

# Order Management

Smarter Commerce Proof Of Technology – February 2013

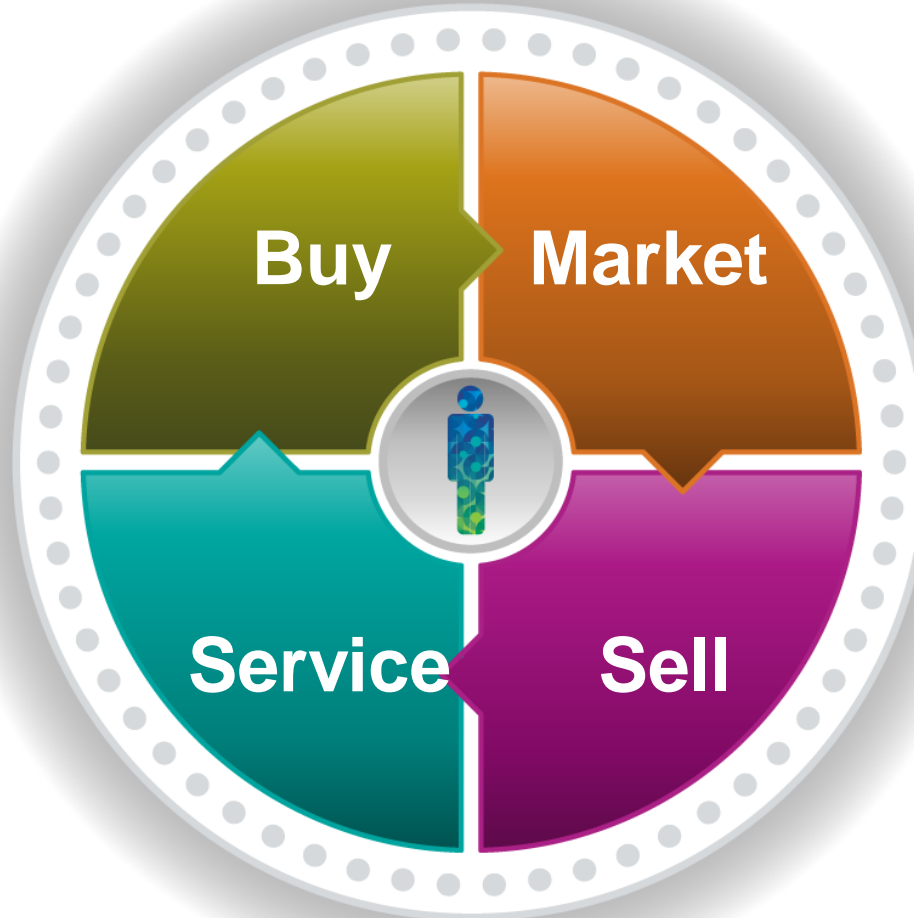


## Buy

*Adaptive procurement and optimized supply chain*

## Service

*Anticipate behavior and deliver flawless customer service*



## Market

*Targeted and personalized marketing across all channels*

## Sell

*Seamless cross-channel customer experience*

---

## Agenda

### **What is Order Management**

Technical Overview

Proof Of Technology Exercise

Retailers must now optimise their customers' brand experience to deepen the brand relationship across the shopping continuum



