

Clarks - Multi Country PCM Technical Guide

Document Revision History

Date	Revision	Author	Summary of Changes
December 07, 2015	0.1	Marc Ludwig	Draft
December 15, 2015	0.2	Marc Ludwig	Issued for review
December 16, 2015	0.3	Gary Bennett	Review comments
December 16, 2015	1.0	Marc Ludwig	Issued version

Table of Contents

1	Summary	4
1.1	PCM Gap Analysis	4
1.2	Multi Country PCM Install steps	5
2	Multi Country Extensions	8
3	Multi Country Data Model	10
3.1	Summary	10
3.2	ProductAvailabilityGroup	11
3.3	ProductAvailabilityAssignment	11
3.4	BaseStore	12
3.5	SolrHeroProductDefinition	13
3.6	AbstractSolrSearchProfile	13
4	Multi Country Restrictions	14
4.1	CMSBaseStoreTimeRestriction	14
4.2	CMSTimeRestriction	15
4.3	PromotionBaseStoreRestriction	15
4.4	PromotionTimeRestriction	15
4.5	BaseStoreVoucherRestriction	15
5	Multi Country Main Components	16
5.1	Multicounty (core)	16
5.2	Multi Country backoffice	20
5.3	Multi Country Extras	21
5.4	Multi Country Search Restrictions	24
5.5	Multi Country Integration Tests	29

1 Summary

Hybris Professional Services have been engaged to deliver the Hybris PS Multi Country PCM package into the Clark's Pangaea2 project, being developed by Accenture.

This document forms part of that delivery, together with the high level activities as follows:

- SCM checkout (from BitBucket), local build - iteration with partner around SCM folder structure, branching strategy etc.
- Upgrade, resolve issues and test PS Multi Country Package with version 5.7.0.3 of hybris
- Move to new folder structure within Bitbucket repository as updated by Accenture
- Integration of the Multi Country model into the "pangaea2" model structure
 - agreed to remove pre-existing multi country types within the "pangaea2" model
- Integration of the Multi Country extensions into the "pangaea2" source structure
- Create Multi Country sample data based upon the Apparel store accelerator
- Create integration tests and resolve test data issues
- Perform online demonstration of Multi Country features within the "pangaea2" project
- Create technical documentation

1.1 PCM Gap Analysis

The engagement is to deliver the base functionality of the Multi Country PCM package. However, the following potential additional requirements are documented.

#	Requirement	GAP/FIT	Comments
1	Single global product catalog	FIT	The core functionality of PCM package is to provide a single product catalog for multiple channels, countries/regions or markets. Due to the timings of the engagement, sample data has been delivered based upon the Apparel accelerator and this has been demonstrated.
2	Country specific categorisation	GAP	Some form of Global and Local product categorisation will be required. However, the full requirements and PCM process to support this are not documented at this point.

3	Pricing	GAP	Multiple prices may be active at the same time for the same country, e.g. a standard product and a markdown product. However, the full requirements and PCM process to support this are not documented at this point. This could be delivered through extension of the existing provider of prices for the User Price Group in force ("UPGPriceValueProvider")
4	SOLR Index and Store	FIT / CONFIG	At the moment, the proposed Clark's solution will define a Base Store per country. The PCM solution links BaseStore to a Solr index, and this relation is used during the Solr indexing to find out the products that are available for the BaseStore. The Apparel Multi Country data delivered, associates a single Solr index with all of the base stores. Should a specific Solr Index per Base Store be required, this is supported OOTB through the updating of the Base Store, however consideration must be taken of the potential performance impact of multiple indexes.
5	SOLR Indexing Level	FIT / CONFIG	At the time of engagement, no SOLR indexes had been defined for the Clark's product catalog. The sample Multi Country Apparel data, defines SOLR Indexes at the leaf level. the "SolrIndexerQuery" should be configured dependent upon the level at which the indexing should be performed. Three sample levels of search restriction are delivered as part of the base package. Currently within the sample data, "Frontend_ProductBaseStoreVariant" is enabled. Further detail is available within the Multi Country Search Restrictions section of this document.
6	Size and Fit	GAP	An enhancement has been discussed whereby a country / market will have a specific size and fit. The source of this mapping, and the manner in which it will be supported has not been documented. Extension of the Multi Country package (in a manner similar to market / country specific pricing) may be appropriate, however the PCM process to support this must be understood.

1.2 Multi Country PCM Install steps

With a running platform including the "pangaea2" extensions, the following steps are required to install the Multi Country PCM package

- add the multi country PCM extensions to the install

```
<!-- PS Multi Country --><extension
dir="$localdirectory/multicountrypackage/multicountry"/><extension
dir="$localdirectory/multicountrypackage/multicountryextras"/>
<extension
dir="$localdirectory/multicountrypackage/multicountrywebaddon"
/><extension
dir="$localdirectory/multicountrypackage/multicountrybackoffice"
/>
```

- install the add on

```
$ ant addoninstall -Daddonnames="multicountrywebaddon" -
DaddonStorefront.yacceleratorstorefront="pangaea2storefront"
```

- build and run

```
$ cd $HYBRIS_HOME/hybris/platform/bin$ ant all$
./hybrisserver.sh
```

- update the install

Within the hybris Administration Console, perform an update of all the "multicountry" extensions

- ☒ Update running system
- ☒ Clear the hMC configuration from the database
- ☐ Create essential data
- ☒ Localize types

- test the storefronts and resolve any issues

Launch the storefronts to ensure that the web addons are installed.

Please note that when using the responsive UI experience, issues were found with the install of the live edit addon. These can be resolved as follows:

- install the live edit addon

```
$ ant addoninstall -Daddonnames="liveeditaddon" -
DaddonStorefront.yacceleratorstorefront="pangaea2storefront"
```

- update "project.properties" within "pangaea2storefront"

```
cmsPageBeforeViewHandler=com.clarks.pangaea2.hybris.storefront  
.interceptors.beforereview.CmsPageBeforeViewHandleruiThemeResour  
ceBeforeViewHandler=com.clarks.pangaea2.hybris.storefront.inte  
rceptors.beforereview.UiThemeResourceBeforeViewHandler
```

- build and run

```
$ cd $HYBRIS_HOME/hybris/platform/bin$ ant all$  
./hybrisserver.sh
```

2 Multi Country Extensions

The MultiCountry module is delivered in the form of 4 extensions.

