

Document Version: 1.0 – 2016-07-25

SAP Configure, Price, and Quote for Solution Sales Configuration

For SAP Hybris Commerce Suite 6.0



Typographic Conventions

Type Style	Description
<i>Example</i>	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.
Example	Emphasized words or expressions.
EXAMPLE	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE	Keys on the keyboard, for example, F2 or ENTER .

Document History

Version	Date	Change
1.0	2016-07-25	Initial version

Contents

1	SAP Configure, Price and Quote for Solution Sales Configuration	6
2	Preparation	7
2.1	System Requirements	7
2.2	Required SAP Notes.....	7
3	Installation on SAP Hybris	8
3.1	Downloading the Software.....	9
3.2	Configuration Runtime	9
3.3	Data Loader	10
3.4	UI Extensions	10
3.5	Installing Product Configuration Addon in the Storefront	10
3.6	Preparing and Starting the SAP Hybris Server	12
3.7	Launch and Enable SAP Hybris	14

4	Setting Up the Page Template for Solution Configuration in WCMS	16
5	Status Display	19
6	Display of Components in Multilevel Configuration.....	20
7	Price Information on Configuration Page	22
8	Settings in the SAP ERP Back End.....	23
9	Price Information for Characteristic Values.....	24
10	Initial Load of Configuration Master Data via Data Loader	25
11	User Authorization for Data Loader User	26
12	Configuring the SAP ECC for Data-Loader Delta Download.....	27
13	Stock Level for Configurable Products	29
14	Extensions	30
15	Supported Characteristic Types	32
16	Pricing Attributes	33
17	Configuration Context Attributes	34
18	Deploying User Exits for Use with SAP Hybris	35
19	Troubleshooting	36
20	Appendix.....	37
20.1	Installing ysapordermgmtb2baddon for SAP Synchronous Order Management	37
20.2	Database Connection Parameters for local.properties.....	38

1 SAP Configure, Price and Quote for Solution Sales Configuration

About This Document

This guide describes how to install SAP Configure, Price, and Quote for solution sales configuration. SAP CPQ for solution sales configuration has the following software components:

- SAP Hybris Commerce Suite release 6.0 or higher
- SAP CPQ Configuration Runtime (SSC)

You can download the software components from SAP Software Download Corner at <http://support.sap.com/swdc>.

2 Preparation

2.1 System Requirements

SAP Configure, Price and Quote for solution sales configuration has the following system requirements:

SAP ECC System Requirements

You must install the ABAP add-on for SAP CPQ for solution sales configuration. For more information, see SAP Note [2090791](#) (Release strategy for the ABAP add-on SLCE 607 onwards).

SAP Hybris System Requirements

Software	Version
Operating system	Windows (Win32, x86, and x64), Linux
Java Runtime Environment	Java 1.8
Database	See the product availability matrix at http://support.sap.com/pam .

More Information

For more information about the general system requirements for SAP Hybris, see SAP Hybris help at <http://help.hybris.com> and choose *Installing & Migrating Hybris → System Requirements*.

For more information about the databases that are supported by the SAP Hybris Commerce Suite, see SAP Hybris help at <http://help.hybris.com> and choose *Installing & Migrating Hybris → System Requirements → Third-Party Compatibility*.

2.2 Required SAP Notes

You must implement SAP Note [2006212](#) (Additional enhancement point in function group V45CU) before the add-on is installed in SAP ECC.

3 Installation on SAP Hybris

SAP Configure, Price, and Quote for solution sales configuration contains the required runtime components and a UI extension to enable you to configure solutions in the SAP Hybris Commerce Suite.

Prerequisites

- You have installed and configured SAP Hybris Commerce 6.0. For more information about the supported releases, see SAP Note 2227752. For information about installing and configuring SAP Hybris, see *Installing & Migrating Hybris* at <http://help.hybris.com>.
- You have configured `b2b accelerator` in SAP Hybris Commerce.

You can configure `b2b accelerator` using the recipes or you can implement it manually. For information about using the recipes, see <http://help.hybris.com> → *Accelerators* → *Installation and Migration* → *B2C/B2B Accelerator* → *B2C/B2B Accelerator Quick Installation Guide*.

Note

In SAP Hybris 6.0, by default the `b2b` recipes create a custom storefront by copying the existing `yacceleratorstorefront` by the name `yb2bacceleratorstorefront` in the `hybris/bin/custom` folder. As a result, if you use recipes, your storefront name will be `yb2bacceleratorstorefront`.

For information about configuring the `b2b accelerator` manually, see <http://help.hybris.com> → *Accelerators* → *Installation and Migration* → *B2C/B2B Accelerator* → *B2C/B2B Accelerator Implementation Guide*.

For information about creating a custom `b2b accelerator` using `modulegen`, see <http://help.hybris.com> → *Accelerators* → *B2B Accelerator* → *Creating a Customized B2B Accelerator with Modulegen*.

- You have configured SAP Synchronous Order Management in SAP Hybris Commerce.
For more information about configuring SAP Synchronous Order Management, see <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Back-End Integration: SAP ERP and SAP CRM* → *Getting Started with SAP ERP Integration*.

Note

Unlike SAP Configure, Price, and Quote for product configuration, SAP Configure, Price, and Quote for solution sales configuration doesn't support asynchronous order management out of the box.

You can configure SAP Synchronous Order Management automatically if you have used `som` recipes such as `sap_som_b2b`, or you can configure it manually using the `ysapordermgmtb2baddon` add-on.

For information about installing this add-on, see the appendix.

- You have created a database that is separate from the SAP Hybris database for CPQ. For a list of databases supported by SAP Configure, Price, and Quote for solution sales configuration, see the product availability matrix at <http://support.sap.com/pam>. The SAP CPQ database can reside on a different database instance and have a different database type than that of the SAP Hybris database. If you decide to use different database types, ensure that the JDBC driver of the CPQ database is placed in `<hybris installation directory>\bin\platform\lib\dbdriver`.

The SAP CPQ database must be case-sensitive (use a case-sensitive collation such as `UTF8_bin`).

