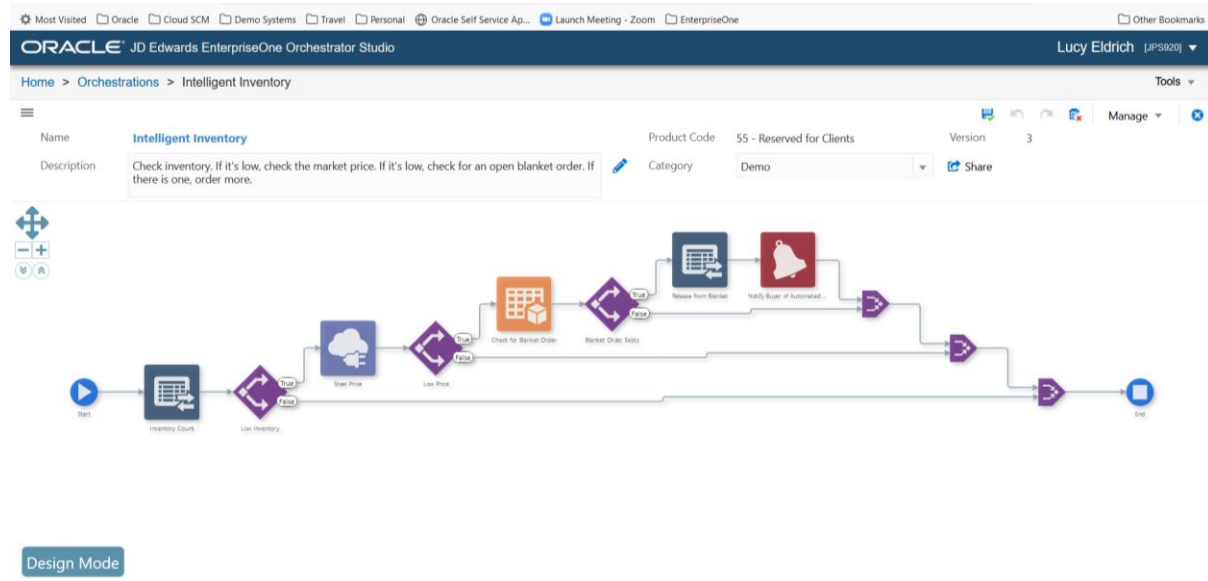


Ex 5 - Intelligent Inventory

Objective:

Create an orchestration that will check inventory. If the inventory is low, check the steel price in the market. If the steel price is low, check if there is a blanket order. If so, automatically create a purchase order for the product and notify the buyer.

Run this orchestration on a schedule, every weekday morning at 6:12 AM.



We will use item 1001 in B/P 30 for this

Current availability is below 350

We need to make some adjustments in the demo data

In P04012 Supplier Master Revisions, change the adjustment schedule to 4343 for Supplier 4343

In P4310 PO Entry, create a Blanket order OB for 10000 EA of 1001 in B/P 30 for Supplier 4343

Step 1: Create Form Request – Inventory Count

Record from P41202:

Input: item and branch/Plant

Summary Only

Return: Quantity Available

Step 2: Create Orchestration from form request: Intelligent Inventory

Under Manage

Step 3: add 2 variables to orchestration:

Inventory Threshold and Price Threshold (both numeric)

Step 4: Add Rule: Standard – Numeric

Rule Type	Value 1	Operator	Literal	Value 2	Literal Value Type
Numeric	Available	<	<input type="radio"/>	Threshold	
		=	<input type="radio"/>		

Transformation:

Input	Available Values	Default Value
Available	Available	
Threshold	Quantity Threshold	

Save Orchestration

Step 5: Create Connection for Steel Price

Home: Connections, add new

The screenshot shows the Oracle JD Edwards EnterpriseOne Orchestrator Studio interface. The top navigation bar includes the Oracle logo, the text 'JD Edwards EnterpriseOne Orchestrator Studio', and the user name 'Lucy Eldrich' with a dropdown arrow. Below the navigation bar, the breadcrumb trail reads 'Home > Connections > BLS PUBLIC DATA API'. The main content area displays the configuration for the 'BLS PUBLIC DATA API' connection. The 'Name' field is 'BLS PUBLIC DATA API' and the 'Type' is 'REST'. The 'Description' field contains 'BLS Public data API'. Below this, there are tabs for 'Service Information', 'Security', 'Client Certificate', 'Proxy', and 'HTTP Headers'. The 'Service Information' tab is active, showing the 'Endpoint' as 'https://api.bls.gov/publicAPI/v2/timeseries/data/'. Above the tabs, there are fields for 'User/Role' (JDE) and 'Environment' (JPS920).

Step 6: Add connector

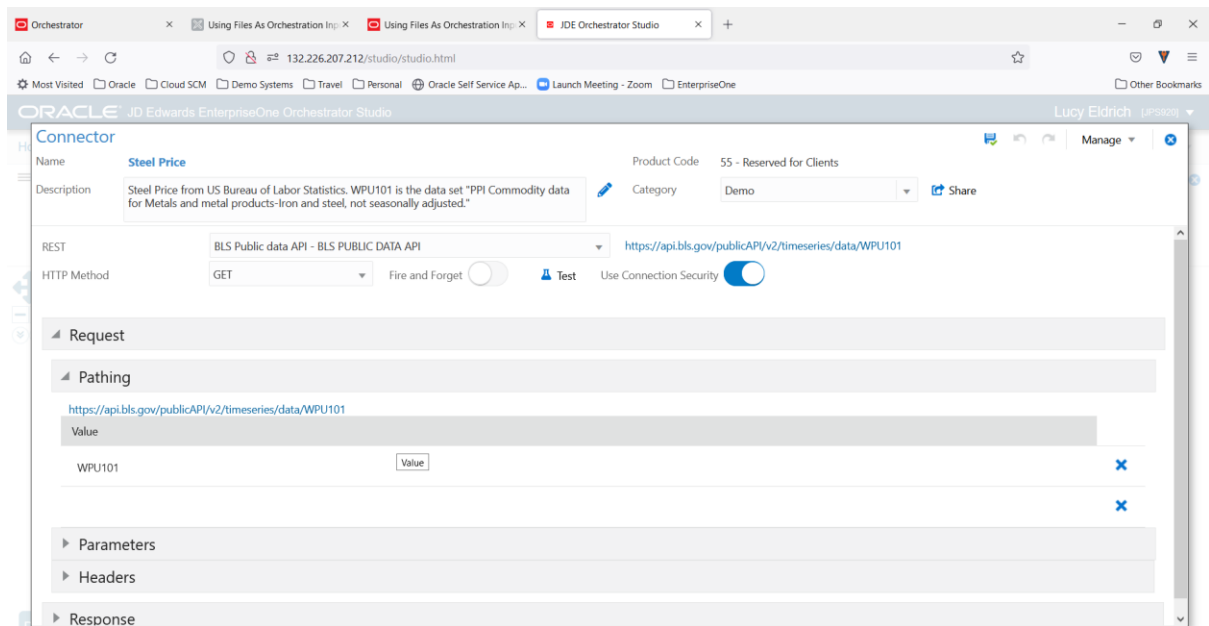
Select Intelligent Inventory Orchestration

