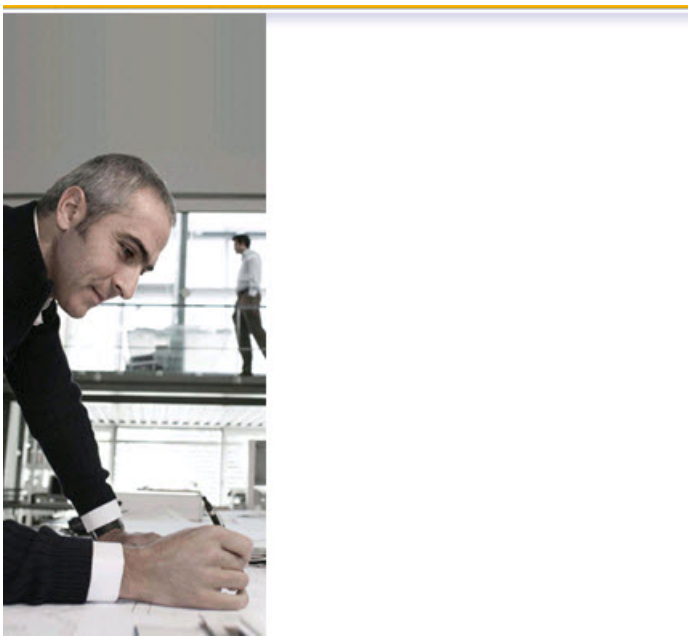
- Customizing the images in the installation program
- Customizing the license agreement

3.6.11.1 Customizing the images in the installation program

You can customize the images in the installation program including the welcome screen, the top image for all screens, and the billboard for the progress dialog. Images are stored as files in the resources folder:

```
dunit\product.businessobjects64-4.0-core-32\setup.ui.framework\resources
```

Table 3-5: Image files in the `resources` folder

Image name	File name	Size (W x H)	Default image
Welcome screen	dialog Full.bmp	500 x 400 px	
Top image for all screens	dialog Top.bmp	500 x 83 px	
Billboard for progress dialog	billboard.bmp	500 x 193 px	

You customize an image by creating a new image file, putting the file in the custom resources folder, and adding the `resources` element to the configuration file.

Example: Customizing the image in the welcome screen on Windows platforms

1. Create a folder called `MyResources` in the location `C:\SAPCustomTool`
2. Create a new image file called `dialogFull.bmp` and place it in the `C:\SAPCustomTool\MyResources` folder
3. Ensure the `resources` element exists in the configuration file as follows:

```
<resources cleanTarget="no" sourcePath="C:\SAPCustomTool\MyResources"/>
```

Related Topics

- Changing resources

3.6.11.2 Customizing the license agreement

You can customize the license agreement that is presented to the user during installation. License agreements are stored as text files in the resources folder:

```
dunit\product.businessobjects64-4.0-core-32\setup.ui.framework\resources\<language code>
```

For example, on Windows platforms, the English license agreement is located here:

```
dunit\product.businessobjects64-4.0-core-32\setup.ui.framework\resources\en\license_en.rft
```

On Unix and Linux platforms, the English license agreement is located here:

```
dunit/product.businessobjects64-4.0-core-32/setup.ui.framework/resources/en/license_en.txt
```

For a list of language codes, see Language codes.

You customize the license agreement by creating a new license file, putting the file in the custom resources folder, and adding the `resources` element to the configuration file.

Example: Customize the Japanese license agreement on Windows platforms

The Japanese license agreement is stored here:

```
dunit\product.businessobjects64-4.0-core-32\setup.ui.framework\resources\ja\license_ja.rtf
```

To customize the Japanese license agreement:

1. Create a folder called `ja` in the location `C:\SAPCustomTool\MyResources`.

2. Create a new license agreement file called `license_ja.rtf` and place it in the `C:\SAPCustomTool\MyResources\ja` folder.
3. Ensure the `resources` element exists in the configuration file as follows:

```
<resources cleanTarget="no" sourcePath="C:\SAPCustomTool\MyResources"/>
```

Related Topics

- Changing resources

3.6.12 Removing items from the Collaterals folder

The SAP BusinessObjects Business Intelligence platform installation program stores tools, samples, and documentation in the `Collaterals` folder of the installation program. By default, a customized installation program will contain the default `Collaterals` folder with the default contents. You can remove unnecessary items from the `Collaterals` folder in order to reduce the size of your customized installation program. Use the `collaterals` element with `cleanTarget="yes"` and `sourcePath="<full path to custom Collaterals folder>"`:

```
<collaterals cleanTarget="yes" sourcePath="<full path to custom Collaterals folder>"/>
```

Note:
You must set the `cleanTarget` attribute to `yes` so the customization tool will replace the original folder with the new folder.

To remove items from the Collaterals folder

1. Copy the contents of the existing `Collaterals` folder to a new location, for example (on Windows) `C:\SAPCustomTool\Utilities`.
2. Remove any items from `C:\SAPCustomTool\Utilities` that are not required by your customized installation program. See below for more information.
3. Add the `<collaterals>` element to the configuration file to specify the location of the custom collaterals folder, for example:

```
<collaterals cleanTarget="yes" sourcePath="C:\SAPCustomTool\Utilities"/>
```

Table 3-6: Description of items in the `Collaterals` folder

Folder	Description	When to remove
`Collaterals > Add-Ons > SAP`	Provides connectivity to SAP systems.	Remove if there is no need to connect to SAP systems.

Folder	Description	When to remove
`Collaterals > Add-Ons > Subversion`	Subversion is the default version control system that is used by Lifecycle Management (LCM).	Remove if the LCM feature is removed.
`Collaterals > Add-Ons > Tivoli Agent`	The server monitoring feature can integrate with IBM Tivoli, and this item provides the connectivity.	Remove if integration with IBM Tivoli is not required.
`Collaterals > Customization Template`	Required template files.	Do not remove this folder.
`Collaterals > DiagnosticsAgent7.3`	SAP Solution Manager Diagnostics (SMD) agent. SMD is used by SAP Support tools to troubleshoot installed product.	Remove if the SMD feature is removed.
`Collaterals > Docs`	Documentation in every language that SAP BusinessObjects Business Intelligence platform supports.	Remove any languages that are not included in the customized installation program. For a list of language codes, see Language codes.
`Collaterals > Tools > CustomizationTool`	The SAP BusinessObjects customization tool.	Remove this folder if the customers do not need to customize their own installation programs.
`Collaterals > Tools > LCM command line tool`	Command-line utility for Lifecycle Management (LCM).	Remove if the LCM feature is removed.
`Collaterals > Tools > wdeploy`	WDeploy is used to deploy web applications to web application servers other than Tomcat.	Not recommended to remove. Remove only if customers will use Tomcat exclusively.

3.7 Running the tool

The SAP BusinessObjects customization tool is included with the SAP BusinessObjects Business Intelligence platform installation package in this location:

`Collaterals\Tools\CustomizationTool`

On Windows platforms, the tool is named `customizationtool.exe`. On Unix and Linux platforms, the tool is named `customizationtool.sh`

This section explains the command line parameters.

Note:
The customization tool may take several minutes to complete. You can check its progress by viewing the log file.

Example:

This example runs the customization tool on a Windows platform. To use this example you must:

- Create a configuration file called `oem.xml` in the location `C:\SAPCustomTool`.
- Download the SAP BusinessObjects Business Intelligence platform installation package to the folder `C:\SAPCustomTool\packages`. See To download the server installation program.
- Create a folder called `output` in the location `C:\SAPCustomTool`.

```
C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool\customizationtool.exe
xml=C:\SAPCustomTool\oem.xml packageDir=C:\SAPCustomTool\packages outputDir=C:\SAPCustomTool\output
logDetail=error > C:\oemlog.log
```

3.7.1 Command line parameters

Table 3-7: Required parameters

Parameter	Description	Example (Windows)
`xml`	Full path to the configuration file.	`xml=example_customization_win_boe.xml`
`packageDir`	Full path to the folder that contains the installation program you are modifying. The installation program is downloaded from SAP Service Marketplace in order to start the installation of SAP BusinessObjects Business Intelligence platform. It contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries.	`packageDir=C:\SAPCustomTool\packages`
`outputDir`	Full path to the folder where the customized installation program will be created. Must be empty before running the tool.	`outputDir=C:\SAPCustomTool\output`

Table 3-8: Optional parameters

Parameter	Description	Example (Windows)

`baselinePath`	Full path to a root folder containing the original, non-customized versions of all previous full and update installation programs you have customized.
	Use a semicolon (`;` - Windows) or colon (`:` - Unix) to separate multiple root folders.
		Assume you want to customize SAP BusinessObjects Business Intelligence platform 4.0 Support Package 5 and you customized the previous programs: `4.0 SP2 (Full installation)`, `4.0 SP4`. Customize 4.0 Support Package 5, and provide the root folder path to the non-customized packages for the 4.0 SP2 full installation and SP4 update installation. For example, if the non-customized packages are contained in the following directory structure:
		`C:\productUpdates\4.0\`
		`\SP2 Full\`
		`\SP4\`
		set the value to `baselinePath=C:\productUpdates\4.0\`
		See Customizing update installation programs for more information and examples of the `baselinePath` parameter.

`logDetail`	The level of logging detail. Default value is info. Accepted values: • `error` • `warn` • `info` • `debug` • `trace`	`logDetail=warn`
`action`	The tool mode. Accepted values are: • `generate` (default value) The tool performs the specified customizations. • `validate` The tool validates the configuration file but does not perform any customizations.	`action=validate`

Related Topics

- Quick start for the Business Intelligence platform (Windows)
- Quick start for the Business Intelligence platform (Unix or Linux)

3.8 Customizing update installation programs

Update installation programs are minor releases, Support Packages or Patches that contain updates to your existing BI platform software. Support Packages contain more updates than Patches but are released less frequently. You can use the SAP BusinessObjects customization tool to customize these update installation programs, but some modifications to the command line and configuration file are required.

3.8.1 Frequently asked questions about update installation programs

Where do I find Support Packages and Patches?

1. Go to https://service.sap.com/bosap-support > **Software Downloads**.
2. On the "Find your software" tab, under the "A–Z Index", click **Support Packages and Patches**.
3. Select **B** > **SBOP BI platform (former SBOP Enterprise)** > **SBOP BI PLATFORM (ENTERPRISE)** > **SBOP BI PLATFORM 4.1** > **Comprised Software Component Versions** > **SBOP BI PLATFORM SERVERS 4.1** > **<***platform***>** .
4. Select your Support Package or Patch, then follow the instructions on the website to download and extract the objects.

What parts of update installation programs can I customize?

You can customize the same aspects of update installation programs as you did in the main installation program. Because minor release, Support Package and Patch updates contain fewer installation screens, not all of the customization steps apply. It is recommended to run the minor release, Support Package or Patch before customizing it to determine what customizations you require.

How do I customize update installation programs?

Update installation programs use the same architecture as a main installation program for the BI platform (full installation), so you can use the customization tool as described inCreating the configuration file and Running the tool, with some modifications to the command line and the configuration file. See How to customize update installation programs in this section.

Is it necessary to customize and install all minor release, Support Package and Patch updates?

No. As with non-customized versions of the BI platform, you only need to install the updates that you want. This may be a minor release, Support Package, a Patch, or any valid combination of the three updates.

Can I install a non-customized update on a customized BI platform installation?

Yes. Both customized and non-customized updates may be applied to your customized installation. However, non-customized minor release, Support Package, or Patch installation programs will not display your branding or installation customizations (such as removed features or shortcut changes) you created for the main installation program.

I have delivered a customized version of BI platform to customers but I want to modify the customizations in an update installation program. Is this possible?

This scenario is not supported. The customizations that you make to update installation programs must be consistent with the original customizations.

3.8.2 Quick start for update installation programs

Ensure you have customized and installed the main installation program, such as SAP BusinessObjects Business Intelligence platform Support Package 4 (full installation) using the instructions in Quick start for the Business Intelligence platform (Windows), and that the non-customized installation program is located in `C:\SAPCustomTool\packages`.

This section shows you how to run the SAP BusinessObjects customization tool to customize the installation program for a Support Package (update installation). It uses the sample configuration file provided with the customization tool. Notice that the sample configuration file contains the `<cloneProduct>` element for the main installation program as well as the `<clonePatchProduct>` element for a Support Package update installation program.

Note:
You can run this example only when a Support Package is available on https://service.sap.com/bosap-support.

1. Download the installation program for the BI Platform 4.0 Support Package to the folder `C:\SAPCustomTool\SupportPackage`.
2. Ensure the `product_version` for the `<clonePatchProduct>` element in the configuration file matches the version number of the Support Package that you downloaded. See Customizing the product name and version number.
3. Customize the BI Platform 4.0 Support Package and place the customized installation program in `C:\SAPCustomTool\output\SupportPackage`. Use the following command:

   ```
   customizationtool.exe xml=example_customization_win_boe.xml packageDir=C:\SAPCustomTool\SupportPackage baselinePath=C:\SAPCustomTool\packages outputDir=C:\SAPCustomTool\output\SupportPackage logDetail=error > C:\oemlog_SP04.log
   ```

4. Use `C:\SAPCustomTool\output\SupportPackage\setup.exe` to run the customized installation program for the BI Platform 4.0 Support Package.

3.8.3 How to customize update installation programs

Use the configuration tool as described in Creating the configuration file and Running the tool to customize update installation programs for minor releases, Support Packages, and Patches, with the following differences:

- The configuration file must use the `clonePatchProduct` element (with the correct product ID), instead of the `cloneProduct` element.
- The configuration file must contain the complete, original `<cloneProduct>` element used when customizing the main installation package that you are updating, with no modifications. Do not add or remove features in the `<cloneProduct>` element as it may cause unpredictable results, especially when customizations involve removing features.
- The configuration file cannot contain more than one `clonePatchProduct`. If you are customizing both a Support Package and a Patch for example, you must create two configuration files: one file containing `cloneProduct` and `clonePatchProduct` for the Support Package, and the other file containing `cloneProduct` and `clonePatchProduct` for the Patch.
- Refer to all prerequisite installation programs using the `baselinePackages` command.

All configuration file elements and command-line parameters can be used to customize update installation programs, but not all of them are applicable to every minor release, Support Package, or Patch. Run the installation program for the update first to determine what you need to customize, then use the information in Creating the configuration file and IDs and codes for BI Platform customization to create the customization file.

To specify the product version in the configuration file

The configuration file for Support Packages and Patches must contain the `product version` in the `clonePatchProduct` element as shown below:

```
<oem name="<any name>">
  <clonePatchProduct sourceId="<product version>">
  ...
  </clonePatchProduct>
</oem>
```

The `product version` in the configuration file must match the version number of the installation program that you are customizing. To find the version number, look in the `dunit` folder for a folder with a name in this format:

`product.boe64.patch-4.x.x.x-core-32`

Use the name of this folder as the `product version`.

Example:

This example configuration file customizes the SAP BusinessObjects Business Intelligence platform 4.1 Patch 1, which has the product version `product.boe64.patch-4.1.0.1-core-32`. The configuration file customizes the product long name to Custom Company Server and the product short name to Custom CS.

```
<oem name="Custom Patch Tool">
 <clonePatchProduct sourceId="product.boe64.patch-4.1.0.1-core-32">
  ...
 </clonePatchProduct>
</oem>
```

To use the baselinePath parameter

Use the command line parameter `baselinePath` to refer to a root folder containing the original, non-customized versions of all previous full or update installation programs you have customized. This means you must keep the original installation packages.

Note:
This parameter replaces the `baselinePackages` parameter introduced in 4.0 Feature Pack 3.

To simplify the `baselinePath` parameter value, reference a single root folder - the customization tool will ignore unneeded files and folders. Otherwise, use a semicolon (`;` - Windows) or colon (`;` - Unix) in the `baselinePath` value to specify multiple root folders. Consider the following examples on Windows.

