



SUMMIT

THE DIGITAL MARKETING CONFERENCE

March 9-13, 2015 | Salt Lake City, Utah

Twitter : #AdobeSummit



Adobe Experience Manager - Commerce Integration Framework

Varun Venkataraman | Senior Technical Consultant

SUMMIT
THE DIGITAL MARKETING CONFERENCE

Varun Venkataraman
SENIOR TECHNICAL CONSULTANT

Adobe
Suite 1001,
1 Queens Road,
Melbourne
Victoria

T 0422521722

vvenkata@adobe.com



Agenda

- 1 | AEM Commerce Integration framework - Overview
- 2 | Introduction to default components and features in AEM
- 3 | Introduction to AEM Commerce API - Interfaces and Classes
- 4 | Introduction to AEM ElasticPath Commerce Provider Architecture, API - Interfaces and Classes

Key Takeaways

1

Understand AEM Commerce Framework and what is available by default.

2

Understand AEM Commerce API and Extension points.

3

Leverage the framework to add simple functionalities

Agenda

- 1 | AEM Commerce Integration framework - Overview
- 2 | Introduction to default components and features in AEM
- 3 | Introduction to AEM Commerce API - Interfaces and Classes
- 4 | Introduction to AEM ElasticPath Commerce Provider Architecture, API - Interfaces and Classes



Introduction to the AEM Commerce Integration framework

Adobe Marketing Cloud

SOLUTIONS



MEDIA
OPTIMIZER



CAMPAIGN



SOCIAL



EXPERIENCE
MANAGER



TARGET



ANALYTICS



PRIMETIME



AUDIENCE
MANAGER

CORE SERVICES



PROFILES &
AUDIENCES



ASSETS



ACTIVATION



MOBILE



EXCHANGE



COLLABORATION



USER
MANAGEMENT &
ADMINISTRATION

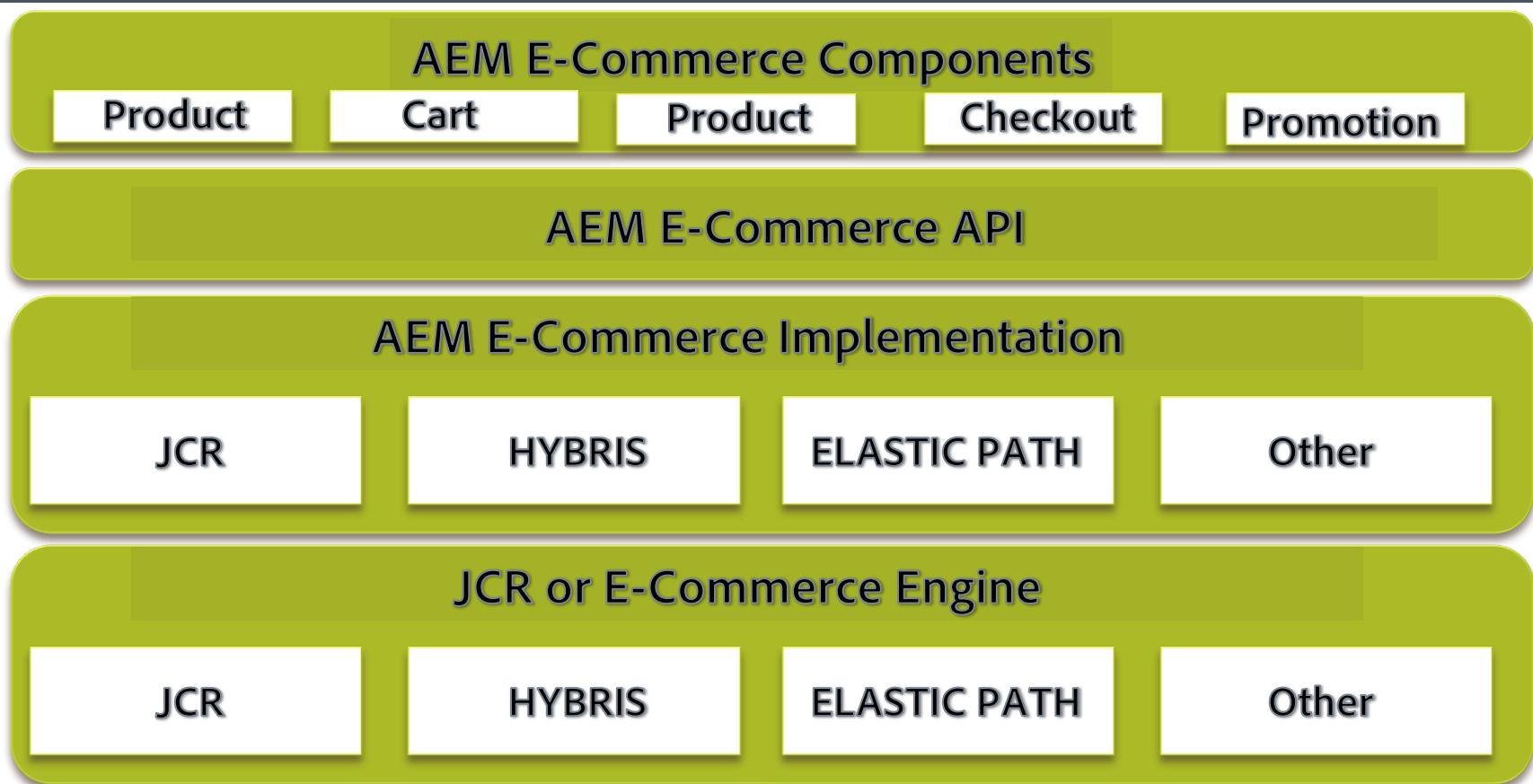
PLATFORM



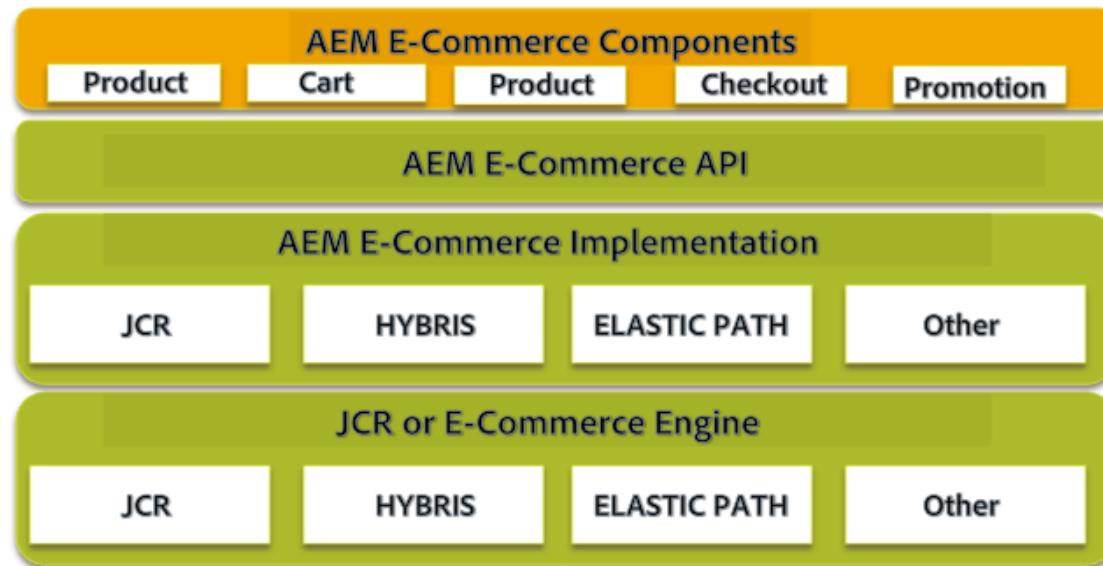
Commercial Commerce Engines Supported Today