Extension name	Description
multicountry	This is the core module of the Multi Country module. It has all the customisations that are required such as the Multi Country Data Model/Types, Solr Index Value Providers, Search Restrictions, Populators, and Product cockpit.
multicountryextras	This module has new data types for CMS Restrictions and Promotion Restrictions that are based on a BaseStore and TimeZone of the BaseStore.
multicountrybackoffice	This module customises the commercesearchbackoffice (hero products, search profiles) so that it is aware of the BaseStore.
multicountrywebaddon	This is a web addon for the Multi Country module, that provides a capability to vary the theme by BaseStore.

The multicountry extension is the core module. This module implements the logic to create assortments of Products from a single Global catalog. These Product assortments are based on pre-defined demarcations such as a Channel or a Country or a Brand or a combination. In the case of Clarks, the demarcation will be based on Country. Thus a Product assortment is created for each of these BaseStores. All the base stores are linked to the single Global catalog.

There are 3 main areas that are affected by this

1. Solr Indexing process
2. Storefront
3. Product cockpit

The **multicountry** extension provides all the functions around those 3 areas. The Solr Index Type has additional attributes that are related to Availability. These attributes are populated during the Solr Indexing. The Storefront uses these attributes to decide on the visibility of a Product. It also uses Search Restrictions to determine the visibility of a Product. A number of Search Restrictions are defined in this extension. Finally, the Product cockpit has additional features to create the assortments for base stores. The UI now has an additional section to show the Base Stores. This can be used to restrict the view of the product set. The Product Manager can select a base store and catalog version and view only those

products that are available for the base store. The Product cockpit also provides capabilities to assign the Availability to a Product, remove Availability from a Product or Edit an existing Availability. The Product cockpit uses Search Restrictions to restrict Product set visibility. Note that the term Product Assortment is the same as Assignment or Availability and will be used interchangeably.

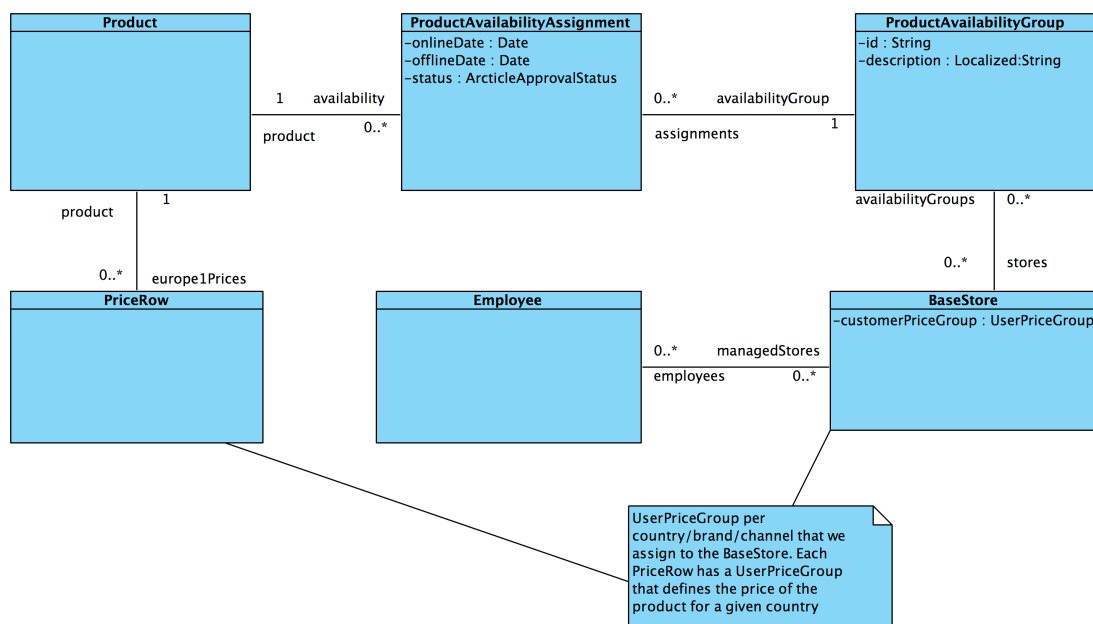
The **multicountrybackoffice** extension provides "multicountry" features to the commercesearch functionality. The OOTB commercesearch functionality provided by hybris links the site to the Solr Index and does not discriminate on the base store. In a multicountry system, the base store is also used in all the functions. E.g. Now the Hero product definition model is extended to link it to a base store. Similarly, the Boost profile is also extended to link it to a base store. Thus the base store has become important within the backoffice extension.

The **multicountryextras** extension provides some "multicountry" features for WCMS and Promotions. Standard hybris promotions are available to all countries and channels. A custom Promotion Restriction is provided that allows filtering of the countries or channels that a specific promotion is applicable for. In addition to being able to restrict which countries promotions are applicable for, the administrator can also specify the time and timezone that the promotion is visible.

A custom WCMS restriction is provided to allow pages and/or components to be filtered. For example we can create a home page specifically for customers from Germany or we can create a specific promotional banner only for customers in the UK. It also provides feature to restrict the content based on the time.

3 Multi Country Data Model

The PCM modules introduces new item types like availability groups, availability assignment etc. In order to allow the solution to have a specific price for a product for each country, brand or channel, the PCM module uses the UserPriceGroup enum is used which is a standard feature of Europe1 price extension which is part of hybris core platform. Each BaseStore is then assigned with a UserPriceGroup which can assigned to a PriceRow to give a specific price for the BaseStore. The following diagram shows the high-level data model of the PCM module.