Example: Customizing 4.0 SP5 Patch 2

Assume you are customizing BI platform 4.0 Support Package 5 Patch 2 and you customized the previous programs: `4.0 SP2 (Full install), 4.0 SP4, 4.0 SP5, 4.0 SP5 Patch 1`. Assume the non-customized installation programs are located in the following directory structure:

```
C:\productUpdates\4.0\
  \SP2 Full\
  \SP4\
  \SP5\
  \SP5 Patch 1\
```

Set the `baselinePath` parameter to the root folder:

```
baselinePath=C:\productUpdates\4.0\
```

Example: Customizing 4.1 SP 1

Assume you are customizing BI platform 4.1 Support Package 1 and you customized the previous programs: `4.0 SP2 (Full install), 4.0 SP4, 4.0 SP5, 4.1`. Assume the non-customized installation programs are located in the following directory structure:

```
C:\productUpdates\
  \4.0\
    \SP2 Full\
    \SP4\
    \SP5\
  \4.1\
    \Full\
```

Set the `baselinePath` parameter to the root folder:

```
baselinePath=C:\productUpdates\
```

3.9 IDs and codes for BI Platform customization

The following section contains a list of all the IDs and codes you can use to customize the installation program:

- Feature IDs
- Shortcut deployment unit IDs (Windows only)
- String IDs
- Language codes
- Installation screen and property IDs

3.9.1 Feature IDs

Use these IDs in the `removeFeature` element to remove features and their components from the installation program and the installed product.

For example, this ID will remove all the web tier components, including `JavaWebApps1` and `IntegratedTomcat`:

```
<removeFeature id="WebTier"/>
```

- - `root`: (remove all features)
    - `WebTier`: (remove all web tier components listed below)
      - `JavaWebApps1` Java Web Applications
      - `IntegratedTomcat` (install bundled Tomcat web application server)

        Note:
        If you remove the web tier feature, the web tier components will be removed from the installation program. However, the WebTier radio button will still be visible from the "Choose Install Type" screen. That is, the user will still see the three radio buttons: Full, Custom/Expand, and WebTier. This is a known issue and will be fixed.
    - `Servers`: (remove all server components listed below)
      - `PlatformServers`: (remove all platform servers listed below)
        - `CMS` (Central Management Server)
        - `FRS` (File Repository Servers)
        - `PlatformServers.IntegratedDB.SQLAnywhere` (removes bundled Sybase SQL Anywhere database server)

        - `PlatformServers.EventServer`
        - `PlatformServers.SystemLandscapeSupplier` (SLD)
        - `PlatformServers.WebAppContainerService` (WACS)
        - `AdaptiveProcessingServer` (platform processing)
        - `AdaptiveJobServer` (scheduling)
        - `Platform.RestWebService`
        - `Platform.Action.Framework.backend` (Insight to Action framework)
        - `Subversion` (Subversion version control system)
    - `ConnectionServices`: (removes all connectivity components listed below)
        - `ConnectionProcService`
    - `DataFederatorServices`: (remove all data federation components listed below)
        - `DataFederatorQueryService`
    - `AdvancedAnalysisServices`: *removes all Analysis components listed below)
        - `MultidimensionalAnalysisServices` (MDAS)
        - `BExWebApplicationsService`
    - `CrystalReportsServers`: (removes all SAP Crystal Reports components listed below)
        - `CrystalReportsProcServices` (SAP Crystal Reports Processing)
        - `CrystalReportSchedulingServices`
        - `CrystalReport2011ProcServices` (SAP Crystal Reports 2011 Processing)
        - `CrystalReport2011SchedulingServices` (SAP Crystal Reports 2011 Scheduling)
    - `WebIServers`: (removes all Web Intelligence components listed below)
        - `WebIProcServer` (Web Intelligence Processing)
        - `WebISchedulingServices` (Web Intelligence Scheduling)
    - `XcelsiusServers` (Dashboards)
    - `MobileServices`
        - `MobileServers`
        - `MobileAddon` (CMS plugin for Mobile)
    - `IntegrationServers`: (removes all integration components listed below)
        - `BWPublisherServer` (SAP BW authentication and SAP BW Publisher support)
- `MultitenancyManager`
- `AdministratorTools`: (removes all administrator tools listed below)
    - `UpgradeManager` (Upgrade management tool)

- `DeveloperTools`: (removes all developer tool components listed below)
    - `BOE64bitNETSDK` (64-bit SAP BusinessObjects Business Intelligence platform .NET SDK)

- `DataAccess` (removes all database access components listed below)
    - `DataAccess.DataFederator`
    - `DataAccess.HPNeoView`

- `DataAccess.MySQL`
- `DataAccess.GenericJDBC`
- `DataAccess.GenericODBC`
- `DataAccess.GenericOLEDB`
- `DataAccess.OptionalDataDirectODBC`
- `DataAccess.MaxDB`
- `DataAccess.SAPHANA`
- `DataAccess.Salesforce` (Salesforce.com)
- `DataAccess.Netezza`
- `DataAccess.Microsoft_AnalyticalServices`
- `DataAccess.MicrosoftExchange`
- `DataAccess.MicrosoftOutlook`
- `DataAccess.Microsoft_SQLServer`
- `DataAccess.Microsoft_Access`
- `DataAccess.Ingres`
- `DataAccess.Greenplum`
- `DataAccess.IBMDB2`
- `DataAccess.Informix`
- `DataAccess.ProgressOpenEdge`
- `DataAccess.Oracle`
- `DataAccess.Sybase`
- `DataAccess.Teradata`
- `DataAccess.SAPBW`
- `DataAccess.SAPERP`
- `DataAccess.XMLWebServices`
- `DataAccess.OData`
- `DataAccess.Excel`
- `DataAccess.SAP` (security and data access for SAP BW and R/3 systems)
- `DataAccess.PersonalFiles`
- `DataAccess.JavaBean`
- `DataAccess.OpenConnectivity`
- `DataAccess.HSQLDB`
- `DataAccess.Derby`
- `DataAccess.HadoopHive`
- `DataAccess.Essbase`
- `DataAccess.Peoplesoft` (PeopleSoft Enterprise)
- `DataAccess.JDEdwards` (JD Edwards EnterpriseOne)
- `DataAccess.Siebel` (Siebel Enterprise Server)
- `DataAccess.OracleEBS` (Oracle E-Business Suite)
- `DataAccess.Universe` (SAP BusinessObjects Universe)
- `DataAccess.MyCube` (OLAP Cube)
- `DataAccess.XML`

    - `DataAccess.ADO.NET`
    - `DataAccess.COMData`
    - `DataAccess.DataSet` (Dataset Consumer)
    - `DataAccess.SymantecACT`
    - `DataAccess.BDE` (IDAPI Database DLL)
    - `DataAccess.CDO` (Crystal Data Objects)
    - `DataAccess.FieldDefinitions`
    - `DataAccess.FileSystem`
    - `DataAccess.NTEventLog`
    - `DataAccess.WebActivityLog`
    - `DataAccess.Btrieve` (Pervasive Database Driver)
    - `DataAccess.dBase`
    - `DataAccess.UWSC` (Universal Web Services Connector (UWSC))
  - `Samples`: (remove sample reports and data sources)

Related Topics

- Removing features

3.9.2 Shortcut deployment unit IDs (Windows only)

Use the deployment unit IDs in the `shortcut` element to change the location and name of the program shortcuts in the Windows **Start** menu.

Table 3-9: Shortcut deployment unit IDs

Shortcut deployment unit ID	Shortcut target
`product.businessobjects64.shortcut.wdeploy-4.0-core`	WDeploy
`product.businessobjects64.shortcut.ccm-4.0-core`	Central Configuration Manager
`product.businessobjects64.shortcut.cmc-4.0-core`	Central Management Console
`product.businessobjects64.shortcut.infoview-4.0-core`	BI launch pad (InfoView)
`product.businessobjects64.shortcut.odbc-4.0-core`	32-bit Data Source Administrator

Shortcut deployment unit ID	Shortcut target
`product.businessobjects64.shortcut.onlinedoc-4.0-core`	Online documentation
`product.businessobjects64.shortcut.tomcat-4.0-core`	Apache Tomcat. See Modifying the tomcat shortcut for additional instructions.
`product.businessobjects64.shortcut.upgrade-4.0-core`	Upgrade management tool
`product.businessobjects64.shortcut.wacs.infoview-4.0-core`	WACs stored in InfoView
`product.businessobjects64.shortcut.wacs-4.0-core`	Web Application Container Server

Related Topics

- Customizing the Windows Start menu shortcuts (Windows only)

3.9.3 String IDs

You can change the value of all strings in the installation program. You can replace a string for all languages or for a specific language. Use the `replaceString` element, for example:

```
<replaceString id="productname" value="Sales Data Insight lang="all"/>
```

Table 3-10: Commonly changed strings

String ID	Description
`product.boe64_name`	Product long name
`product.boe64_shortname`	Product short name
`product_version`	Product version
`product_majorversion`	Product major version

Related Topics

- Customizing the product name and version number

3.9.4 Language codes

The SAP BusinessObjects customization tool uses these language codes to represent supported languages:

language	Code
English	EN
Czech	CS
Danish	DA
Dutch	NL
Finnish	FI
French	FR
German	DE
Hungarian	HU
Italian	IT
Japanese	JA
Korean	KO
Norwegian Bokmal	NB
Polish	PL
Portuguese	PT
Romanian	RO
Russian	RU
Simplified Chinese	zh_CN
Slovak	SK
Spanish	ES
Swedish	SV
Thai	TH

language	Code
Traditional Chinese	`zh_TW`
Turkish	`TR`

Related Topics

- Customizing the product name and version number
- Customizing the Windows Start menu shortcuts (Windows only)
- Customizing the Windows Add Remove Program utility (Windows only)
- Removing language packs
- Customizing the license agreement

3.9.5 Installation screen and property IDs

Use the installation screen IDs in the `removeDialog` element to remove screens from the installation program. For example, use this element to remove the "User Information" screen:

```
<removeDialog id="EnterProductKey.dialog"/>
```

Use the properties and the property values to prepopulate user input. For example, use this element to set the default installation type to "custom":

```
<replaceProperty id="InstallType" defaultValue="custom"/>
```

Note:
Property values are case-sensitive.

Table 3-12: Installation screen IDs and associated properties

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Check Prerequisites"	`CheckPreRequisites.dialog`	Not applicable	Not applicable

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Select Installer Language"	`SelectUILanguage.dialog`	`SortedAvailableSetupLanguages`	Set of language codes that the installation program can be run in, for example `"en;ja"`
		`SetupUILanguage`	Single language code describing the language that the installation program will be run in, for example `"en"`
"Welcome to the installation wizard ...."	`ShowWelcomeScreen.dialog`	Not applicable	Not applicable
"License Agreement"	`ShowLicenseAgreement.dialog`	Not applicable	Not applicable
"Configure Product Registration"	`EnterProductKey.dialog`	`RegisteredUser`	"Username"
		`RegisteredCompany`	"Company name"
		`ProductKey`	"Product keycode"
"Select Language Packages"	`SelectLanguagePack.dialog`	`SelectedLanguagePacks`	The set of language packs to be installed, for example `"en;ja"` For a list of language codes, see Language codes.
"Select Install Type"	`ChooseInstallType.dialog`	`InstallType`	• `default` (Full) • `custom` • `webtier`
"Configure Destination Folder"	`ChooseInstallDir.dialog`	`InstallDir`	Installation folder
"Select Default or Existing Database"	`SelectDataSource.dialog`	`SelectIntegratedDatabase`	• `0` (Use an existing database) • `1` (Install and use the default database)
"Expand Installation"	`ExpandInstallMessage.dialog`	Not applicable	Not applicable
"Select Java Web Application Server"	`ChooseWebAppServer.dialog`	`WebAppServerType`	• `tomcat` • `manual` • `wacs`

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Select Features"	`SelectFeatures.dialog`	Not applicable	Not applicable
"Select Version Management"	`SelectLCM.dialog`	`NewOrExistingLCM`	• `existing` • `new`
"Select New or Expand Installation"	`ChooseExpandInstall.dialog`	`NewOrExpandInstall`	• `new` • `expand`
"Configure Subversion"	`SetLCMConfig.dialog`	`LCMName`	Repository name
		`LCMPort`	Repository port
		`LCMUserName`	Repository user
		`LCMPassword`	Repository password
		`LCMPasswordConfirm`	Confirm password
"Configure Server Intelligence Agent (SIA)"	`GetSIAInfo.dialog`	`SIAPort`	SIA port
		`SIAName`	Node name
"Configure Central Management Server (CMS)"	`GetCMSInfo.dialog`	`CMSPort`	Any valid port number
"Configure CMS Account"	`GetCMSPassword.dialog`	`CMSPassword`	The CMS password
		`CMSPasswordConfirm`	The CMS password
		`ClusterKey`	The CMS cluster key
		`ClusterKeyConfirm`	The CMS cluster key
"Configure Sybase SQL Anywhere"	`GetSQLAnywhereInfo.dialog`	`SQLAnywhereServerName`	The SQL Anywhere server name (Unix and Linux only)
		`SQLAnywherePort`	The SQL Anywhere port
		`SQLAnywhereAdminPassword`	The SQL Anywhere administrator password (username is `dba`)

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Select Automatic Server Start"	`ChooseToEnableServers.dialog`	`EnableServers`	• `0` (Stop servers upon installation) • `1` (Start servers upon installation)
"Configure Tomcat"	• `ShowTomcatInfo.dialog` • `GetTomcatInfo.dialog` Both dialog IDs must be included in the configuration file in order to remove the "Configure Tomcat" screen. That is, you must include two `removeDialog` elements in your configuration file.	`TomcatConnectionPort`	Connection port
		`TomcatShutdownPort`	Shutdown port
		`TomcatRedirectPort`	Redirect port
"Select Connectivity for Solution Manager Diagnostics (SMD) Agent"	`SelectSMDIntegrate.dialog`	`ChooseSMDIntegration`	• `nointegrate` (Do not integrate) • `integrate` (Integrate)
"Configure Connectivity to SMD Agent"	`ConfigureSMDAgent.dialog`	`SMDAgent_HOST`	SMD agent host
		`SMDAgent_PORT`	SMD agent port
"Select Connectivity to Introscope Enterprise Manager"	`SelectIntroscopeIntegrate.dialog`	`ChooseIntroscopeIntegration`	• `nointegrate` (do not integrate) • `integrate` (integrate)
		`Introscope_ENT_HOST`	Introscope host name
		`Introscope_ENT_PORT`	Introscope port number

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure Connectivity to Introscope Enterprise Manager"	`ConfigureIntroscope.dialog`	`IntroscopeENT_HOST`	Enterprise manager host
		`IntroscopeENT_PORT`	Enterprise manager port
		`IntroscopeENT_INSTRUMENTATION`	Set to `true` to indicate that you configured this installation screen
"Configure HTTP Listening port"	`GetWACSPort.dialog`	`WACSPort`	Port number for the web application container service
"Select Existing Auditing Database Type"	`SelectAuditDatabase.dialog`	`UsingAuditDBType`	• `sybase` • `db2` • `oracle` • `mysql` • `mssql` • `maxdb` • `none`
"Select Existing CMS Database Type"	`SelectCMSDatabase.dialog`	`UsingCMSDBType`	• `sybase` • `db2` • `oracle` • `mysql` • `mssql` • `maxdb`
"Existing CMS Deployment Information"	`SetRemoteCMSInfo.dialog`	`RemoteCMSName`	Name of the existing CMS
		`RemoteCMSPort`	Port number for the existing CMS
		`RemoteCMSAdminName`	Administrator's username
		`RemoteCMSAdminPassword`	Administrator's password
"SAP BusinessObjects BI platform has been successfully installed"	`ShowInstallCompleteLaunchWDeploy.dialog`	`LaunchWDeploy`	• `0` (Do not launch WDeploy tool after install) • `1` (Automatically launch WDeploy tool after install)