AEM E-Commerce Architecture – 4 Layers



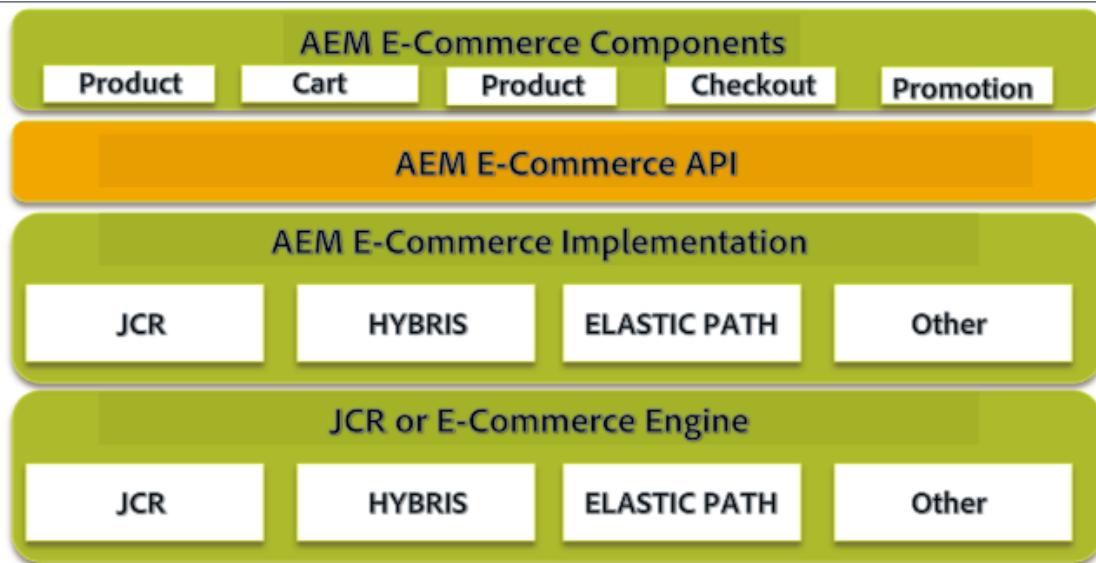
AEM E-Commerce Architecture – AEM E-Commerce Components – Layer 1



1. E-Commerce components

- Front-end functionality and the shop as viewed by the user.
 - Rely on the eCommerce API to retrieve product data and perform actions related to the commerce engine.
-
- **Labs 2 and 3**

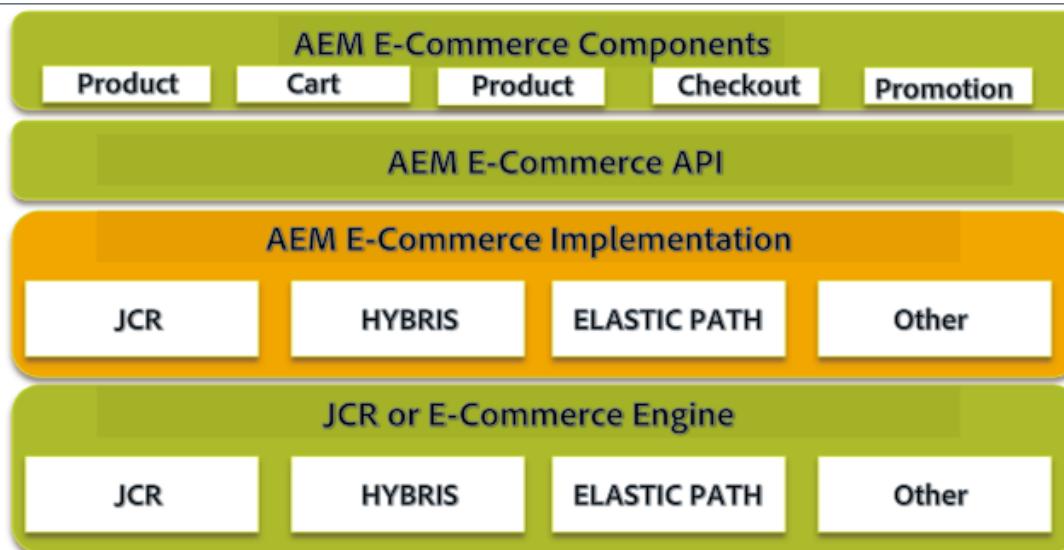
AEM E-Commerce Architecture – AEM E-Commerce Components – Layer 2



2. E-Commerce API

- The central piece of AEM E-Commerce Architecture.
 - Frontend components can be built to implement an E-Commerce application.
 - The API is fixed by Adobe Experience Manager and cannot be adapted.
-
- **Labs 3**

AEM E-Commerce Architecture – AEM E-Commerce Components – Layer 3



3. E-Commerce provider implementation

- The eCommerce API can be implemented to integrate with a 3rd party eCommerce application (or) to create a custom (native) implementation.
 - Default : AEM implementation based on GRANITE (formerly CRX)
 - Reference Implementations by Adobe : Hybris, Elastic Path, Intershop
 - Reference Implementation through Partners : Oracle ATG, IBM Commerce.
-
- **Labs 3 and 4**

AEM E-Commerce Architecture – AEM E-Commerce Components – Layer 4



4. E-Commerce Engine

- E-Commerce backend integrated in the provider implementation.
- Example: JCR, Hybris, Elastic Path, Oracle ATG, IBM Commerce etc.
- **Labs 4**



AEM Commerce – Key Terms

AEM Commerce – Key Terms – Part 1/3

1. Product

- Basic item in the shop.
- **Example:** Running shoe Nairobi, Coat Edmonton Winter, Honda Accord etc.

2. Variant Axis

- Attribute fine-grained property of a product can be implemented as a variant axis.
- Common variant axes : **Color** and **Size**
- **Example:** Blue Nairobi running shoes, Nairobi running shoes size 10, Automatic Honda Accord, Red Honda Accord etc.

3. Variant

- A Type of product and has one or more variant axis specified. Generally a Stock keeping Unit (SKU)
- **Example:** Blue Nairobi Shoe of Size 10, Red Automatic Honda Accord etc.

AEM Commerce – Key Terms – Products, Variants and Variant Axis – AS A SNAPSHOT



AEM Commerce – Key Terms – Part 2/3

4. Shopping Cart

- When a commerce user intends to buy an item, he adds the product (variant) to the shopping cart.
- For each product, the quantity of items added to the cart is specified.

5. Checkout

- When a commerce user is done shopping, he proceeds to the **Checkout**.
- In the following steps, he can actually order the items in his shopping cart.

6. Promotion

- Promotions are discounts a commerce user receives when specific requirements are met.
- Example: it is also possible to create **Perfect Partner** promotions, where a discount is given if a user buy two specific products.

7. Voucher

- Vouchers are codes distributed to selected users which can be entered in the **checkout** process to receive a discount.

AEM Commerce – Key Terms – Part 3/3

8. Product Information Management Data (PIM)

- Product information is provided by a commerce engine.
- Contains all the product data required for the commerce application.
- PIM data is stored in /etc/commerce/products (for default implementations – Geometrixx, Hybris, ElasticPath)

9. Catalog Blueprint

- When a commerce user is done shopping, he proceeds to the **Checkout**.
- In the following steps, he can actually order the items in his shopping cart.

