



Technical Design documentation

Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Updated By | Date | Comments |
| 1.1 | Vamshi Vushakola  Rohit John Alex | 29/02/2016 | Initial Draft |
| 1.2 | Vamshi Vushakola  Rohit John Alex | 07/03/2016 | Added Class diagram, Sequence diagram and Use case diagram |
|  |  |  |  |

Document Sign-Off

|  |  |  |
| --- | --- | --- |
| Name | Role | Signature/Date |
| Gyaneshwar Dubey | Manager |  |
|  |  |  |
|  |  |  |

Table of Contents

[1. Introduction 3](#_Toc452044662)

[1.1. Key Features 3](#_Toc452044663)

[1.2. Key Benefits 4](#_Toc452044664)

[1.3. Order Management Services Extensions 4](#_Toc452044665)

[1.4. Order Management Services Workflow 6](#_Toc452044666)

[Sourcing 6](#_Toc452044667)

[Allocation 6](#_Toc452044668)

[Pick, Pack, Label, and Ship 6](#_Toc452044669)

[Payment Capture 6](#_Toc452044670)

[Commit Taxes 6](#_Toc452044671)

[1.5. Order Management Services User Interfaces 7](#_Toc452044672)

[Order Management Cockpit 7](#_Toc452044673)

[Order Fulfillment Cockpit 7](#_Toc452044674)

[2. Installation and Configuration 8](#_Toc452044675)

[2.1 Setup 8](#_Toc452044676)

[2.2 Initialize 8](#_Toc452044677)

[2.3 Start 8](#_Toc452044678)

[2.4 Import into Eclipse 9](#_Toc452044679)

[3. Modification of the ApprovalProcessCompleteAction 11](#_Toc452044680)

[4. Test the OMS Installation 13](#_Toc452044681)

1. Introduction

Order Management Services Version 5.7.0 is built on top of the hybris Core platform. Previous versions of Order Management Services (5.6.0 and prior), were built on top of the hybris Core+ microservice platform. It allows you to view and manage customer orders, review and edit shipping information, manage returns and refunds, search for and view your inventory information.

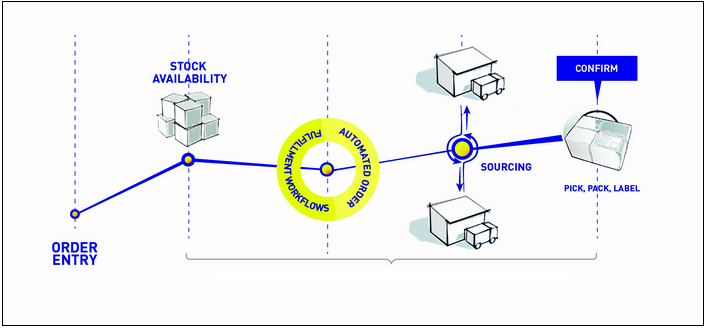
Order Management Services (OMS) is built on top of the hybris Core platform. It allows you to view and manage customer orders, review and edit shipping information, manage returns and refunds, search for and view your inventory information.

Basically **Order Management** **Services** out of the box works for B2C scenarios. In order to enable Order Management Services with a B2B scenario, a little customization is required

## Key Features

OMS includes the following key features:

* Real-time inventory management across all stock locations
* Automated order workflow
* Sourcing
* Order splitting
* Ability to pick, pack, label, and confirm shipments through the Order Fulfillment Cockpit
* Ability to create and manage returns and refunds
* Ability to view fraud reports, payment details, and customer information
* Accelerator integration

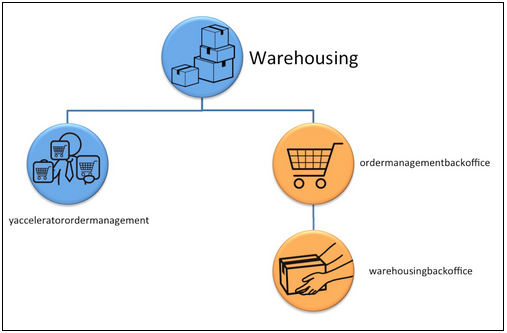


## Key Benefits

OMS offers the following key benefits:

* Fulfil from anywhere
* Save the sale by preventing inventory stock-outs
* A centralized view of inventory
* Reduce shipping time and shipping costs

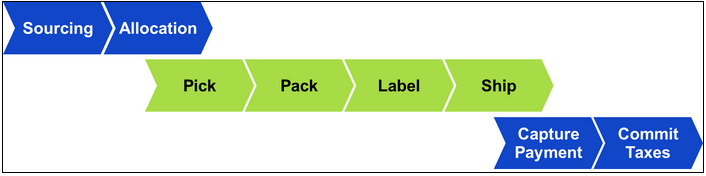
## Order Management Services Extensions



|  |  |  |
| --- | --- | --- |
| Extension | Layer | Description |
| [**ordermanagementbackoffice**](https://wiki.hybris.com/display/release5/ordermanagementbackoffice+Extension) | Consumer | Order Management Backoffice - Graphical user interface designed for customer service agents.   * Access to order information * Order cancellation * Create a return * Manage existing returns |
| [**warehousingbackoffice**](https://wiki.hybris.com/display/release5/warehousingbackoffice+Extension) | Consumer | Warehousing Backoffice - Graphical user interface designed for warehouse users and inventory managers.   * Access to consignments, Points of Service and stock level information * Print pick, pack, and shipping label * Confirm shipping / Confirm pickup * Edit Points of Service and stock level information |
| [**yacceleratorordermanagement**](https://wiki.hybris.com/display/release5/yacceleratorordermanagement+Extension) | Business Layer | Template Model Extension - Main orchestration for orders, consignments and returns.   * Defines the different process engines (order, consignment and return) * Defines the actions executed by these process engines |
| [**warehousing**](https://wiki.hybris.com/display/release5/warehousing+Extension) | Business Layer | Model Extension - Contains the warehousing business logic.   * Order allocation * Order cancellation * Order sourcing * Shipping * Inventory events * Availability To Promise (ATP) |

## Order Management Services Workflow

When an order is received from the hybris Accelerator, OMS processes and completes the order by performing the steps as shown in the following figure.



## Sourcing

Sourcing is the action of determining which physical location (or locations) will supply all items for an order.

## Allocation

Allocation is the action of grouping order items into consignments (that is, determining which items of an order will be shipped from a single location in a single box or package).

## Pick, Pack, Label, and Ship

Once an order is allocated, the following steps in the order workflow take place:

* A pick list is printed, which lists the item(s) that need to be collected from the store or warehouse shelf.
* A pack slip is printed, which indicates all the items that need to be packed in a single shipping box or package.
* A label is printed, which displays the customer's shipping address. Details vary depending on the shipping carrier.
* The order is shipped and confirmation of the shipment is logged in OMS.

## Payment Capture

After the order is packed and ready to be shipped, the payment is charged to the customer. OMS communicates with a third-party payment service provider to complete the payment capture.

## Commit Taxes

The final step of the workflow is to determine the tax amount of an order and to log the information, so that a tax invoice can be created. OMS communicates with a third-party tax invoicing service to complete the tax invoicing.

## Order Management Services User Interfaces

OMS has two user interfaces (perspectives) in the Backoffice for customer service and warehouse employees.

## Order Management Cockpit

The Order Management Cockpit allows a customer service user to:

* Search for and view information about open orders (to be shipped).
* Search for and view information about all orders, that is, shipped orders, cancelled orders, picked orders, packed orders, and orders to be shipped.
* View order information including order details, consignments, payment transactions, and fraud reports.
* Cancel an item or an order.
* Create and manage returns.
* Confirm a manual or automatic refund.
* Cancel a refund and perform a manual tax reverse.

## Order Fulfillment Cockpit

The Order Fulfillment Cockpit allows a warehouse user to:

* View and manage customer orders, such as print pick slips, pack slips, and shipping labels, as well as review and edit package and shipping information.
* View and manage to be picked up store orders.
* Search for and manage stockroom locations.
* View and manage inventory.

1. Installation and Configuration

From your root hybris installation folder:

1. Go to **installer/recipes**
2. Create a folder called **"b2b\_acc\_oms"**
3. Within this new folder, create a new file named**"build.gradle"**and add the following content into it:

|  |
| --- |
| Hybris\_5.7.0.5\installer\recipes\b2b\_acc\_oms/build.gradle |
|  |

## Setup

To setup the hybris platform with the OMS extensions, run the following command:

**Windows: install.bat -r b2b\_acc\_oms setup**

**Unix: ./install.sh -r b2b\_acc\_oms setup**

## Initialize

To initialize the platform, run the following command:

**Windows: install.bat -r b2b\_acc\_oms initialize**

**Unix: ./install.sh -r b2b\_acc\_oms initialize**

## Start

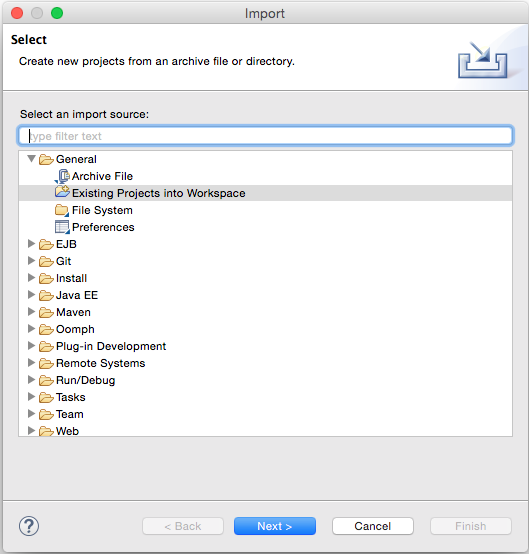
To start the platform, run the following command:

**Windows: install.bat -r b2c\_acc\_oms start**

**Unix: ./install.sh -r b2c\_acc\_oms start**

## Import into Eclipse

 Select **Import...** from the file menu:



You need to have an extension generated from yacceleratorordermanagement. Use command:**ant extgen** on hybris/bin/platform folder and choose the option to create an extension out of the yacceleratorordermanagement template and follow the instructions.

**Important info**

During the generation process you will be asked to choose an extension name please make sure to name it: omsb2b

When asked for the package name input de.hybris.platform.omsb2b

If you choose different naming, you are responsible to modify the code to correspond to your package and naming.

For this example, we named the newly generated extension **omsb2b** but you can name it according to your needs.

In this generated extension (**omsb2b**), you need to add a required extension into the extensioninfo.xml.

|  |
| --- |
| **extensioninfo.xml** |
| ...      <requires-extension name="b2bapprovalprocess"/>  ... |

**Important Step:**

After successfully generating your new extension you have to go back to your **localextensions.xml** and replace the yacceleratorordermanagement extension by the newly created **omsb2b** extension as follows:

|  |
| --- |
| **localextensions.xml** |
| ...      <!--B2B list of extensions -->      <extension name='b2bacceleratoraddon' />      <extension name='b2bacceleratorservices' />      <extension name='b2bapprovalprocess' />      <extension name='b2bcommerce' />      <extension name='b2badmincockpit' />      <extension name='b2bcommercebackoffice' />      <!--ordermanagement list of extensions -->      <extension dir="{$commerceSuiteHome}/hybris/bin/custom/omsb2b"/>      <extension name='warehousingbackoffice' />      <extension name='warehousing' />      <extension name='ordermanagementbackoffice' />  ... |
| Go to /hybris/bin/platform home and build your system by calling **ant clean all**. |

1. Modification of the ApprovalProcessCompleteAction

In B2B, the placement of an order goes through an approval process. You could have several ways to approve an order but each one of them is ending with an ApprovalProcessCompleteAction.

The customization goal is to overwrite this final step so that it will:

1. Publish an event of type ApprovalProcessCompleteEvent (insurance for backward compatibility).
2. Create/Start the order process engine defined in the OMS extension.

Below is the java file that you need to create:

|  |
| --- |
| **OmsApprovalProcessCompleteAction.java** |
|  |

Your IDE might not know where to find ApprovalProcessCompleteEvent (mentioned in line 27 above), we suggest to add b2bapprovalprocessserver.jar to your build path.

You can find it in, **commerceSuiteHome/hybris/bin/ext-commerce/b2bapprovalprocess/bin**.

Then, you must declare this new bean into your Spring context:

|  |
| --- |
| **omsb2b-spring.xml** |
| ...      <alias name="defaultApprovalProcessCompleteAction" alias="approvalProcessCompleteAction"/>      <bean id="defaultApprovalProcessCompleteAction"            class="de.hybris.platform.omsb2b.actions.b2b.OmsApprovalProcessCompleteAction"              parent="abstractAction">          <property name="businessProcessService" ref="businessProcessService"/>          <property name="eventService" ref="eventService"/>      </bean>  ... |

This overwrites the default behavior of the approval process complete action and from now, the orders placed from B2B will be managed by the order process engine.

After you make the changes above, you can go to your platform home and build by calling **ant clean all**.

Your build should be successful, initialize and run the hybris server then test that an order made from the asianpaints website will have consignments in backoffice.

1. Test the OMS Installation

Go to <https://localhost:9002/asianpaintsb2bstorefront/> place an order for an item i.e. Apcolite 10L (Green). Follow instruction to continue your checkout.

To test if OMS has received the order you placed, just point your browser to: <http://localhost:9001/backoffice/>, then click on the Order Fulfillment icon at the top of the window. Then go to "Order Shipping" and click on "All Consignments".