3.1 Summary

- ProductAvailabilityAssignment provides a link between ProductAvailabilityGroup and Product. Also provides online, offline date and approval status
- 1-N Relation between Product and ProductAvailabilityAssignment to define product assortment
- ProductAvailabilityGroup represents an assortment of products, this is a separate type to the BaseStore so we can share assortments
- BaseStore represents a Country/Brand/Channel

- N-M Relation between Employee and BaseStore (required for the back office visibility)
- UserPriceGroup attribute on the BaseStore (UserPriceGroup for every Country/Brand/Channel)

3.2 ProductAvailabilityGroup

This entity is used to define a group with an id and description. It is related to the BaseStore in a many-many type of relation. Using this relation a single BaseStore can relate to many different product availability groups and therefore different assortments. The simple model is to have a single BaseStore related to a single availability group. E.g. A BaseStore representing UK can relate to a single Availability Group for UK.

3.3 ProductAvailabilityAssignment

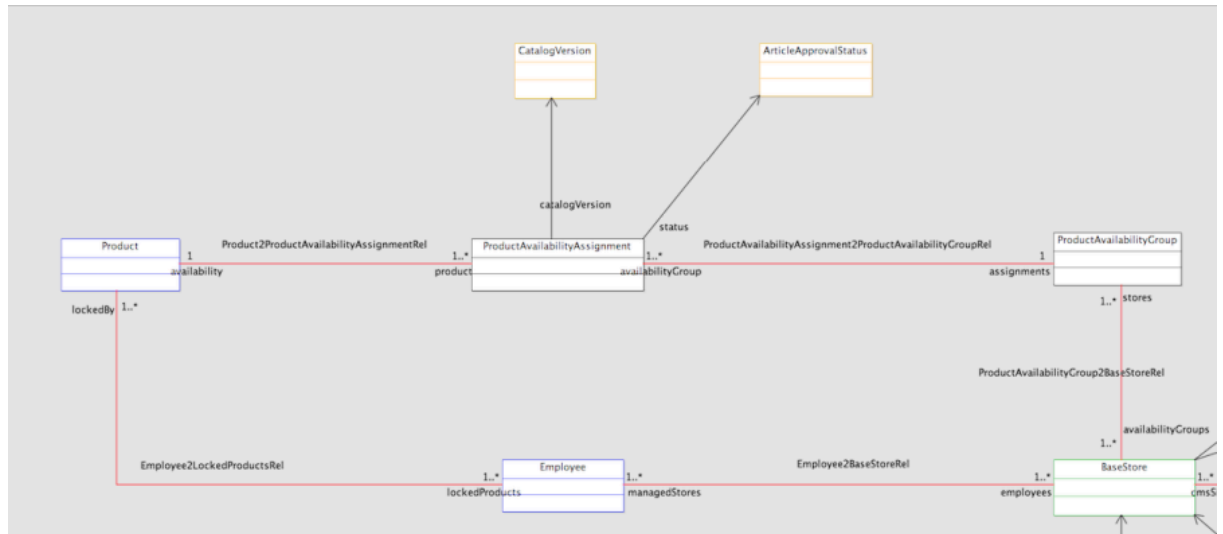
It creates a direct link between the ProductAvailabilityGroup and the Product entities. This entity allows creating sets of Products. It allows a BaseStore (that represents a channel/country/brand), to associate with a Product.

The ProductAvailabilityGroup represents the BaseStore, so the direct association is between the ProductAvailabilityGroup and the Product. Each Product-ProductAvailabilityGroup association has additional attributes – approvalStatus, online date, offline date. These attributes determine the eligibility to display a Product on the storefront, i.e. the approvalStatus must be "approved" and the online and offline date must be valid vis-à-vis the current date.

Once an association is created, there is no need to remove it. If required, only the status or the dates can be changed to effectively remove it from showing at the storefront.

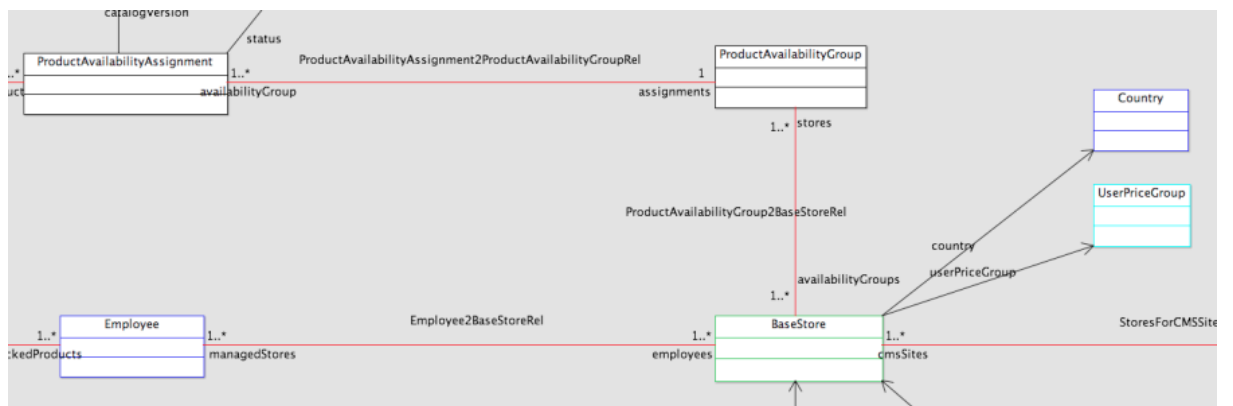
There is a one-many association between the Product and the ProductAvailabilityAssignment. There is a one-many association between the ProductAvailabilityGroup and the ProductAvailabilityAssignment. Effectively it means that there is a many-many relation between Product and ProductAvailability Group. The **Product2ProductAvailabilityAssignmentRel** defines the one-many relationship between the Product and the ProductAvailabilityAssignment.

The **ProductAvailabilityAssignment2ProductAvailabilityGroupRel** defines the one-many relationship between the ProductAvailabilityGroup and the ProductAvailabilityAssignment. The Multi Country module has defined several Search Restrictions that use these relationships to find if a product is eligible for any of the client applications/processes such as the storefront or the backend cockpits or the solr indexing process. These entities are joined in the SQL queries of the Search Restrictions.