- You have maintained `<hybris install dir>:\hybris\config\local.properties` and added the relevant database connection parameters. For a list of connection parameters and an explanation of what each one means, see the appendix.

3.1 Downloading the Software

For installation on SAP Hybris, you can always use the latest patch of your Support Package. Patches contain the complete software, which means you do not need to install the initial delivery ("Support Package 0") and upgrade to your Support Package. Similarly, you don't need to install patch level 0 and upgrade to the latest patch level.

Download the installation .zip files from SAP Software Download Center for the following software components. The files are available at <http://support.sap.com/swdc> → *Support Packages and Patches*. In the alphabetical list of products choose "C" and then *SAP CPQ SOL. SALES CONFIG* → *SAP CPQ SOL. SALES CONFIG 2.0 (SUPPORT PACKAGES AND PATCHES)* → *Entry by Component* → *CPQ SOL SALES CONFIG HYBRIS 6.0*.

Download the following:

- FBS_SOLCONF_DATALOADER
- FBS_SOLCONF_ENGINE
- FBSSOLHYBUIEXT

More Information

SAP Note [2132595](#)

3.2 Configuration Runtime

1. Place the downloaded file `FBSSLCEngine<Support Package>_<Patch Level>.ZIP` into the following directory of your SAP Hybris installation:
`${HYBRIS_HOME}/hybris/bin/ext-integration/sap/productconfig/
sapproductconfigruntimeessc/lib`
2. Extract the content of the `FBSSLCEngine.zip` file to the directory and delete the .zip file. The extracted file contains all the necessary Java archives in the version required.

Result

The directory should now contain the following files:

- `com.sap.custdev.projects.fbs.slc.engine-2.<support package>.<patch level>.jar`
- `com.sap.custdev.projects.fbs.slc.database-2.<support package>.<patch level>.jar`
- `com.sap.custdev.projects.fbs.slc.common-2.<support package>.<patch level>.jar`

The directory must not contain any other versions of these files.

3.3 Data Loader

Procedure

1. Extract the file FBSSOLDATBS<support package>_<patch level>.ZIP to the following directory of your SAP Hybris installation:

```
${HYBRIS_HOME}/hybris/bin/ext-integration/sap/productconfig/  
saproductconfigmodel/lib
```

2. Delete the .zip file. The extracted files contain the necessary Java archive in the version required.
3. Extract the file FBSSLCENGINE<Support Package>_<Patch Level>.ZIP to the following directory of your SAP Hybris installation:

```
${HYBRIS_HOME}/hybris/bin/ext-integration/sap/productconfig/  
saproductconfigmodel/lib
```

4. Delete the .zip file. The extracted file contains all the necessary Java archives in the version required.

Result

The sapproductconfigmodel/lib directory now contains the following files:

- com.sap.custdev.projects.fbs.slc.dataloader.lib-2.<support package>.<patch level>.jar
- com.sap.custdev.projects.fbs.slc.engine-2.<support package>.<patch level>.jar
- com.sap.custdev.projects.fbs.slc.database-2.<support package>.<patch level>.jar
- com.sap.custdev.projects.fbs.slc.common-2.<support package>.<patch level>.jar

The sapproductconfigruntimessc/lib directory now contains the following files:

- com.sap.custdev.projects.fbs.slc.engine-2.<support package>.<patch level>.jar
- com.sap.custdev.projects.fbs.slc.database-2.<support package>.<patch level>.jar
- com.sap.custdev.projects.fbs.slc.common-2.<support package>.<patch level>.jar

The directories must not contain any other versions of these files.

3.4 UI Extensions

1. Extract the file FBSSOLHYBUIEXT<Support Package>_<Patch Level>.ZIP from the content of the downloaded installation .zip file.
2. Place it in the following directory on your SAP Hybris installation: {HYBRIS_INSTALL_DIR}\bin\custom. You may have to create the "custom" folder first.

3.5 Installing Product Configuration Addon in the Storefront

SAP CPQ for solution sales configuration extends SAP CPQ for production configuration. Consequently, production configuration needs to be installed here.

1. **Optional but recommended:** Copy the ysaproductconfigb2addon AddOn and tailor it to your needs.

1. To enable extgen to find the ysaproductconfigb2baddon extension template for product configuration, you must add the following path to the localextension.xml file:

```
<path dir= '${HYBRIS_BIN_DIR}/ext-integration/sap' autoload='false' />
```

2. In hybris platform directory run the following ant command:

```
ant extgen -Dinput.template=ysaproductconfigb2baddon -Dinput.name=<your addon name> -Dinput.package=de.hybris.platform.sap.cproductconfig.frontend
```

where:

- o input.name = Name of the extension
- o input.package = Java package of the extension
- o input.template = Extension template to use

For more information about copying AddOns, see <http://help.hybris.com> → Platform → Extension Concept in SAP Hybris Commerce → Creating a New Extension.

For more information about creating your own extension based on the ysaproductconfigb2baddon extension template, see <http://help.hybris.com> → Integrations and Data Management → SAP Integrations → SAP Configure, Price, and Quote for Product Configuration → Extensions and AddOns for SAP CPQ for Product Configuration → ysaproductconfigb2baddon AddOn.

2. Add the following extensions to localextensions.xml:

```
<!-- product config extension -->
<your copy of ysaproductconfigb2baddon or if you want to activate the B2B
accelerator directly without any customization, ysaproductconfigb2baddon>
<extension name=' <your storefront extension>' />
<extension name='sapproductconfigb2b' />
<extension name='sapproductconfigbackoffice' />
<extension name='sapproductconfigruntimeessc' />
```

Note

Depending on whether you used a recipe to install the storefront, you may already find the extensions of SAP CPQ for production configuration in your localextensions.xml file. For example, this is the case if you use recipe sap_b2b_som.

3. Ensure that the Java package in the following files points to your storefront.

1. Change file ListAddToCartActionController located in folder
\${HYBRIS_BIN_DIR}/custom/<your product config
addon>/acceleratoraddon/web/src/<java package of your storefront>/controllers.



