



AEM & eCommerce Integration

Lokesh BS | TechAspect Solutions

AGENDA

- eCommerce Framework
 - Integration framework (Hybris implementation)
 - Native AEM framework (JCR implementation)
- Architecture
- Commerce Components
- Commerce APIs
- Custom JCR Commerce implementation
- Demo
- Q&A

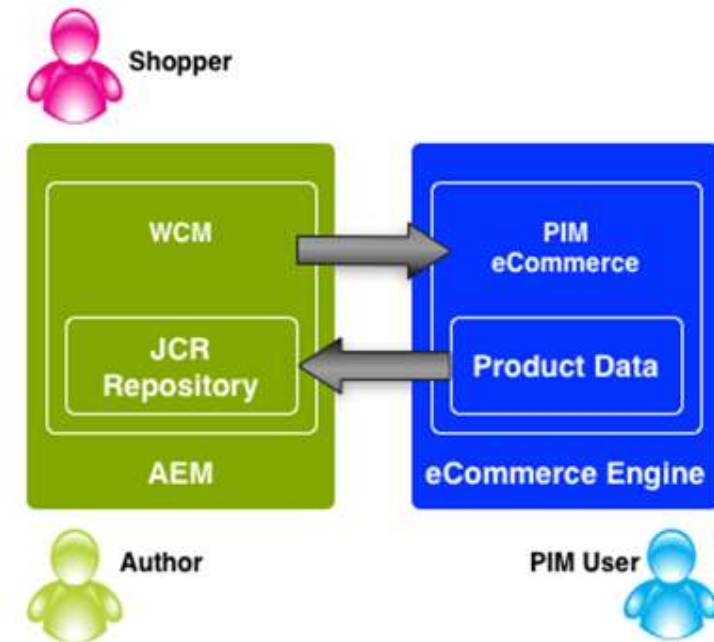
AEM – eCommerce Engine

AEM can:

- Request:
 - Product Information from the eCommerce engine.
- Provide:
 - User views for product information, shopping cart and checkout.
 - Shopping cart and checkout information to the eCommerce engine.
 - Search Engine Optimization (SEO).
 - Community functionality.

eCommerce engine can:

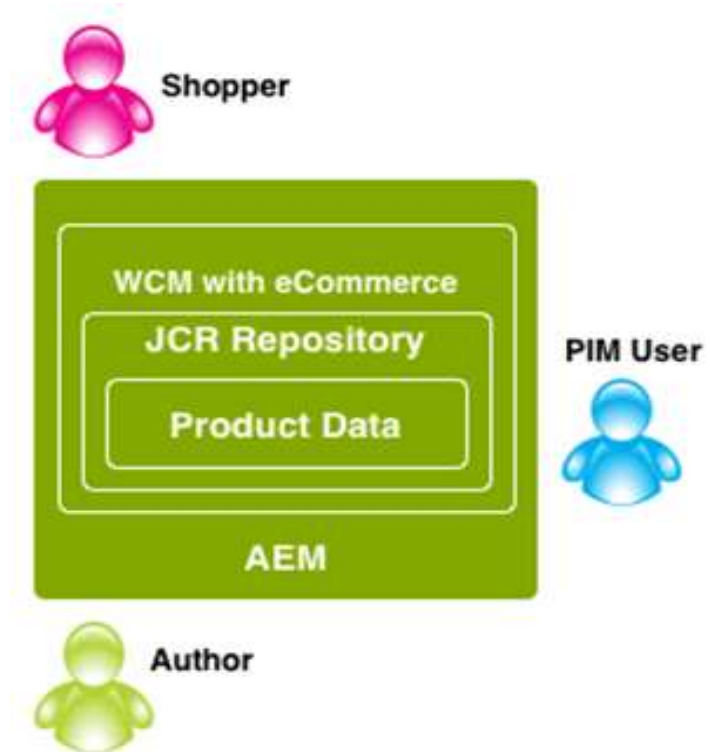
- Provide:
 - Product information from the database.
 - Product variant management.
 - Order management.
 - ERP (Enterprise Resource Planning).
 - Search within the product information.
- Process:
 - The shopping cart.
 - The checkout.
 - Order fulfillment.