3.4 BaseStore

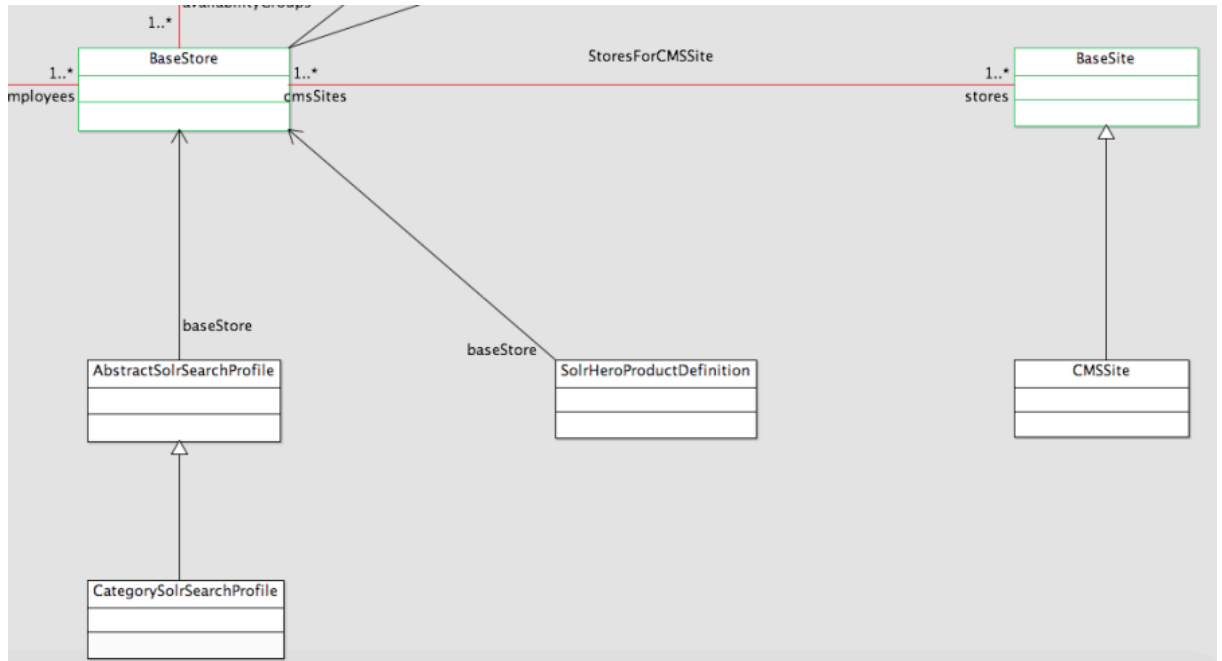
The BaseStore entity is defined in the basecommerce extension. It is extended by other extensions such as commerceservices with additional attributes. The Multi Country module also extends the BaseStore with additional attributes – userPriceGroup and country. PriceRows are defined on a Product and can be linked to Customer specific price groups. This functionality is provided out of the box (OOTB). The Multi Country module allows linking these Price Groups to a BaseStore. Therefore these prices will be available. Note that it is not mandatory to have PriceRows linked to a UserGroup.



The Multi Country module defines relations between an Employee and BaseStore. This enables the control of access to employees, e.g. a Global product manager can have access to all the stores, whereas a UK based product manager should have access only to the UK BaseStore and a EU product manager to a EU BaseStore. Thus these employees would be given control to manage only specific product assortments. The **ProductAvailabilityGroup2BaseStoreRel** defines a many-many relation between ProductAvailabilityGroup and the BaseStore.

The **Employee2BaseStoreRel** defines a many-many relation between an Employee and the BaseStore. This is used to map an employee to manage products in multiple channels/countries/brands via the BaseStore. Also it allows many employees access to a single channel/country/brand via the BaseStore. E.g. this entity

is always checked when the user logs into the product cockpit and also when the user selects the BaseStore in the product cockpit.



3.5 SolrHeroProductDefinition

This entity is defined in the hybris commercesearch module. It is used to boost products so that they appear on top in the search results.

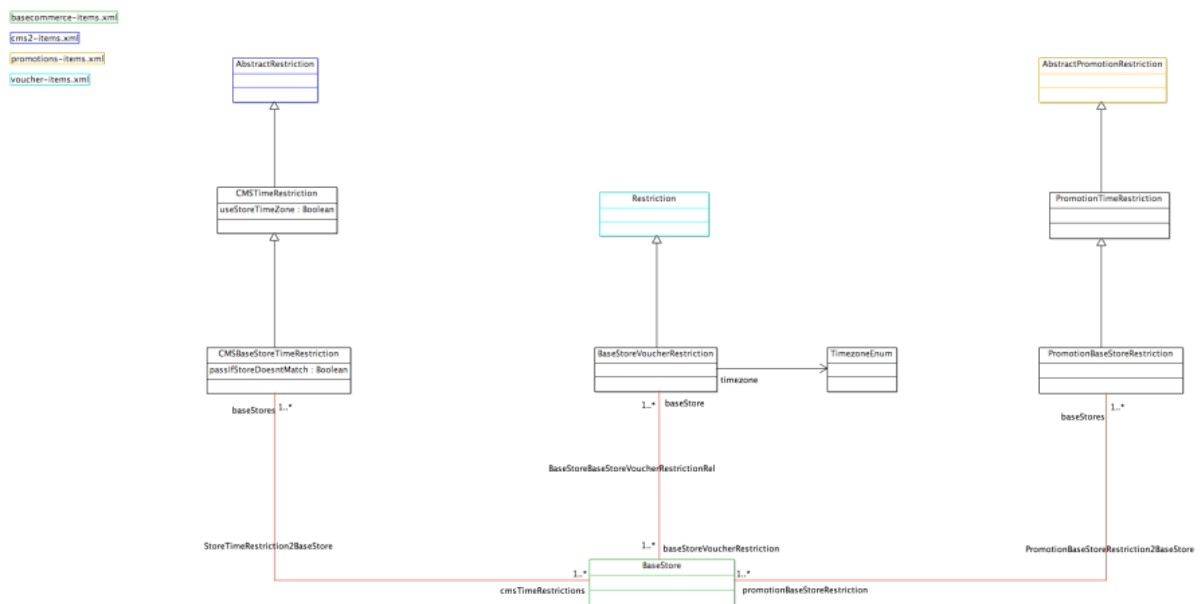
The Multi Country module has extended this to make it aware of the BaseStore. This type has a new attribute called **basestore**. The **SolrHeroProductDefinition** will be created and retrieved on the basis of the category, indextype and the **basestore**.