Example

Change the line

```
import
de.hybris.platform.yacceleratorstorefront.controllers.cms.DefaultCMSComponentCon
troller; to import <package of your
storefront>.controllers.cms.DefaultCMSComponentController;
```

2. Replace the placeholder in the example with the actual package of your own storefront to which you will add the extension.

4. If you have created your own storefront extension in the earlier step, ensure that the `project.properties.template` file of your product config addon or `ysapproductconfigb2baddon` (if you aren't customizing it) has the following configuration for your storefront:

```
<your storefront extension>.additionalWebSpringConfigs.<your product config addon>=classpath:/.<your product config addon>/web/spring/<your product config addon>-web-spring.xml
```

For more information, see <http://help.hybris.com> → *Accelerators* → *Core Accelerator* → *Hybris AddOn Concept* → *Installing an AddOn for a Specific Storefront*.
5. To register your copy of the `ysapproductconfigb2baddon` AddOn, enter the `addoninstall` command in the command line dialog box.



```
ant addoninstall -Daddonnames="<your copy of ysapproductconfigb2baddon>" -  
DaddonStorefront.yacceleratorstorefront="<your storefront extension>"
```

6. Execute `ant clean all initialize` to generate the initial data for the new product configuration front-end extension.

3.6 Preparing and Starting the SAP Hybris Server

All additional extensions used in SAP Hybris, as well as information about the path and data structure of the SAP Hybris Commerce Suite, is written in the `localextensions.xml` file, which is located under `${HYBRIS_CONFIG_DIR}` (for example, `D:\hybris\config`). The original file contains only the basic settings and entries. These do not include entries related to SAP extensions or data-structure entries.

To use the extensions provided by SAP CPQ for solution sales configuration, you have to add the solution configuration extension manually to the file `localextensions.xml`.

Procedure

1. Add the following extension of solution configuration to `localextensions.xml`:

```
<extension name="ysapcustdevsolutionconfigb2bfrontend"/>
```
2. If you have customized your product config addon, ensure that the `extensioninfo.xml` file of the `ysapcustdevsolutionconfigb2bfrontend` extension points to your product config addon instead of the default `ysapproductconfigb2baddon`.

```
<requires-extension name="<your copy of ysapproductconfigb2baddon or if you want to  
activate B2B accelerator directly without customization  
ysapproductconfigb2baddon"">/>
```
3. If you have customized product config addon, ensure that the Java package in the following files points to your product configuration package.
 1. Change file `AddProductToSolutionController` located in folder `${HYBRIS_BIN_DIR}/custom/ysapcustdevsolutionconfigb2bfrontend` extension.
 - o Change the following lines:

```
import de.hybris.platform.sap.productconfig.frontend.UiStatus;  
  
import  
de.hybris.platform.sap.productconfig.frontend.constants.Sapproductconfigb2bad  
donConstants;
```

```
import
de.hybris.platform.sap.productconfig.frontend.controllers.AbstractProductConfigController;
```

to the following:

```
import <java package of your custom product config add on>.UiStatus;
import <java package of your custom product config add on>.constants.<your
constant file at this location>;
import <java package of your custom product config add
on>.controllers.AbstractProductConfigController;
```

1 Note

Replace the placeholder in the example with the actual package of your own product config add on which your `ysapcustdevsolutionconfigb2bfrontend` is based.

2. Change file `FindProductForSolutionController.java` located in folder `${HYBRIS_BIN_DIR}/custom/ysapcustdevsolutionconfigb2bfrontend` extension.

- o Change the following lines:

```
import de.hybris.platform.sap.productconfig.frontend.UiStatus;
import
de.hybris.platform.sap.productconfig.frontend.controllers.AbstractProductConfigController;
import
de.hybris.platform.yb2bacceleratorstorefront.controllers.pages.SearchPageController;
```

to the following:

```
import <java package of your custom product config add on>.UiStatus;
import <java package of your custom product config add
on>.controllers.AbstractProductConfigController;
import <<java package of your custom
storefront>.controllers.pages.SearchPageController;
```

3. Change file `ListProductOfSolutionController.java` located in folder `${HYBRIS_BIN_DIR}/custom/ysapcustdevsolutionconfigb2bfrontend` extension.

- o Change the following lines:

```
import de.hybris.platform.sap.productconfig.frontend.UiStatus;
import
de.hybris.platform.sap.productconfig.frontend.controllers.AbstractProductConfigController;
```

to the following:

```
import <java package of your custom product config add on>.UiStatus;
import <java package of your custom product config add
on>.controllers.AbstractProductConfigController;
```

4. Change file `SolutionConfigControllerInterceptor.java` located in folder `${HYBRIS_BIN_DIR}/custom/ysapcustdevsolutionconfigb2bfrontend` extension.

- o Change the following line:

```
import
de.hybris.platform.sap.productconfig.frontend.constants.sapproductconfigb2bad
donConstants;
```

to the following:

```
import <java package of your custom product config add
on>.frontend.constants.<your sapproductconfigb2baddonConstants filename>;
```

- o Replace occurrences of Sapproductconfigb2baddonConstants with your sapproductconfigb2baddonConstants file name.

5. Change file menuNode.tag located in

```
${HYBRIS_BIN_DIR}/\custom\ysapcustdevsolutionconfigb2bfrontend\acceleratoraddon\
web\webroot\WEB-INF\tags\responsive\configuration
```

- o Change the following lines:

```
<%@ taglib prefix="cssConf" uri="/WEB-
INF/tags/addons/ysapproductconfigb2baddon/responsive/configuration/sapproduct
config.tld"%>
```

```
<jsp:useBean id="cons"
class="de.hybris.platform.sap.productconfig.frontend.util.impl.ConstantHandle
r" scope="session" />
```

to the following:

```
<%@ taglib prefix="cssConf" uri="/WEB-INF/tags/addons/<your product config
add on>/responsive/configuration/sapproductconfig.tld"%>
```

```
<jsp:useBean id="cons" class="<java package of your product config
addon>.frontend.util.impl.ConstantHandler" scope="session" />
```

4. If you created your own storefront extension in the earlier step, ensure that the `project.properties` file of `ysapcustdevsolutionconfigb2bfrontend` has the following configuration for your storefront:

```
<your storefront
extension>.additionalWebSpringConfigs.ysapcustdevsolutionconfigb2bfrontend=classpat
h:/
ysapcustdevsolutionconfigb2bfrontend/web/spring/ysapcustdevsolutionconfigb2bfronten
d-web-spring.xml
```

For more information, see <http://help.hybris.com> → *Accelerators* → *Core Accelerator* → *Hybris AddOn Concept* → *Installing an AddOn for a Specific Storefront*.