# Helping balance customer brand experience with operational efficiency

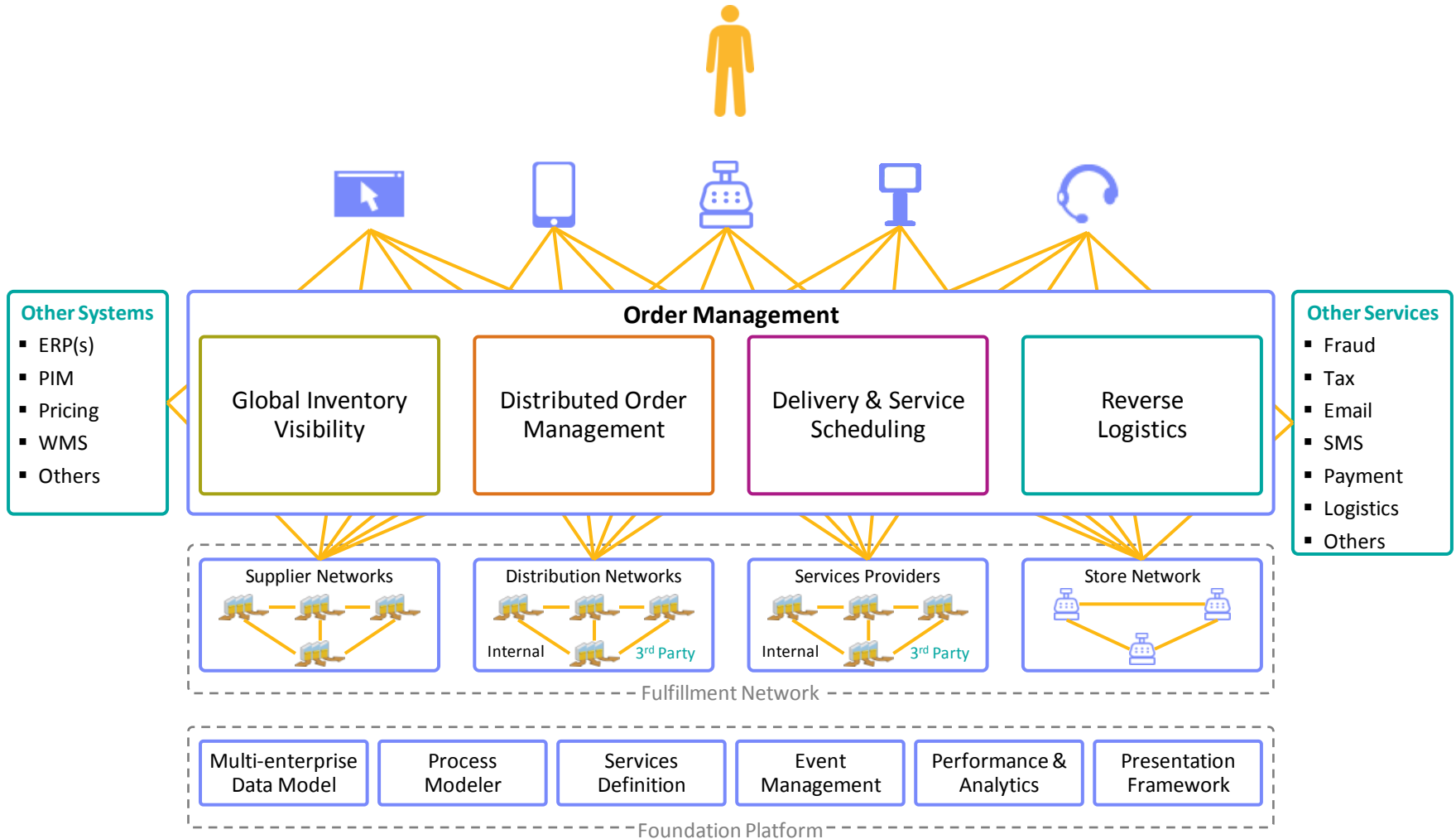
Inventory Visibility  
 Order Sourcing and Promising  
 Order Visibility  
 Dynamic Order Orchestration  
 Monitoring, Event, Action Framework