3.6 AbstractSolrSearchProfile

This entity is defined in the hybris commercesearch module. It is used to boost products so that they appear on top in the search results. The commercesearch feature allows creating search profiles that are based on rules. These rules can be configured and they allow multiple conditions to be defined. Each condition is based on a product attribute and value.

The Multi Country module has made this type aware of the BaseStore. The **CategorySolrSearchProfile** will be created and retrieved on the basis of the category, indextype and the **basestore**.

4 Multi Country Restrictions



4.1 CMSBaseStoreTimeRestriction

This type is defined in the Multi Country module. It extends the CMSTimeRestriction, which is defined in the cms2 extension of hybris.

The purpose is to give the capability to filter CMS by using this restriction, which is mapped to CMSBaseStoreRestrictionEvaluator for the evaluation logic.

It adds 2 additional attributes useStoreTimeZone and passIfStoreDoesntMatch. These attributes are used to configure the behavior of the restriction.

It also defines a period of time in which the CMS page or component is visible to the BaseStore. To do this check it uses the timezone configured on the basestore. This restriction can be applied just like any CMS restriction i.e. on a CMS page or component.

The **StoreTimeRestriction2BaseStore** defines a many-many relation between the CMSBaseStoreTimeRestriction and the Base Store. . This relation is used to configure the BaseStores on a CMS restriction.

4.2 CMSTimeRestriction

This type is defined in the cms2 extension of hybris. The Multi Country module has customized this type with an additional attribute called useStoreTimeZone. The Multi Country module also defines a new type CMSBaseStoreTimeRestriction that extends the CMSTimeRestriction.

4.3 PromotionBaseStoreRestriction

Promotions are filtered by a custom Promotion restriction called PromotionBaseStorePromotion. The evaluation logic of the restriction is in the class.

PromotionBaseStoreRestriction extends from PromotionTimeRestriction so it can have defined a period of time in which the promotion applies.

The **PromotionBaseStoreRestriction2BaseStore** defines many-many relation between the PromotionBaseStoreRestriction and the BaseStore. This relation is used to configure the BaseStores on a Promotion restriction.

4.4 PromotionTimeRestriction

This type defines the time-based attributes such as activeFrom and activeTo. It also defines a configuration attribute useStoreTimeZone.

This type acts as a base type to the main type called PromotionBaseStoreRestriction.

4.5 BaseStoreVoucherRestriction

Vouchers are filtered by a custom restriction. So Vouchers can apply to a specific BaseStore, meaning it can be redeemed at that specified BaseStore.

The BaseStoreBaseStoreVoucherRestrictionRel defines a many-many relation between the BaseStore and the BaseStoreVoucherRestriction. This relation is used to configure the BaseStores on a Voucher restriction.

5 Multi Country Main Components

5.1 Multicountry (core)

Customisation is done on every point where the system must be aware of the base store and the availability groups. The MultiCountryStoreSessionFacade is a key point where a session is initialized. In addition several search restrictions are defined in the system. These restrictions restrict the Product type, so that a Product is made available by following the rules defined by the data model of the Multi Country module, i.e. all the types and relations defined in the Multi Country such as Base store, Product Availability Group, Product Availability Assignment and the Product are used in the Queries so that the right set of products are visible to any client component such as the storefront, the Solr Indexing process and the Product cockpit.

This extension also defines several ImpEx files that are examples. Executing these ImpEx files (see SampleDataSystemSetup) will create a sample catalog called apparelMultiCountryCatalog, and a set of products. It also creates all the relations for the Multi Country module.

This extension also has all the custom classes that are used to modify the Product cockpit UI and behavior. The product cockpit now allows the user to select the base store. A Search restriction is applied so that the right product set is visible to the employee who uses the cockpit. The Multi Country module has defined a relationship between the Employee and the Base store and this relation is used to restrict the Product set for the employee. This restriction also uses the other Multi Country entities such as Base store, Product Availability Group, Product Availability Assignment.

This extension depends on the following extensions

```
<requires-extension name="basecommerce"/>
<requires-extension name="cms2"/>
<requires-extension name="cockpit"/>
<requires-extension name="accelerator services"/>
<requires-extension name="commerce search"/>
<requires-extension name="commerce facades"/>
<requires-extension name="product cockpit"/>
<requires-extension name="csc cockpit"/>
<requires-extension name="commerce search backoffice"/>
<requires-extension name="multicountry extras"/>
```


Class	Description
DefaultActiveProductAvailabilityGroupsStrategy	This class provides methods to return a list of all the Active Availability Groups for a Product. If the Product is a Variant, then it collects all the Availability Groups in the hierarchy and retains only the Approve Availability Groups. This strategy class is used internally by the 3 Solr value Provider classes - OnlineDateProvider, OfflineDateProvider, ProductAvailabilityGroupValueProvider.
UPGPriceValueProvider	This is a Solr value provider and is used to create attribute values in the Solr Index on Price and Price Range. The Multi Country module has customised this so that the availability information is embedded in the attribute name. For example, the typical entries in the Solr Index would be of the form below "priceValue_pricegroup-uk_gbp_double": 101.96,"priceValue_pricegroup-eu_eur_double": 116.85.
OnlineDateProvider	This is a Solr value provider and is used to create attribute values in the Solr Index on entity ProductAvailabilityAssignment.online attribute. These attribute entries are of the form "onlineDate_pricegroup-uk_date_mv": ["2014-02-01T01:00:00Z"], "onlineDate_pricegroup-uk_date_mv": ["2014-02-01T01:00:00Z"]. The onlineDate has availability information in its name. At the time of searching, the Solr query uses these attributes & values and sends it to the solr server that filters out the records. A related attribute is the ProductAvailabilityAssignment.offline attribute.

OfflineDateProvider	<p>This is a Solr value provider and is used to create attribute values in the Solr Index on entity</p> <p>ProductAvailabilityAssignment.offline attribute. This attribute entries are of the form "offlineDate_pricegroup-uk_date_mv": ["2014-02-01T01:00:00Z"], "offlineDate_pricegroup-uk_date_mv": ["2014-02-01T01:00:00Z"], The offlineDate has availability information in its name. At the time of searching, the Solr query uses the online and offline attributes & values and sends it to the solr server that filters out the records.</p>
ProductAvailabilityGroupValueProvider	<p>This is a Solr value provider and is used to create attribute values in the Solr Index on entity ProductAvailabilityGroup. A Product could be assigned to multiple Product Availability groups belonging to one or more Base Stores. This value provider aims to populate the Product index in Solr, with the list of availability groups. The Solr Query embeds the availability groups from the user's session into its query and passes it to the Solr Server. The attribute entries are of the form "availability_string_mv": ["pricegroup-de_-availability", "pricegroup-uk-availability"].</p>
ProductAvailabilityGroupPopulator	<p>This class is used while constructing the Solr Query. It adds the facet for availability into the Solr Query.</p>
MultiCountryFieldNameProvider	<p>This class is used to translate the name of the index property according to the User Price Group and currency linked to the current Base Store prefix, eg: "pricegroup-uk_gbp" etc.</p>
OnlineDatePostProcessor	<p>This is a post processor used in the Solr Query. It adds a filter query that helps to remove all the offline products, also handles the online and offline boundaries.- (-onlineDate:[* TO now] AND onlineDate:[* TO]) -(-offlineDate:[now TO *] AND offlineDate:[TO *]).</p>

MultiCountryStoreSessionFacade	Extends the DefaultStoreSessionFacade, for adding the user price group to the session when the later is initialized. Based on the current BaseStore, the SessionFacade initialises the PriceGroup, ProductAvailability, TimeZone.
DefaultMulticountryRestrictionService	This service is used to get and set Product availability groups into the session. It also provides methods to disable certain Search Restrictions.
MultiCountrySolrIndexerJob	This class extends the SolrIndexerJob. This job uses the Search Restriction on Product that is dependent on the availability groups. This class finds all the availability groups applicable to all the base stores that are using this Solr Facet Configuration. These Availability groups are set into the session to be used by the Search Restriction.
MultiCountrySolrUpdatingWorker	During the Indexing process, Solr workers are invoked. Each Solr Worker Thread executes this class. This class initialises the availability groups in the session. This is essential as the Search Restrictions use the parameter <i>session.availabilityGroups</i> .
MultiCountryEditorAreaNewButtonComponent	Creates the lock/unlock button to prevent a product from being synchronised. It uses the type Employee2LockedProductsRel that represents the relationship between an Employee and the Products that it has locked.
ProductCockpitUIAccessRightService	When a user locks a Product, then it is read only to all other users. But the other user can access and edit the userPriceGroups and Product Availability group that belongs to him/her base store. This class provides those checks.
MultiCountryDynamicQueryServiceImpl	This class extends the DynamicQueryServiceImpl. It provides the Queries related to Lock/Unlock of products such as "All Locked products" and "My Locked products".

ProductCockpitSearchProvider	Filters products to only show products for the currently selected BaseStore. This class uses Search Restriction queries that are defined within this class. The Multi Country package only shows Products that are allowed for the Employee (product cockpit user). To do this it joins the ProductAvailabilityGroup2BaseStoreRel, with the ProductAvailabilityGroup, ProductAvailabilityAssignment. By default, the capability to restrict at the variant level is disabled, but can be overridden within Spring configuration.
BaseStoreSectionSelectionSection	Creates a list selector of BaseStores that the current user is allowed to edit.

5.2 Multi Country backoffice

The multicountry backoffice extension customizes the commercesearchbackoffice.

Main points of integration are with respect to the types HeroProductDefinition and Search Profiles. These integration points are not aware of the Base Store that is the fundamental design block in the Multi Country module. So all classes such as MCHeroProductDefinitionService, MCBoostService, MCHeroProductDefinitionDaoImpl and others have been made "base store aware", so that whenever these types are created or retrieved it will always involve the base store as well.

It requires the following extensions

```
<requires-extension name="backoffice"/>
<requires-extension name="cockpit"/>
<requires-extension name="commercesearchbackoffice"/>
```

Class	Description
MultiCountryHeroProductDefinitionService	This class extends the OOTB DefaultHeroProductDefinitionService. The SolrHeroProductDefinitionModel has been customised in Multi Country module to include the BaseStore. This class creates the SolrHeroProductdefinition on the basis of the Caetgory, IndexedType, CatalogVersion and the BaseStore. It provides methods to fetch SolrHeroProductDefinition objects based on category and BaseStore.

MultiCountryHeroProductDefinitionDaoImpl	This class extends the DefaultHeroProductDefinitionDao.It overrides methods and includes the BaseStore parameter
MultiCountryHeroProductPrepareInterceptor	This intercepts requests to create a new SolrHeroProductDefintion.
MultiCountryGlobalSearchProfileService	This class extends the DefaultGlobalSearchProfileService.The GlobalSolrSearchProfileModel has been extended with the BaseStore attribute. This class has methods to create and retrieve the SearchProfile based on the base store.
MultiCountryBoostService	This class extends the DefaultBoostService.This service is used by the BoostSearchQueryPopulator to fetch the boost Profile based on category, indexedtype and the base store.
MultiCountryCategorySolrSearchProfileValidator	This is a ValidateInterceptor. Validates a Search Profile before saving. It validates on 3 parameters the indexedType, categorycode and the baseStore.
MultiCountryGlobalSolrSearchProfileValidator	This is a ValidateInterceptor. Validates a Search Profile before saving. It validates on 2 parameters the indexedType and the baseStore.
DefaultMulticountryImpersonationService	The Multi Country module extends the DefaultImpersonationService. In addition to the setting up the ImpersonationContext with properties such as user, site, taxgroup, currency, language, it also sets the availabilityGroups into the session.