3.7 Launch and Enable SAP Hybris

Procedure

1. Open a command prompt (shell in Unix family systems) and navigate to folder `${HYBRIS_BIN_DIR}\platform`.
2. Run `setantenv.bat`. Do not close the command window/shell.
3. Execute command `ant clean all` to build the SAP Hybris solution.
4. After the ANT build has finished ("BUILD SUCCESSFUL"), start the SAP Hybris server by executing file `hybrisserver.bat` (also located in the platform directory).
5. Perform an "update" in the SAP Hybris administration console for the following extensions:

-
- o ysapcustdevsolutionconfigb2bfrontend
 - o sapcustdevordermgmtbol
 - o sapcustdevordermgmtfacades
 - o sapcustdevsolutionconfigservices
 - o sapcustdevsolutionconfigfacades
6. Complete the settings for the SAP base store configuration and set up a user account as explained at <http://help.hybris.com>.

i Note

If you are integrating SAP, generate the Customizing indexes based on your SAP ECC settings and run them in SAP Hybris. This ensures that all required data is synchronized between SAP ERP and SAP Hybris.

For more information, see SAP Note 1983231 and Synchronizing Data between SAP ERP and hybris.

4 Setting Up the Page Template for Solution Configuration in WCMS

If you do not use the powertools accelerator, you have to set up the page template for product configuration in the WCMS Cockpit. The cockpit provides a page template called "Product Configuration Template". This template has the page type "Product Configuration Page".

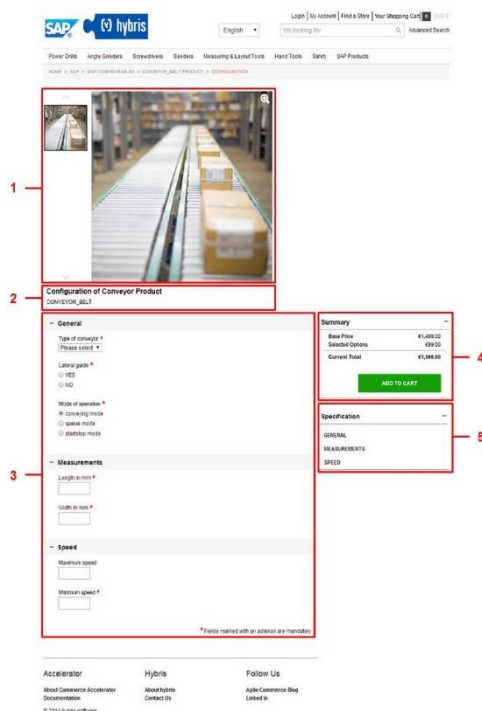


Figure 1: Product Configuration Page

The following components are available on this page:

No.	Component	Explanation
1	Product Configuration - Image	This component displays one large and several small product images as maintained in the Backoffice and the product cockpit.
2	Product Configuration - Title	This component displays the product name as the title for the interactive configuration area and displays a short description of the product. You can hide the short description, which is taken from the Summary field in the Backoffice and the product cockpit.

No.	Component	Explanation
3	Product Configuration – Form	<p>This component displays the characteristics and corresponding characteristic values so that customers in the online store can configure the product according to their needs. Note that your settings for the Product Configuration - Form component should depend on your settings for the Product Configuration - Tab Navigation component:</p> <ul style="list-style-type: none"> • If you have set the tab navigation to visible, you should display the levels in the form from level 1 onwards. • If you have set the tab navigation to Not visible, you should display the levels in the form from level 0 onwards. <p>Note that the default delivery of the product configuration page template contains two versions of Product Configuration Form – one starting at level 1, the other starting at level 0. These are meant to be combined with the Product Configuration - Tab Navigation either being visible or not.</p> <p>If you configure the restrictions using product categories and products in the right way, you can take advantage of the tab navigation for complex multilevel products, whereas simpler configurable products are displayed without tab navigation.</p> <p>For multilevel products, you can hide the short description for configurable components, which is taken from the Summary field in the Backoffice and the product cockpit. Note that this option is relevant only if a product has been created for the configurable component in the</p>
4	Product Configuration - Price Summary	<p>This component displays the price summary containing the base price of the configurable product, the sum of the selected price-relevant options and the total price (net value without shipping costs excluding tax). This means that the customer always has an overview of how the selection of options affects the total price of the product.</p> <p>Note that in multilevel configuration, the system displays the price of the highest level (main or root instance). Prices of the individual levels (subinstances) are not taken into consideration.</p> <p>Right below the price summary, this component displays the Add to Cart button.</p> <p>You have the option to hide the base price and the price for the selected options.</p> <p>Note that if you use this component, you should not use the Product Configuration - Add Configuration to Cart component in addition.</p>
5	Product Configuration - Specification	<p>For multilevel products, this component displays an overview of the components and their error or conflict status. This facilitates quick navigation to characteristic values causing errors or conflicts.</p>
Not shown in screenshot	Product Configuration - Add Configuration to Cart	<p>This component displays the button for adding the configured product to the cart.</p> <p>You can show this button only for configuration forms that are more than twice as long as the browser window.</p>

No.	Component	Explanation
Not shown in screenshot	Product Configuration - Tab Navigation	<p>This component is relevant if you use multilevel configuration with a large number of characteristics. It displays the configuration components (groups/ instances) on hierarchy level 0 as tabs, which makes navigation for products with multilevel configuration easier. Each tab contains the components from level 1 onwards. Note that your settings for the Product Configuration - Form component should depend on your settings for the Product Configuration - Tab Navigation component:</p> <ul style="list-style-type: none"> • If you have set the tab navigation to visible, you should display the levels in the form from level 1 onwards. • If you have set the tab navigation to Not visible, you should display

In addition, the component [Product Configuration - Configure Button](#) is available with the [Product Page](#) page type. For configurable products, this component replaces the [Add to Cart](#) button with the [Configure](#) button, which takes customers to the product configuration page.