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure Auditing Database - DB2"	`ExistingAuditDB2.dialog`	`ExistingAuditingDBServer`	DB2 alias name
		`ExistingAuditingDBUser`	Username
		`ExistingAuditingDBPassword`	Password
"Configure CMS Repository Database - SQL Anywhere (ODBC)"	`ExistingCMSSQLAnywhere.dialog`	`ExistingCMSDBDSN`	Data source name
		`ExistingCMSDBUser`	Username for existing database
		`ExistingCMSDBPassword`	Users's password
"Configure Auditing Database - SQL Anywhere (ODBC)"	`ExistingAuditSQLAnywhere.dialog`	`ExistingAuditingDBDatabase`	Name of existing auditing database
		`ExistingAuditingDBUser`	Username for existing database
		`ExistingAuditingDBPassword`	User's password
"Configure Auditing Database - MaxDB"	`ExistingAuditMaxDB.dialog`	`ExistingAuditingDBDatabase`	Name of existing auditing database
		`ExistingAuditingDBUser`	Username for existing database
		`ExistingAuditingDBPassword`	User's password
		`ExistingAuditingDBPort`	Port number for existing database
		`ExistingAuditingDBServer`	MaxDB server name

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure Auditing Database - SQL Server (ODBC)"	`ExistingAuditMSSQL.dialog`	`ExistingAuditingDBDatabase`	SQL database name
		`ExistingAuditingDBServer`	SQL server name
		`ExistingAuditingDBUser`	Username
		`ExistingAuditingDBPassword`	Password
		`ExistingAuditingDBUseTrustedConnection`	Use trusted connection
		`ExistingAuditingDBDSN`	Data source name
		`ExistingAuditingDBShowSysDB`	Show system database
"Configure Auditing Database - MySQL"	`ExistingAuditMySQL.dialog`	`ExistingAuditingDBDatabase`	Auditing database name
		`ExistingAuditingDBUser`	Username
		`ExistingAuditingDBPassword`	Password
		`ExistingAuditingDBPort`	MySQL Port
		`ExistingAuditingDBServer`	MySQL Server
"Configure Auditing Database - Oracle"	`ExistingAuditOracle.dialog`	`ExistingAuditingDBUser`	Username
		`ExistingAuditingDBPassword`	Password
		`ExistingAuditingDBServer`	Oracle TNSNAME

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure Auditing Database - Sybase"	`ExistingAuditSybase.dialog`	`ExistingAuditingDBUser`	Username
		`ExistingAuditingDBPassword`	Password
		`ExistingAuditingDBServer`	Sybase service name
"Configure CMS Repository Database - DB2"	`ExistingCMSDB2.dialog`	`ExistingCMSDBServer`	DB2 Alias Name
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	• `0` (Do not reset existing database) • `1` (Reset existing database)
"Configure CMS Repository Database - MaxDB"	`ExistingCMSMaxDB.dialog`	`ExistingCMSDBServer`	CMS Database Name
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	• `0` (Do not reset existing database) • `1` (Reset existing database)
		`ExistingCMSDBDatabase`	MaxDB Server
		`ExistingCMSDBPort`	MaxDB Port

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure CMS Repository Database - SQL Server"	`ExistingCMSMSSQL.dialog`	`ExistingCMSDBServer`	Existing server name
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	• `0` (Do not reset existing database) • `1` (Reset existing database)
		`ExistingCMSDBDatabase`	CMS Database Name
		`ExistingCMSDBUseTrustedConnection`	Use trusted connection
		`ExistingCMSDBDSN`	Data source name
		`ExistingCMSDBShowSysDB`	Show system database
"Configure CMS Repository Database - MySQL"	`ExistingCMSMySQL.dialog`	`ExistingCMSDBServer`	MySQL Server
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	• `0` (Do not reset existing database) • `1` (Reset existing database)
		`ExistingCMSDBDatabase`	CMS Database Name
		`ExistingCMSDBPort`	MySQL Port

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"Configure CMS Repository Database - Oracle"	`ExistingCMSOracle.dialog`	`ExistingCMSDBServer`	Oracle TNSNAME
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	• `0` (Do not reset existing database) • `1` (Reset existing database)
"Configure CMS Repository Database - Sybase"	`ExistingCMSSybase.dialog`	`ExistingCMSDBServer`	Sybase service name
		`ExistingCMSDBUser`	Username
		`ExistingCMSDBPassword`	Password
		`ExistingCMSDBReset`	Reset existing database
"Configure Subversion"	`SetLCMConfig.dialog`	`LCMName`	Repository Name
		`LCMPort`	Repository Port
		`LCMUserName`	Repository User
		`LCMPassword`	Password
		`LCMPasswordConfirm`	Confirm password
"SAP BusinessObjects BI platform has been successfully installed"	`ShowInstallComplete.dialog`	Not applicable	Not applicable

Title of installation screen	Installation screen ID	Property ID(s)	Allowed property value(s)
"SAP BusinessObjects BI platform has been successfully installed "	`ShowInstallCompleteMultiCheckbox.dialog`	`LaunchWDeploy`	• `0` (Do not launch WDeploy tool after install) • `1` (Automatically launch WDeploy tool after install)
		`LaunchSSW`	• `0` (Do not launch System Setup Wizard after install) • `1` (Automatically launch System Setup Wizard after install)
		`ViewLogButton`	• `0` (Do not view log file after install) • `1` (Automatically view log file after install)
"Start Installation"	`ShowInstallSummary.dialog`	Not applicable	Not applicable
"Post Installation Steps"	`ShowPostInstall.dialog`	Not applicable	Not applicable
"Uninstall Confirmation"	`VerifyToRemove.dialog`	Not applicable	Not applicable
"SAP BusinessObjects BI platform has been successfully uninstalled"	`ShowUninstallComplete.dialog`	Not applicable	Not applicable

Related Topics

- Customizing user input
- Removing installation screens

Web Application Customization

4.1 Introduction

You can apply your own branding (or “skinning”) to the BI launch pad, OpenDocument, and Crystal Reports JavaScript viewer web applications. For example, you can customize your BI platform systems by applying your own corporate identity elements.

You can customize the following web and graphic elements:

- Favicon (the icon displayed in the browser URL bar)
- Logos
- Certain background patterns and colors
- Certain animated gifs (for example, the progress indicator)
- Certain CSS styles (borders, padding, margins, and so on)
- JavaScript files for the Crystal Reports JavaScript viewer

You can customize many aspects of the BI platform web applications, and you may choose to customize only a subset of these options.

Who should use this information?

This section is intended for web application designers, developers, and system administrators who are customizing BI platform web applications. Familiarity with the basics of CSS design and Java web application archives is required. If you are deploying customizations, then you should also be familiar with the methods to install and deploy BI platform web applications to an application server.

For information on installing the SAP BusinessObjects Business Intelligence platform, see the *Business Intelligence Platform Installation Guide*.

For information on deploying BI platform web applications using the WDeploy tool, see the *Business Intelligence Platform Web Application Deployment Guide*.

4.1.1 Key concepts

To make and deploy your customizations, you should understand the following concepts:

Installation package

The installation package is the set of binaries that are downloaded from SAP Service Marketplace in order to start the installation of SAP BusinessObjects Business Intelligence platform. It contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries.

Customization template

The `template.zip` file is located in the `Collaterals\CustomizationTemplate` folder of your installation package, and contains the branding bundles (JAR files) to customize before installing SAP BusinessObjects Business Intelligence platform. This file is the starting point for customizing your web applications.

BOE WAR file

`BOE.war` is the primary web application archive for the BI platform. BI launch pad, OpenDocument, the Crystal Reports JavaScript viewer, and the changes you make in `template.zip` to their respective branding bundles are included by the installation program in `BOE.war`. To deploy your customizations and make these applications available to your customers, you must deploy `BOE.war` to your Java application server either during or after the installation process.

Branding bundles

A branding bundle is a JAR file that contains the custom resources (CSS, icons, images, JavaScripts) that you include in the installation program within `template.zip`. The following branding bundles are included:

- `com.businessobjects.webpath.InfoViewBranding.jar` (BI launch pad)

  This branding bundle consists of two main folders: a `css` folder containing a custom CSS file, and an `images` folder containing a favicon and theme subfolder with custom logos, images, and animated GIFs.

  ```
  \com.businessobjects.webpath.InfoViewBranding\web
    \css
      customize.css
    \images
      favicon.ico
      \theme
        *.png, *.gif
  ```

- `com.businessobjects.webpath.OpenDocumentBranding.jar` (OpenDocument)

  This branding bundle consists of two main folders: a `css` folder containing a custom CSS file, and an `images` folder containing a `theme` subfolder with custom logos and images.

  ```
  \com.businessobjects.webpath.InfoViewBranding\web
    \service
      \css
        customize.css
      \images
        \theme
          *.png
  ```

- `com.businessobjects.webpath.CrystalReports_oem.jar` (Crystal Reports JavaScript viewer)

This branding bundle consists of two main resources: a JavaScript file with custom listeners where you can define new behavior for certain viewer events, and a JSON properties file to reference any custom JavaScript files or images you choose to use for the viewer.

```
\com.businessobjects.webpath.CrystalReports_oem\web
  CustomListener.js
  \WEB-INF\classes
    JSAPI-properties.json
  \images
    *.png
```

Web application deployment

The SAP BusinessObjects Business Intelligence platform installation program can deploy `BOE.war` only to the bundled Tomcat web application server. Other supported web application servers require that you deploy the web applications after the installation is complete. It is recommended that you use the WDeploy tool.

4.1.2 Testing your customizations

Before performing customizations on your production systems, it is good practice to test your customizations first on a test installation. In a default installation that uses the bundled Tomcat server, you can instantly see the effects of your changes by making temporary modifications to the `webpath.InfoViewBranding`, `webpath.OpenDocumentBranding`, and `webpath.CrystalReports_oem` folders in the Tomcat `work` directory: `\SAP BusinessObjects\tomcat\work\Catalina\localhost\BOE\eclipse\plugins\webpath.OpenDocumentBranding\web\service`. These folders have the same structure as the branding resources contained in `template.zip`.

Note:
The Tomcat `work` directory is not permanent and your temporary changes are deleted after a Tomcat restart.

4.2 Quick start

Before you start, back up `\Collaterals\Tools\CustomizationTemplate\template.zip` from your installation package.

This section shows you the basic steps required to customize and deploy one of the BI platform web applications: BI launch pad. The steps shown are also applicable to OpenDocument and the Crystal Reports JavaScript viewer.

Note:
This quick start describes the end-to-end customization, including performing a full installation of SAP BusinessObjects Business Intelligence platform and the deployment of `BOE.war` to an application server. These steps may take considerable time.

1. Locate `template.zip` in your SAP BusinessObjects Business Intelligence platform installation package at: `\Collaterals\Tools\CustomizationTemplate`.
2. Extract the contents of `template.zip` to a working folder.

   `template.zip` contains the branding bundles, for example, `\SAP BusinessObjects Enterprise XI 4.0\warfiles\webapps\BOE\WEB-INF\eclipse\plugins\com.businessobjects.webpath.InfoViewBranding.jar`
3. Unpackage the BI launch pad branding bundle `com.businessobjects.webpath.InfoViewBranding.jar`.

   ```
   jar xf com.businessobjects.webpath.InfoViewBranding.jar
   ```
4. Customize the default favicon that is displayed in the browser URL bar for BI launch pad.

   The BI launch pad branding bundle contains a sample favicon. Copy `\web\sample\images\favicon.ico` up one level to `\web\images\favicon.ico`.
5. Repackage `com.businessobjects.webpath.InfoViewBranding.jar` containing your new favicon, and include it in `template.zip`.

   To repackage the `web` and `META-INF` folder contents back into `com.businessobjects.webpath.InfoViewBranding.jar`:

   ```
   jar cf com.businessobjects.webpath.InfoViewBranding.jar web META-INF
   ```
6. Create a subfolder called `\OEMZips` at: `\dunit\product.businessobjects64.oemzips-4.0-core-nu`
7. Add `template.zip` to the `\OEMZips` folder.

   Your customized zip file is now at: `\dunit\product.businessobjects64.oemzips-4.0-core-nu\OEMZips\template.zip`
8. Install and deploy `BOE.war` to your Java application server using one of the following options:

Option	Description
Use the bundled Tomcat server	Selected during the installation process.
Use your own supported Java application server	Performed after the installation program completes. Use the WDeploy tool.

   Use `setup.exe` (Windows) or `setup.sh` (Unix) to start the installation process.
9. After successful installation and deployment, test your changes by accessing BI launch pad: `http://<web server name>:<port>/BOE/BI`

   You can see the new favicon in your browser URL bar: 

4.3 Customizing BI launch pad

For BI launch pad, you can customize the favicon, logo, backgrounds, styles, and more. Most of these customizations involve changing the CSS rules in the `customize.css` file. All customizations must be made available in the `web` folder of `com.businessobjects.webpath.InfoViewBranding.jar` to take effect, as follows:

```
\web
  \css
    customize.css
  \images
    favicon.ico
    \theme
      *.png, *.gif
```

Note:
Sample customizations are provided in the JAR file. For example, when you open `com.businessobjects.webpath.InfoViewBranding.jar`, there is a `web\sample` folder which includes a sample CSS file, sample images, and a README file.

4.3.1 To customize the favicon image

The favicon is the small icon displayed in a browser's address bar when viewing the BI launch pad.

- Replace the `favicon.ico` file stored in the `web\images` folder with your own `favicon.ico` image.

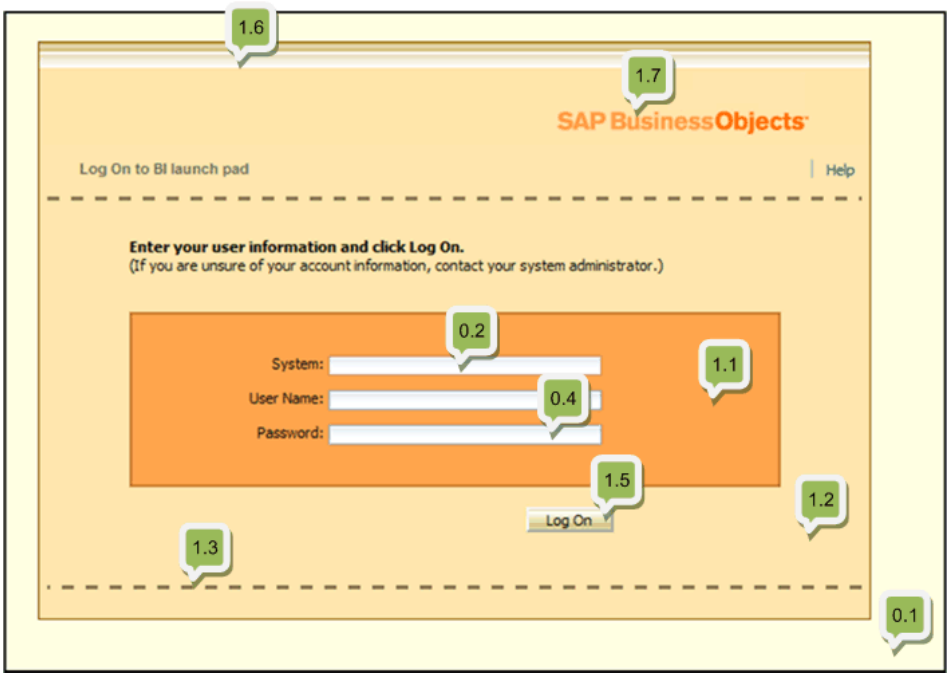
4.3.2 To customize logos

Logos used in the BI launch pad can be customized by editing the CSS rules in the `web\css\customize.css` file. If you are using custom images and referring to them in the `customize.css` file, make sure you place them in the `web\images\theme` folder.

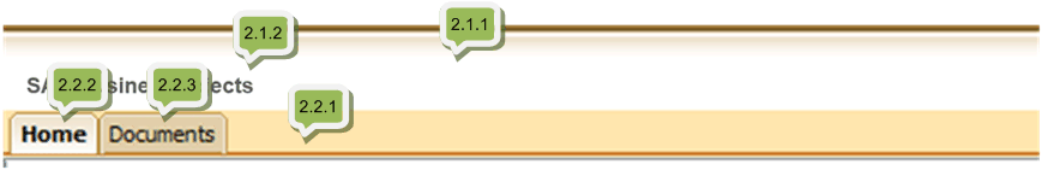
4.3.3 Customizing other user interface elements

Logos, background, styles, and other user interface elements of the BI launch pad can be customized by editing the CSS rules in the `web\css\customize.css` file. If you are using custom images and referring to them in the `customize.css` file, make sure you place them in the `web\images\theme` folder.

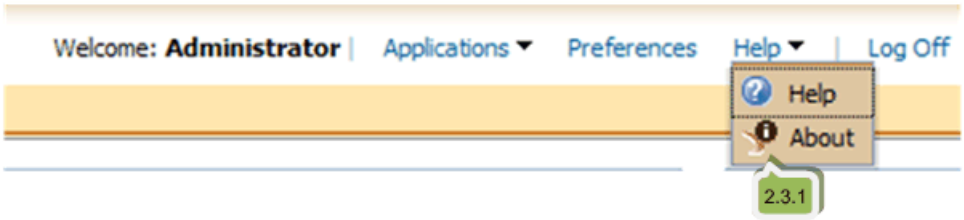
The following diagrams show the elements customized in the sample branding bundle, for reference purposes. The numbers in the balloons refer to sections in the bundle's `customize.css` file.



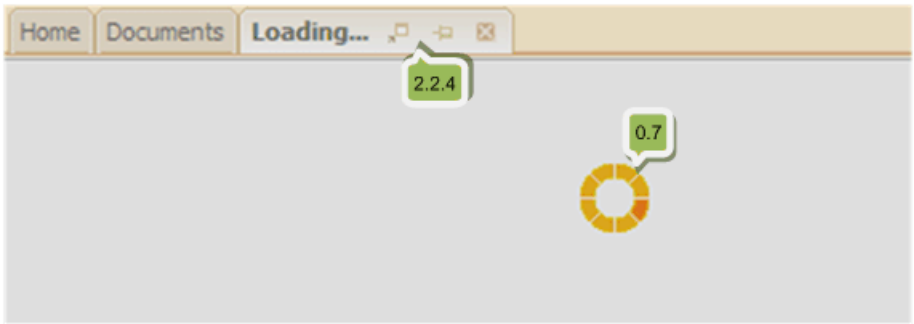
1. (0.1) customize the background of pages and sub-pages (inside frames)
2. (0.2) input text field
3. (0.4) password field
4. (1.1) authentication fields container
5. (1.2) container of everything
6. (1.3) horizontal rules
7. (1.5) "Log On" button
8. (1.6) banner background pattern
9. (1.7) logo



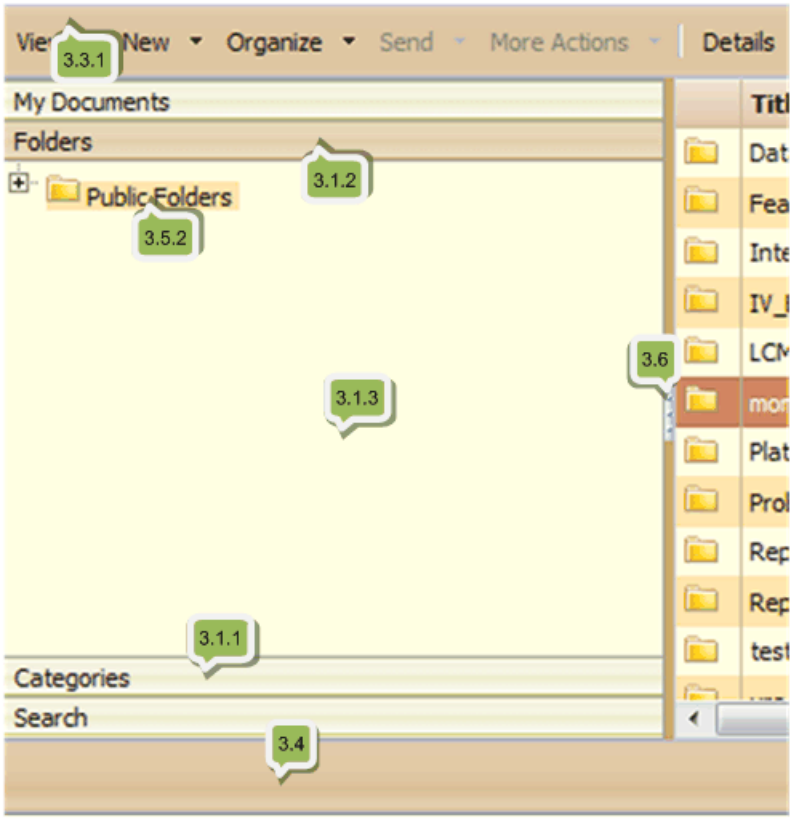
10. (2.1.1) top banner area (background pattern)
11. (2.1.2) banner logo
12. (2.2.1) tab container
13. (2.2.2) active tab
14. (2.2.3) inactive tab
15. (2.3.1) BI launch pad icon



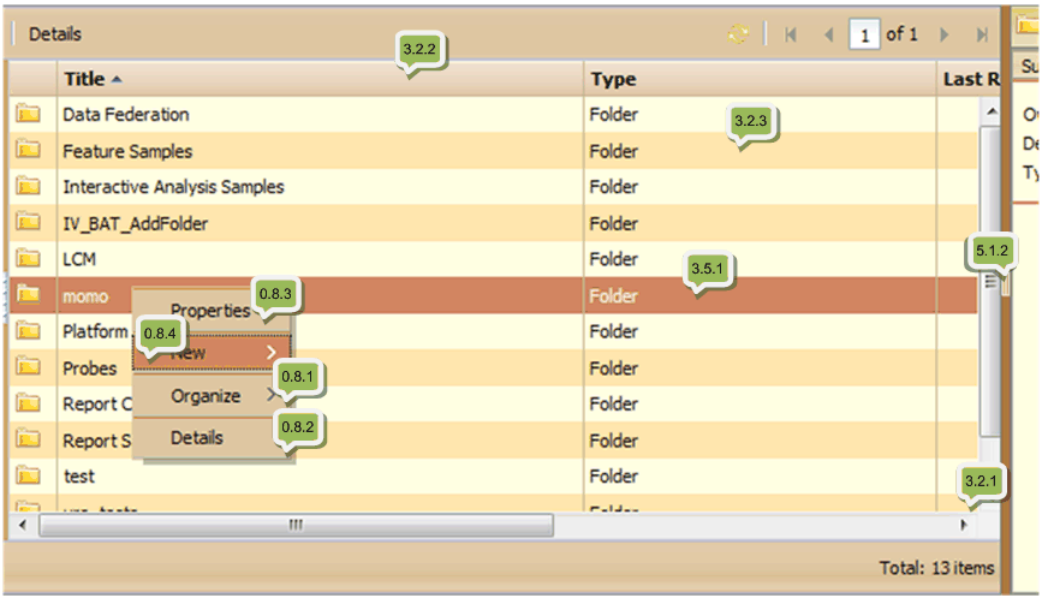
16. (2.2.4) tab buttons
17. (0.7) spinner



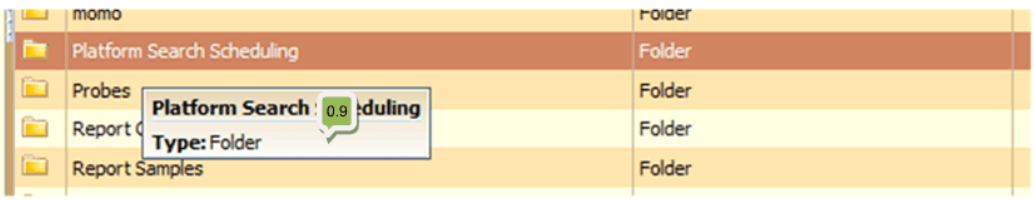
18. (3.1.1) accordion inactive header
19. (3.1.2) accordion active header



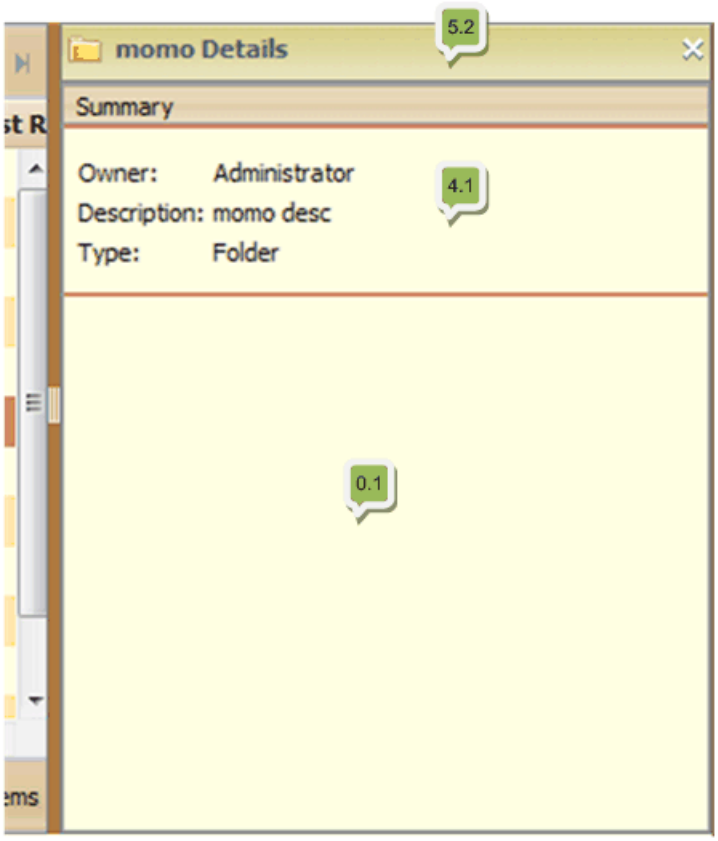
20. (3.1.3) accordion drawer/tree background
21. (3.3.1) toolbar background
22. (3.4) footer background
23. (3.5.2) selected unfocused row in both the list pane and the tree view (on the left)
24. (3.6) accordion-list pane resize bar



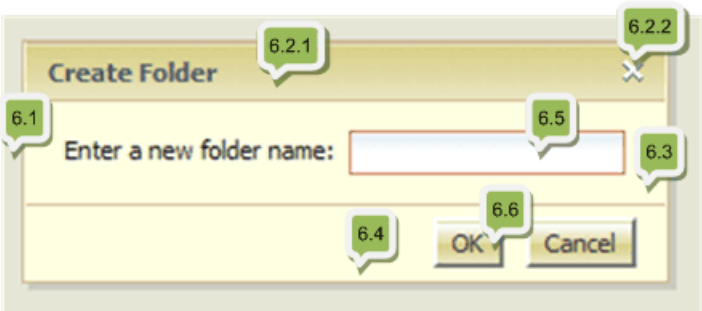
25. (3.2.1) list pane container
26. (3.2.2) list pane heading
27. (3.2.3) list pane rows
28. (3.5.1) selected focused row in both the list pane and tree view (on the left)
29. (5.1.2, 5.1.3, 5.1.4) resize handle and knob
30. (0.8.1) context menu container
31. (0.8.2) context menu body
32. (0.8.3) context menu item
33. (0.8.4) context menu selected item



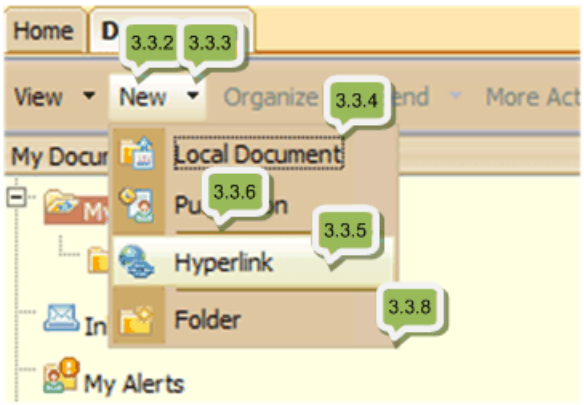
34. (0.9) tooltip



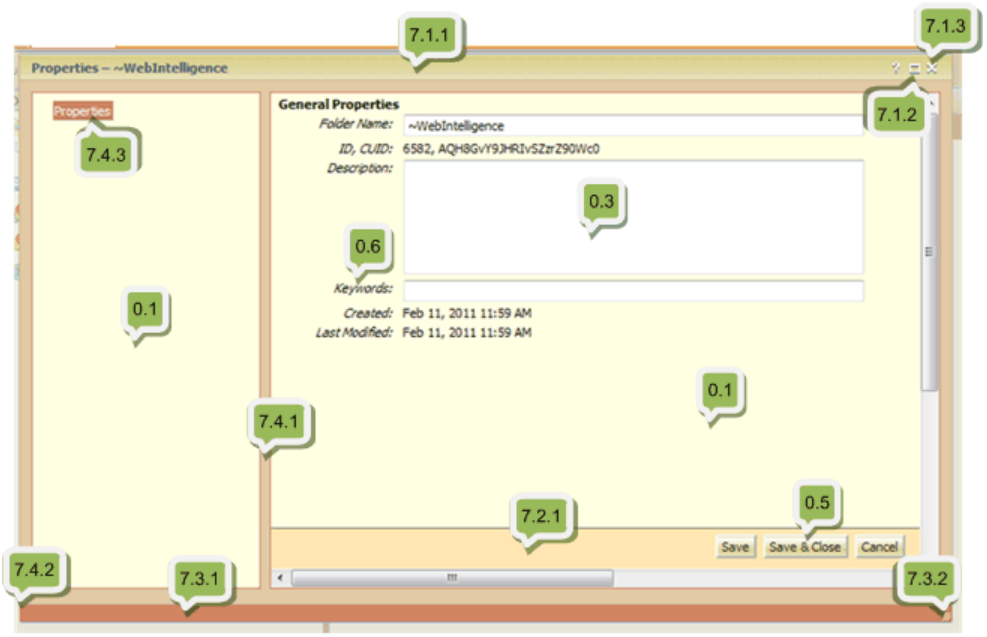
35. (4.1) details container
36. (5.2) details header



37. (6.1) simple dialog container
38. (6.2.1) simple dialog header
39. (6.2.2) close button dialog header
40. (6.3) simple dialog body
41. (6.4) simple dialog footer
42. (6.5) simple dialog text field (overrides general text field)
43. (6.6) simple dialog buttons



44. (3.3.2, 3.3.3) toolbar button hover/press
45. (3.3.4) toolbar menu item
46. (3.3.5) toolbar menu item hovered
47. (3.3.6) toolbar menu separator
48. (3.3.7) toolbar menu frame
49. (3.3.8) toolbar menu refresh icon



50. (7.1.1) large dialog header
51. (7.1.2) large dialog header – maximize button (hover)
52. (7.1.3) large dialog header – close button (hover)
53. (7.2.1) large dialog button panel
54. (7.3.1) large dialog footer
55. (7.3.2) large dialog footer resize handle
56. (7.4.1) large dialog body container

57. (7.4.2) large dialog frame
58. (7.4.3) large dialog left navigation pane selected item
59. (0.3) text box
60. (0.5) button
61. (0.6) form labels

4.3.4 Working with BI workspaces and compound modules

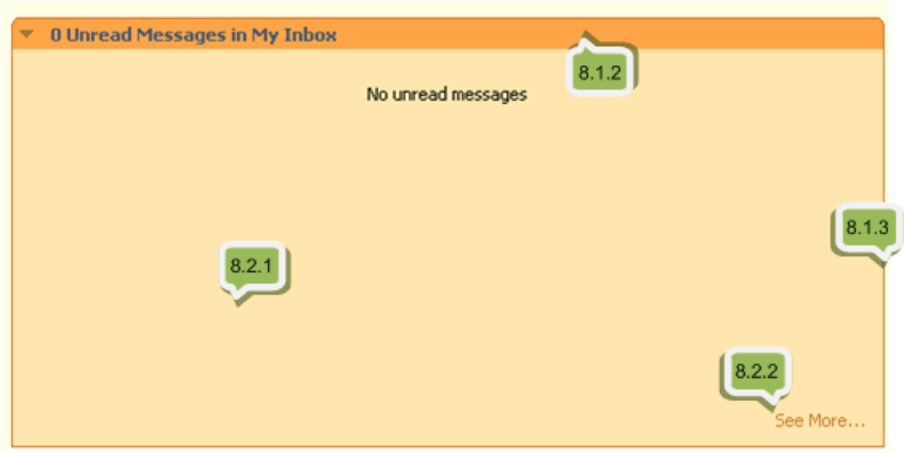
You can also use a BI workspace or a compound module as your BI launch pad home page. You can customize the workspace or compound module to match the style of your BI launch pad.

Note:
The customized style is reflected only on the home page. If the same workspace or compound module is opened outside of the home page (in regular view), the standard style will be used.

The following diagrams show the elements customized in the sample branding bundle, for reference purposes. The numbers in the balloons refer to sections in the bundle's `customize.css` file.

For the default home page or module

The following settings can be used to customize the default home page, or any BI workspace or compound module that has been set as the home page.



1. (8.1.2) module title background
2. (8.1.3) module border
3. (8.2.1) background of BI launch pad module
4. (8.2.2) color of **See More** text

For the regular view of a BI workspace

The following settings can be used to customize the appearance of a BI workspace in regular view.

1. (8.3.1) customized top tab container

2. (8.3.2) customized subtab container
3. (8.3.4) active top tab
4. (8.3.5) inactive top tab
5. (8.3.6) subtab

4.3.4.1 To match the style of a BI workspace to the style of BI launch pad

1. Open the BI workspace for editing.
2. From the first tab of the workspace, click **Properties**.
   The "Properties" dialog box appears.
3. Select the icon of the option immediately before the (last) **Default style** option.
4. Click **OK**.

4.3.4.2 To match the style of a compound module to the style of BI launch pad

1. On the BI launch pad home page, click **Preferences**.
   The "Preferences – Administrator" dialog box appears.
2. Select **BI workspaces** from the **Preferences** list.
3. From the list in the main pane, select **BI Launch Pad**.

4.3.5 To change the name of BI launch pad

You may want to change the name of BI launch pad so that the application blends in with an existing set of applications that your company uses.

Note:
You do not need to change the branding bundles in order to change the name of BI launch pad, but you may want to also change related images, which will require changes in the branding bundles.

1. Copy the `BIlaunchpad.properties` file from

   ```
   <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\warfiles\webapps\BOE\WEB-INF\config\default
   ```

   to

   ```
   <INSTALLDIR>\SAP BusinessObjects Enterprise XI 4.0\warfiles\webapps\BOE\WEB-INF\config\custom
   ```

Note:
Do not change files in the `default` folder. You should always make your changes to copies of the files stored in the `custom` folder.

2. Modify the following properties:

```
app.name=BI launch pad
app.name.greeting=BusinessObjects
app.name.short=BI launch pad
app.url.name=/BI
```

3. Re-deploy `BOE.war` to your Java application server.

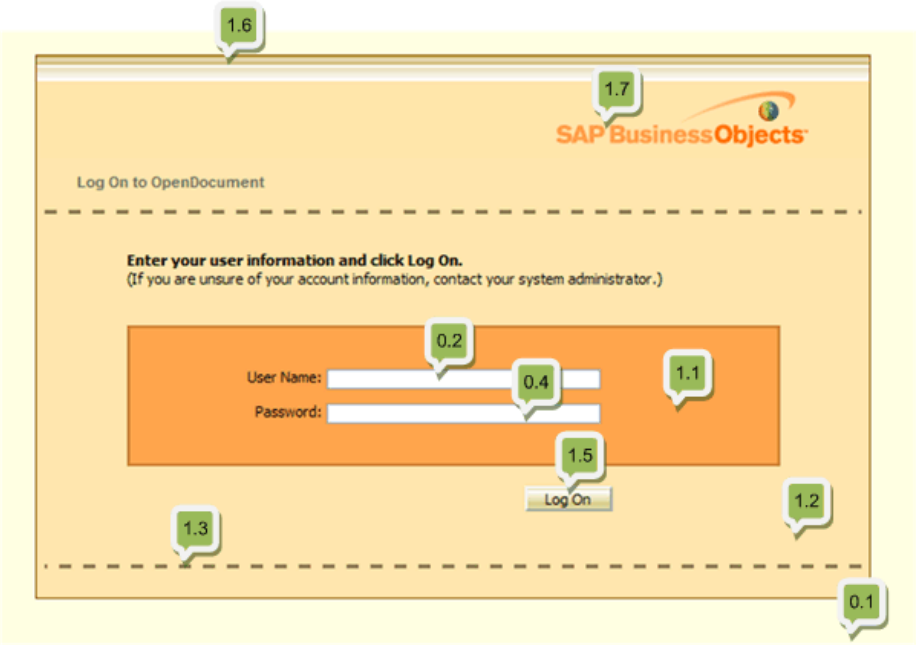
4.4 Customizing OpenDocument

For OpenDocument, you can customize the logo, backgrounds, and styles of the logon page. Most of these customizations involve changing the CSS rules in the `customize.css` file. All customizations must be made available in the `web\service` folder of `com.businessobjects.webpath.OpenDocumentBranding` to take effect, as follows:

```
\web
  \service
    \css
      customize.css
    \images
      \theme
        *.png
```

Note:
Sample customizations are provided in the JAR file. For example, when you open `com.businessobjects.webpath.OpenDocumentBranding`, there is a `web\sample` folder which includes a sample CSS file, sample images, and a README file.

The following diagram shows the elements customized in the sample branding bundle, for reference purposes. The numbers in the balloons refer to sections in the bundle's `customize.css` file.



1. (0.1) customize the background of pages and sub-pages (inside frames)
2. (0.2) input text field
3. (0.3) password field
4. (1.1) authentication fields container
5. (1.2) container of everything
6. (1.3) horizontal rules
7. (1.5) "Log On" button
8. (1.6) banner background pattern
9. (1.7) logo

4.5 Customizing the Crystal Reports JavaScript viewer

This section shows how to customize the report viewer included in your BI platform deployment.

The viewer can be customized by adding the following:

- A custom logo
- SAP Crystal Reports JavaScript API event and action listeners
- CSS files
- External JavaScript files or libraries

4.5.1 Customizing the viewer

You can repackage the `template.zip` file with the files you are using to customize the report viewer.

The workflow is as follows:

1. Extract the contents of the `template.zip` file.
2. Modify the `com.businessobjects.webpath.CrystalReports_oem.jar` file.
3. Enable custom viewer behavior, by setting the `crystal_enable_jsapi` property to `true` in the `SAP BusinessObjectsEnterprise XI 4.0\warfiles\webapps\config\custom\CrystalReports.properties` file.
4. Recreate the `template.zip` file.

Note:
It is recommended to make a backup copy of the `template.zip` file before modifying its contents.

Within the `template.zip` file, the following file must be modified:

template.zip	Modify
`SAP BusinessObjectsEnterprise XI 4.0\warfiles\webapps\BOE\WEB_INF\eclipse\plugins\com.businessobjects.webpath.CrystalReports_oem.jar`	Unzip and modify.

Within the `com.businessobjects.webpath.CrystalReports_oem.jar` file, the following may be modified:

CrystalReports_oem.jar	Modify
`\web`	Add custom or external JavaScript and CSS files to this folder.
`\web\CustomListener.js`	Add SAP Crystal Reports JavaScript API event listeners to the `OnViewerInit` and `OnViewerFail` functions in the `CustomListener.js` file. For more information, see the *SAP Crystal Reports JavaScript API Guide*.
`\WEB-INF\classes\JSAPI-properties.json`	Add a relative path to all images, JavaScript files, and CSS files added to the `\web` folder. You can also change the logo that will be displayed by the viewer. In the following example, a logo, a JavaScript file, a folder and its JavaScript contents, and a CSS file are added: `{ "logo" : { "img" : "images/logo.gif", "tooltip" : "SAP Crystal Reports", "url" : "http://www.businessobjects.com/ipl/default.asp?destination=ViewerLogoLink&product=crystalreports&version=14%2E0" }, "scripts" : [ CustomListener.js \CustomFiles\*.js ], "styles" : [ \CustomStyle.css ] }` **Note:** All files referenced in the `JSAPI-properties.json` file must be included in the `\web` folder.

SAP Crystal Reports 2011 Customization

5.1 Introduction

SAP Crystal Reports 2011 can be repackaged and sold by partners. You can customize the installed product and the installation program to create a seamless experience for customers. The SAP BusinessObjects customization tool customizes SAP Crystal Reports and its installation program with changes such as the following:

- Reducing the product size
- Renaming the product
- Changing default properties in the installation program
- Hiding screens in the installation program

To make customizations, you write a configuration file to specify the customizations then run the SAP BusinessObjects customization tool to create a customized installation program. Customers can then use this installation program to install a customized version of the product.

The customization tool can be used to customize a full installation program, a Support Package installation program, and a Patch installation program.

5.2 Quick start for Crystal Reports

This section shows you how to run the customization tool to create a customized installation program for SAP Crystal Reports. It uses the sample configuration file that is provided with this tool. When you finish this tutorial, you can run your customized installation package and install a customized version of Crystal Reports.

The customizations include changing the default installation type, removing features, hard-coding the product keycode, changing the default installation folder, renaming the product, and changing the Windows **Start** menu shortcut. These customizations are described in more detail in the configuration file.

1. Set up the SAP BusinessObjects customization tool.
   a. Create a working folder on your development machine, for example: `C:\SAPCustomTool\packages`.
   b. Copy the contents of the Crystal Reports installation package to `C:\SAPCustomTool\packages`.

The installation package contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries. See To download the installation program for instructions.

c. (Optional) Add your keycode to the sample configuration file.

In an XML editor, open the file `C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool\example_customization_win_cr.xml` and replace the phrase `PLEASE SET` in `<replaceProperty id="ProductKey" defaultValue="PLEASE SET" />` with your Crystal Reports keycode.

d. Create the folder `C:\SAPCustomTool\output`.

Note:
This folder must be empty.

e. Run the following command from the command prompt: `cd C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool`

The folder `CustomizationTool` contains the executable `customizationtool.exe` and the sample configuration file `example_customization_win_cr.xml`.

2. Run the following command from the command prompt:

```
customizationtool.exe xml=example_customization_win_cr.xml packageDir=C:\SAPCustomTool\packages outputDir=C:\SAPCustomTool\output logDetail=error > C:\oemlog.log
```

Verify that the customized installation package was created at `C:\SAPCustomTool\output`. Ensure no errors were reported in the log file `C:\oemlog.log`.

Note:
The customization tool may take several minutes to complete. You can check its progress by viewing the log file.

3. Use `C:\SAPCustomTool\output\setup.exe` to run the customized Crystal Reports installation program.

Crystal Reports is installed with the customizations described in the configuration file `C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool\example_customization_win_cr.xml`.

5.3 To download the installation program

1. Go to https://service.sap.com/bosap-support > **Software Downloads**.
2. On the "Find your software" tab, under the "A–Z Index", select **Installations and Upgrades**.
3. Select **C** > **CRYSTAL REPORTS** > **CRYSTAL REPORTS 2011**.
4. Select **Installation and Upgrade** > **WINDOWS**.
5. Select the object titled "SAP Crystal Reports 2011 <version> Windows (32B)", and then follow the instructions on the website to download and extract the objects.

Note:
The software may take a long time to download, and you may need to contact the system administrator to ensure that your company's firewall will not terminate the download process.

Support Packages and Patches are installation programs that contain updates to SAP Crystal Reports. You can download them from https://service.sap.com/bosap-support. On the "Find your software" tab, under the "A–Z Index", click **Support Packages and Patches**. For more information on installing Support Packages and Patches, see Customizing update installation programs.

5.4 Planning the customization process

To use the SAP BusinessObjects customization tool:

1. Download the installation program. See To download the installation program.
2. Decide what customizations are required. See Creating the configuration file.
3. Write the configuration file to specify the customizations.
4. Run the customization tool to create a customized installation program.
5. Run the customized installation program to install a customized version of SAP Crystal Reports.

5.4.1 Best practices

This section provides recommendations for creating a customized installation program.

Validate the configuration file

You may want to validate the configuration file before running the tool. Use the `validate` command-line parameter.

Reduce product size

Customers prefer a smaller installation program and a smaller installed product. To keep the product as small as possible:

- Remove any language packs that are not required.
- Remove any features that are not required.
- Remove any items from the `Collaterals` folder that are not required.

Apply customized names consistently

The product name and version number appear in several places in the installation program and in the installed product. Ensure you verify customizations in the following locations:

- Product name, product version, and product major version
- Windows **Start** menu entry and all feature shortcuts

- Windows "Add Remove Program" utility
- Default installation folder

Consider name changes in all languages

It is good practice to consider how the customized name appears in all supported languages.

Modify patch installation programs to be consistent with the main installation program

You must apply the same customizations to Support Packages and Patches that you applied to the main release. If you release a customized main installation program and then try to release a Support Package or Patch installation program with different customizations, you might see unpredictable results which might not be repairable using standard rollback procedures.

Test rollback, modify, and repair installations for Support Packages and Patches

Rollback, modify, and repair installations are supported for customized Support Packages and Patches, provided they have been customized in a manner consistent with the main installation package. It is recommended to test these scenarios.

Related Topics

- Command line parameters

5.5 Creating the configuration file

The following section describes the customizations you can make to the installation program by editing the configuration file:

- Renaming the product
  - Customizing the product name and version number
  - Customizing the Windows **Start** menu shortcuts
  - Customizing the Windows "Add Remove Program" utility
  - Customizing the installation folder
- Customizing default user input
- Removing installation screens
- Embedding a keycode
- Removing features
- Preventing prerequisite checks
- Removing language packs
- Changing resources
  - Customizing the images in the installation program
  - Customizing the license agreement
- Removing items from the Collaterals folder

5.5.1 Configuration file overview

The SAP BusinessObjects customization tool uses information in the configuration file to perform the customizations. The configuration file is an XML document, and you use XML elements to describe your customizations. The sample configuration file is contained in this folder in the installation program:

`Collaterals\Tools\CustomizationTool\example_customization_win_cr.xml`

The file must have this format:

```
<oem name="<Any name>">
  <cloneProduct sourceId="product.crystalreports-4.0-core-32">
  ...
  ...
  </cloneProduct>
</oem>
```

The configuration file for the full installation program can have any name, for example, `oem.xml`.

The configuration file for the Support Package installation program is described in the section How to customize update installation programs.

Note:
The configuration file must be written in correct XML syntax. Use an XML editor to create and edit the file, and verify that the format is correct before running the tool.

Example:

This example specifies the following customizations:

- Change the product's long name to “Custom Company Crystal Reports” for all languages.
- Change the product's short name to “Custom CR” for all languages.
- Change the publisher and product name for the "Windows Add Remove Program" entry.
- Remove the installation screen titled "Choose Installation Type" and set the installation type to "Custom".
- Specify that the only language packs that are included in the installation package are English, French, German, Italian, and Chinese.

```
<oem name="CustomCompanyCrystalReports">
  <cloneProduct sourceId="product.crystalreports-4.0-core-32">

    <replaceString id="product.cr_name" value="Custom Company Crystal Reports" lang="all"/>
    <replaceString id="product.cr_shortname" value="Custom CR" lang="all"/>

    <arp duSourceId="product.crystalreports.arp-4.0-core">
      <arg id="publisher" value="Custom Company"/>
      <arg id="display_name" value="Custom Company Crystal Reports"/>
    </arp>

    <replaceProperty id="InstallType" defaultValue="custom"/>
    <removeDialog id="ChooseInstallType2.dialog"/>

    <languageIncludeList value="en;fr;de;it;zh_CN"/>

   </cloneProduct>
</oem>
```

5.5.2 Renaming the product

You can rename the product by customizing the following:

- The product name and version number
- The Windows "Add or Remove Programs" entry
- The "Start" menu entry for feature shortcuts
- The default installation folder

The following sections explain these customizations.

5.5.2.1 Customizing the product name and version number

You can customize the product name and version number. Use the `replaceString` element with the desired string ID:

```
<replaceString id="<string id>" value="<new value>" lang="<language list>"/>
```

There are four strings that represent the product name and version number: the product long name, the product short name, the product version number, and the product major version number. The full product name is composed of the product long name and the version number. The product short name and product major version are used in the Windows shortcut menu.

Table 5-1: Product name and version number

String description	String ID	Default value
Product long name	`product.cr_name`	Crystal Reports
Product short name	`product.cr_shortname`	Crystal Reports
Product version	`product_cr_version`	2011
Product major version	`product_cr_majorversion`	2011

Note:
You should customize the product version and product major version together. For example, if you change product version to “1.0” you should also customize product major version to “1”. Otherwise the version number in the menus will not match the version number in the product.