5.3 Multi Country Extras

This extension, as the name suggests, provides extra types and functions relating to those types. It defines the types CMSBaseStoreTimeRestriction, PromotionBaseStoreRestriction and its relations with the Base Stores.

The PromotionBaseStoreRestriction class helps to restrict the Promotions to apply on configured base stores. Similarly, the CMSBaseStoreTimeRestriction helps to restrict the Content to configured base stores. These restrictions can optionally be made dependent on the timezone of the basestore. In which case, the current time of the timezone is matched with the configured valid from and valid to dates.

It is dependent on the following extensions

```
<requires-extension name="cms2"/>
<requires-extension name="admincockpit"/>
<requires-extension name="cmscockpit"/>
<requires-extension name="promotions"/>
<requires-extension name="commerceservices"/>
```

Class	Description
PromotionBaseStoreRestriction	This is a jalo class. It is also a type and defined in the items.xml. It extends the PromotionTimeRestriction. It is used in the relation - PromotionBaseStoreRestriction2BaseStore, which is a relation between the PromotionBaseStoreRestriction and the BaseStores. It has an evaluate method that is used to check if the restriction is applicable. The check involves finding if the order's basestore is also in the current site's base stores. In addition, it also checks if the base store's timezone and current time match with the promotion's active from and active to dates.
PromotionTimeRestriction	This is a jalo class. It is also a type and defined in the items.xml. It extends the AbstractPromotionRestriction. It defines the properties activeFrom, activeTo and useStoreTimeZone. It has an evaluate method that is used to check if the restriction is applicable. It checks if the base store's timezone and current time match with the promotion's active from and active to dates.

CMSBaseStoreTimeRestriction	<p>This extends the CMSTimeRestriction. It is used to create a relation between the CMSTimeRestriction and the BaseStores. After this relation is defined, it is used by the mapped evaluator MultiCountryCMSBaseStoreTimeRestrictionEvaluator, to evaluate if the content component is eligible. The eligibility criteria involves checking if the current base store matches the list of configured base stores (of the CMSBaseStoreRestriction) and if also by invoking the MultiCountryCMSTimeRestrictionEvaluator, it checks if the current base store's time zone's current time matches the configured valid from and valid to time.</p>
CMSTimeRestriction	<p>This is a Type, which is already defined in the cms2 extension. The type defined in Multi Country module simply adds an additional attribute ; a switch to useStoreTimeZone. This class is extended by another type CMSBaseStoreTimeRestriction. This restriction class is mapped to the MultiCountryCMSTimeRestrictionEvaluator, which handles the check to match the base store's time zone's current time with the configured valid from and valid to timings.</p>
MultiCountryCMSTimeRestrictionEvaluator	<p>This class is of type CMSRestrictionEvaluator. It checks for edibility by checking the configured settings on the CMSTimeRestriction with the current conditions. The criteria to check is the configured valid from and valid to times. The current base store's timezone's current time is matched with the configured valid from and valid to timings. If all conditions match then the content is ok to be displayed.</p>

MultiCountryCMSBaseStoreTimeRestrictionEvaluator	This class is of type CMSRestrictionEvaluator. It checks for edibility by checking the configured settings on the CMSBaseStoreRestriction with the current conditions. The criteria to check are the configured base stores and the configured valid from and valid to times. The current base store is matched with the configured base store. The current base store's timezone's current time is matched with the configured valid from and valid to timings. For checking the timings, it uses another evaluator class called MultiCountryCMSTimeRestrictionEvaluator. If all conditions match then the content is ok to be displayed.
--	--

5.4 Multi Country Search Restrictions

Multi Country has created new Search Restrictions that take into account the Product's relationships with the base store, availability groups, assignments and even the employee.

Following set of Search Restrictions is created by the system setup.

Type	Name	User Group	Active	Restricted To	Query
Product	Frontend_ProductBaseStore ¹	customergroup	false	Product Availability Assignment and Product visibility are checked only on the Base Product. Product visibility Restriction is added to {Product} of the main flexible search	see "productSearchRestriction/multicountry.impe x"
Product	Frontend_ProductBaseStore Variant ²	customergroup	true	Allows all Variant Products whose parent Product is also in the Product Assignment. The Front end as well as the Solr Indexing process uses this.	see "productSearchRestriction/multicountry.impe x"

Product	Frontend_ProductBaseStoreVariantVariant ²	customergroup	false	Allows two levels of Variant Products whose parent Product and Parent Variant is also in the Product Assignment. The Front end as well as the Solr Indexing process uses this.	see "productSearchRestriction/multicountry_two_variantlevels.impex"
ProductAvailabilityAssignment	Frontend_ProductAvailabilityAssignmentApproved	customergroup	true	Restricts the Assignment to only approved	{status} = _ARTICLE_APPROVAL_STATUS_
ProductAvailabilityAssignment	Frontend_ProductAvailabilityAssignmentOnline	customergroup	true	Allows only currently valid productavailabilityassignment.	({onlineDate} IS NULL OR {onlineDate} <= (?session.user.currentDate)) AND ({offlineDate} IS NULL OR {offlineDate} >= (?session.user.currentDate))
BaseStore	Backend_BaseStore	employeeegroup	true	Allows access to only specific employees who have been granted access to the base stores	EXISTS ({ SELECT {pk} FROM {Employee2BaseStoreRel} WHERE {target}={item:pk} AND {source}=(?session.user) }) OR NOT EXISTS ({ SELECT {pk} FROM {Employee2BaseStoreRel} WHERE {source}=(?session.user) })

ProductAvailabilityGroup	Backend_ProductAvailabilityGroup	employeeegroup	true	Allows Availability groups that are related to the base stores that the employee manages	EXISTS ({{ SELECT {pk} FROM { ProductAvailabilityGroup2BaseStoreRel as pag2bs JOIN Employee2BaseStoreRel as ebs ON {ebs.target}={pag2bs.target}} WHERE {ebs.source}=(?session.user) AND {pag2bs.source} = {item:pk} }})
UserPriceGroup	Backend_UPG	employeeegroup	true	Allows prices that are related to the base stores that the employee manages	EXISTS({{ SELECT * FROM {BaseStore as bs JOIN Employee2BaseStoreRel as ebs on {bs.pk} = {ebs.target}} WHERE {ebs.source}=(?session.user) and {bs.userpricegroup} = {item:pk} }})
CMSSite	Backend_CMSSites	employeeegroup	true	Allows only sites which are related to the base stores that the employee manages	EXISTS ({{ SELECT {pk} FROM {Employee2BaseStoreRel as ebs JOIN StoresForCMSSite as ss ON {ebs.target} = {ss.target} } WHERE {ss.source}={item:pk} AND {ebs.source}=(?session.user) }}) OR NOT EXISTS ({{ SELECT {pk} FROM {Employee2BaseStoreRel as ebs} WHERE {ebs.source}=(?session.user) }})

Product	Backend_ProductBaseStore	employeegroup	true	Allows only the products that are related to the base stores that the employee manages,	(EXISTS ({ SELECT {pk} FROM {Employee2BaseStoreRel as ebs JOIN ProductAvailabilityGroup2BaseStoreRel as pag2bs ON {ebs.target}={pag2bs.target} JOIN ProductAvailabilityGroup as pag ON {pag2bs:source} = {pag:pk} JOIN ProductAvailabilityAssignment as paa ON {paa:availabilityGroup} = {pag:pk} } WHERE {paa.product}={item:pk} AND {ebs.source}={?session.user} })) OR NOT EXISTS ({ SELECT {pk} FROM {Employee2BaseStoreRel as ebs} WHERE {ebs.source}={?session.user} }}))
Product	VARIANT_LEVEL1_QUERY ³ Refer to class: ProductCockpitSearchProvider	customergroup	true	Filters products to only show products for the currently selected BaseStore.	Refer to class: ProductCockpitSearchProvider

¹ This is disabled. Instead of this the Frontend_ProductBaseStoreVariant is enabled.

² This restriction is the default one shipped with PCM package.

³ All other restrictions excepting this, are defined in the impex file multicountry.impex

This restriction is not part of that impex file. The search restriction is defined and used only by the Product Cockpit application. Hence it is defined in the class ProductCockpitSearchProvider. Its function is to restrict products so that when a product manager selects the base store in the cockpit, the search restriction should allow only those products that belong to the base store(s) that are managed by the product manager. In addition, if a Product is available but for some reason its parent/grandparent is not available then the product will not be shown. Another way of saying is that if a Product is not available then none of its Child products will be available or visible to the product manager.

5.5 Multi Country Integration Tests

As part of the engagement, the integration tests that form part of the package have been updated and extended as documented below.

Unit and integration tests should be executed as part of the Continuous Delivery system that should be setup to support the development project. In order to execute these, the following ant task was run locally after initialising the JUnit tenant.

```
$ ant yunitinit
$ ant integrationtests -Dtestclasses.extensions="multicountry"
```

5.5.1 DefaultMultiCountryImpersonationServiceTest

Method	Description
availabilitiesGroupsShouldBeEmpty	Adding empty availability group list into impersonation context and making sure the list is the same in the current session after session has been configured.
availabilitiesGroupsShouldNotBeEmpty	Adding availability group list with single element into impersonation context and making sure the list is the same in the current session after session has been configured.

5.5.2 ProductCockpitUIAccessRightServiceTest

Method	Description
productLockedByOtherUser	Locking one product with single user and verifying that it's not possible to modify with anonymous user.
productLockedBySameUser	Locking one product with single user and verifying that it's possible to modify with same user.
productNotLocked	Unlocking one product with single user and verifying that it's possible to modify with single user.

5.5.3 AddToCartHookTest

Method	Description
addProductWithoutGroups	Adding two products into cart without availability assigned and no availability groups in the session.

addProductWithGroups	Adding two products into cart with availability assigned but no availability groups in the session.
addProductWithGroupsAndProductAvailability	Adding two products into cart with availability assigned and availability groups in the session.
addProductWithGroupsAndProductAvailabilityBlocked	Adding two products into cart with availability assigned as blocked and availability groups in the session.

5.5.4 SearchRestrictionTest

The test data associated with this test mirrors that delivered within the sample data. The current test implementation is using "Frontend_ProductBaseStoreVariantVariant" search restrictions.

Method	Description
allProductsVisibleWhenBaseEnabled	Adding approved availability to the base product and verifying that all variants are visible via search.
noProductVisibleWhenBaseDisabled	Adding availability with status "check" to the base product and verifying that none are visible via search.
noProductVisibleWhenNoAssignment	Having base product without availability assigned and verifying that none are visible via search.
noProductVisibleWhenOnlyVariantAssignments	Adding approved availability to variant, and asserting that none are visible via search as base product is not assigned
noProductVisibleWhenBaseDisabledOnlyVariantAssignments	Adding approved availability to variant, and asserting that none are visible via search as base product is at check status
someProductsVisibleWhenSomeVariantsDisabled	Adding approved availability to base, setting 1 mid variant to check status, and one leaf to unapproved, and asserting search count
someProductsVisibleWhenSomeVariantDisabledTwo	Adding approved availability to base, setting both mid variants to approved, and two leaves to unapproved, and asserting search count

5.5.5 SimpleActiveProductAvailabilityGroupsStrategyTest

Method	Description
emptyIfNoAssignment	With no assignment for the Product, there should be no active groups.
emptyIfCheckAssignment	With a CHECK assignment for the Product, there should be no active groups.
singleIfAssignedToBase	With a valid assignment for the Base Product, there should be one active groups for the Base Product.
singleIfAssignedToBaseForVariant	With a valid assignment for the Base Product, there should be one active groups for the Variant Product.