As of version 5.7, this is achieved via the `CMSConfigurableProductRestriction` restriction.

More Information

For more information about how to adjust the page to your needs, see <http://help.hybris.com> → [Channel](#) → [WCMS Module](#).

For more information about the available page layouts, see <http://help.hybris.com> → [Integrations and Data Management](#) → [SAP Integrations](#) → [SAP Configure, Price, and Quote for Product Configuration](#) → [Extensions and AddOns for SAP CPQ for Product Configuration](#) → [ysaproductconfigb2baddon](#) AddOn.

5 Status Display

You can set up the status display at the level of characteristics and characteristic groups so that you can immediately see whether your characteristic value assignment is allowed (for example, it has a green icon). You can also see if your specified values are not allowed or if they conflict with other selected characteristic values (for example, they have a red icon). The following assignment statuses can be displayed for characteristics:

Value Assignment	Meaning	Style Class Suffix
Allowed value assignment	The customer has processed the characteristic and assigned an allowed value to it.	success
Value assignment with conflict	The specified value conflicts with other selected characteristic	warning
Value assignment not allowed	The specified value is not allowed.	error

The following assignment statuses can be displayed for characteristic groups:

Value Assignment	Meaning	Style Class Suffix
Value assignment with conflict	At least one characteristic within the group conflicts with another selected characteristic value.	Warning
Value assignment not allowed	At least one characteristic within the group has a value assignment that is not allowed.	Error

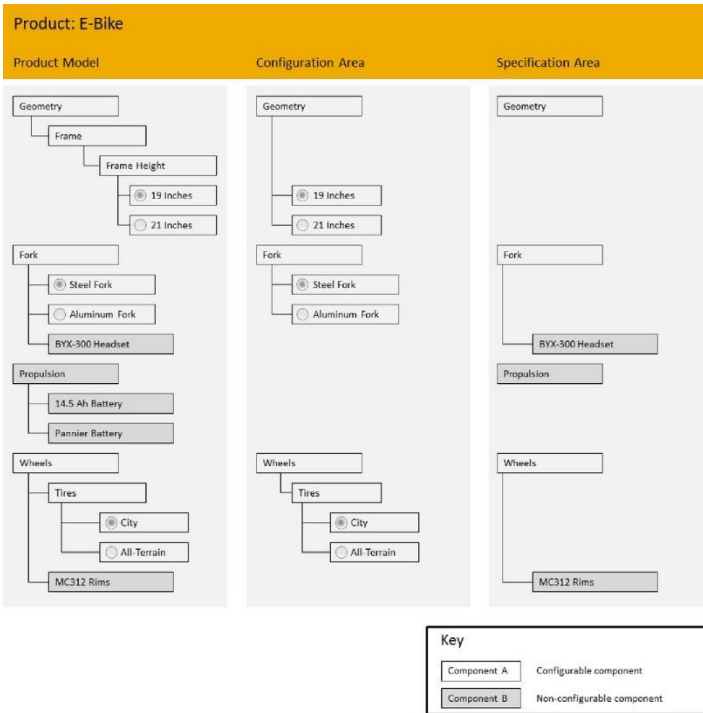
6 Display of Components in Multilevel Configuration

The following rules apply to the display of components in multilevel configuration:

Type of Component	Display in the Interactive Configuration Area	Display in the Specification Area
Configurable components	The component is displayed if there are other configurable components or characteristic groups on the same level. If there are no other configurable components or characteristic groups on the same level, the characteristics of the components are displayed, but the components themselves are not.	The component is displayed if there are other configurable components or characteristic groups on the same level. No characteristics are displayed in the specification area.
Non-configurable components	No display	The component is displayed if there are other configurable components or characteristic groups on the same level. In the case of subhierarchies that consist completely of non-configurable components, only the top node is displayed.

Example

The following example illustrates which components are displayed where:



7 Price Information on Configuration Page

The product price (total price) on the configuration page is the net value of the product without shipping costs excluding tax.

Note that in multilevel configuration, the system displays the price of the highest level (main or root instance). Prices of the individual levels (subinstances) are not taken into consideration.

To show the product price on the configuration page, you must select the option *Enable Pricing* in the Backoffice (see the Price Summary section under <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Configure, Price, and Quote for Product Configuration* → *Configuring SAP CPQ for Product Configuration* → *Making Settings in Backoffice for Product Configuration*), and you must include the WCMS component *Product Configuration - Price Summary* in your page template (see the *WCMS Components for Product Configuration* section under <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Configure, Price, and Quote for Product Configuration* → *Configuring SAP CPQ for Product Configuration*).

If you want to display the price summary on your product configuration page, you need to activate the *Enable Pricing* option and specify the relevant pricing procedure and purposes of condition functions in the Backoffice (versions 5.3 to 5.5.1). You can find these settings under *SAP Integration* → *SAP Base Store Configurations*. Select your store configuration and specify the following data on the *Product Configuration* tab under *SAP Product Configuration Pricing Procedure*:

Field	Description
Enable Pricing	Activate this option if you want to display the price summary on your product configuration page.
Pricing Procedure	Enter the pricing procedure that SAP CPQ Configuration Runtime (SSC) uses to calculate the product prices in the online store.
Base Price	Enter the purpose assigned to the condition function relevant for determining the base price of configurable products (see <i>Configuring SAP CPQ for Product Configuration – Updated for 5.7</i> , section <i>Settings in the SAP ERP Back End</i>). This is the price of the product if no price-relevant options are selected.
Selected Options	Enter the purpose assigned to the condition function relevant for determining the total of the price-relevant options selected (see <i>Configuring SAP CPQ for Product Configuration – Updated for 5.7</i> , section <i>Settings in the SAP ERP Back End</i>).