10. Catalog

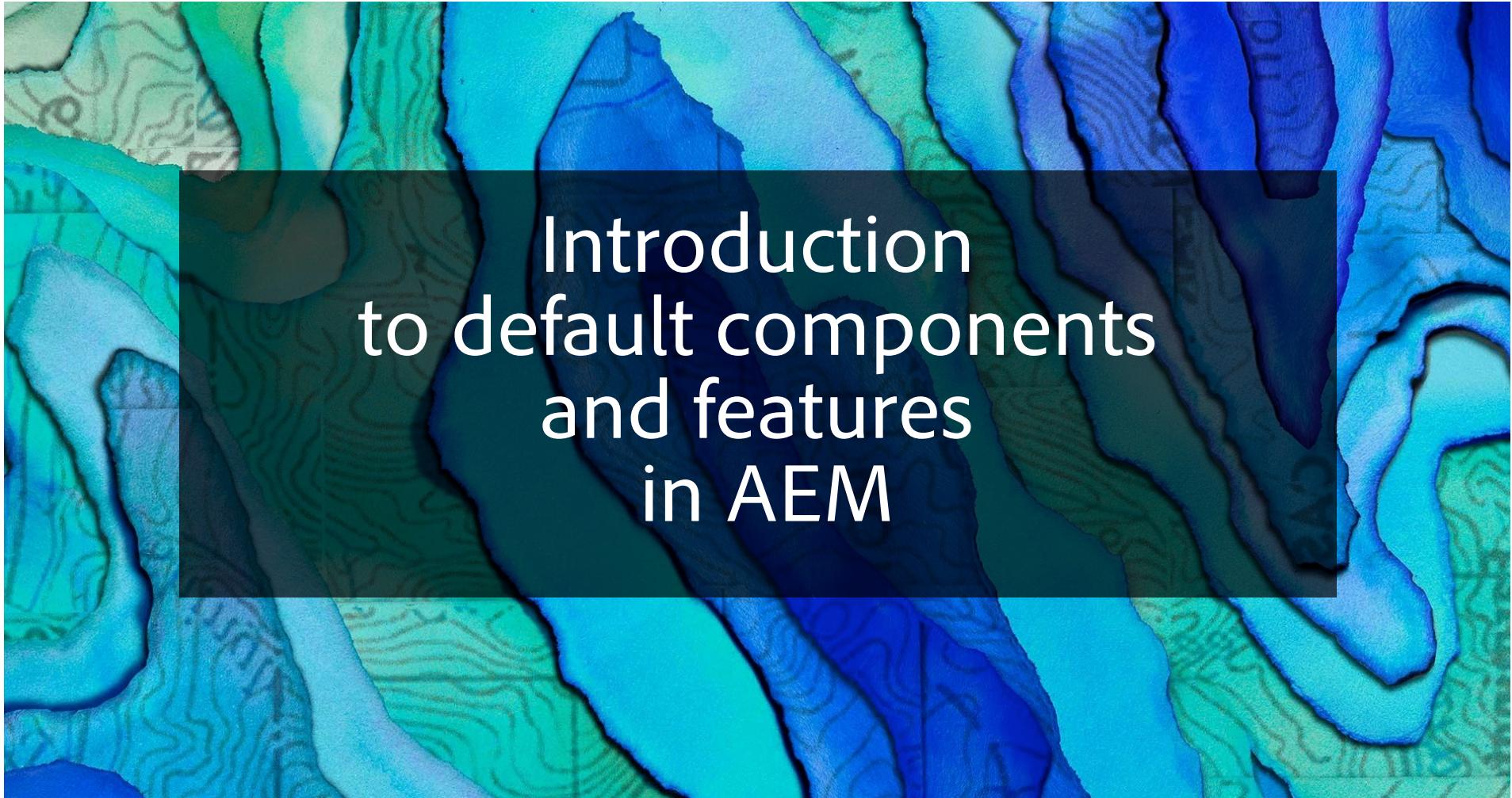
- The generated catalog is usually located in /content and is created as a CQ **live copy**.

Exercise 1 – Summary and Key Takeaway

- Import and Manage Products in Adobe Experience Manager !!
- Use Cases : As a user, I would like to manage my companies products in my content management system, so I will be able to use this information to present to my customers.
- Example :
 1. Product management.
 2. Product related pages.
 3. Compare feature.

Agenda

- 1 | AEM Commerce Integration framework - Overview
- 2 | **Introduction to default components and features in AEM**
- 3 | Introduction to AEM Commerce API - Interfaces and Classes
- 4 | Introduction to AEM ElasticPath Commerce Provider Architecture, API - Interfaces and Classes



Introduction to default components and features in AEM

AEM Commerce – Default Features

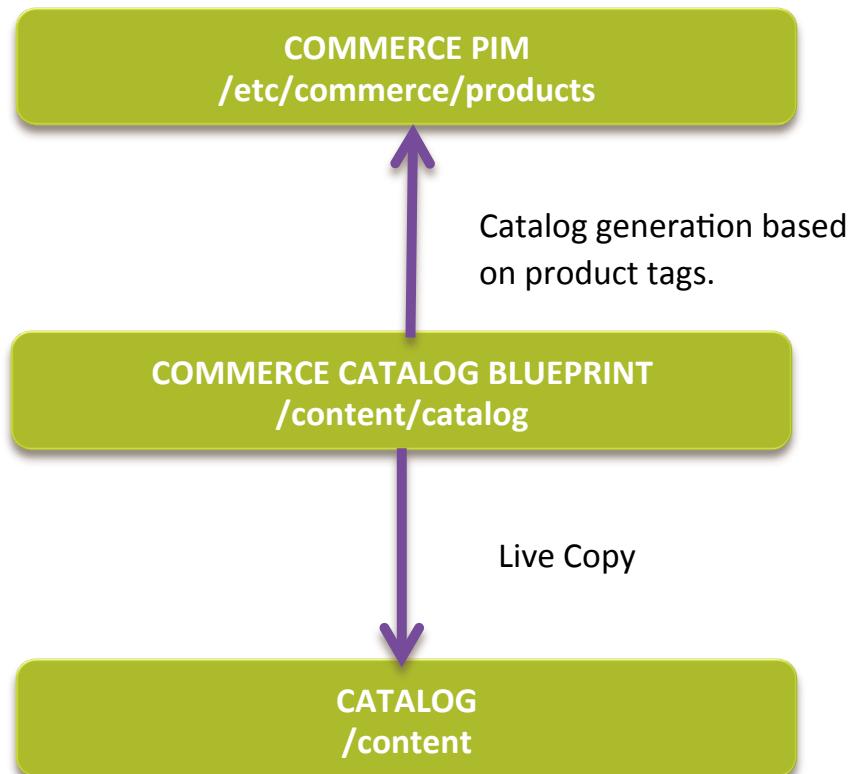
Stores	Comments
Catalogue Management and Rollout	Catalog management and rollout
Templates	/libs/commerce/templates
Components	/libs/commerce/components
ContextHub	/libs/commerce/contexthub/clientlibs

Things to Notes:

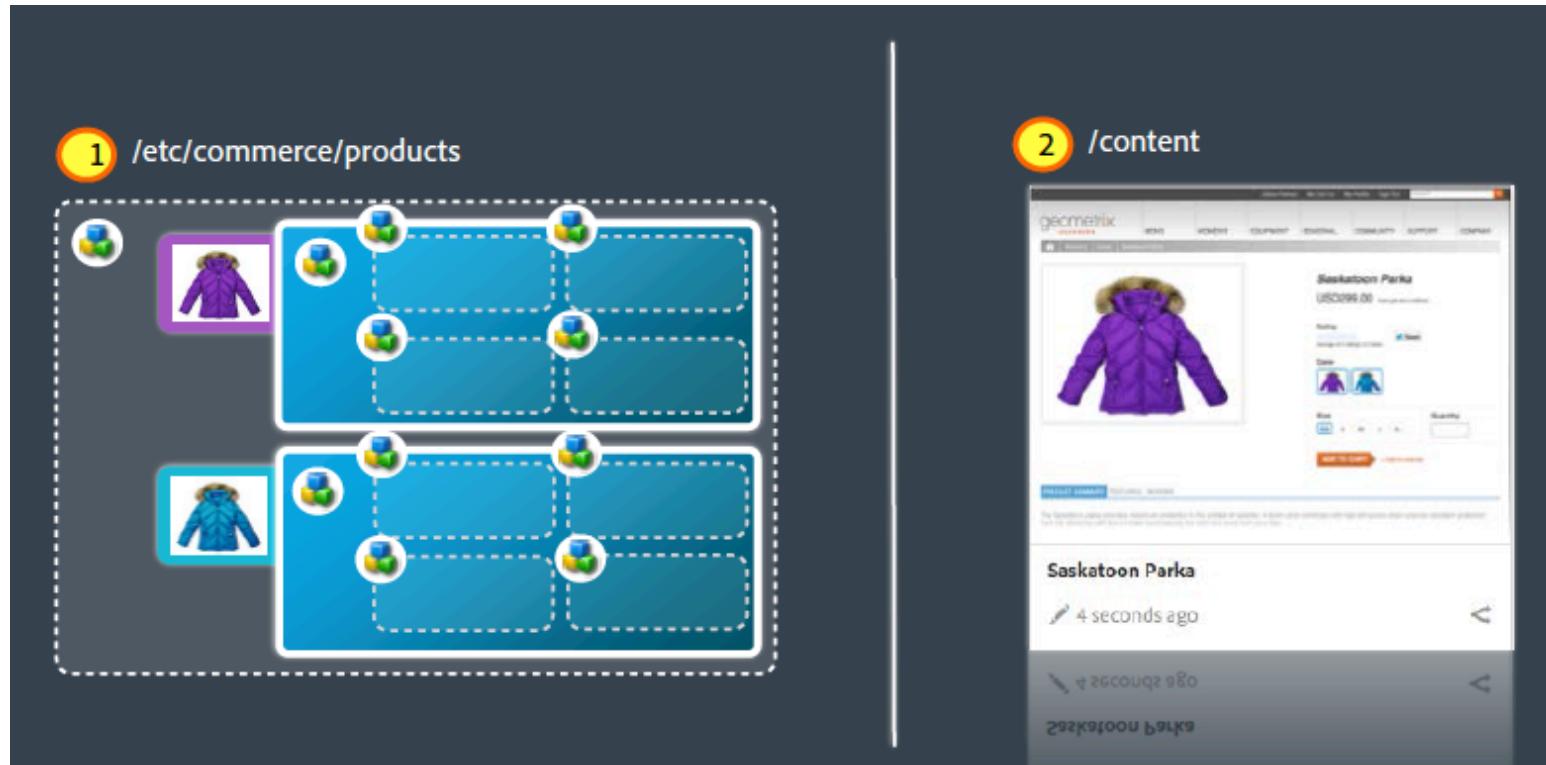
- Default Templates should/can only be used in very specific locations of the JCR repository.
- The default commerce components very likely need to be adapted for a customer environment.
- Still, they can be used for guidance when implementing custom components and in some cases may be extended to reuse some functionality.

AEM Commerce – Catalogue Management

- Adobe AEM provides a default structure how the elements of the commerce application are placed in the JCR repository.
- This applies to the commerce implementations provided by Adobe (Granite, Hybris, Elastic Path). If a custom commerce implementation were created, the points below do not necessarily apply.



AEM Commerce - Commerce Management - AS A SNAPSHOT



AEM Commerce – Components – 1/2

Component	Comments
Call to Action	Setup of call to action components with images.
Image with Add to Cart Link	Annotate a image with Add to Cart buttons
Last-Viewed Call to Action	Tracking of the Call to action clicks
Navigation Product	A navigation list of products
Order History	History of the user's orders
Payment Methods	Available payment methods for user
Product	Display of the product
Recently Viewed Products	A list of products viewed by user tracked through ContextHub
Recommended Products	Related products which are recommended.
Search	Search the product data
Shipping Methods	Available shipping methods for the product

AEM Commerce – Components – 2/2

Component	Purpose
Products Display	Product component used in the product data (/etc/commerce) and in the catalog content. Product nodes in the catalog content reference the PIM product node with the productData property.
Shopping Cart	Component that allows to add, delete, modify and display items in the current users shopping cart.
Voucher	Text box to use vouchers and mange them

AEM Commerce – Templates

Template name	Description	Allowed/suggested location
catalog	Catalog blueprint to create a new catalog from product data	/content/catalog
section	Catalog section based on product data	/content/catalog
voucher	Voucher which can be redeemed for a discount on a shopping cart. Default voucher implementation uses a promotion to specify the actual discount.	
promotion	Promotion to give a discount based on the user segment or redeemed vouchers.	/content/campaigns

AEM Commerce – ClientContext and ContextHub Modules

Stores	Comments	Type
Recent Products	Records all products recently viewed. This can be used for personalization.	Store and Module
Related Products	Record related products for a given product.	Store and Module
Cart	Record what is in the cart in ContextHub	Store and Module
Abandoned Cart	Items added to cart which has been previously abandoned.	Store
Promotions	Promotions applicable based on campaign.	Module
Vouchers	Vouchers available to use for cart and user.	Module

Exercise 2 – Summary and Key Takeaway

The PIM managed from Exercise 1 is rolled out as pages using Page Rollout Hook in AEM Commerce Integration framework.

Use Cases : As a user, I would like to create consistent pages using blueprint template for product categories, sections and product pages, so I will be able to reduce effort in replicating page creation process for each page in authoring interface.

Example :

Product management.

Product related pages.

Compare feature.

Agenda

- 1 | AEM Commerce Integration framework - Overview
- 2 | Introduction to default components and features in AEM
- 3 | Introduction to AEM Commerce API - Interfaces and Classes**
- 4 | Introduction to AEM ElasticPath Commerce Provider Architecture, API - Interfaces and Classes

AEM Commerce – Key Bundles

The following key bundles:

1. com.adobe.cq.commerce.cq-commerce-core
2. com.adobe.cq.commerce.cq-commerce-pim

AEM Commerce – Key Packages

The following key packages in the key bundles:

- com.adobe.cq.commerce.api
- com.adobe.cq.commerce.api.promotion
- com.adobe.cq.commerce.common
- com.adobe.cq.commerce.common.promotion
- com.adobe.cq.commerce.pim.api

AEM Commerce – Key Interface and Classes – Commerce Management

Class	Purpose
CommerceServiceFactory.	OSGi entry point into the commerce provider.
CommerceService	Root of the Commerce API; other classes are fetched through it.
CommerceSession	Represents a user's shopping session
CommerceSession.CartEntry	A CartEntry is a product with a given quantity in the user's shopping cart.

AEM Commerce – Key Interface and Classes – Commerce Management - UML

<<interface>> CommerceSession	<<interface>> CommerceService	<<interface>> CommerceSession.CartEntry
logout() setUserLocale(Locale) getUserLocale() getAvailableCountries() getAvailableShippingMethods() getAvailablePaymentMethods() getProductPriceInfo(Product) getProductPriceInfo(Product, Predicate) getProductPrice(Product, Predicate) getProductPrice(Product) getCartEntryCount() getCartEntries()	login(SlingHttpServletRequest, SlingHttpServletResponse) isAvailable(String) getServer() setContext(Map<String, Object>) getContext() getProduct(String) getPromotion(String) getVoucher(String) getProductCollection(String) isActivated(Product) catalogRolloutHook(Page, Page) sectionRolloutHook(Page, Page) productRolloutHook(Product, Page, Product) search(CommerceQuery) getAvailablePromotions(ResourceResolver) getOrderPredicates() getCountries() getCreditCardTypes()	getCartPriceInfo(Predicate) getCartPrice(Predicate) addCartEntry(Product, int) addCartEntry(Product,int,Map<String, Object>) modifyCartEntry(int, int) modifyCartEntry(int,Map<String, Object>) deleteCartEntry(int) addVoucher(String) removeVoucher(String) getVoucherInfos() supportsClientsidePromotionResolution() addPromotion(String) removePromotion(String) getPromotions() getOrderId() getOrderDetails(String) getOrder() updateOrderDetails(Map<String, Object>, String) updateOrder(Map<String, Object>) placeOrder(Map<String, Object>) getPlacedOrders(String,int,int, String) getPlacedOrder(String) getPriceInfo(Product) getCartPreTaxPrice() getCartTax() getCartTotalPrice() getOrderShipping() getOrderTotalTax() getOrderTotalPrice() getVouchers() updateOrderDetails(Map<String, String>) getOrderDetails() submitOrder(Map<String, String>)
<<interface>> CommerceServiceFactory		
getCommerceService(Resource)		

AEM Commerce – Key Classes and Interface – Product Management

Class	Purpose
PriceInfo	Provides information about prices for products and cart line items.
Product	Reflects resources that represent products and their variants. Class is adaptable from resources with sling:ResourceType commerce/components/product.

AEM Commerce – Key Interface and Classes - Product

<<interface>> Product
getPath() getPagePath() getSKU() getTitle() getTitle(String) getDescription() getDescription(String) getThumbnailUrl() getThumbnailUrl(int) getThumbnailUrl(String) getAsset() getAssets() getImage() getImages() getProperty(String, Class<T>) getProperty(String, String, Class<T>) getVariantAxes() axisIsVariant(String) getVariants() getVariants(VariantFilter) getBaseProduct() getPIMProduct() getImageUrl() getImagePath() getThumbnail()

PricelInfo
getAmount() getQuantity() getLocale() getCurrency() getFormattedString()

AEM Commerce – Key Interfaces – Search

Class	Purpose
CommerceQuery	Holds properties about commerce searches such as the search text, the page size, the page to be retrieved of the result and the sortId to sort the results with.
CommerceResult	Describes a search result and contains info about the original search query, about applied sorts and about the current/total pages of the search

AEM Commerce – Key Interfaces – Search - UML

<<interface>> CommerceQuery
Builder
getQueryText()
getPage()
getPageSize()
getSortId()
getFacets()

<<interface>> CommerceResult
getPaginationInfo()
getSorts()
getProducts()
getOriginalQuery()
getActualQuery()
getFacets()

AEM Commerce – Key Interfaces – Vouchers and Promotions

Class	Purpose
Promotion	The Promotion interface represents a Promotion, which can be applied to certain segments and has a specific promotion type (e.g. free shipping, discount,...).
PromotionHandler	The PromotionHandler interface is implemented by services providing business logic for actual promotion. Services should define a property PromotionHandler.TYPE with a unique identifier used to reference the service from the configured promotions.
PromotionManager	An Interface that should be implemented by services, which provides access to promotions and vouchers.
Voucher	Interface that describes a voucher. The basic commerce framework requires that a voucher has at least a code and a concept of validity.
PromotionInfo	