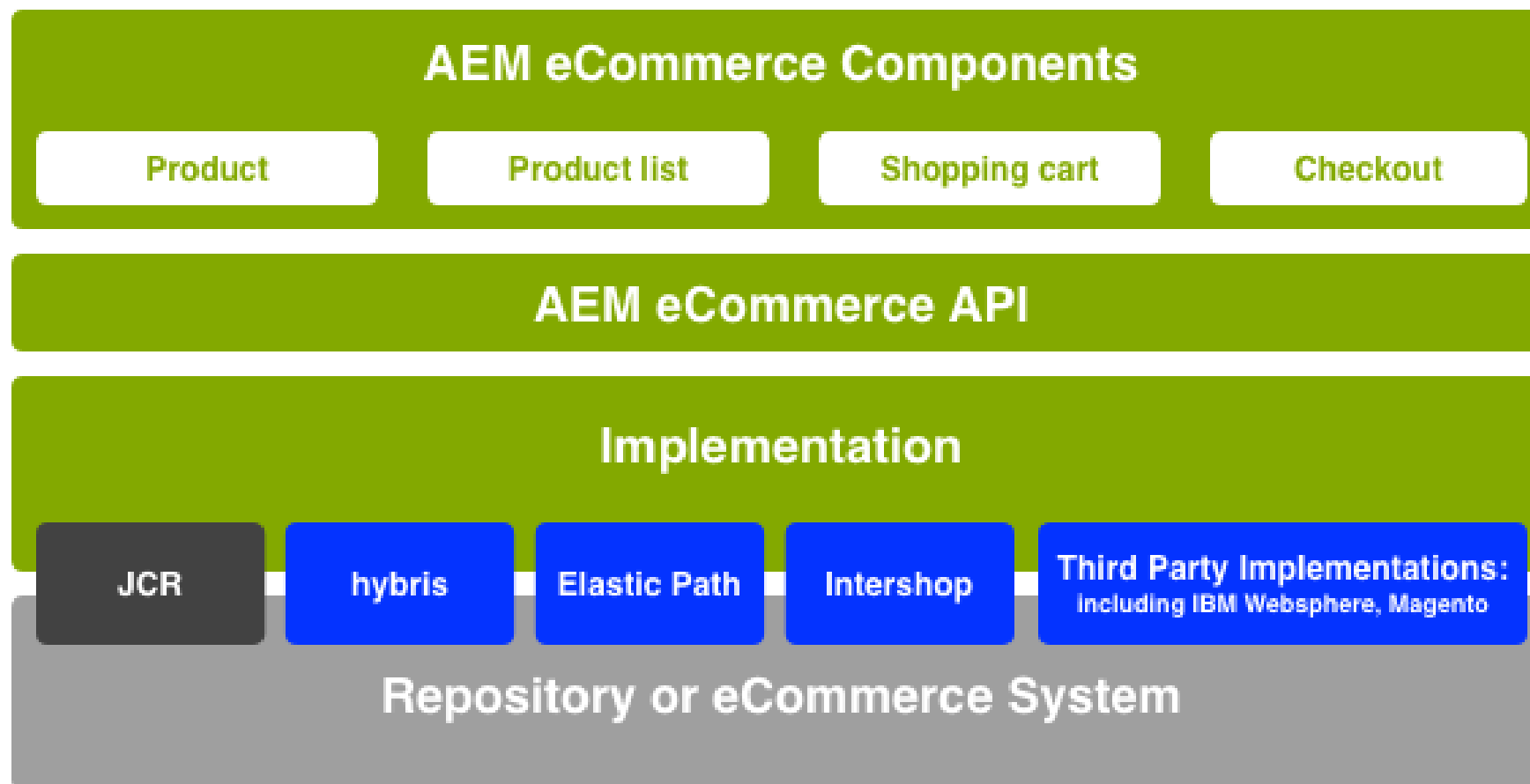


AEM-native (JCR) eCommerce Engine

AEM can:

- Provide:
 - Product information from the database.
 - Product variant management.
 - Order management.
 - Search within the product information.
 - User views for product information, shopping cart and checkout.
 - Shopping cart and checkout information.
 - Search Engine Optimization (SEO).
 - Community functionality.
- Process:
 - The shopping cart.
 - The checkout.
 - Order fulfillment.





Key Ingredients



Product

- Product Data
- Product Variant
- Product Attributes
- Product Assets



Cart

- Selected Products
- Quantity
- Product Attributes

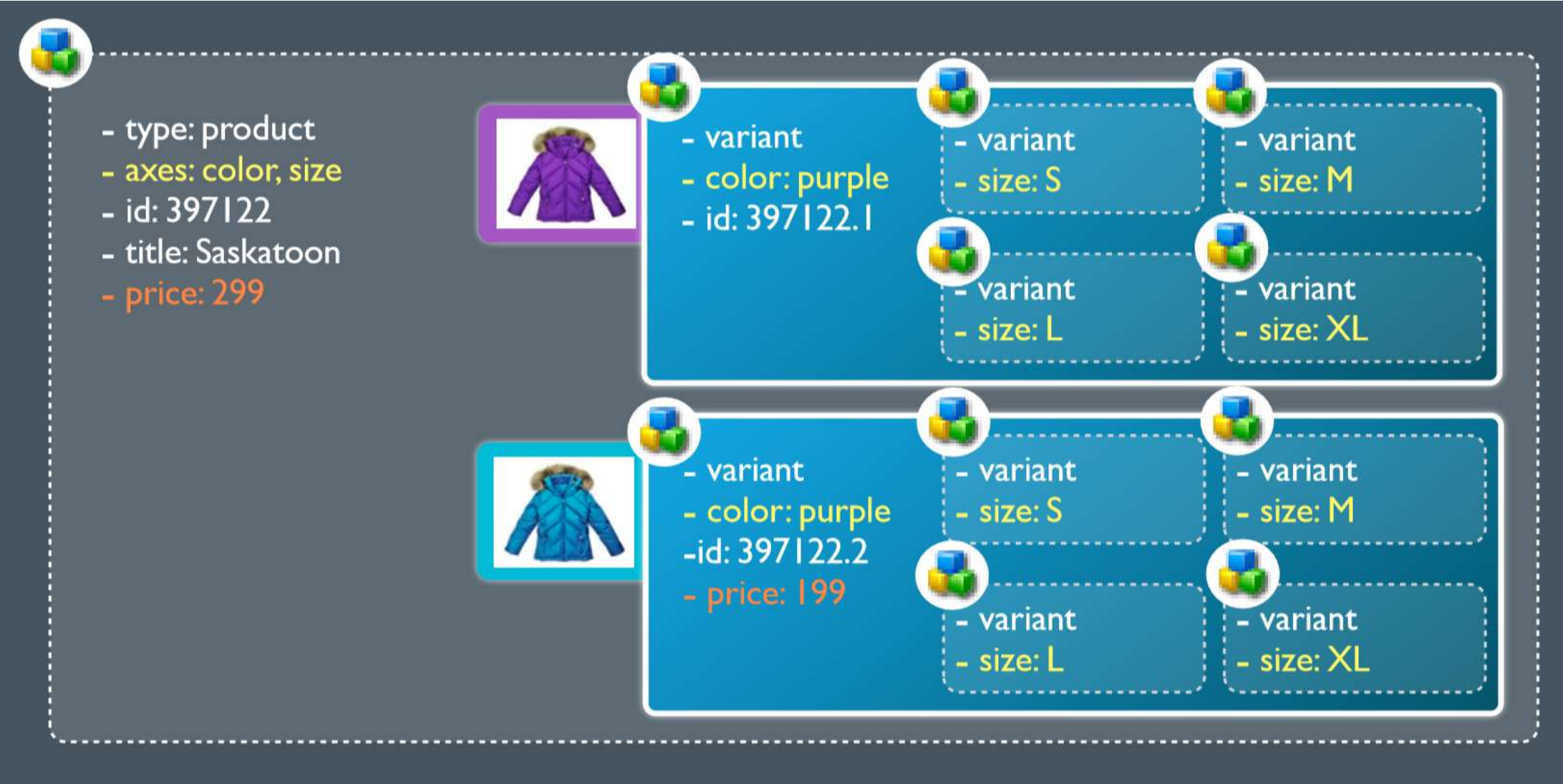


Checkout

- Shipping Method
- Payment Details
- Order Information



Products and Variants Architecture



Product References and PIM Data

Generally,

- PIM data is located under /etc
- Product references under /content.
- There must be a 1:1 map between product variations and product data nodes.

`/etc/commerce/products/geometrixx-outdoors/wm/wmap/wmapsp`



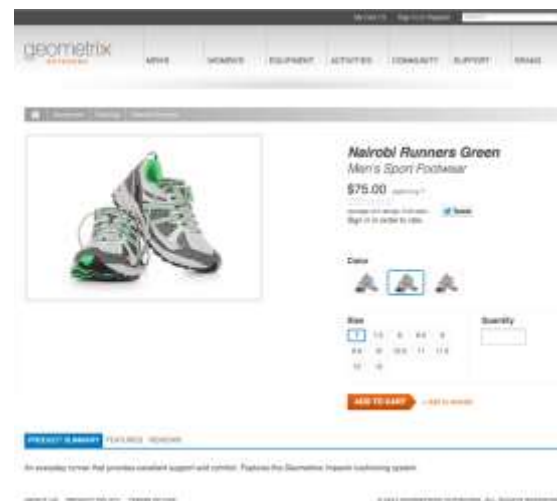
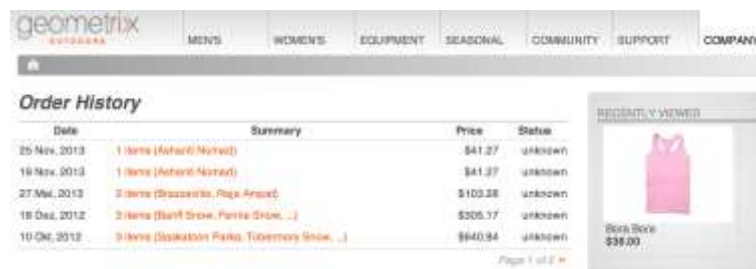
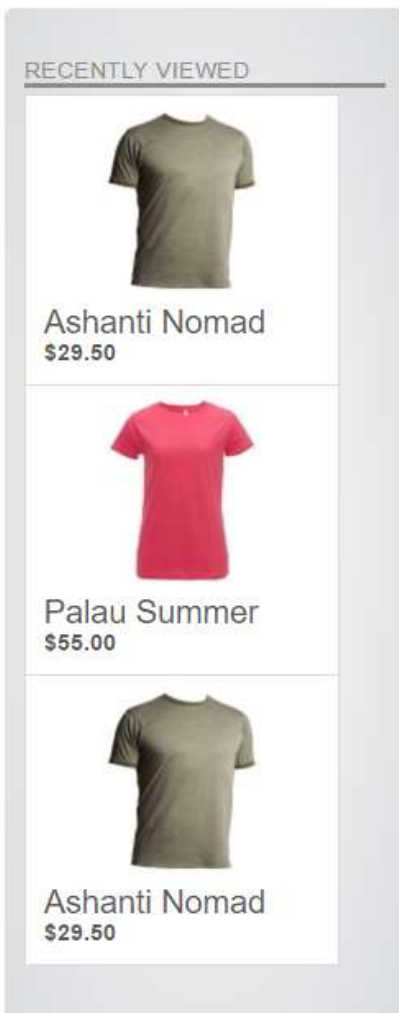
`/geometrixx-outdoors/en/women/coats/saskatoon-parka.html`



Commerce Components

A number of out-of-the-box AEM components to illustrate what can be achieved for your project:

Product
Shopping cart
Payment Methods
Recently viewed products
Order History
Search
Recommended Products
.....and others



RECOMMENDED

0 products

RECENT

Ashanti Nomad

ABANDONED

Fulani Nomad

CART

3 items, \$227.90

PROMOTIONS

1 applied

VOUCHERS

0 applied

HISTORY

1 orders, \$79.22

Messages

My Account

Sign Out

Search

ACTIVITIES

COMMUNITIES

SUPPORT

geometrix

OUTDOORS

My Cart

Edmonton Winter

1

\$160.00

Bora Bora

1

\$38.00

Maui Marine

1

\$29.90

\$227.90

Ashley Thompson's friends and followers (social graph):

Chrome

0:0:0:0:0:1

Windows 10

No keywords

1920x1080

Desktop

Chrome

777

731

US

UNITED STATES

NY

40.714353

-74.005973

Active campaign:

(simulation)

Active experience:

(simulation)

My Cart

3 items in shopping cart.

Product	Quantity	Price
<div>Edmonton Winter</div> <div>Men's Winter Coat</div> <div>Size: S</div> <div>Add gift wrapping...</div>	1	\$160.00
<div>Bora Bora</div> <div>Women's Fitness Top</div> <div>Size: XS</div> <div>Add gift wrapping...</div>	1	\$38.00
<div>Maui Marine</div> <div>Women's Cotton Top</div> <div>Size: XS</div> <div>Add gift wrapping...</div>	1	\$29.90

Commerce APIs

- CommerceService
- CommerceSession
- Product
- CommerceQuery
- CommerceResult
- and many more

```
public abstract interface CommerceServiceFactory {
```

```
    public static final String PROPERTY_COMMERCE_PROVIDER = "commerceProvider";  
    public static final String PROPERTY_COMMERCE_PROVIDER_LABEL = "commerceProviderLabel";  
  
    public abstract CommerceService getCommerceService(Resource paramResource);
```

Commerce
Provider

```
}
```

```
public abstract interface CommerceService {
```

```
    public abstract CommerceSession login(SlingHttpServletRequest paramSlingHttpServletRequest, SlingHttpServletResponse  
    paramSlingHttpServletResponse)  
    public abstract boolean isAvailable(String paramString);  
    public abstract Product getProduct(String paramString)
```

```
}
```

com.adobe.cq.commerce.api.CommerSession

```
public abstract interface CommerceSession {
```

```
    public abstract void addCartEntry(Product product, int quantity);  
    public abstract void modifyCartEntry(int entryNumber, int quantity);  
    public abstract void deleteCartEntry(int entryNumber);  
    public abstract int getCartEntryCount();
```

Cart Details

```
    public abstract String getCartPreTaxPrice();  
    public abstract String getCartTax();  
    public abstract String getCartTotalPrice();  
    public abstract String getOrderShipping();  
    public abstract String getOrderTotalTax();  
    public abstract String getOrderTotalPrice();
```

Price Details

```
    public abstract void updateOrderDetails(Map<String, String> orderDetails);  
    public abstract Map<String, String> getOrderDetails();  
    public abstract void submitOrder();
```

Order Details

```
}
```


public abstract interface Product extends Adaptable {

```
public abstract String getPath();           // path to specific variation
public abstract String getPagePath();       // path to presentation page for all variations
public abstract String getSKU();           // unique ID of specific variation

public abstract String getTitle();          // shortcut to getProperty(TITLE)
public abstract String getDescription();    // shortcut to getProperty(DESCRIPTION)
public abstract String getImageUrl();      // shortcut to getProperty(IMAGE_URL)
public abstract String getThumbnailUrl();  // shortcut to getProperty(THUMBNAIL_URL)
```

Product Details

```
public abstract Iterator<String> getVariantAxes();
public abstract boolean axisIsVariant(String axis);
public abstract Iterator<Product> getVariants(VariantFilter filter) throws CommerceException;
```

Variant Details

}

Custom Commerce Provider

- Define the commerce provider name
 - CommerceServiceFactory
- Implement (minimum)
 - CommerceService
 - CommerceSession
 - Product
- You can add your own
 - addCart()
 - loadCart()
 - PaymentMethods
 -

For AEM 6.2

```
<dependency>
  <groupId>com.adobe.aem</groupId>
  <artifactId>uber-jar</artifactId>
  <version>6.2.0</version>
  <scope>provided</scope>
  <classifier>apis</classifier>
</dependency>
```

DEMO



ADOBE EXPERIENCE MANAGER

Q & A

THANK YOU



Adobe Document Cloud



Adobe Creative Cloud



Adobe Marketing Cloud