8 Settings in the SAP ERP Back End

If you want to display the price summary on your product configuration page, you need to assign condition functions to the condition types you use to determine the following price components of configurable products:

- Base price (net value)
- Total of the price-relevant options selected

Note that business function `ERP_IPC_INT_PRC` must be activated (transaction `SFW5`) before you can assign condition functions to condition types.

Next, you must assign a purpose to each of these condition functions. You do so in view `PRCV_CONDPURP` (transaction `SM30`) in your SAP ERP system.

Later, you enter the purposes in the Backoffice (see <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Configure, Price, and Quote for Product Configuration* → *Configuring SAP CPQ for Product Configuration* → *Making Settings in Backoffice for Product Configuration*).

Procedure

1. In view `PRCV_CONDPURP` (transaction `SM30`), maintain condition functions for the following price component products:
 - Base price (net value)
 - Total of the price-relevant options selected
2. In Customizing for SAP ERP under *Sales and Distribution* → *Basic Functions* → *Pricing* → *Define Condition Types*, assign these condition functions to the condition types you use for the base price and the selected options.

Use these condition types in the pricing procedure that you use for configurable products.

3. In the Backoffice, enter the pricing procedure and the purposes of the condition functions for the base price and the selected options.

You can find these settings in the Backoffice under *SAP Integration* → *SAP Base Store Configurations*.

4. Select your store configuration and enter the relevant data.

Note

These back-end settings are available in SAP Hybris only after you have run the data loader. For more information, see <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Configure, Price, and Quote for Product Configuration* → *Configuring SAP CPQ for Product Configuration* → *Making Settings in Backoffice for Product Configuration*.

9 Price Information for Characteristic Values

The price information shown is based on the way prices are defined in configuration maintenance (transaction CU50) of the SAP ERP back end. Note that you must define surcharges and discounts as fixed amounts, not as percentages.

The following features are available:

- With single value characteristics, the price information on the interactive configuration page next to each option is dynamic. It shows the price difference between each option and the one that is selected (delta price), and changes when a different option is selected.
- With multiple-value characteristics, the price of each individual option is displayed right next to the option. When the option is selected, the price of the option is added to the total price of the product, and the text *included in price* is shown next to the option.

10 Initial Load of Configuration Master Data via Data Loader

When you set up product configuration, you need to download your configuration master data from the SAP ERP Central Component (ECC) to SAP Hybris Commerce. This includes the product model data in your knowledge base as well as Customizing data that is relevant for pricing.

To start the initial load, go to the Backoffice under *SAP Integration → SAP Base Store Configurations*. Select your SAP base store configuration and specify the following data on the *Product Configuration* tab under *Data Loader Parameters*:

Setting	Description
RFC Destination	RFC destination created in Backoffice SAP Integration under SAP Integration → RFC Destination. For more information, see <i>Setting Up System Connections for SAP Integrations</i> at http://help.hybris.com .
RFC Connection	RFC destination created during the TCP/IP connection setup in the SAP ERP system (transaction SM59). For more information, see <i>Setting Up System Connections for SAP Integrations</i> at http://help.hybris.com . Note that if you have multiple data loader instances running (for example, several SAP Hybris systems), you need to define an RFC connection for each of those instances.
Knowledge	You only need to enter data in the filter fields if you want to filter the data that is loaded by the data loader. If no filter files are uploaded, all of the knowledge bases, conditions, and materials are downloaded from the back-end system. For more information on how to use the filter fields, see SAP Note 2006212 .
Conditions	
Material Filter	

When you choose *Start Initial Load*, the system first creates the required tables in the database and then starts to load the data. If the required tables already exist in the database, the system doesn't replace them.

11 User Authorization for Data Loader User

The data loader user that you enter in the RFC destination must have the following authorizations:

S_RFC with RFC_TYPE = 'FUGR' and RFC_NAME = 'SYST, RFC1, CRM0, CFIL, SDTX, ORFC'.

12 Configuring the SAP ECC for Data-Loader Delta Download

Note

You only need to complete this activity if you use the delta replication function to replicate master data changes from SAP ECC to your local database. If you only use the initial download, you can skip this activity.

Add the following data to enable the delta download. Use the transactions mentioned below or `SM30` to create or modify the table entries.

Procedure

1. Check whether the replication of configuration knowledge bases from the old data model (SCE tables) to the new data model within SAP ECC has already been activated. To do so, check the content of table `COMM_CFGDBVERS`. If the table contains data, replication is already activated. For more information, see SAP Note [1044682](#).

2. Call transaction `BF11` and modify table `TBE11` (*BTE Application Indicator*) as follows:

Application Indicator	Active	Text
BC-MID	X	Middleware

3. Call transaction `BF31` and modify table `TBE31` (*Publish and Subscribe BTE: SAP Enhancement*) as follows:

Event	Application Indicator	Function Module
00001250	BC-MID	CRS_MATERIAL_COLLECT_DATA
00503301	BC-MID	CRS_PRICES_COLLECT_DATA
00550002	BC-MID	COM_SCE_COLLECT_DATA_CHANGE or CRS_SCE_COLLECT_DATA_CHANGE (see SAP Note 1044682)

4. Add (or modify) table `CRMCONSUM` as follows:

User (CONSUMER)	Active (AKTIV)	30 Characters (TEXT)
IPC	X	Data Loader

To connect multiple data loader instances to the same ECC, add another entry with a new `CONSUMER` name in table `CRMCONSUM`.

Note

The name of the `CONSUMER` must contain "IPC" (for example, IPC2).