AEM Commerce – Key Interfaces – Vouchers and Promotions - Methods

<<interface>> Voucher
<code>getCode()</code>
<code>getPath()</code>
<code>getTitle()</code>
<code>getDescription()</code>
<code>isValid(SlingHttpServletRequest request)</code>
<code>getPriority()</code>
<code>getMessage(SlingHttpServletRequest request)</code>
<code>getConfig()</code>

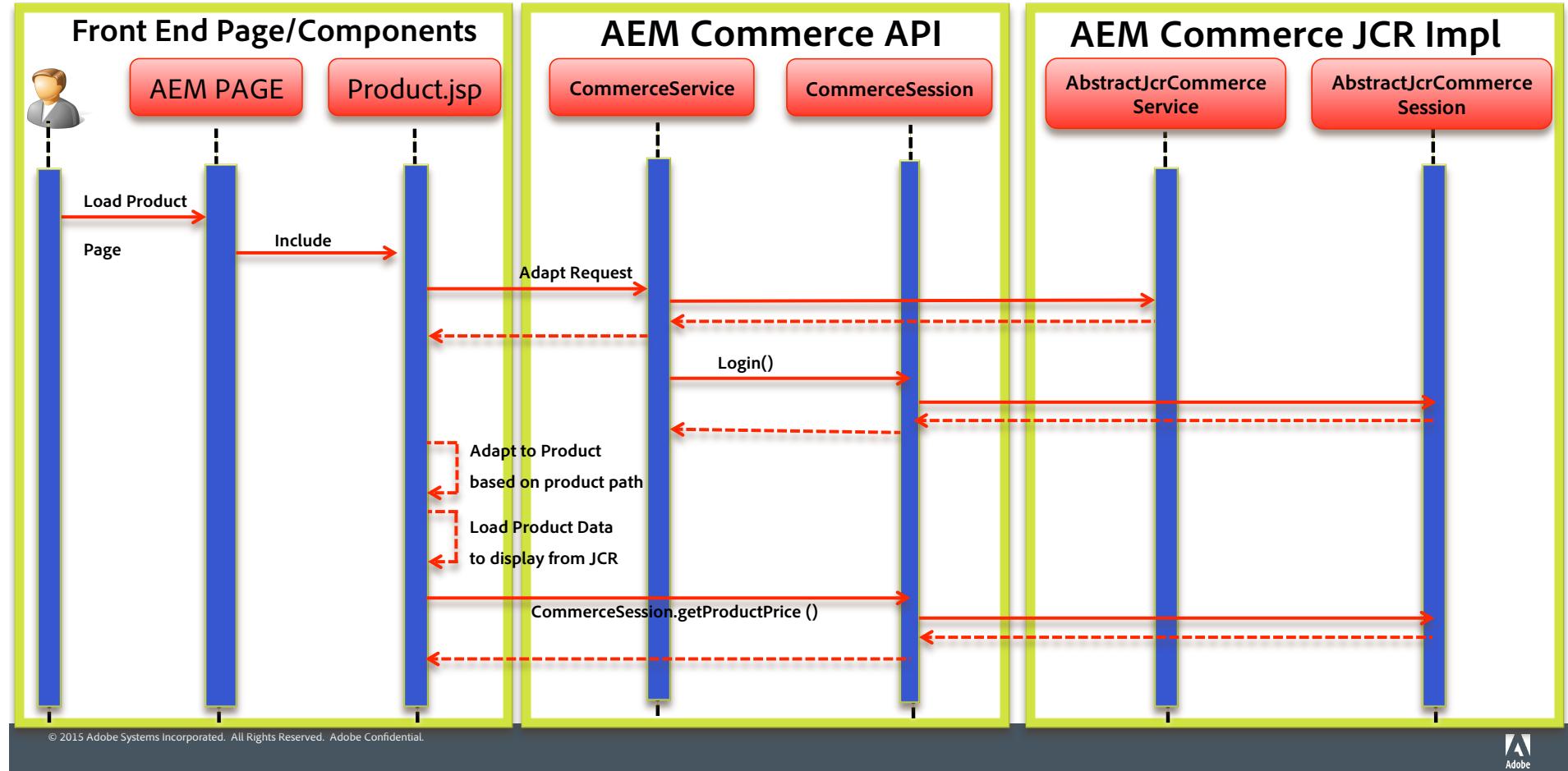
<<interface>> PromotionManager
<code>getPromotionsMap(SlingHttpServletRequest)</code>
<code>getAvailablePromotions(ResourceResolver)</code>
<code>getVoucher(SlingHttpServletRequest, String)</code>
<code>findVoucher(SlingHttpServletRequest, String)</code>
<code>getHandler(String)</code>

<<interface>> Promotion
<code>getPath()</code>
<code>getTitle()</code>
<code>getDescription()</code>
<code>getType()</code>
<code>getPriority()</code>
<code>getSegments()</code>
<code>isValid()</code>
<code>getConfigResource()</code>
<code>getConfig()</code>

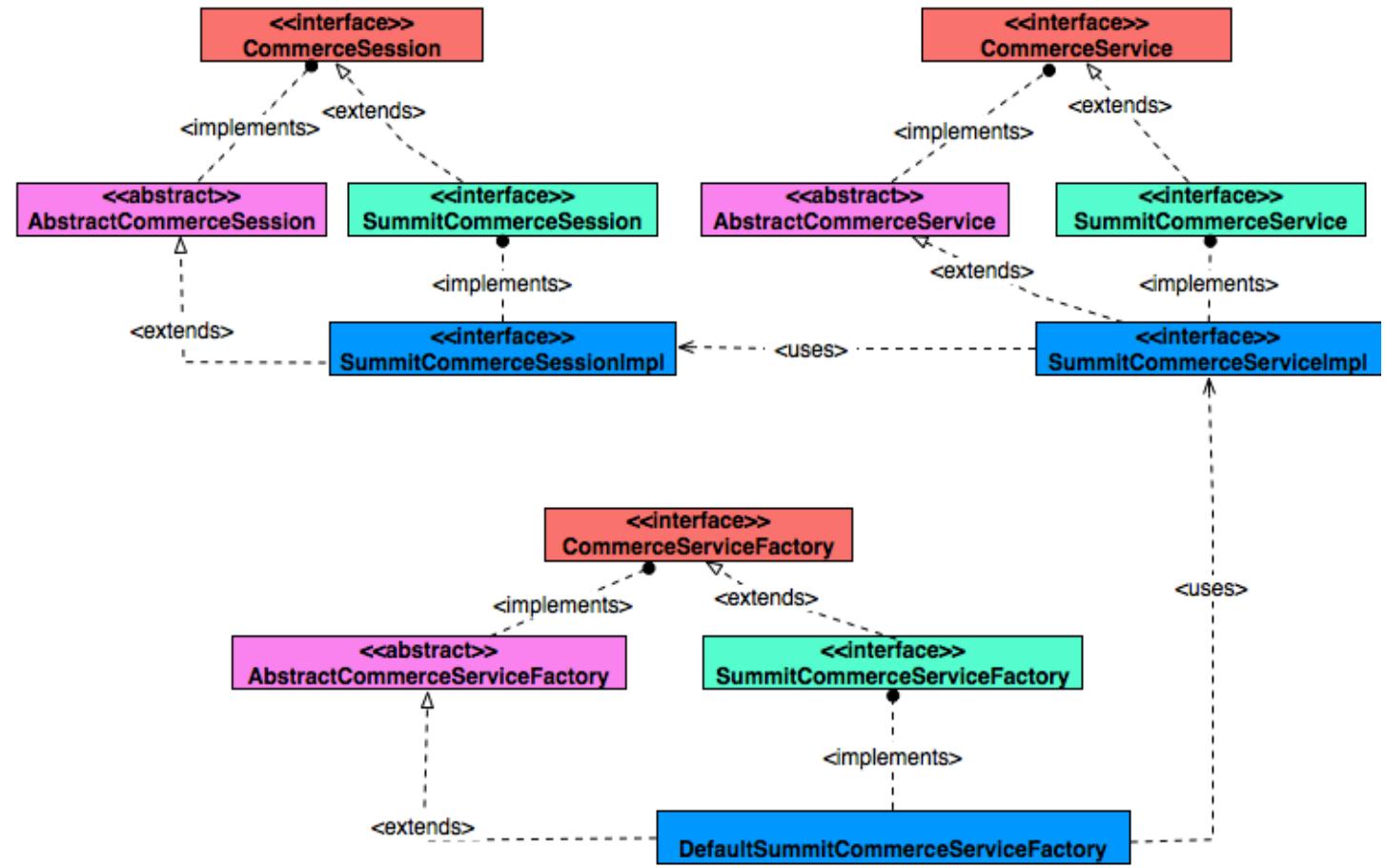
<<interface>> PromotionHandler
<code>applyCartEntryPromotion(CommerceSession, Promotion, CartEntry)</code>
<code>applyOrderPromotion(CommerceSession, Promotion)</code>
<code>applyShippingPromotion(CommerceSession, Promotion)</code>
<code>getMessage(SlingHttpServletRequest, CommerceSession, Promotion)</code>
<code>getDescription(SlingHttpServletRequest, CommerceSession, Promotion)</code>
<code>getMessages(SlingHttpServletRequest, CommerceSession, Promotion)</code>
<code>invalidateCaches()</code>

<<interface>> PromotionInfo
<code>getPath()</code>
<code>getTitle()</code>
<code>getStatus()</code>
<code>getDescription()</code>
<code>getMessage()</code>
<code>getCartEntryIndex()</code>

Adobe Commerce API – Key Classes Interaction – Product Page



Ex 3 - Summary



Exercise 3 – Summary and Key Takeaway

Custom Commerce Implementation based on AEM Commerce Integration framework.

Use Cases : As a user, I would like to create my own commerce implementation, so that I can use my own commerce engine or extend and manage the products from JCR.

Example :

Custom Implementations

Custom Products

Agenda

- 1 | AEM Commerce Integration framework - Overview
- 2 | Introduction to default components and features in AEM
- 3 | Introduction to AEM Commerce API - Interfaces and Classes
- 4 | Introduction to AEM ElasticPath Commerce Provider Architecture, API - Interfaces and Classes**

AEM Commerce – Reference Implementations

Implementation	Purpose	Availability
JCR	Reference implementation for JCR based commerce provider	Available through AEM Package manager
Elastic Path Commerce Implementation	Reference implementation for Elastic Path Commerce engine.	Available through AEM Package manager
Hybris Commerce Implementation	Reference Implementation for Hybris Commerce Engine	Available through AEM Package manager
Intershop Commerce Implementation	Reference implementation for Intershop Commerce Engine	Arriving soon !
ATG Commerce Implementation	Reference implementation for ATG Commerce Engine	Custom
Google Commerce Implementation	Reference implementation for Google Commerce	Arriving Soon !

AEM Commerce – Reference Implementations - Overview

- Uses AEM Commerce Integration Framework.
- Commerce components, Commerce API, and Commerce Implementation.
- Commerce capabilities and business logic powered by Commerce Engine.

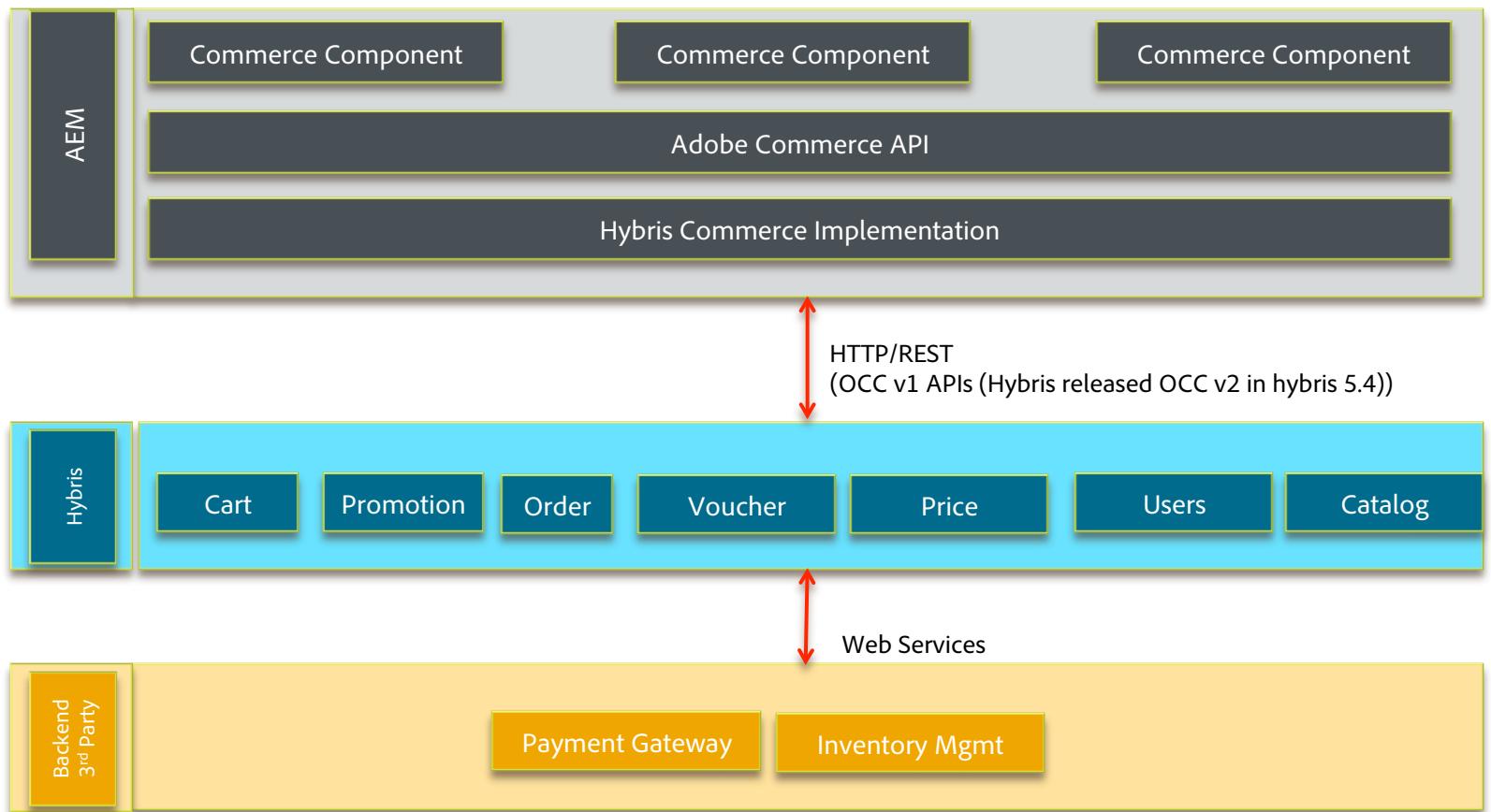
AEM Commerce – Brief Introduction to Hybris

- Uses AEM Commerce Integration Framework.
- Commerce components, Commerce API, and Commerce Implementation.
- Commerce capabilities and business logic powered by Commerce Engine.
- Synchronizes Users



(x) **hybris**

AEM - Hybris Commerce - Architecture Overview



Adobe Commerce – Hybris Commerce Reference Implementation – Key Features



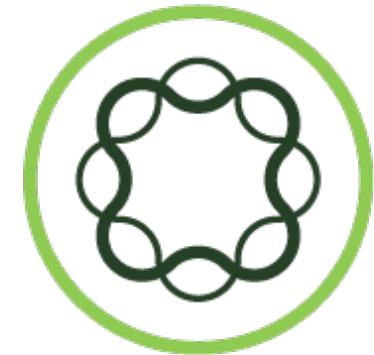
Feature	Purpose
User Integration and Synchronization	Synchronizes Users from hybris and AEM
Product import and Synchronization	Products from Hybris imported and synchronized on regular basis.
Checkout	Built on AEM and maintained from Hybris
Dynamic Price	Catered from Hybris
Cart	Dynamic Cart loaded from Hybris

Adobe Commerce – Hybris – Key Classes



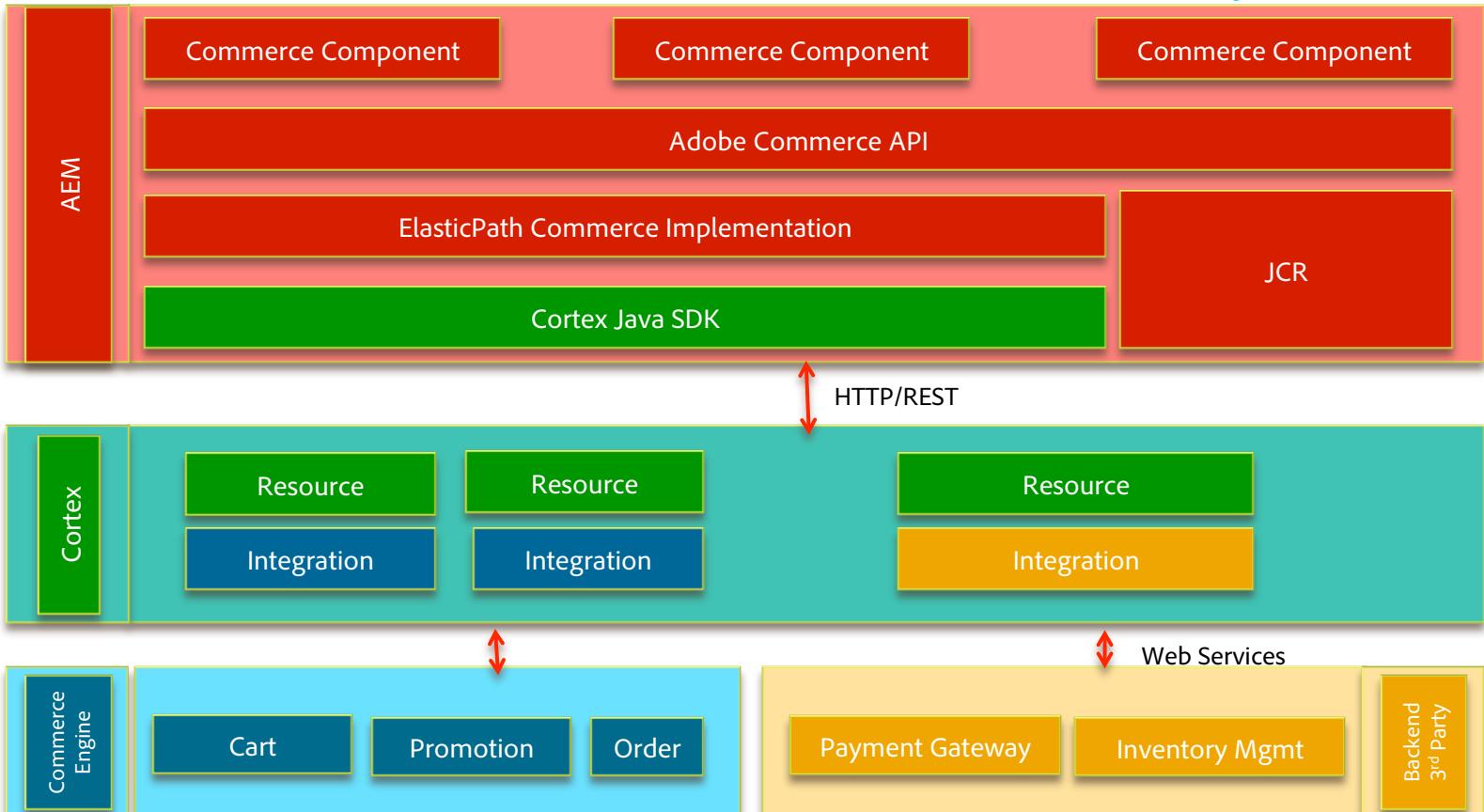
Class	Purpose
DefaultHybrisService	An extension of the core CommerceService interface to add specific Hybris-related functions.
<u>DefaultHybrisFactory</u>	An extension of the core CommerceServiceFactory interface to add specific Hybris-related APIs.
<u>DefaultHybrisSession</u>	An extension of the core CommerceSession interface to add specific Hybris-related functions.
DefaultHybrisConfigurationService	Specification of server and connection.
DefaultHybrisConnection	Handle connections with Hybris and executes Command triggered through command patterns
<u>DefaultHybrisImporter</u>	This service helps importing product catalogs.
DefaultResponseParser	Parses the response received from REST call with Hybris and returns to calling services and components.

- Uses AEM Commerce Integration Framework.
- Uses Commerce components, Commerce API, and Commerce Implementation.
- EP uses **Cortex** which is a **dynamic REST API layer** which exposes commerce capabilities
- **Cortex Java SDK** marshals and un-marshalls request between Cortex REST API and the integration.
- Commerce capabilities and business logic powered by Elastic Path Commerce Engine.
- Additional Cortex resources can be created to integrate with other backend services, and exposed in Adobe Experience Manager through Cortex REST API.



AEM - Elastic Path Commerce - Architecture Overview

elasticpath™



Level 3 REST API (HATEOAS) - API responses include links and information needed by client applications to perform key scenarios and transactions.

Concept	Definition
Resources	A service URL that provides data or performs an action
Links	Self-describing links to related data and actions in different resources
Forms	Describe information to be collected from the client
Zoom	Consolidates related link requests into one request.
Selectors	Allow for interaction by describing choices
NeedInfo	Indicates data required for workflow when it is missing



Discoverable hypermedia API - Concepts and data are “accessible,” easily related to common personal experiences of shopping online

Resource	Description
/profiles	Shopper's profile
/authentication	Login/Logout
/items	Saleable goods
/itemdefinitions	Deep definition of items
/itemselections	Item Configurations
/prices	Prices of items
/availabilities	Inventory of items
/media	Rich media of items
/searches	Product Searches
/navigation	Product Navigation
/slots	Personalized Content
/recommendations	Recommended items

Resource	Description
/carts	Shopper's basket of items
/coupons	Promotional coupon codes
/discounts	Promotional discount amounts
/promotions	Shopper incentives
/orders	Shopper's orders
/shipments	Shipments for a given purchase
/deliverymethods	Shipping options
/paymentmethods	Payment Methods
/taxes	Tax calculations
/totals	Cart calculations
/purchases	Shopper's purchases
/geographies	Valid countries and regions

AEM - Elastic Path Commerce - Key Classes



Class	Purpose
CortexContext	Encapsulates Cortex's authentication context. To use Cortex resources, you must have a context. Cortex allows both anonymous and registered shoppers !
AemCortexContextManager	Manages the persistence and retrieval of contexts (using Cookies in AEM).
CortexClientFactory	Provides access to Cortex Java SDK and the CortexClient facade for user-scoped Cortex calls. Most interaction with Cortex will be through a CortexClient instance.
JAX-RS Client	Provides direct connections to other HTTP services and/or Cortex for cases that do not require a user-scoped connection.
CortexSdkServiceFactory	Provides access to services that manage Cortex