# Supply

# Demand

# IBM Sterling OM enables unified, cross-channel order management



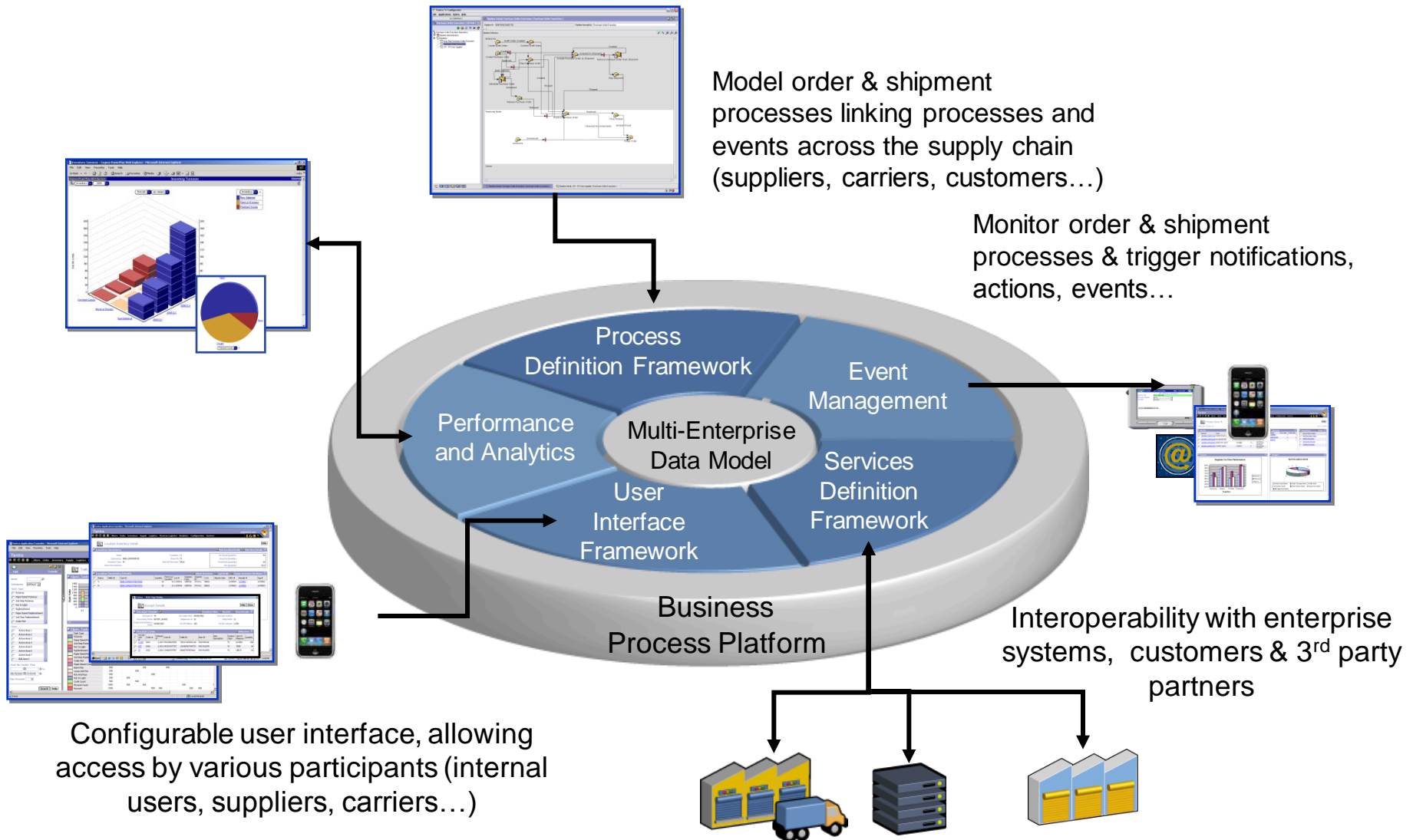
---

## Agenda

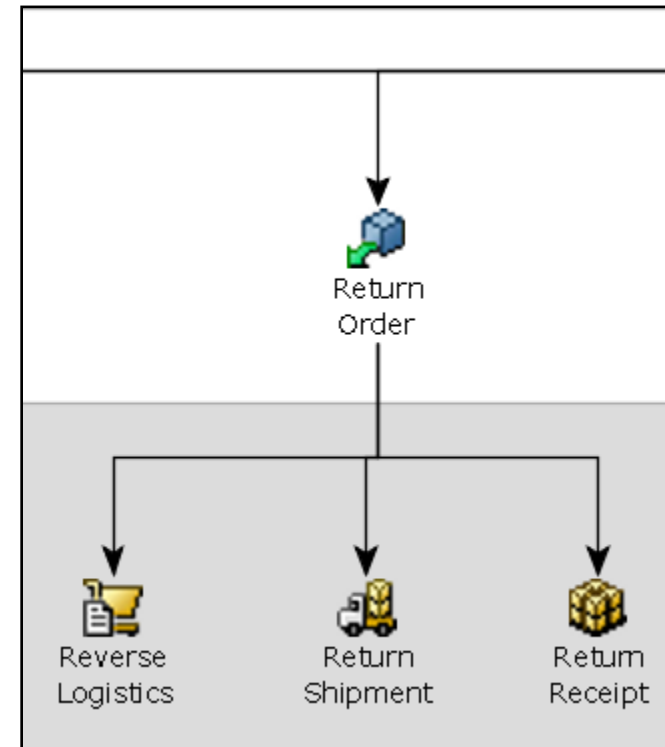
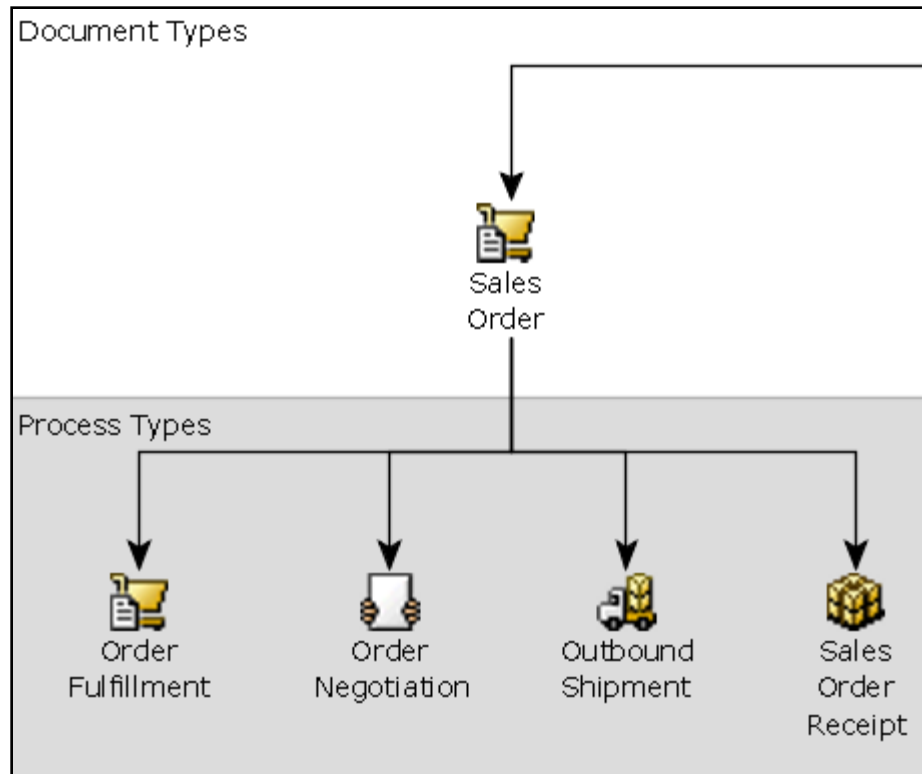
What is Order Management

**Technical Overview**

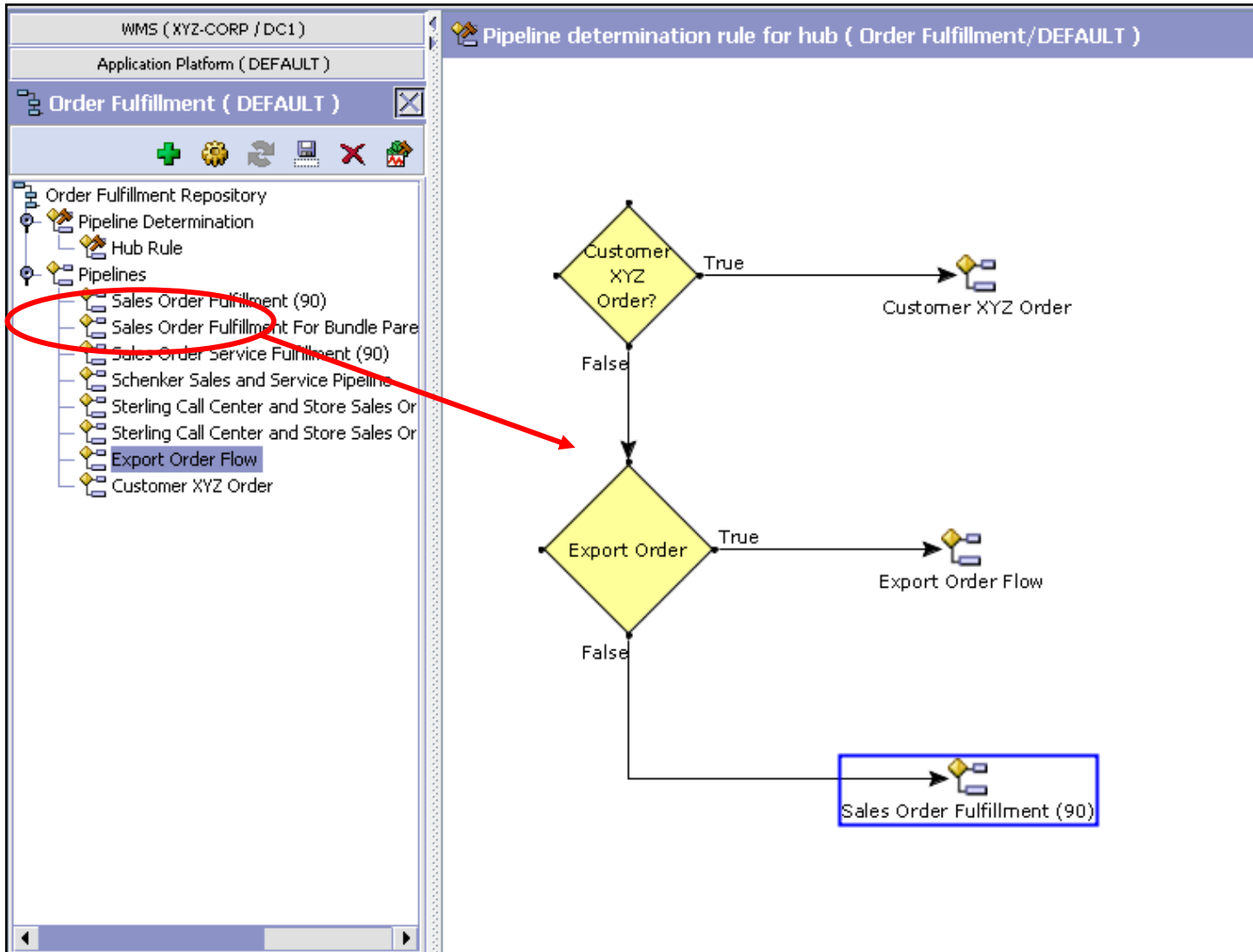
Proof Of Technology Exercise



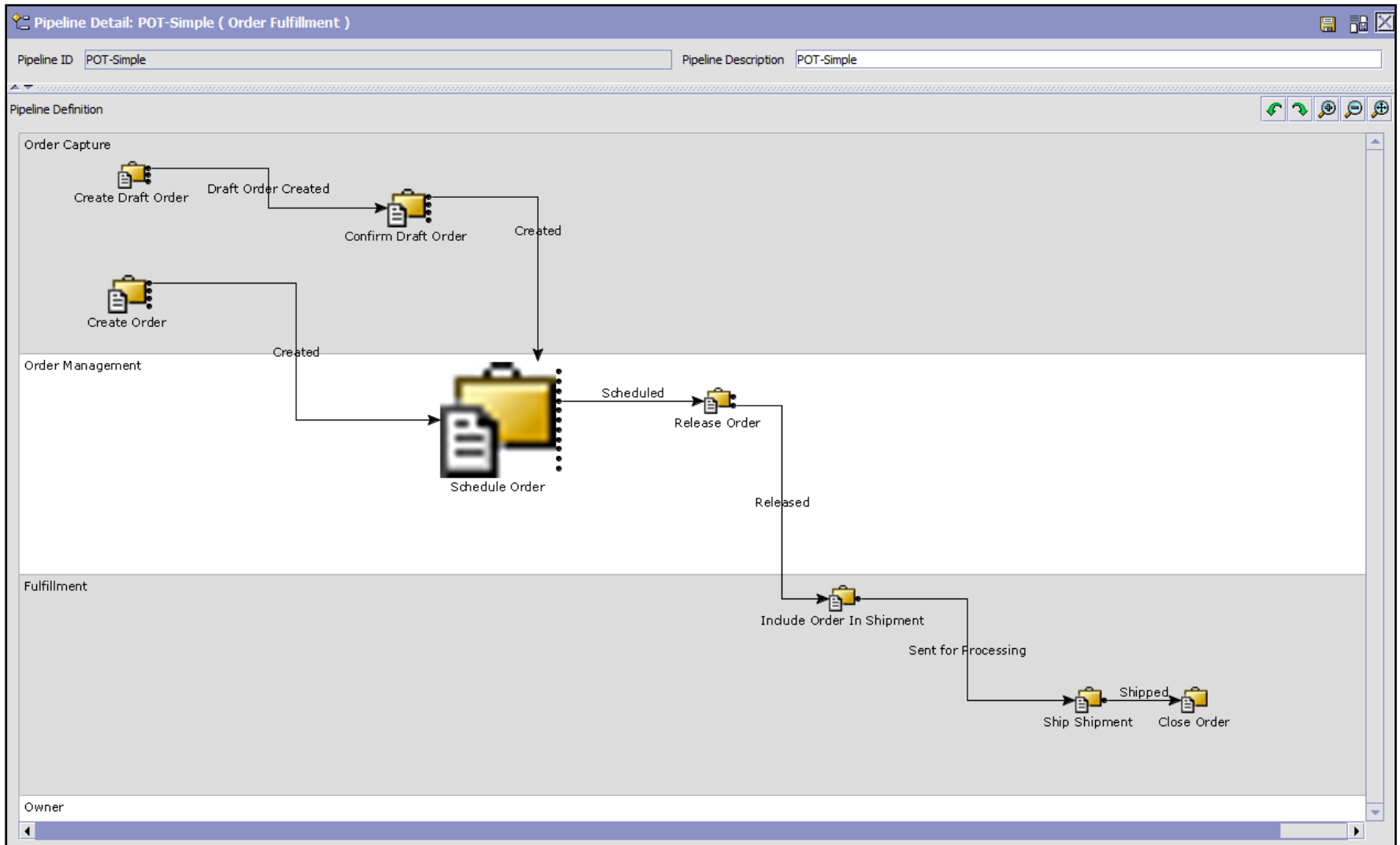




# Pipeline Determination Assigns a Pipeline To An Order

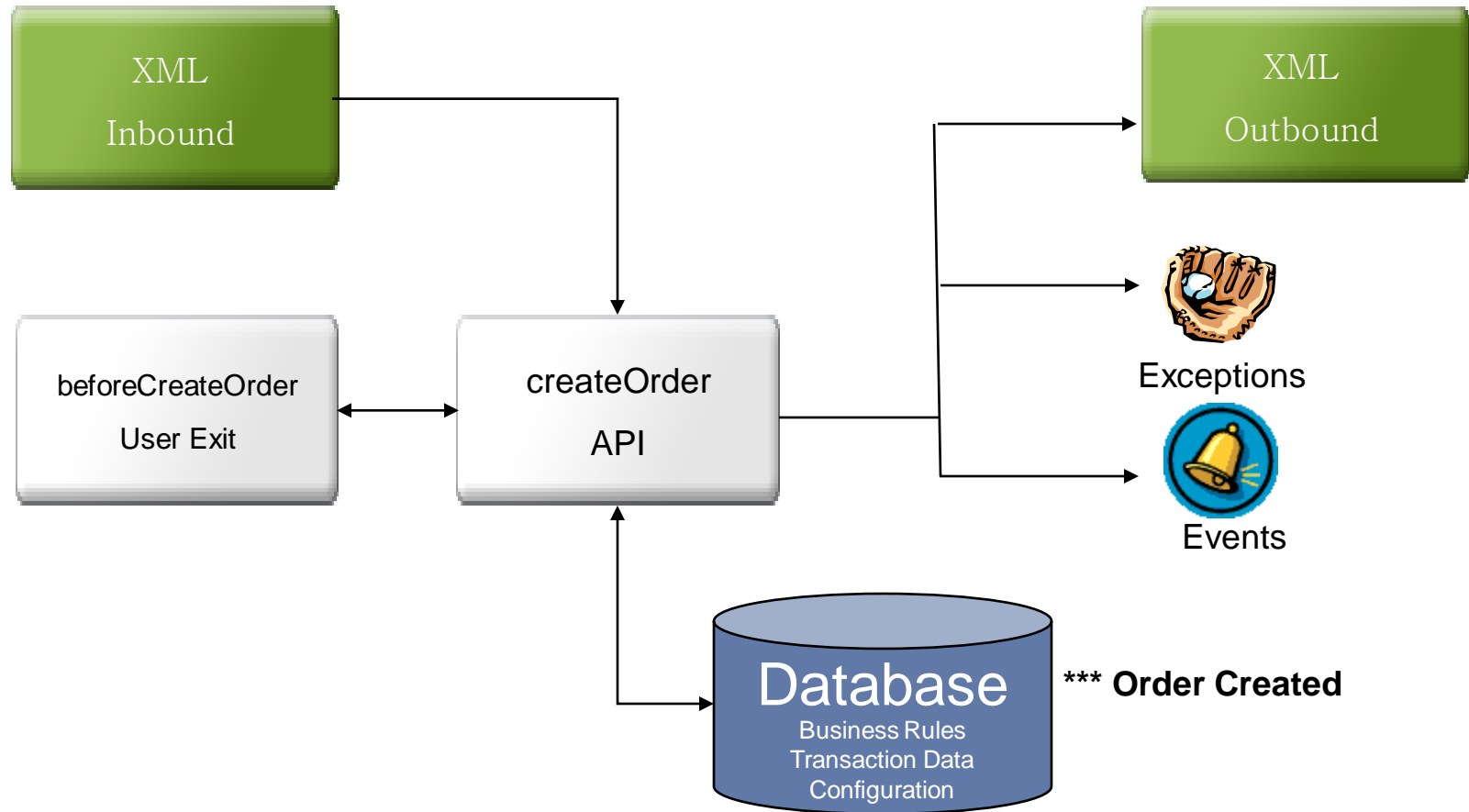


# Process Modeling Engine



# Application Level Services

## The Building Blocks of a Service Oriented Architecture



Transaction Detail: Create Order(Order Fulfillment)

Transaction ID: ORDER\_CREATE      Transaction Name: Create Order

Externally Triggered    Time Triggered    User Triggered    Others

This Transaction is triggered from external systems

---

Events    Pickup Statuses    Drop Statuses    User Exits

Events

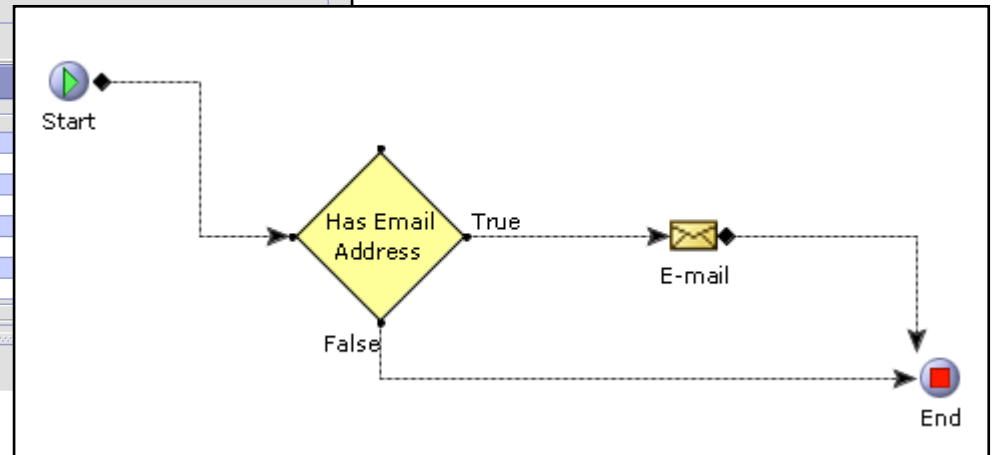
Event Id	Event Name	
BACKORDER_AT_CREATE	On Backorder	N
INCOMPLETE_PAYMENT_INFORMATION	On Incomplete Payment Information	N
ON_AWAITING_RESERVATION_ACCEPTANCE	On Awaiting Reservation Acceptance	N
ON_FAILURE	On Failure	N
ON_HOLD_TYPE_STATUS_CHANGE	On hold type status change	N
ON_ORDER_LINE_HOLD_TYPE_STATUS_CHANGE	On Order Line Hold Type Status Change	N
ON_SUCCESS	On Success	Y
ON_ZERO_UNIT_COST	On Zero Unit Cost	N

---

Event Handler Definition: ON\_SUCCESS

```

    graph LR
      A{Is a draft order} -- True --> B[DoNothing]
      A -- False --> C[Create Order]
      C -- Continue --> D[Email Order Confirmation]
      D -- Continue --> E{Store Pickup}
      E -- True --> F[Auto Schedule Release  
Create Shipment]
      E -- False --> C
  
```



---

## Agenda

What is Order Management

Technical Overview

**Proof Of Technology Exercise**

## Proof Of Technology Order Management Handout

### Activity:

*Our company, Aurora offers Click and Collect. They wish to enhance the customer experience by emailing customers when the item(s) have been picked in store and are ready for collection. Additionally, they would like the order status to update to 'Customer Collected' when the customer collects the item(s).*

- Section 1 – Summary of task and login information
- Section 2 – Create a BOPIS order in WCS and execute in OM 15-20 Minutes
- Section 3 – Create and assign 'Ready For Collection' Email 10-15 Minutes
  
- Section 4 – Extension 1 – Create and assign customer status 5 Minutes
- Section 5 – Extension 2 – Create and assign Listener 5-10 Minutes
  
- Section 6 – Create a BOPIS order again to test changes 10-15 Minutes

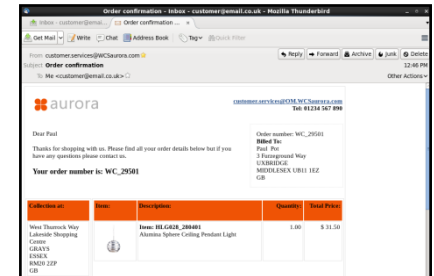
# Create a BOPIS order in WCS and execute in OM



Place Order



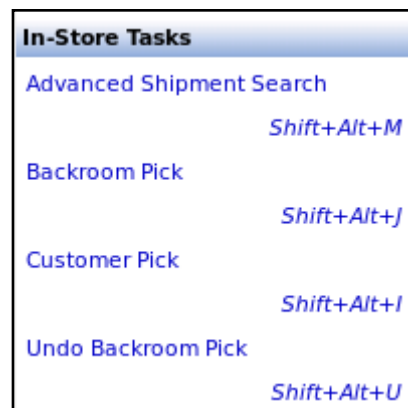
Check out



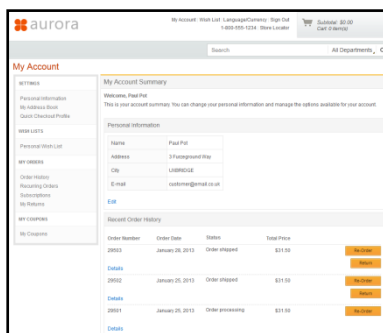
Order received



Login as a Store User



Perform pick and customer collection



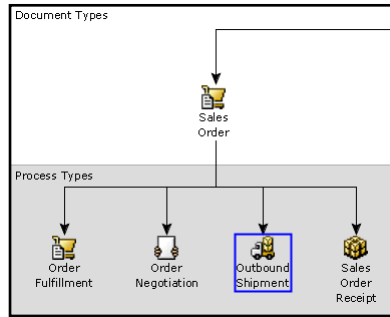
Check My Account



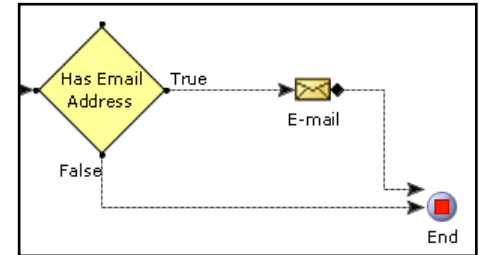
# Create and assign 'Ready For Collection' Email



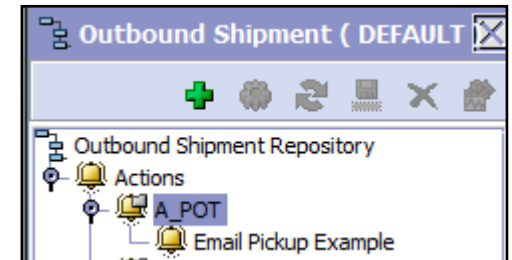
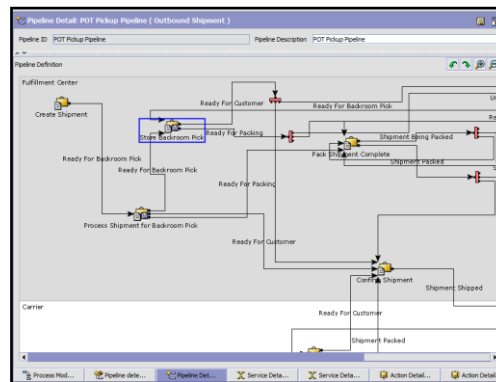
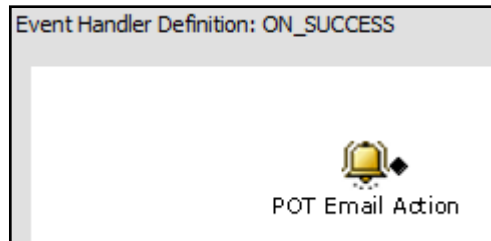
Login to OM Console