5. Add (or modify) table `CRMRFPCPAR` as follows.

User (CONSUMER)	Object Name (OBJNAME)	Destination (RFCDEST)	Load Type (DOWNLOAD)	Queue Name (RFC_QUEUE)	Queue Name (RFC_IN_QUEUE)
IPC	SCE	<dest>	*	CRM_DELT_SC	CRM_DELT_SC
IPC	MATERIAL	<dest>	*	CRM_DELT_MATERI AL	CRM_DELT_MATERI AL
IPC	CONDITIONS	<dest>	*	CRM_DELT_CO NDITIONS	CRM_DELT_CO NDITIONS

Use the CONSUMER name maintained in table CRMCONSUM table and replace <dest> with the RFC destination that you created.

The [BAPINAME](#) field must remain empty and none of the other options must be selected. In the [Send XML](#) field, select [No XML](#).

Note

The queue names given in columns RFC_QUEUE and RFC_IN_QUEUE must be set exactly to the values shown above – CRM_DELT_SCE, CRM_DELT_MATERIAL, and CRM_DELT_CONDITIONS. The data loader uses these fixed queue names to stop and start them.

In case filter files are changed, initial downloads should be done again.

13 Stock Level for Configurable Products

If you use the B2B Commerce Accelerator directly and without modifications, we recommend that you set the stock level for configurable products generally to *in stock* so that they do not appear to be out of stock in the online store. You can find the relevant setting in the Backoffice under *Base Commerce → Stock Level*.

Enter the relevant configurable product and for the *In Stock Status* field, choose the value *forceInStock*.

14 Extensions

SAP CPQ for product configuration comes with the following extensions:

- `ysapcustdevsolutionconfigb2bfrontend` extension
- `ysapproductconfigb2baddon` extension
- `sapcustdevsolutionconfigfacades` extension
- `sapproductconfigfacades` extension
- `sapcustdevsolutionconfigservices` extension
- `sapcustdevordermgmtbol` extension
- `sapcustdevordermgmtfacades` extension
- `sapproductconfigservices` extension
- `sapproductconfighmc` extension (versions 5.3 to 5.5.1)
- `sapproductconfigbackoffice` extension (as of version 5.6)
- `sapproductconfigruntimeessc` extension
- `sapproductconfigb2b` extension
- `sapcustomerb2b` extension
- `sapproductconfigruntimeinterface` extension
- `sapproductconfigmodel` (as of version 5.7)

Some of these extensions are dependent on each other. The graphic illustrates these dependencies.

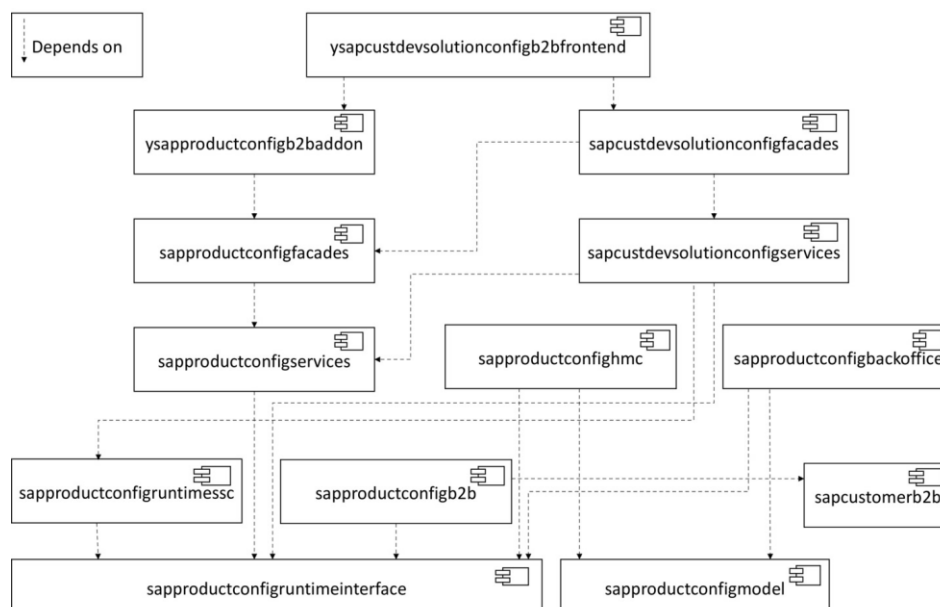


Figure 3: Extension Dependencies

`ysapproductconfigb2baddon` add-on, for example, is dependent on the following extensions:

-
- `sapproductconfigacades`
 - `sapproductconfigservices`
 - `sapproductconfigruntimeinterface`

This means that when you activate the `ysapproductconfigb2baddon` add-on, these three extensions are activated automatically.

For more information about the product configuration extensions on which the solution configuration is based, see <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Configure, Price, and Quote for Product Configuration* → *Extensions and AddOns for SAP CPQ for Product Configuration*.

15 Supported Characteristic Types

Both alphanumeric and numeric characteristic types are supported by the B2B Commerce Accelerator add-on UI, both for single and multiple-value characteristics.

Free-entry fields (“additional values”) with alphanumeric or numeric values are supported only for single-value characteristics.

Interval domains (that is, characteristics that have an interval as the domain) are not supported.

16 Pricing Attributes

In the B2B Commerce Accelerator, the following attributes are used for pricing:

Level	Field in	Description	Field in SAP Hybris
Item (product)	PMATN	Pricing reference material	Article number Backoffice path: Catalog → Products
Header (business partner)	VKORG	Sales organization	Sales organization Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
-	VTWEG	Distribution channel	Distribution channel Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
-	SPART	Division	Division Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
-	LAND1	Country key	Country Backoffice path B2B Commerce → B2B Unit, General tab
-	KUNNR	Customer number	ID (company) Backoffice path B2B Commerce → B2B Unit, General tab
-	KONDA	Price group (customer) / condition group	Price group Backoffice path B2B Commerce → B2B Unit, Customer Prices tab
-	KONWA	Currency	Currency currently used in the online store

17 Configuration Context Attributes

In the B2B Commerce Accelerator, the following configuration context attributes are used:

Field in SAP	Description	Field in SAP Hybris
VBAK-ERDAT	Creation date	Taken from order
VBAP-KWMENG	Order quantity	Taken from order
VBAP-MATNR	Material number	Article number Backoffice path Catalog → Products
VBAK-SPART	Division	Division Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
VBAK-VTWEK	Distribution channel	Distribution Channel Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
VBAK-VKORG	Sales organization	Sales Organization Backoffice path SAP Integration → SAP Base Store Configuration, Common tab
VBPA_AG-LAND1	Country key Sold-to party	Country Backoffice path B2B Commerce → B2B Unit, General tab
VBPA_RG-LAND1	Country key payer	Country Backoffice path B2B Commerce → B2B Unit, General tab
VBAK-KUNNR	Customer number Sold-to party	ID (company) Backoffice path B2B Commerce → B2B Unit, General tab
VBPA_AG-KUNNR	Customer number Sold-to party	ID (company) Backoffice path B2B Commerce → B2B Unit, General tab
VBPA_RG-KUNNR	Customer number payer	ID (company) Backoffice path B2B Commerce → B2B Unit, General tab

18 Deploying User Exits for Use with SAP Hybris

User exits for the configuration and pricing engines in SAP Configure, Price, and Quote for solution sales configuration can also be deployed for use within SAP Hybris.

Procedure

Copy the .jar files that contain the user exit classes to the following path:

`${HYBRIS_BIN_DIR}\ext-integration\sap\productconfig\sapproductconfigruntime\lib`

More Information

SAP Note [809820](#)

19 Troubleshooting

For information about frequently asked questions and issues, see SAP Note [2246019](#).

20 Appendix

20.1 Installing ysapordermgmtb2baddon for SAP Synchronous Order Management

6. Add the following extensions and extension paths to `localextensions.xml`:

```
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/productconfig' autoload='false' />
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/synchronousOM' autoload='false' />
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/core' autoload='true' />
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/creditCheck' autoload='false' />
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/testsupport' autoload='true' />
<!--test support -->
<path dir='${HYBRIS_BIN_DIR}/ext-integration/sap/masterdata' autoload='true' /> <!--
master data replication -->

<!--Synchronous order management-->
<extension name='secureportaladdon' />
<extension name='sapcoreaddon' />
<extension name='ysapordermgmtb2baddon' />
<extension name='sapordermgmtb2bfacades' />
<extension name='sapordermgmtb2bservices' />
<extension name='sapordermgmtservices' />
<extension name='sapordermgmtbol' />
<extension name='sapproductconfigruntimeinterface' />
<extension name='sapproductconfigfacades' />
<extension name='sapcreditcheck' />
```

7. If you have created your own storefront extension, ensure that the `project.properties.template` file of `ysapordermgmtb2baddon` under `{hybris-bin-directory}/ext-integration\sap\synchronousOM/` has the following configuration for your storefront:

```
<your storefront
extension>.additionalWebSpringConfigs.ysapordermgmtb2baddon=classpath:/ysapordermgmtb2baddon/web/spring/ysapordermgmtb2baddon-web-spring.xml
```

For more information, see <http://help.hybris.com> → *Accelerators* → *Core Accelerator* → *Hybris AddOn Concept* → *Installing an AddOn for a Specific Storefront*.

8. Register the AddOn with your storefront by entering the `addoninstall` command in the command line dialog box.



Example

```
ant addoninstall -Daddonnames="ysapordermgmtb2baddon" -  
DaddonStorefront.yb2bacceleratorstorefront="<your storefront extension> "
```

More Information

For more information about how to install this add-on, see <http://help.hybris.com> → *Integrations and Data Management* → *SAP Integrations* → *SAP Back-End Integration: SAP ERP and SAP CRM* → *Synchronous Order Management* → *Extensions and AddOns for Synchronous Order Management* → *SAP ERP Extensions and AddOns for Synchronous Order Management* → *ysapordermgmtb2baddon AddOn*.

20.2 Database Connection Parameters for local.properties

Example of Connection Parameters for SSC Database using MySQL

```
crm.system_type=MySQL  
crm.database_hostname=YOURHOSTNAME  
crm.database=DATABASENAME  
crm.database_port=3306  
crm.database_user=USERNAME  
crm.database_password=PASSWORD  
crm.ssc_jndi_usage=false  
crm.ssc_jndi_datasource=
```

Example of Connection Parameters for SSC Database using MS SQL

```
crm.system_type=Microsoft SQL Server 2  
crm.database_hostname=YOURHOSTNAME  
crm.database=DATABASENAME  
crm.database_port=1433  
crm.database_user=USERNAME  
crm.database_password=PASSWORD  
crm.ssc_jndi_usage=false  
crm.ssc_jndi_datasource=
```

Example of Connection Parameters for SSC Database using Oracle

```
crm.system_type=Oracle  
crm.database_hostname=YOURHOSTNAME  
crm.database=DATABASENAME  
crm.database_port=1521  
crm.database_user=USERNAME  
crm.database_password=PASSWORD  
crm.ssc_jndi_usage=false
```

crm.ssc_jndi_datasource=

Parameter Descriptions

Parameter	Description
crm.system_type	Type of the database management system. Possible values are: <ul style="list-style-type: none">• MySQL (for MySQL)• Microsoft SQL Server 2 (for MS SQL Server)• Oracle (for Oracle database, SAP Hybris version 5.6 or higher)
crm.database_hostname	URL of the host providing the database
crm.database	Database name
crm.database_port	Port
crm.database_user	Database logon user
crm.database_password	Password
crm.ssc_jndi_usage	Controls whether the JNDI source is used for the SSC database connection. Possible values are: <ul style="list-style-type: none">• true: JNDI source is used, user and password logon data do not have to be maintained directly• false: JNDI source is not used, user and password logon data must be maintained directly
crm.ssc_jndi_datasource	JNDI source name



www.sap.com/contactsap

© 2016 SAP SE 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 SE or an SAP affiliate company.
SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies. Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> for additional trademark information and notices.

Material Number:

