**Customizing Existing perspective:**

1. First we will look into the **ordermanagementbackoffice-backoffice-widgets.xml** how the existing widget will be.

<widget-extension widgetId=*"backofficeMainSlot"*>

<widget id=*"omsBackofficePerspective"* widgetDefinitionId=*"com.hybris.cockpitng.borderlayout"* slotId=*"perspectives"* title=*"Order Management"* template=*"false"*>

<widget id=*"omsExplorerTree"* widgetDefinitionId=*"com.hybris.cockpitng.widgets.common.explorertree"* slotId=*"leftSlot"* template=*"false"*>

<widget id=*"openOrderConditionEvaluator"* widgetDefinitionId=*"com.hybris.cockpitng.conditionevaluator"*

slotId=*"cockpitWidgetChildrenInvisible"* template=*"false"*>

<widget id=*"openOrderStatus"* widgetDefinitionId=*"de.hybris.ordermanagementbackoffice.widgets.OpenOrderStatus"* slotId=*"cockpitWidgetChildrenInvisible"*

template=*"false"*>

<setting key=*"widgetStyleAttribute"* type=*"String"*></setting>

<setting key=*"widgetStyleClass"* type=*"String"*></setting>

<virtual-sockets />

</widget>

<setting key=*"expression"* type=*"String"*>#root.id EQ 'ordermanagementbackoffice.typenode.order.open'

</setting>

<setting key=*"socketDataType\_$T"* type=*"String"*>com.hybris.backoffice.navigation.NavigationNode</setting>

<virtual-sockets />

</widget>

</widget>

</widget>

</widget-extension>

**Here**

1. *omsBackofficePerspective = this is hybris given perspective*
2. title=*"Order Management" = hybris give title*
3. *omsExplorerTree = this is the hybris given widget*
4. widgetDefinitionId=*"com.hybris.cockpitng.widgets.common.explorertree" = we have to override this widget id in our customized definition.xml*
5. existing **config.xml** file will be

<context component=*"oms-explorer-tree"*>

<n:explorer-tree xmlns:n=*"http://www.hybris.com/cockpitng/config/explorertree"*>

<n:navigation-node id=*"ordermanagementbackoffice.treenode.orders"*>

<n:type-node id=*"ordermanagementbackoffice.typenode.order.open"*/>

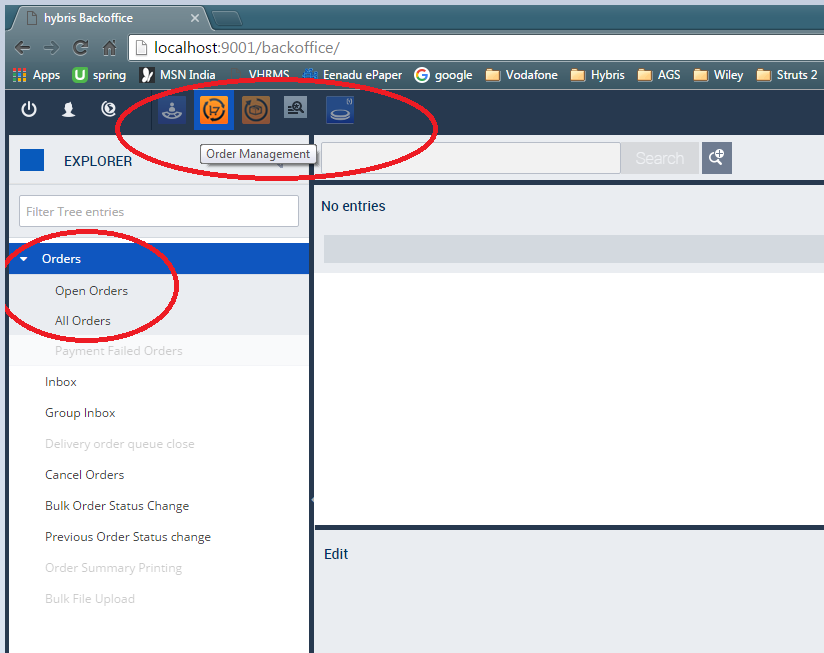
</n:navigation-node>

</n:explorer-tree>

</context>

Here

1. If you see in above *openOrderConditionEvaluator = widget you will find type node ordermanagementbackoffice.typenode.order.open = this will give the name*
2. *Here* navigation-node = is the main tab and type-node is the sub tab
3. *oms-explorer-tree* = in below image there is a tree structure in explorer left side. Whatever you specified in the *oms-explorer-tree those will display in explorer left panel*



**Now we will see the customization**

1. In our own extension **config.xml** we will add new entry cancel orders in *oms-explorer-tree*

<?xml version=*"1.0"* encoding=*"UTF-8"* standalone=*"yes"*?>

<config xmlns=*"http://www.hybris.com/cockpit/config"*

xmlns:wz=*"http://www.hybris.com/cockpitng/config/wizard-config"*>

<context component=*"oms-explorer-tree"* merge-by=*"module"*>

<n:explorer-tree xmlns:n=*"http://www.hybris.com/cockpitng/config/explorertree"*>

<n:navigation-node id=*"ordermanagementbackoffice.treenode.cancelorders"* />

</n:explorer-tree>

</context>

</config>

1. Now we have to create respective widget for cancel order in our own extension **widget.xml**

So here we have to override existing perspective *omsBackofficePerspective*

<widget-extension widgetId=*"omsBackofficePerspective"*>

// here we are deleting existing widget

<remove widgetId=*"omsExplorerTree"* />

<widget id=*"vodafoneOmsExplorerTree"* widgetDefinitionId=*"hu.vodafone.widgets.vodafoneomsexplorertree"*

slotId=*"leftSlot"* template=*"false"*>

<widget id=*"cancelOrdersConditionEvaluator"* widgetDefinitionId=*"com.hybris.cockpitng.conditionevaluator"*

slotId=*"cockpitWidgetChildrenInvisible"* template=*"false"*

access=*"DDTTL,DDTRep,MASPEDTL,MASPEDRep,admin,OCVTL"*>

<widget id=*"cancelOrdersPopUpWidget"* widgetDefinitionId=*"hu.vodafone.widgets.VodafoneCancelOrdersPopUpWidget"*

slotId=*"cockpitWidgetChildrenInvisible"* template=*"true"*>

<widget id=*"wizardNotificationWidget1"* widgetDefinitionId=*"com.hybris.backoffice.notificationarea"* slotId=*"wizardNotificationarea"* template=*"false"*>

<setting key=*"notificationId"* type=*"String"*>creationWizard</setting>

<setting key=*"widgetStyleAttribute"* type=*"String"*></setting>

<setting key=*"debugLog"* type=*"Boolean"*>false</setting>

<setting key=*"widgetStyleClass"* type=*"String"*></setting>

<setting key=*"stackSize"* type=*"Integer"*>3</setting>

<virtual-sockets/>

</widget>

<instance-settings socketEventRoutingMode=*"LAST\_USED"*>

<create onInit=*"false"* reuseExisting=*"false"*>

<all-incoming-events/>

</create>

<close>

<all-outgoing-events/>

</close>

<select onInit=*"false"*>

<all-incoming-events/>

</select>

</instance-settings>

<setting key=*"notificationId"* type=*"String"*>creationWizard</setting>

<setting key=*"widgetStyleAttribute"* type=*"String"*></setting>

<setting key=*"widgetStyleClass"* type=*"String"*></setting>

<setting key=*"\_width"* type=*"String"*>1200px</setting>

<setting key=*"\_height"* type=*"String"*>700px</setting>

<setting key=*"sortableListHeader"* type=*"Boolean"*>true</setting>

<setting key=*"itemRenderer"* type=*"String"*>listViewRenderer</setting>

</widget>

<setting key=*"expression"* type=*"String"*>#root.id EQ

'ordermanagementbackoffice.treenode.cancelorders'

</setting>

<setting key=*"socketDataType\_$T"* type=*"String"*>com.hybris.backoffice.navigation.NavigationNode

</setting>

<setting key=*"navigationNodeId"* type=*"String"* value=*"ordermanagementbackoffice.treenode.cancelorders"* />

<virtual-sockets/>

</widget>

</ widget-extension>

Here

1. *hu.vodafone.widgets.VodafoneCancelOrdersPopUpWidget = this widget id will be configure in definition.xml*

***definition.xml***

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<widget-definition xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xsi:noNamespaceSchemaLocation=*"http://www.hybris.com/schema/cockpitng/widget-definition.xsd"*

id=*"hu.vodafone.widgets.VodafoneCancelOrdersPopUpWidget"*>

<name>Cancel orders</name>

<description>Cancel orders</description>

<defaultTitle>Cancel orders</defaultTitle>

<author>hybris</author>

<version>1.0</version>

<view src=*"vodafonecancelorders.zul"* />

<sockets>

<input id=*"input"* type=*"lang.java.Object"*/>

<input id=*"nodeSelected"* type=*"java.lang.Object"* />

<output id=*"sortData"* type=*"com.hybris.cockpitng.search.data.SortData"*/>

</sockets>

<keywords>

<keyword>Wizard</keyword>

</keywords>

<controller class=*"hu.vodafone.widgets.VodafoneCancelOrdersController"* />

</widget-definition>

***vodafonecancelorders.zul***

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<widget xmlns=*"http://www.zkoss.org/2005/zul"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xmlns:h=*"http://www.w3.org/1999/xhtml"*

xmlns:w=*"http://www.zkoss.org/2005/zk/client"*

xmlns:zk=*"http://www.zkoss.org/2005/zk"*

xsi:schemaLocation=*"http://www.zkoss.org/2005/zul http://www.hybris.com/schema/cockpitng/zul/zul.xsd"*

height=*"100%"*>

<vlayout width=*"100%"* height=*"100%"* spacing=*"0"* style=*"overflow: scroll;"*>

<vlayout width=*"100%"* height=*"100%"* spacing=*"0"* >

<div vflex=*"1"* width=*"100%"* sclass=*"yw-listview-listcontainer"* id=*"mainListBoxContainer"* style=*"margin-bottom: 40px;"*>

<hlayout>

<label sclass=*"yw-listview-unassigned-label"*/>

</hlayout>

<listbox height=*"100%"* vflex=*"false"* hflex=*"false"* id=*"mainListBox"*

mold=*"paging"* pageSize=*"10"* emptyMessage=*"${labels.emptylist}"*>

</listbox>

</div>

<div sclass=*"yw-listview-buttons"* >

<button label=*"${labels.vodafone.cancleorder.checkall.lable}"* id=*"checkAllSection"*/>

<button label=*"${labels.vodafone.cancleorder.clearall.lable}"* id=*"clearAllSection"*/>

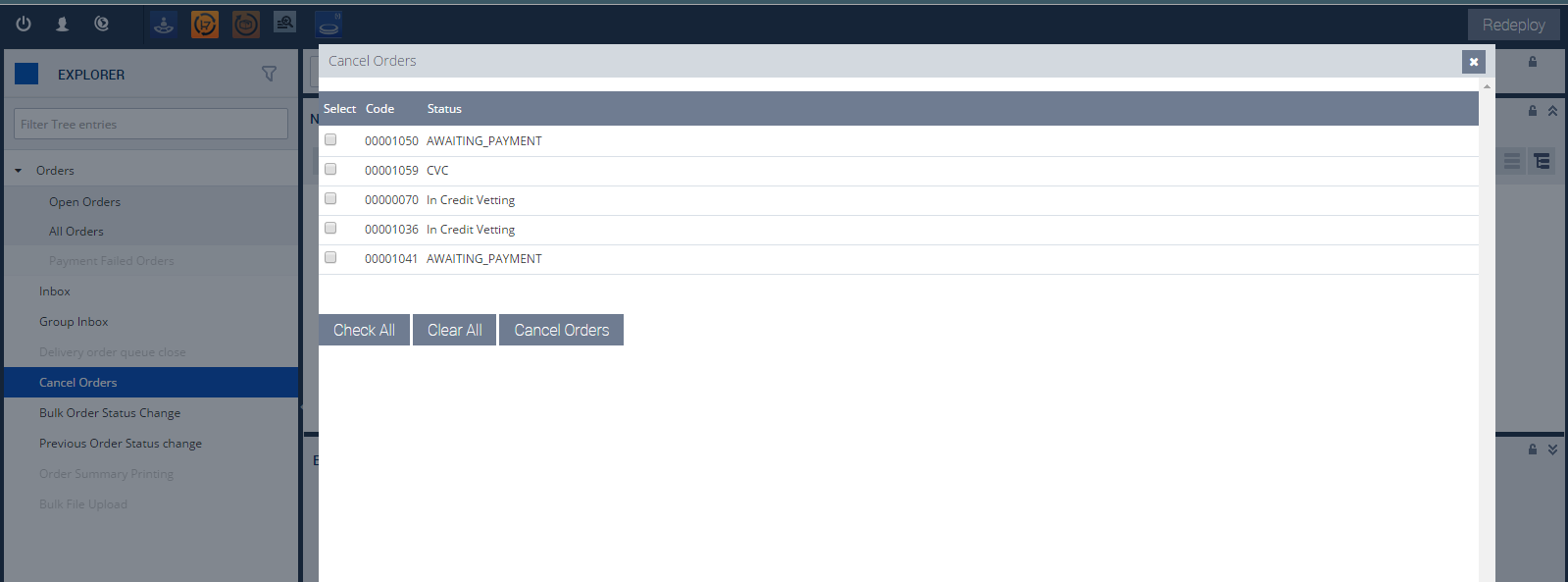
<button label=*"${labels.vodafone.cancleorderbutton.lable}"* id=*"applyChangesSection"*/>

</div>

</vlayout>

</vlayout>

</widget>



**We will do the handling logic in controller like if we check one of the order and click on cancle order button we have to cancel the order**

**Controller class:**

**public** **class** VodafoneCancelOrdersController **extends** DefaultWidgetController {

**private** **static** **final** Logger ***LOG*** = Logger

.*getLogger*(VodafoneCancelOrdersController.**class**);

**private** **static** **final** String ***SELECTED\_OBJECTS\_SECTION*** = "selectedObjects";

**private** **static** **final** String ***VODAFONE\_CANCEL\_ORDER\_NODE*** ="ordermanagementbackoffice.treenode.cancelorders";

**private** **static** **final** String ***LIST\_VIEW\_STRUCTURE***="listview-structure";

**private** **static** **final** String ***ALL\_OBJECTS*** = "allSectionObjects";

**private** **static** **final** String ***VODAFONE\_CANCEL\_ORDER\_TITLE***="vodafone.cancleorder.title.lable";

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

@Wire

**private** Listbox mainListBox;

@Wire

**private** Listhead mainListBoxHead;

@Wire

**private** Button checkAllSection;

@Wire

**private** Button clearAllSection;

**etc**

@Override

**public** **void** initialize(Component comp) {

Boolean isNodeSelected = getValue(VodafonebackofficeConstants.***VODAFONE\_SELECT\_NODE***, Boolean.**class**);

**if** (isNodeSelected == **null**) {

**return**;

}

**super**.initialize(comp);

getWidgetInstanceManager().setTitle(Labels.*getLabel*(***VODAFONE\_CANCEL\_ORDER\_TITLE***));

renderView();

}

**protected** **void** renderView() {

setValue(VodafonebackofficeConstants.***VODAFONE\_COLUMN\_CONFIG***,

loadColumnConfiguration(VodafonebackofficeConstants.***VODAFONE\_ORDER\_STR***, ***LIST\_VIEW\_STRUCTURE***));

renderSection(loadColumnConfiguration(VodafonebackofficeConstants.***VODAFONE\_ORDER\_STR***, ***LIST\_VIEW\_STRUCTURE***));

}

**private** **void** renderSection(**final** ListView config) {

getPagableData(checkAllSection, clearAllSection, applyChangesSection,

***SELECTED\_OBJECTS\_SECTION***, config, VodafonebackofficeConstants.***VODAFONE\_CHECKBOX\_SECTION***);

}

**private** **void** getPagableData(**final** Button checkAllButton,

**final** Button clearAllButton, **final** Button applyChangesButton,

**final** String selectedObjectsSection, **final** ListView config,

**final** String checkBoxKey) {

***LOG***.info("OMS::VodafoneCancelOrdersController:: getPagableData::Start");

Pageable pageable = **null**;

**try** {

pageable = orderSearchService.search(**null**, **null**, **true**);

} **catch** (**final** Exception e) {

***LOG***.error("" + e);

}

setValue("pageable", pageable);

buildHeaders(config, pageable);

render(pageable, config);

checkAllButton.addEventListener("onClick", **new** EventListener() {

@Override

**public** **void** onEvent(**final** Event event) **throws** Exception {

VodafoneCancelOrdersController.**this**.toggleCheckBoxs(

getValue(checkBoxKey, List.**class**), **true**);

}

});

clearAllButton.addEventListener("onClick", **new** EventListener() {

@Override

**public** **void** onEvent(**final** Event event) **throws** Exception {

VodafoneCancelOrdersController.**this**.toggleCheckBoxs(

getValue(checkBoxKey, List.**class**), **false**);

}

});

applyChangesButton.addEventListener("onClick", **new** EventListener() {

@Override

**public** **void** onEvent(**final** Event event) **throws** Exception {

VodafoneCancelOrdersController.**this**.applyChanges(

getValue(selectedObjectsSection, Set.**class**), **false**);

}

});

***LOG***.info("OMS::VodafoneCancelOrdersController:: getPagableData::End");

}