Process Modelling



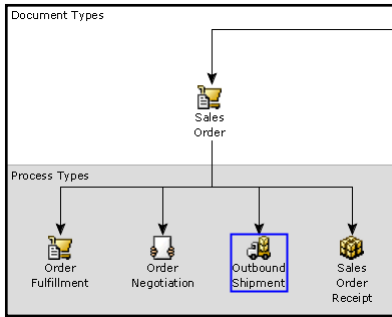
Create Email Service



Create an action

Assign action to a transaction event

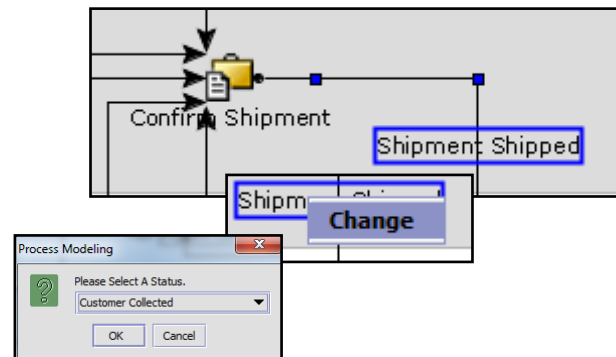
# Extension 1 – Create and assign customer status



Process Modelling

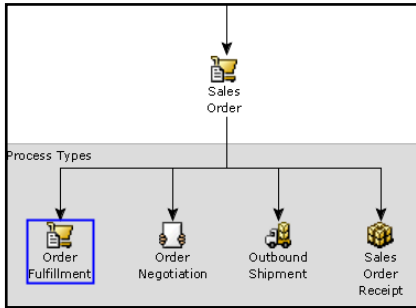


Create custom status



Change a status in our existing pipeline to the new custom status

# Extension 2 – Create and assign Listener



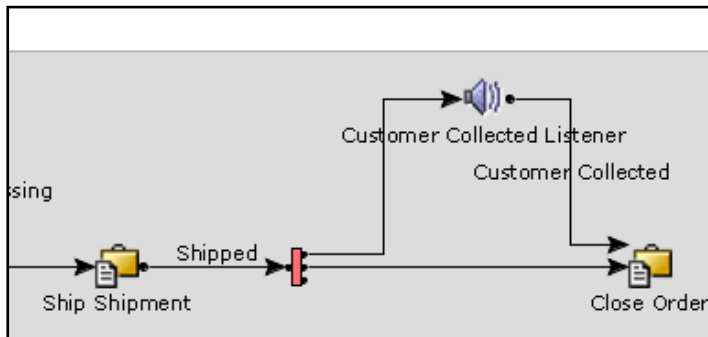
Process Modelling



Create custom status



Create a Listener

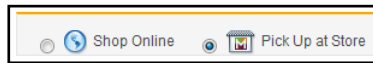


Add the listener and status to the pipeline

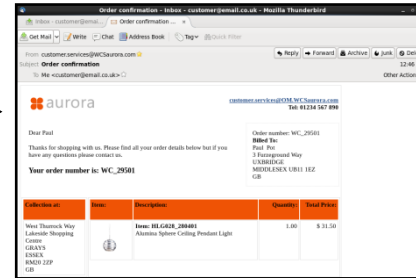
# Create a BOPIS order again to test changes



Place Order



Check out



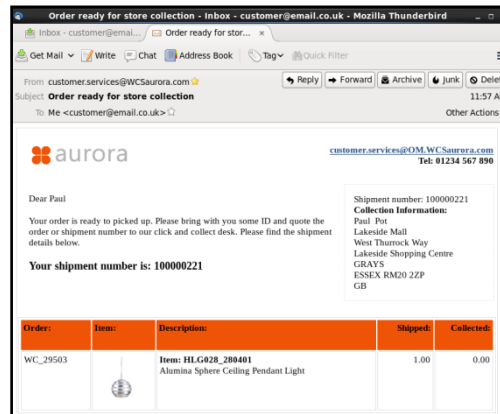
Order received



**In-Store Tasks**

- Advanced Shipment Search Shift+Alt+M
- Backroom Pick Shift+Alt+J
- Customer Pick Shift+Alt+I
- Undo Backroom Pick Shift+Alt+U

Perform the customer collection



Ready for Collection Email



**In-Store Tasks**

- Advanced Shipment Search Shift+Alt+M
- Backroom Pick Shift+Alt+J
- Customer Pick Shift+Alt+I
- Undo Backroom Pick Shift+Alt+U

Perform Backroom Pick

## Proof of Technology Summary

- Created a store pick up order in the web channel
- Integration with WebSphere Commerce
  - Inventory
  - Order
  - Order History
- Performed store picks and collections using Store
- Process Modelling
  - Flexibility (changing the process)
  - Re-use of services and actions
  - Pipeline determination
  - Linking pipelines with listeners