You can specify a new name for each language. For a list of language codes, see Language codes.

Example:

This example makes the following customizations:

- Change the product long name to “Custom Company Crystal Reports” and the product short name to “Custom CR” for English.
- Change the product long name to “Custom Company Crystal Reports (French)” and the product short name to “Custom CR (French)” for French.
- Change the product version to “1.0” and the product major version to “1” for all languages.

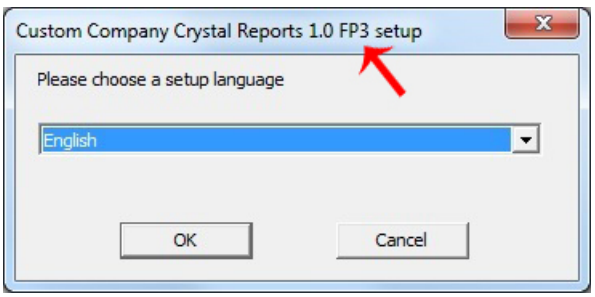
The product name in languages other than English and French will remain as the default value, but the product version and major version will be changed for all languages.

```
<replaceString id="product.cr_name" value="Custom Company Crystal Reports" lang="en"/>
<replaceString id="product.cr_shortname" value="Custom CR" lang="en"/>

<replaceString id="product.cr_name" value="Custom Company Crystal Reports (French)" lang="fr"/>
<replaceString id="product.cr_shortname" value="Custom CR (French)" lang="fr"/>

<replaceString id="product_cr_version" value="1.0" lang="all"/>
<replaceString id="product_cr_majorversion" value="1" lang="all"/>
```

The result of the customization appears below. Notice the version number “FP3” is not removed:



To remove instances of “FP3” from the installation program

When you run the installation program, you may see instances of “FP3” in the product name. To remove “FP3”, modify the lines in the following files:

File name	Original line	Modified line
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\uitext\CrystalReports\product.lang_<language code>.uitext.xml	<string id="product name_patch" value=" FP3"/>	<string id="product name_patch" value=""/>

File name	Original line	Modified line
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\uitext\framework\setup.ui.framework.lang_<language code>.uitext.xml	<string id="product_patch" value="FP3"/>	<string id="product_patch" value=""/>
Same as above	<string id="product_patch_prespace" value=" FP3"/>	<string id="product_patch_prespace" value=""/>

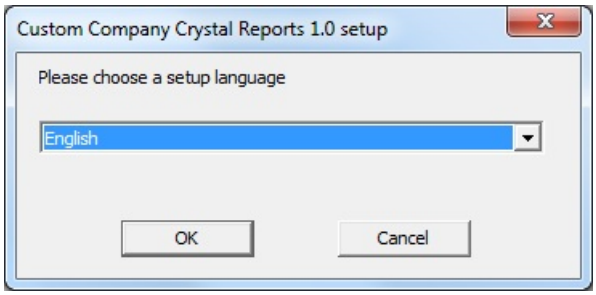
You must modify one file for every language that the installation program supports. For a list of language codes, see Language codes. When you run the customization tool, and then run the installation program, all instances of “FP3” will be removed. This process will be simplified in a future release.

Example:

To remove “FP3” from the English installation program, modify the following files:

- `product.lang_en.uitext.xml`
- `setup.ui.framework.lang_en.uitext`

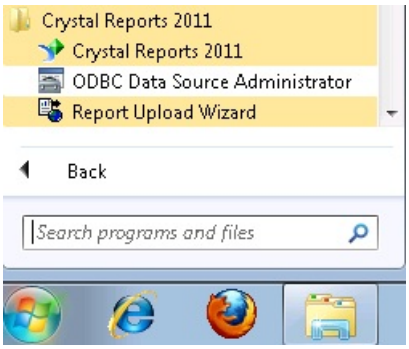
The result of the customization appears below:



5.5.2.2 Customizing the Windows Start menu shortcuts

The Windows **Start** menu contains shortcuts for features such as the ODBC Data Source Administrator. You can customize the name, location, and tooltip for each shortcut. Any shortcut that you do not customize will be grouped under the default **Start** menu entry, "Crystal Reports 2011".

The default **Start** menu in English installations looks like this:



Use the `shortcut` element to customize the location, shortcut name, and tooltip for each feature:

```
<shortcut duSourceId="<shortcut deployment unit ID>">
  <arg id="linkFullPath" value="<full path to shortcut link>" lang="<language list>"/>
  <arg id="description" value="<tooltip string>" lang="<language list>"/>
</shortcut>
```

Attribute	Value
`duSourceId`	The shortcut deployment unit ID that you want to modify. Typical values in clude: • `product.crystalreports.shortcut.crw-4.0-core` Crystal Reports 2011 • `product.crystalreports.shortcut.odbc-4.0-core` ODBC Data Source Administrator • `product.crystalreports.shortcut.rptpubwiz-4.0-core` Report Upload Wizard For a complete list of `duSourceId` values, see Shortcut deployment unit IDs.
`linkFullPath`	The full path to the shortcut link. Be sure to add .lnk to the shortcut link or the link will not be created. You can put the link in the **Start** menu or you can put it on the desktop. The SAP BusinessObjects customization tool will create the links correctly. You can specify one link for each language. For a list of language codes, see Language codes.
`description`	The tooltip string to display when the user hovers the mouse over the shortcut. You can specify one tooltip for each language.

Example:

This example makes the following customizations:

- Change the name of the Crystal Reports 2011 shortcut to “Custom Company CR” for English.
- Customize the Crystal Reports 2011 tooltip to “Launch Custom Company CR” for English.
- Change the name of the “ODBC Data Source Administrator” shortcut to “Custom ODBC” for English.
- Customize the “ODBC Data Source Administrator” tooltip to “Custom ODBC” for English.
- Place the “Custom Company CR” and “Custom ODBC” shortcuts under the **Start** menu entry called “Company Programs”.
- Change the name of the “Report Upload Wizard” shortcut to “Custom Wizard” for English.
- Place the “Custom Wizard” shortcut under the **Start** menu entry called “Custom Wizard”.
- Customize the “Custom Wizard” tooltip to “Launch Custom Wizard” for English.

The shortcut name and tooltip will remain unchanged for all other languages.

Note:

To use this example you must create the following links and folders:

- `Custom Company CR.lnk`
- `Custom ODBC.lnk`
- `Custom Wizard.lnk`
- `Company Programs`
- `Custom Wizard`

Place `Custom Company CR.lnk` and `Custom ODBC.lnk` in the folder `Company Programs` and place `Custom Wizard.lnk` in the folder `Custom Wizard`. Place these folders in the same location you plan to redirect the installation folder to.

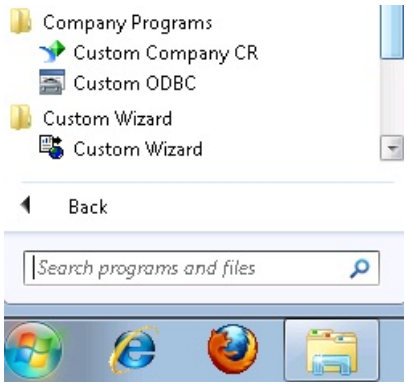
In this example, the installation folder has been redirected to the `C:\` drive. See Customizing the installation folder for more information.

```
<shortcut duSourceId="product.crystalreports.shortcut.crw-4.0-core">
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\Custom Company CR.lnk" lang="en"/>
  <arg id="description" value="Launch Custom Company CR" lang="en"/>
</shortcut>

<shortcut duSourceId="product.crystalreports.shortcut.odbc-4.0-core">
  <arg id="linkFullPath" value="[programmenufolder]\Company Programs\Custom ODBC.lnk" lang="en"/>
  <arg id="description" value="Custom ODBC" lang="en"/>
</shortcut>


<shortcut duSourceId="product.crystalreports.shortcut.rptpubwiz-4.0-core">
  <arg id="linkFullPath" value="[programmenufolder]\Custom Wizard\Custom Wizard.lnk" lang="en"/>
  <arg id="description" value="Launch Custom Wizard" lang="en"/>
</shortcut>
```

The result of the customization appears below:



Example:

This example keeps the default name of the “Report Upload Wizard” feature shortcut, but places it under the **Start** menu entry called “Custom Wizard” for all languages. It also changes the “Custom Wizard” tooltip to “Launch Custom Wizard” for all languages.

Note:
To use this example you must place the `Report Upload Wizard.lnk` in the folder `Custom Wizard`. Place this folder in the same location as the installation folder.

In this example, the installation folder has been redirected to the `C:\` drive.

```
<shortcut duSourceId="product.crystalreports.shortcut.rptpubwiz-4.0-core">
  <arg id="linkFullPath" value="[programmenufolder]\Custom Wizard\Report Upload Wizard.lnk" lang="all"/>
  <arg id="description" value="Launch Custom Wizard" lang="all"/>
</shortcut>
```

5.5.2.3 Customizing the Windows Add Remove Program utility

You can customize the display name, the publisher, and the icon in the Windows "Add Remove Program" (ARP) utility. Use the following element:

```
<arp duSourceId="product.crystalreports.arp-4.0-core">
  <arg id="publisher" value="<publisher name>"/>
  <arg id="display_name" value="<product name>" lang="<language list>"/>
  <arg id="display_icon" value="<full path to icon>"/>
</arp>
```

Note:
The display name must have a `lang` tag to specify a different display name for each language. Multiple languages using the same display name must be separated by a semi-colon. Any unspecified languages will use the default value.

For a list of language codes, see Language codes.

Note:

You must take additional steps in order for the customized value of the publisher to display correctly in the ARP. Follow these steps:

1. Open the file `dunit\product.crystalreports.arp-4.0-core-32\seed.xml`.
2. Find the element `<action id="AddARPEntry">`. You will see multiple nested `<arg>` elements.
3. Add the following line under `<arg name="DISPLAY_NAME" value="[ARP.DISPLAYNAME]" />`:

   ```
   <arg name="PUBLISHER" value="SAP" />
   ```

4. Save the file.

After you customize the publisher name in the configuration file, run the customization tool, and then install the customized product, your customized value for the publisher displays in the ARP. This process will be simplified in a future release.

Icons displayed in the Windows ARP utility are typically 16x16. Refer to Windows documentation for complete information on creating the icon.

Example:

This example makes the following customizations in the Windows ARP utility:

- Change the product name to “Custom Company Crystal Reports Patch 1” for English and French.
- Change the product name to “Custom Company Crystal Reports (German)” for German.
- Change the publisher to “Custom Company”.
- Replace the display icon with the icon `C:\SAPCustomTool\CC_logo.ico`.

Note:

To use this example you must put an icon called `CC_logo.ico` in the location `C:\SAPCustomTool`.

```
<arp duSourceId="product.crystalreports.arp-4.0-core">
  <arg id="publisher" value="Custom Company"/>
  <arg id="display_name" value="Custom Company Crystal Reports Patch 1" lang="en;fr"/>
  <arg id="display_name" value="Custom Company Crystal Reports (German)" lang="de"/>
  <arg id="display_icon" value="C:\SAPCustomTool\CC_logo.ico"/>
</arp>
```

The result of the customization appears below:

5.5.2.4 Customizing the installation folder

You can customize the location of the default installation folder. Use the `replaceProperty` element with `id="<installation folder file path>"`:

```
<replaceProperty id="InstallDir" defaultValue="<default installation folder>"/>
```

Example:

Change the default install folder to `C:\MyInstallDIR\CustomCompanyCrystalReports`.

```
<replaceProperty id="InstallDir" defaultValue="C:\MyInstallDIR\CustomCompanyCrystalReports"/>
```

5.5.3 Customizing default user input

You can customize the default value of the user input that is collected by the installation program. Use the `replaceProperty` element with `id="<property id>"` and the new default value:

```
<replaceProperty id="<property id>" defaultValue="<value to use as default value>"/>
```

For a list of property IDs, see Installation screen and property IDs.

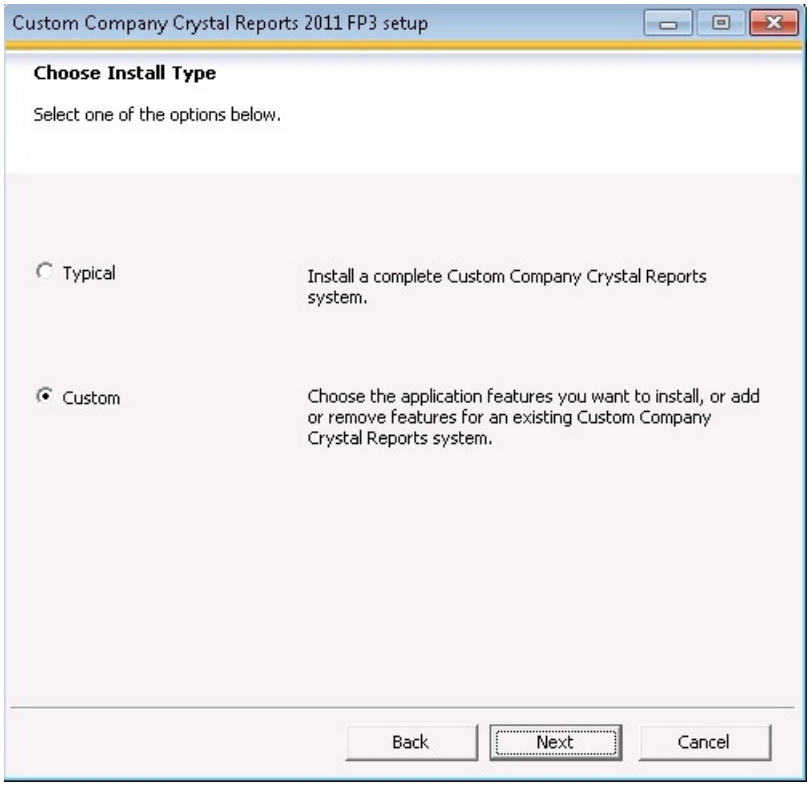
The Windows installation program collects user input using dialog boxes, radio buttons, and other user interface elements.

Example:

On the installation screen called "Choose Install Type", the default install type is "Typical". This example changes the default install type to "Custom".

```
<replaceProperty id="InstallType" defaultValue="custom"/>
```

The result of the customization appears below:



5.5.4 Removing installation screens

You can remove installation screens from the installation program. Use the `removeDialog` element with the installation screen ID:

```
<removeDialog id="<installation screen ID>"/>
```

For a list of installation screen IDs, see Installation screen and property IDs.

Example:

This example shows how to remove the installation screen titled "Select Features".

```
<removeDialog id="SelectFeatures.dialog"/>
```

5.5.5 Embedding a keycode

You can embed a keycode in the installation program so the customer does not need to enter one. This task involves:

- Providing a default value for the keycode
- Removing the installation screen in which the user enters a keycode

Example:

Use the `replaceProperty` element with `id="ProductKey"` to provide a default keycode. Keycodes must have the format `XXXXX-XXXXXXX-XXXXXXX-XXXXXXX-XX`.

Use the `removeDialog` element with `id="CREnterProductKey.dialog"` to remove the installation screen for the license key.

```
<replaceProperty id="ProductKey" defaultValue="XXXXX-XXXXXXX-XXXXXXX-XXXXXXX-XX"/>
<removeDialog id="CREnterProductKey.dialog"/>
```

Related Topics

- Installation screen and property IDs
- Customizing default user input
- Removing installation screens

5.5.6 Removing features

SAP Crystal Reports includes many optional features. You can remove a feature from the installation program. Use the `removeFeature` element with `id="<feature id>"`:

```
<removeFeature id="<Feature ID>"/>
```

For a list of feature IDs, see Feature IDs.

When you specify a feature to be removed, the SAP BusinessObjects customization tool removes all executables, installation screens, and other files that belong to that feature. Removing unnecessary features is a good way to reduce the size of the customized product.