AEM - Elastic Path Commerce - Key Classes



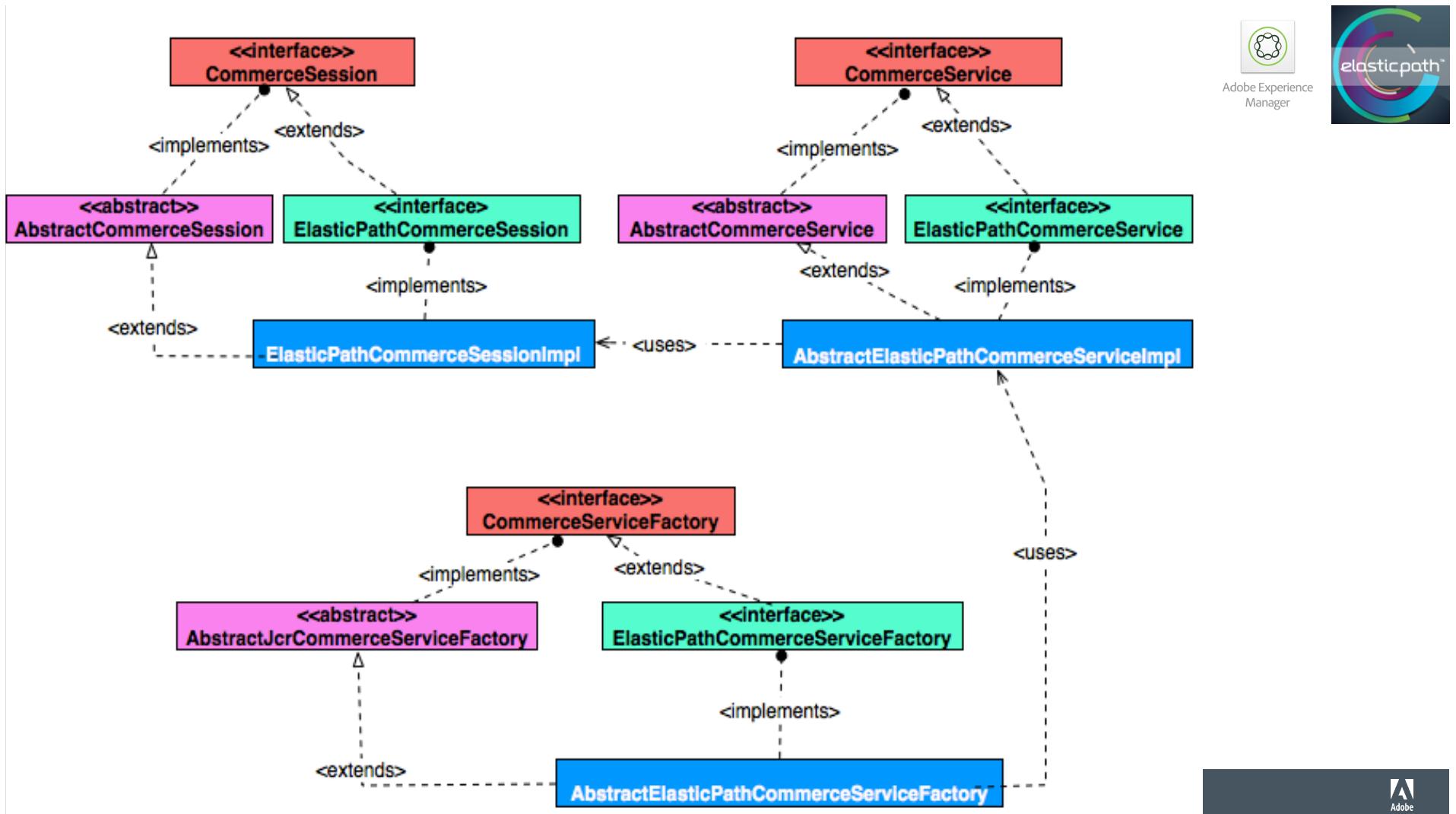
Class	Purpose
ElasticPathCommerceService	An extension of the core CommerceService interface to add specific ElasticPath-related functions.
ElasticPathCommerceServiceFactory	An extension of the core CommerceServiceFactory interface to add specific ElasticPath-related APIs.
ElasticPathCommerceSession	An extension of the core CommerceSession interface to add specific ElasticPath-related functions.

https://cms-developers.elasticpath.com/6%20v1.2.0.PREV/Configuration/EP_AEM_OSGi_Configuration

AEM Commerce – Elastic Path Key Configurations



Elastic Path ActiveMQ Connection Configuration	-	
Elastic Path Cortex Client Factory	-	
Elastic Path Cortex Context Manager	-	
Elastic Path Demo Password Configuration	-	
Elastic Path JAX-RS Client Apache Connector	-	
Elastic Path JAX-RS Configured Client Provider	-	
Elastic Path Product Importer Configuration	-	



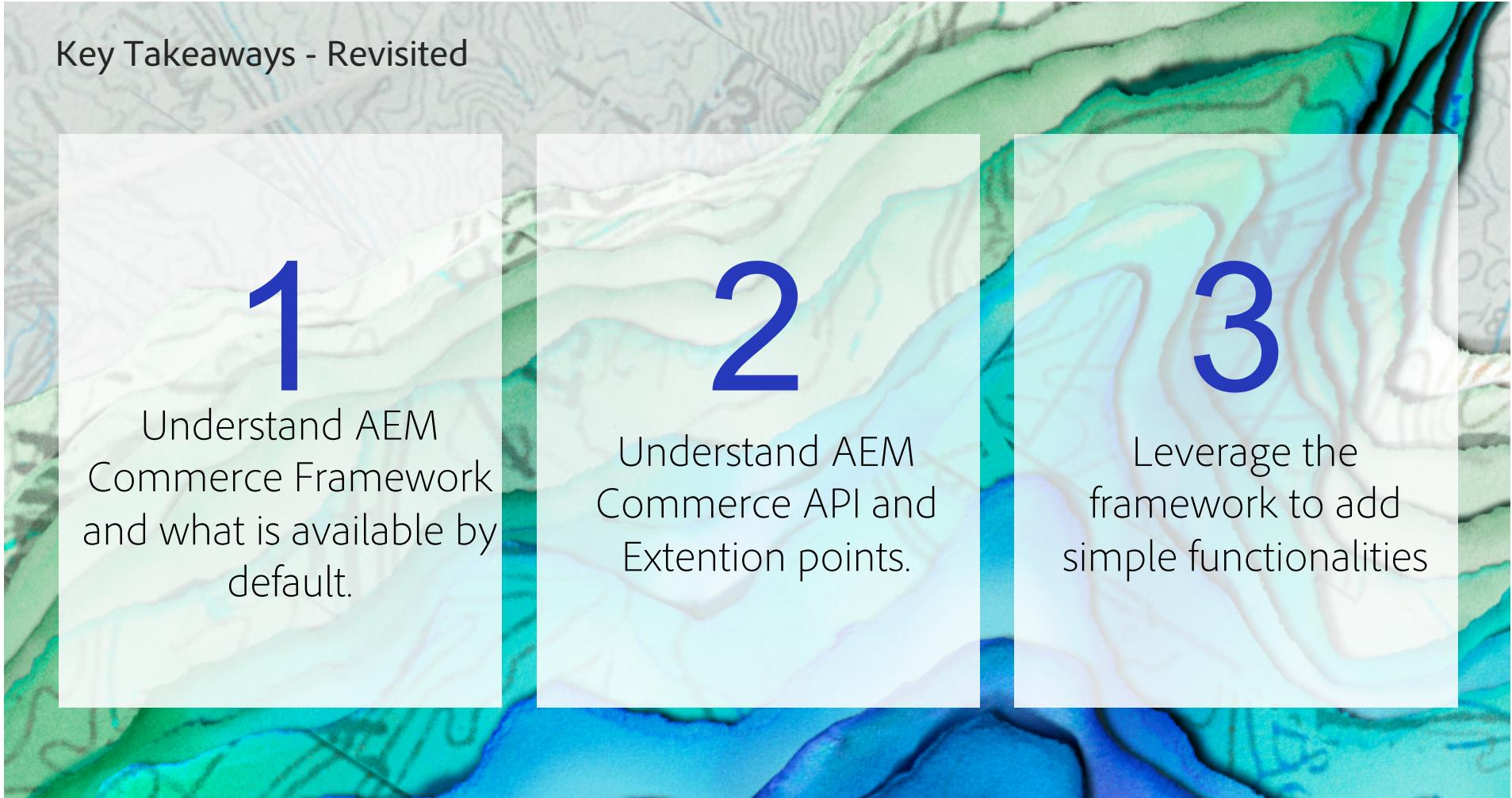
Exercise 4 – Summary and Key Takeaway

Abstract implementations for Hybris or Elastic Path in AEM Commerce Framework is a scalable framework for extension based on client use cases.

Use Cases : As a user, I would like to extend my commerce framework for my commerce servers provided by AEM such as Hybris or Elastic Path to create customizations, so I can address my customer requirements.

Example :

1. Create additional attributes in my commerce service for products and make sure this appears in AEM.
2. Customize cart data received from my commerce service to appear in AEM.



Key Takeaways - Revisited

1

Understand AEM Commerce Framework and what is available by default.

2

Understand AEM Commerce API and Extension points.

3

Leverage the framework to add simple functionalities

References

1. <https://docs.adobe.com/docs/en/cq/current/ecommerce/eCommerce-framework.html>
2. http://docs.adobe.com/docs/en/cq/current/developing/java_api_reference.html
3. <https://docs.adobe.com/docs/en/cq/current/ecommerce/eCommerce-hybris.html>
4. <https://docs.adobe.com/docs/en/cq/current/ecommerce/eCommerce-intershop.html>
5. <https://docs.adobe.com/docs/en/cq/current/ecommerce/eCommerce-native.html>
6. <http://docs.adobe.com/docs/en/cq/current/javadoc/index.html>
7. <http://www.elasticpath.com/adobe-ecommerce>
8. <https://developers.elasticpath.com/>
9. <https://github.com/Adobe-Marketing-Cloud/aem-commerce-framework-extensions>

Q&A

Varun Venkataraman

vvenkata@adobe.com

Twitter: @varunstech

Summit Twitter Tag : #AdobeSummit

LinkedIn: <http://au.linkedin.com/in/varunv1>



Take the Survey to WIN!

("surveys" section of mobile app)



\$10
Gift Card

Every Session



SKI
Swag

Thursday