Example:

Remove the geographic mapping feature. This ID will remove the program's ability to display relationships between data and geographic regions:

```
<removeFeature id="Mapping"/>
```

5.5.7 Preventing prerequisite checks

Prerequisites are conditions that must exist on the host machine in order for the installation program to succeed. The installation program verifies the existence of these prerequisites before starting, and displays the results in the "Prerequisite check" screen. Removing the "Prerequisite check" screen prevents prerequisite checks from being performed. Use the `removeDialog` element with `id="CheckPreRequisites.dialog"`

Note:
It is recommended that you remove this installation screen only if you are performing the prerequisite checks by some other means. If the prerequisites are not met, the installation program will fail.

Example:

This example removes the "Prerequisite check" screen and prevents prerequisite checks from being performed.

```
<removeDialog id="CheckPreRequisites.dialog"/>
```

5.5.8 Removing language packs

The installation program allows the user to select which language packs to install. A language pack contains translated versions of all the strings that are used by the installed product. By default, all possible language packs are included in the installation program. You can specify which language packs to include. Use the `languageIncludeList` element with a list of language codes:

```
<languageIncludeList value="<list of language codes>"/>
```

For a list of language codes, see Language codes.

Note:
Language packs can be large. The installation program will be smaller if fewer language packs are included.

Example:

Include English, French, and German language packs in the installation program. The user can select from this list during installation.

```
<languageIncludeList value="en;fr;de"/>
```

5.5.9 Changing resources

The installation program stores image and text files as resources in this folder:

```
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\resources
```

You can customize the resources in this folder. Resources that are commonly customized include:

- Images in the installation program
- License agreement in the installation program

To customize a resource:

1. Create a custom resources folder, for example `C:\MyResources`. The folder can have any name, but note that it will be visible to customers. Use the same folder for all resources that you customize.
2. Create a new resource with the same name and filepath as the original resource, and place it into the custom resources folder. See the related topics section for specific examples.
3. Add the `<resources>` element to the configuration file to specify the location of the custom resources folder, for example:

   ```
   <resources cleanTarget="no" sourcePath="C:\MyResources"/>
   ```

cleanTarget attribute

If you set `cleanTarget='yes'`, the customization tool will delete the original `resources` folder and use only those resources in the custom resources folder. This option is not recommended.

Related Topics

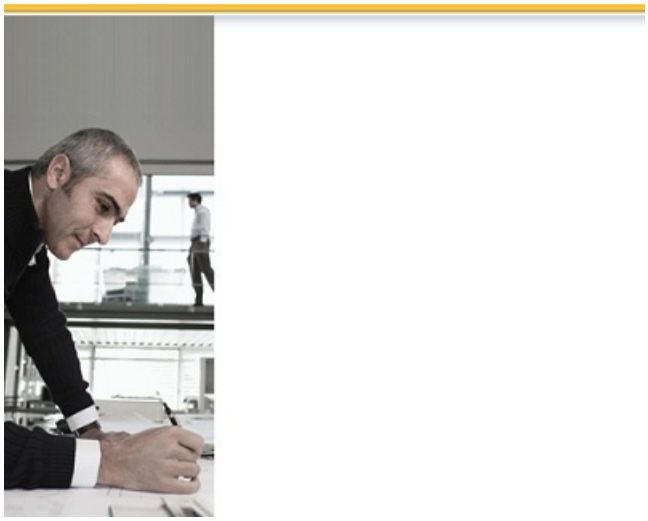
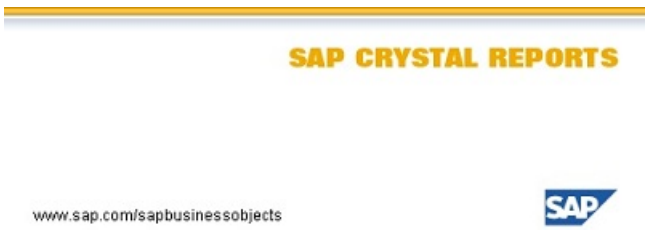
- Customizing the images in the installation program
- Customizing the license agreement

5.5.9.1 Customizing the images in the installation program

You can customize the images in the installation program including the welcome screen, the top image for all screens, and the billboard for the progress dialog. Images are stored as files in the resources folder:

```
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\resources
```

Table 5-4: Image files in the `resources` folder

Image name	File name	Default image
Welcome screen	`dialogFull.bmp`	
Top image for all screens	`dialogTop.bmp`	
Billboard for progress dialog	`billboard.bmp`	

You customize an image by creating a new image file, putting the file in the custom resources folder, and adding the `resources` element to the configuration file.

Example: Customizing the image in the welcome screen

1. Create a folder called `MyResources` in the `C:\` drive.
2. Create a new image file called `dialogFull.bmp` and place it in the `C:\MyResources` folder.
3. Ensure that the `resources` element exists in the configuration file as follows:

```
<resources cleanTarget="no" sourcePath="C:\MyResources"/>
```

Related Topics

- Changing resources

5.5.9.2 Customizing the license agreement

You can customize the license agreement that is presented to the user during installation. License agreements are stored as text files in the resources folder:

```
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\resources\<lan
guage code>
```

For example, the English license agreement is located here:

```
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\resources\en\li
cense_en.rft
```

For a list of language codes, see Language codes.

You customize the license agreement by creating a new license file, putting the file in the custom resources folder, and then adding the `resources` element to the configuration file.

Example: Customize the English license agreement

The English license agreement is stored here:

```
dunit\product.crystalreports-4.0-core-32\setup.ui.framework\resources\en\li
cense_en.rtf
```

To customize the English license agreement:

1. Create a folder called `MyResources` in the `C:\` drive.
2. Create a folder called `en` and place it in the `C:\MyResources` folder.
3. Create a new license agreement file called `license_en.rtf` and place it in the `C:\MyRe sources\en` folder.
4. Ensure that the `resources` element exists in the configuration file as follows:

   ```
   <resources cleanTarget="no" sourcePath="C:\MyResources"/>
   ```

Related Topics

- Changing resources

5.5.10 Removing items from the Collaterals folder

The SAP Crystal Reports installation program stores tools, samples, and documentation in the `Collaterals` folder of the installation program. By default, a customized installation program that is delivered to customers will also contain the `Collaterals` folder with the same contents. You can remove unwanted items from the `Collaterals` folder in order to reduce the size of your customized installation program. Use the `collaterals` element with `cleanTarget="yes"` and `sourcePath="<full path to custom Collaterals folder>"`:

```
<collaterals cleanTarget="yes" sourcePath="<full path to custom Collaterals folder>"/>
```

Note:
You must set the `cleanTarget` attribute to `yes` so that the customization tool will replace the original folder with the new folder.

To remove items from the Collaterals folder

1. Copy the contents of the existing `Collaterals` folder to a new location, for example `C:\MyCollaterals`.
2. Remove any items from `C:\SAPCustomTool\Collaterals` that are not required by your customized installation program.
3. Add the `<collaterals>` element to the configuration file to specify the location of the custom collaterals folder, for example:

```
<collaterals cleanTarget="yes" sourcePath="C:\MyCollaterals"/>
```

Table 5-5: Description of items in the `Collaterals` folder

Folder	Description	When to remove
`Collaterals > Add-Ons > SAP`	Provides connectivity to SAP systems.	Remove this folder if there is no need to connect to SAP systems.
`Collaterals > CustomizationTemplate`	Contains the sample `template.zip` file for customizations to the report designer.	Remove this folder if the customers do not need to provide a sample `template.zip` file.
`Collaterals > Docs`	Documentation in every language that Crystal Reports supports.	Remove any languages that are not included in the customized installation program. For a list of language codes, see Language codes.
`Collaterals > Tools > CustomizationTool`	The SAP BusinessObjects customization tool.	Remove this folder if the customers do not need to customize their own installation programs.

5.6 Customizing the report designer

You can customize the following properties of the report designer:

- Splash screen
- Start page
- String values on the menu

You can make these customizations after installing the program. You can also deploy your customizations to your customized installation package, so that the customizations are applied when users install the program.

5.6.1 Customizing the splash screen

When Crystal Reports is run, a splash screen loads. You can replace this splash screen with your own bitmap.

Note:
The following steps assume you already installed Crystal Reports. If you want to deploy the splash screen into your customized installation program, rename the bitmap you want to use for the splash screen to `splash.bmp` and follow the instructions in Deploying the OEM customization file.

1. Rename the bitmap you want to use for the splash screen to `splash.bmp`.

   Note:
   The bitmap must be a valid `.bmp` file and can be of any size.

2. Place `splash.bmp` in the same folder as `crw32.exe`.

   By default, `crw32.exe` is found in the following location:

   ```
   C:\Program Files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise
   XI 4.0\win32_x86
   ```

When Crystal Reports is run, `splash.bmp` should load. If it does not load, then the default splash screen loads instead.

5.6.2 Customizing the start page

You can modify the content of the start page with your own HTML file. Most of your customizations will affect the top part of the start page. You can also remove the bottom part, which contains links to SAP Crystal Reports web pages.

Note:
The following steps assume you already installed Crystal Reports. If you want to deploy the start page into your customized installation program, rename the HTML file you want to use for the start page to `start.html` and follow the instructions in Deploying the OEM customization file.

1. Rename the HTML file you want to use for the start page to `start.html`.
2. Place `start.html` in the `Start Page\<language code>` sub-folders, depending on the languages you want to support.

   Note:
   By default, the file path of the sub-folder is:

   ```
   C:\Program Files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\win32_x86\Start Page\<language code>
   ```

   For a list of all language codes, see Language codes.

   Tip:
   If `start.html` uses images, place them in the following location:

   ```
   C:\Program Files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\win32_x86\Start Page\image
   ```

   Your HTML file should use relative paths to point to this folder.

When Crystal Reports is run, the start page will display the customizations you made in `start.html`.

5.6.3 Customizing menu strings

You can modify string values on the menu that contain the product name SAP Crystal Reports. Such values are:

Property name	Description	Location	Default value
`ProductName`	Product name	Window title	SAP Crystal Reports
`CrystalReportHelp`	Product help	**Help** menu	SAP Crystal Reports Help
`AboutCrystalReport`	About product help	**Help** menu	About SAP Crystal Reports

To customize these strings, an XML file is required. The XML file name must have the following format:

```
crw_oem_res_<language code>.xml
```

For example, the English XML file name is:

`crw_oem_res_en.xml`

For a list of language codes, see Language codes.

Note:
If you have already installed Crystal Reports, then the XML file should be placed in the same folder as `crw32.exe`. By default, this is found in:

`C:\Program Files (x86)\SAP BusinessObjects\SAP BusinessObjects Enterprise XI 4.0\win32_x86`

When Crystal Reports is run in a specific language, the corresponding language XML file is loaded.

Note:
If you want to deploy the customized strings into your customized installation program, follow the instructions in Deploying the OEM customization file.

Example:

This example makes the following customizations:

- Change the `ProductName` value to `Custom CR`
- Change the `CrystalReportHelp` value to `Custom CR help`
- Change the `AboutCrystalReport` value to `About Custom CR`

```
<Root>
 <ProductName>Custom CR</ProductName>
 <MainFrameMenu>
 <Help>
 <CrystalReportHelp>Custom CR help</CrystalReportHelp>
 <AboutCrystalReport>About Custom CR</AboutCrystalReport>
 </Help>
 </MainFrameMenu>
</Root>
```

Note:

- To support multiple languages, the attribution encoding should be UTF-8: `<?xml version="1.0" encoding="UTF-8"?>`. In addition, when the XML file is saved with a text editor, select **UTF-8** from the **Encoding** menu.
- Keep the property name and value in the same line. For example the following is acceptable:

  ```
  <ProductName>Custom CR</ProductName>
  ```

  The following is not acceptable. There will be unrecognizable characters in the modified strings when Crystal Reports is run:

  ```
  <ProductName>
  Custom CR
  </ProductName>
  ```

5.6.4 Deploying the OEM customization file

After you prepare your customized files (splash images, start page, and menu strings), you can deploy your customizations into the installation package. Place the customized files in a zip file.

1. Create a zip file named `template.zip`.
2. Place the customized files into the zip file.

   Note:
   The folder structure within the zip file must match the structure of the folder where you want the files placed, relative to the installation folder. Files must be placed in the following location in `template.zip`: `SAP BusinessObjects Enterprise XI 4.0\win32_x86`.

   For example, the following customized files are placed in these locations in `template.zip`:

Customized file	Location in `template.zip`
`splash.bmp`	`SAP BusinessObjects Enterprise XI 4.0\win32_x86`
`start.html` (for English)	`SAP BusinessObjects Enterprise XI 4.0\win32_x86\Start Page\en`
`crw_oem_res_en.xml`	`SAP BusinessObjects Enterprise XI 4.0\win32_x86`

3. Copy the zip file to the following location:
   `dunit\product.crystalreports.oemzips-4.0-core-nu\OEMZips`

   Note:
   The OEMZips folder may need to be created manually.

4. Run the installer.

The contents of `template.zip` will be unzipped to the installation folder.

Note:
In the install package of SAP Crystal Reports, there is a sample zip file located in:

`Collaterals\CustomizationTemplate\template.zip`

5.7 Running the tool

The SAP BusinessObjects customization tool `customizationtool.exe` is included with the SAP Crystal Reports installation package in this location:

`Collaterals\Tools\CustomizationTool`

This section explains the command line parameters used for the tool.

Note:
The SAP BusinessObjects customization tool may take several minutes to complete. You can check its progress by viewing the log file.

Example:

This example runs the customization tool and creates a log file located in the `C:\` drive. To use this example, you must do the following:

- Create a configuration file called `oem.xml` in the location `C:\SAPCustomTool`.
- Download the Crystal Reports installation package to the location `C:\SAPCustomTool\packages`. See To download the installation program.
- Create a folder called `output` in the location `C:\SAPCustomTool`.
- Run the following command from the command prompt: `cd C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool`

```
customizationtool.exe xml=C:\SAPCustomTool\oem.xml packageDir=C:\SAPCustomTool\packages
outputDir=C:\SAPCustomTool\output logDetail=error > C:\oemlog.log
```

For more information on how to run the SAP BusinessObjects customization tool, see Quick start for Crystal Reports.

5.7.1 Command line parameters

Table 5-8: Required parameters

Parameter	Description	Example
`xml`	Full path to the configuration file. The configuration file for the full installation program can have any name.	`xml=C:\SAPCustomTool\oem.xml`

Parameter	Description	Example
`packageDir`	Full path to the folder that contains the installation program you are modifying. The installation program is downloaded from SAP Service Marketplace in order to start the installation of SAP Crystal Reports. It contains the folders `Collaterals`, `dunit`, `langs`, and `setup.engine` in addition to other binaries.	`packageDir=C:\SAPCustomTool\packages`
`outputDir`	Full path to the folder where the customized installation program will be created. Must be empty before running the tool.	`outputDir=C:\SAPCustomTool\output`

Table 5-9: Optional parameters

Parameter	Description	Example

`baselinePath`	Full path to a root folder containing the original, non-customized versions of all previous full and update installation programs you have customized.
	Use a semicolon (`;`) to separate root folders.
		Assume you want to customize SAP Crystal Reports 2011 Support Package 5 and you customized the previous programs: `2011 (Full install)`, `2011 SP4`. Customize 2011 Support Package 5, and provide the root folder path to the non-customized packages for the major 2011 release and Support Package 4 release. For example, if the non-customized packages are contained in the following directory structure:
		`C:\productUpdates\2011\`
		`\2011 Full\`
		`\SP4\`
		set the value to `baselinePath=C:\productUpdates\2011\`
		See Customizing update installation programs for more information and examples of the `baselinePath` parameter.

Parameter	Description	Example
`logDetail`	The level of detail tracked in the log file. The default value is `info`. The following are the accepted values: • `error` • `warn` • `info` • `debut` • `trace`	`logDetail=warn`
`action`	The tool mode. The following are the accepted values: • `generate` (default value) The tool performs the specified customizations. • `validate` The tool validates the configuration file but does not perform any customizations.	`action=validate`

Related Topics

• Quick start for Crystal Reports

5.8 Customizing update installation programs

Update installation programs are Support Packages or Patches that contain updates to your existing SAP Crystal Reports software. Support Packages contain more updates than Patches but are released less frequently. You can use the SAP BusinessObjects customization tool to customize these installation programs, but some modifications to the command line and configuration file are required.

5.8.1 Frequently asked questions about update installation programs

Where do I find Support Packages and Patches?

1. Go to https://service.sap.com/bosap-support > **Software Downloads**.
2. On the "Find your software" tab, under the "A–Z Index", click **Support Packages and Patches**.

3. Select **C > CRYSTAL REPORTS > CRYSTAL REPORTS 2011 > Comprised Software Component Versions > CRYSTAL REPORTS 2011 > Windows Server on IA32 32bit**.
4. Select your Support Package or Patch, then follow the instructions on the website to download and extract the objects.

What parts of update installation programs can I customize?

You can customize the same aspects of update installation programs as you did in the main installation program. Because Support Package and Patch updates contain fewer installation screens, not all of the customization steps apply. It is recommended to run the Support Package or Patch before customizing it to determine what customizations you require.

How do I customize update installation programs?

Update installation programs use the same architecture as a main installation program for Crystal Reports (full installation), so you can use the customization tool as described in Creating the configuration file and Running the tool, with some modifications to the command line and the configuration file. For more information, see How to customize update installation programs in this section.

Is it necessary to customize and install all Support Package and Patch updates?

No. As with non-customized versions of Crystal Reports, you only need to install the updates that you want. This may be a Support Package, a Patch, or both.

Can I install a non-customized update on a customized Crystal Reports installation?

Yes. Both customized and non-customized Support Packages or Patches may be applied to your customized installation. However, non-customized Support Package or Patch installation programs will not display your branding or installation customizations you created for the main installation program.

I have delivered a customized version of Crystal Reports to customers but I want to modify the customizations in an update installation program. Is this possible?

This scenario is not supported. The customizations that you make to Support Packages and Patches must be consistent with the original customizations.

5.8.2 Quick start for update installation programs

Ensure that you have customized and installed the main (full) installation program (SAP Crystal Reports) using the instructions in Quick start for Crystal Reports, and that the non-customized installation package is located in `C:\SAPCustomTool\packages`.

This section shows you how to run the SAP BusinessObjects customization tool to customize the installation program for a Support Package. It uses the sample configuration file provided with the customization tool. Notice that the sample configuration file contains the `<cloneProduct>` element for the main installation program as well as the `<clonePatchProduct>` element for a Support Package installation program.

Note:
You can run this example only when a Support Package is available on https://service.sap.com/bosap-support.

1. Download the installation program for the Crystal Reports Support Package to the folder `C:\SAP CustomTool\SupportPackage`.
2. Ensure the `product_cr_version` for the `<clonePatchProduct>` element in the configuration file matches the version number of the Support Package that you downloaded. See Customizing the product name and version number.
3. Run the following command from the command prompt: `cd C:\SAPCustomTool\packages\Collaterals\Tools\CustomizationTool`
4. Customize the Crystal Reports Support Package and place the customized installation program in `C:\SAPCustomTool\output\SupportPackage` by using the following command:

   ```
   customizationtool.exe xml=example_customization_win_cr.xml packageDir=C:\SAPCustomTool\SupportPackage baselinePath=C:\SAPCustomTool\packages outputDir=C:\SAPCustomTool\output\SupportPackage logDetail=error > C:\oemlog_SP02.log
   ```

5. Use `C:\SAPCustomTool\output\SupportPackage\setup.exe` to run the customized installation program for the Crystal Reports Support Package.

5.8.3 How to customize update installation programs

Use the configuration tool as described in Creating the configuration file and Running the tool to customize update installation programs for Support Packages and Patches, with the following differences:

- The configuration file must use the `clonePatchProduct` element (with the correct product ID), instead of the `cloneProduct` element.
- The configuration file must contain the complete `<cloneProduct>` element for the main installation package that you are updating. If it does not, it may cause unpredictable results, especially when customizations involve removing features.
- The configuration file cannot contain more than one `clonePatchProduct`. If you are customizing both a Support Package and a Patch, you must create two configuration files: one file containing `cloneProduct` and `clonePatchProduct` for the Support Package, and the other file containing `cloneProduct` and `clonePatchProduct` for the Patch.
- Refer to all prerequisite installation programs using the `baselinePath` command.

All configuration file elements and command-line parameters can be used to customize update installation programs, but not all of them are applicable to every Support Package or Patch. Run the installation program for the Support Package or Patch first to determine what you need to customize, then use the information in Creating the configuration file and IDs and codes for Crystal Reports customization to create the customization file.

To specify the product version in the configuration file

The configuration file for update installation programs must contain the `product version` in the `clonePatchProduct` element as shown below:

```
<oem name="<any name>">
  <clonePatchProduct sourceId="<product version>">
  ...
  </clonePatchProduct>
</oem>
```

The `product version` in the configuration file must match the version number of the installation program that you are customizing. To find the version number, look in the `dunit` folder for a folder with a name in this format:

`product.cr.patch-4.x.x.x-core-32`

You can use the name of this folder as the `product version`.

Example:

This example configuration file customizes SAP Crystal Reports 2011 Patch 1, which has the product version `product.cr.patch-4.1.0.1-core-32`. The configuration file customizes the product long name to Custom Company Crystal Reports and the product short name to Custom CR.

```
<oem name="Custom Patch Tool">
 <clonePatchProduct sourceId="product.cr.patch-4.1.0.1-core-32">
  ...
 </clonePatchProduct>
</oem>
```

To use the baselinePath parameter

Use the command line parameter `baselinePath` to refer to a root folder containing the original, non-customized versions of all previous full or update installation programs you have customized. This means you must keep the original installation packages.

Note:
This parameter replaces the `baselinePackages` parameter introduced in 2011 Feature Pack 3.

To simplify the `baselinePath` parameter value, reference a single root folder - the customization tool will ignore unneeded files and folders. Otherwise, use a semicolon (`;`) in the `baselinePath` value to specify multiple root folders. Consider the following examples.

Example: Customizing Crystal Reports 2011 SP5

Assume you are customizing Crystal Reports 2011 Support Package 5 and you customized the previous programs: `2011 (Full install), 2011 SP4`. Assume the non-customized installation programs are located in the following directory structure:

```
C:\productUpdates\2011\
  \2011 Full\
  \SP4\
```

Set the `baselinePath` parameter to:

`baselinePath=C:\productUpdates\2011\`

Example: Customizing Crystal Reports 2011 SP5 Patch 2

Assume you are customizing Crystal Reports 2011 Support Package 5 Patch 2 and you customized the previous programs: `2011 (Full install), 2011 SP4, 2011 SP5, 2011 SP5 Patch 1`. Assume the non-customized installation programs are located in the following directory structure:

```
C:\productUpdates\2011\
  \2011 Full\
  \SP4\
  \SP5\
  \SP5 Patch 1\
```

Set the `baselinePath` parameter to:

```
baselinePath=C:\productUpdates\2011\
```

Related Topics

- Command line parameters

5.9 IDs and codes for Crystal Reports customization

The following section contains a list of all the IDs and codes you can use to customize the installation program:

- Feature IDs
- Shortcut deployment unit IDs
- String IDs
- Language codes
- Installation screen and property IDs

5.9.1 Feature IDs

Use feature IDs in the `removeFeature` element to remove features and their components from the installation program and the installed product.

For example, this ID will remove support for displaying relationships between data and geographic regions: `<removeFeature id="Mapping"/>`

You can remove features for the following components:

- Data access
- Enterprise system integration
- Export support

- Other

Table 5-10: Data access

Feature ID	Description
DataAccess	Data access
Access	Microsoft Access
ADO.NET	ADO.NET
BDE	IDAPI Database DLL
Btrieve	Pervasive Database Driver (Betrieve)
COMData	COM Data Provider
Comm_Rep	Command in repository
DB2	IBM DB2
dBase	dBase
Exchange	Microsoft Exchange
FieldDefinitions	Field Definitions
FileSystem	File System
HPNeoview	HP Neoview
Informix	Informix
JavaData	Java Data Provider
JDBC	JDBC (JNDI) Data Driver
MicrosoftOutlook	Microsoft Outlook
MyCube	OLAP Cube
MYSQL_DataAccess	MySQL
NCRTeradata	NCRTeradata
NETEZZA	NETEZZA
NTEventLog	NT Event Log
OLE_DB_Data	OLE DB Data
OptionalDataDirectODBC	DataDirect ODBC

Feature ID	Description
`Oracle`	Oracle
`Progress.OpenEdge`	Progress OpenEdge
`RDO`	ODBC RDO
`SFORCE`	Salesforce.com Driver
`SIEBEL`	Siebel
`Sybase`	Sybase
`SymantecACT`	ACT!
`Universe`	BusinessObjects Universe
`UWSC`	Universal Web Services Connector
`WebActivityLog`	Web Activity Log
`XML`	XML Driver

Table 5-11: Enterprise system integration

Feature ID	Description
`IntegrationOptions`	Integration options
`EBS`	Oracle E-Business Suite
`JDE`	JD Edwards EnterpriseOne
`PSFT`	Peoplesoft Enterprise
`SAP`	SAP Solutions
`SIEBEL`	Siebel

Table 5-12: Export support

Feature ID	Description
`Application`	Export to an application
`CharacterSeparated`	Character Separated format
`CrystalReports`	Crystal Reports format

Feature ID	Description
DiskFile	Export to a file
Excel	Microsoft Excel 97-2003 format
ExchangeFolders	Exchange Folder
HTML	HTML 3.2 and HTML 4.0 (DHTML) formats
LegacyXMLExport	Legacy XML format
LotusNotes	Lotus Notes document
LotusNotesMail	Lotus Domino
ODBC	Export to any of your installed ODBC formats
PDF	PDF format
Record	Record Style format
ReportDefinition	Report Definition format
RichTextFormat	Rich Text Format
Text	Text formats
WordforWindows	Microsoft Word 97-2003 format
XMLExport	XML document

Table 5-13: Other

Feature ID	Description
CrystalReportsRoot	Crystal Reports 2011
crw	Crystal Reports Designer
Mapping	Geographic mapping
MicrosoftMail	Microsoft Mail Destination
PGEditor	Custom charting
UploadWizard	Report Upload Wizard

Related Topics

- Removing features

5.9.2 Shortcut deployment unit IDs

Use the deployment unit IDs in the `shortcut` element to change the location and name of the program shortcuts in the Windows **Start** menu.

Table 5-14: Shortcut deployment unit IDs

Shortcut deployment unit ID	Shortcut target
`product.crystalreports.shortcut.crw-4.0-core`	Crystal Reports 2011
`product.crystalreports.shortcut.odbc-4.0-core`	ODBC Data Source Administrator
`product.crystalreports.shortcut.rptpubwiz-4.0-core`	Report Upload Wizard

Related Topics

- Customizing the Windows Start menu shortcuts

5.9.3 String IDs

You can change the value of all strings in the installation program. You can replace a string for all languages or for a specific language. Use the `replaceString` element, for example:

```
<replaceString id="product.cr_name" value="Custom Company Crystal Reports
lang="all"/>
```

Table 5-15: Commonly changed strings

String ID	Description
`product.cr_name`	Product long name
`product.cr_shortname`	Product short name
`product_cr_version`	Product version

String ID	Description
product_cr_majorversion	Product major version

Related Topics

- Customizing the product name and version number

5.9.4 Language codes

The SAP BusinessObjects customization tool uses these language codes to represent supported languages:

Language	Code
English	EN
Czech	CS
Danish	DA
Dutch	NL
Finnish	FI
French	FR
German	DE
Hungarian	HU
Italian	IT
Japanese	JA
Korean	KO
Norwegian Bokmal	NB
Polish	PL
Portuguese	PT
Romanian	RO
Russian	RU

Language	Code
Simplified Chinese	`zh_CN`
Slovak	`SK`
Spanish	`ES`
Swedish	`SV`
Thai	`TH`
Traditional Chinese	`zh_TW`
Turkish	`TR`

Related Topics

- Customizing the product name and version number
- Customizing the Windows Start menu shortcuts
- Customizing the Windows Add Remove Program utility
- Removing language packs
- Customizing the license agreement
- Customizing menu strings

5.9.5 Installation screen and property IDs

Use the installation screen IDs in the `removeDialog` element to remove screens from the installation program. For example, use this element to remove the "Select Features" screen:

```
<removeDialog id="SelectFeatures.dialog"/>
```

Use the property IDs and the property values to prepopulate user input. For example, use this element to set the default installation type to "custom":

```
<replaceProperty id="InstallType" defaultValue="custom"/>
```

Table 5-17: Screen IDs

Title of installation screen	Installation screen ID	Property IDs in installation screen	Property values
"Please choose a setup language"	`SelectUILanguage.dialog`	`SortedAvailableSetupLanguages`	Set of language codes representing supported setup languages
		`SetupUILanguage`	Single language code representing the setup language
"Install cannot proceed"	`SharedAlwaysFailure.dialog`	Not applicable	Not applicable
"Choose Install Type"	`ChooseInstallType2.dialog`	`InstallType`	• `default` (Typical) • `custom`
"Prerequisite check"	`CheckPreRequisites.dialog`	Not applicable	Not applicable
"Welcome to the installation wizard...."	`ShowWelcomeScreen.dialog`	Not applicable	Not applicable
"License Agreement"	`ShowLicenseAgreement.dialog`	Not applicable	Not applicable
"User Information"	`CREnterProductKey.dialog`	`RegisteredUser`	Your "Username"
		`RegisteredCompany`	Your "Company name"
		`ProductKey`	Your "Product keycode"
"Specify the Destination Folder"	`ChooseInstallDir.dialog`	`InstallDir`	Filepath of the installation folder
"Choose Language Packs"	`SelectLanguagePack.dialog`	`SelectedLanguagePacks`	Array of language codes
"Select Features"	`SelectFeatures.dialog`	Not applicable	Not applicable

Title of installation screen	Installation screen ID	Property IDs in installation screen	Property values
"SAP Crystal Reports 2011 has been successfully installed"	`ShowInstallComplete.dialog`	Not applicable	Not applicable
"SAP Crystal Reports 2011 has been successfully installed"	`ShowInstallComplete_PatchUpdate.dialog`	Not applicable	Not applicable
"Start Installation"	`ShowInstallSummary.dialog`	Not applicable	Not applicable
"Uninstall Confirmation"	`VerifyToRemove.dialog`	Not applicable	Not applicable
"Web Update Service Option"	`ShowPrivacyStatement.dialog`	`DisableWebUpdateService`	• `0` (Enable Web Update Service) • `1` (Disable Web Update Service)
"SAP Crystal Reports 2011 has been successfully uninstalled"	`ShowUninstallComplete.dialog`	Not applicable	Not applicable

More Information

Information Resource	Location
SAP product information	http://www.sap.com
SAP Help Portal	http://help.sap.com/analytics Access the most up-to-date English documentation covering all SAP Analytics products at the SAP Help Portal: • http://help.sap.com/bobi (BusinessObjects Business Intelligence) • http://help.sap.com/boepm (Enterprise Performance Management) • http://help.sap.com/boeim (Enterprise Information Management) Certain guides linked to from the SAP Help Portal are stored on the SAP Service Marketplace. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative. To find a comprehensive list of product documentation in all supported languages, visit:http://help.sap.com/boall.
SAP Support Portal	http://service.sap.com/bosap-support The SAP Support Portal contains information about Customer Support programs and services. It also has links to a wide range of technical information and downloads. Customers with a maintenance agreement have an authorized user ID to access this site. To obtain an ID, contact your customer support representative.
Developer resources	http://www.sdn.sap.com/irj/sdn/bi-sdk-dev https://www.sdn.sap.com/irj/sdn/businessobjects-sdklibrary (BI SDK Developer Library)
Articles and eLearning on the SAP Community Network	http://scn.sap.com/docs/DOC-19311 These articles were formerly known as technical papers.

Information Resource	Location
Notes	https://service.sap.com/notes These notes were formerly known as Knowledge Base articles.
Forums on the SAP Community Network	https://www.sdn.sap.com/irj/scn/forums
Training	http://www.sap.com/services/education From traditional classroom learning to targeted e-learning seminars, we can offer a training package to suit your learning needs and preferred learning style.
Consulting	http://www.sap.com/services/bysubject/businessobjectsconsulting Consultants can accompany you from the initial analysis stage to the delivery of your deployment project. Expertise is available in topics such as relational and multidimensional databases, connectivity, database design tools, and customized embedding technology.