Add True line, add Connector

New – REST – select BLS Public data API in REST Field

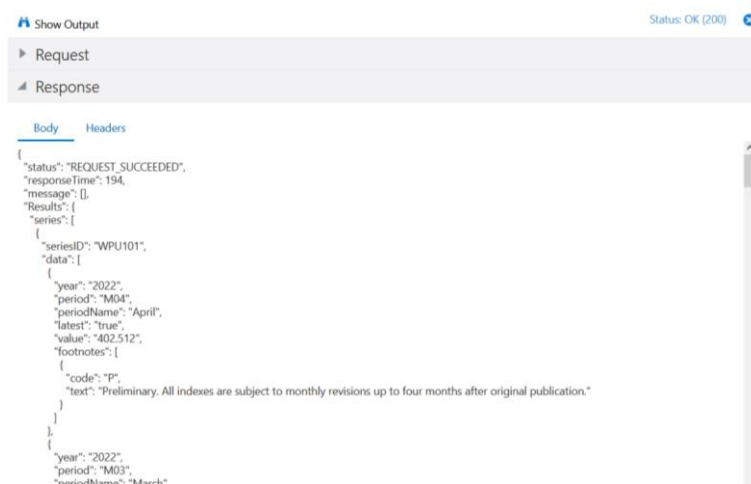
The screenshot shows the Oracle JD Edwards EnterpriseOne Orchestrator Studio interface with the 'Connector' configuration window open. The window title is 'Connector'. The 'Name' field is 'Steel Price'. The 'Description' field contains 'Steel Price from US Bureau of Labor Statistics. WPU101 is the data set "PPI Commodity data for Metals and metal products-Iron and steel, not seasonally adjusted."'. The 'Product Code' is '55 - Reserved for Clients' and the 'Category' is 'Demo'. The 'REST' field is set to 'BLS Public data API - BLS PUBLIC DATA API' and the 'Endpoint' is 'https://api.bls.gov/publicAPI/v2/timeseries/data/WPU101'. The 'HTTP Method' is 'GET'. There are checkboxes for 'Fire and Forget' (disabled) and 'Use Connection Security' (checked). Below the configuration fields, there is a 'Request' section with expandable options: 'Pathing', 'Parameters', 'Headers', and 'Response'. The 'Pathing' option is expanded, showing a text input field with the value 'WPU101'.

Expand Pathing and add value WPU101 as shown



Click Test

Result should be like this:



Open Response Section, Output

Select JSON Output

Enter first line only (data) – link to variable Market Price

Connector

Name: **Steel Price**

Description: Steel Price from US Bureau of Labor Statistics. WPU101 is the data set "PPI Commodity data for Metals and metal products-Iron and steel, not seasonally adjusted."

Product Code: 55 - Reserved for Clients

Category: Demo

Share

Headers

Manipulate Body

Output

JSON Output ☒ Output to a File ☐

JSON

Output	Variable Name	Array	Named Object
Results.series[0].data[0].value	Market Price	<input type="radio"/>	<input checked="" type="checkbox"/>
Results.series[0].data[0].year	Year	<input type="radio"/>	<input checked="" type="checkbox"/>
Results.series[0].data[0].periodName	Period	<input type="radio"/>	<input checked="" type="checkbox"/>
		<input type="radio"/>	<input checked="" type="checkbox"/>

Step 7: add rule to check steel price

Rule

Name: **Low Price**

Description: Check to see if a price is below average.

Product Code: 55 - Reserved for Clients

Category: Demo

Share

Match Type: Match All

Rule Type	Value 1	Operator	Literal	Value 2	Literal Value Type
Numeric	Price	<	<input type="radio"/>	Price Threshold	<input checked="" type="checkbox"/>
		=	<input type="radio"/>		<input checked="" type="checkbox"/>

Map variables in transformation

Step 8: Data request for blanket orders

At true line, add new data request

Use view V4311B

Load

Filter on Order Type, Branch and Item

Change values as shown:

Data Request

Name: **Check for Blanket Order**

Description: Check for Blanket Order. Inputs are Branch and Item Number. Outputs are the Supplier, Quantity Open, and Order Number.

Product Code: 55 - Reserved for Clients

Category: [Dropdown]

Table/View Name: V4311B Purchase Order Detail Update - POE

Filter: [Type to filter]

Description	Data Dictionary	Options
Order Number	DOCO	[Icons]
Or Ty	DCTO	[Icons]
Order Co	KCOO	[Icons]
Ord Suf	SFXO	[Icons]

Description	Operator	Values
Or Ty	is equal to	OB
Business Unit	is equal to	\$(Branch)
2nd Item Number	is equal to	\$(Item Number)

Select Address Number, Quantity open and Order Number as return fields and rename as shown:

Data Request

Name: **Check for Blanket Order**

Description: Check for Blanket Order. Inputs are Branch and Item Number. Outputs are the Supplier, Quantity Open, and Order Number.

Product Code: 55 - Reserved for Clients

Category: [Dropdown]

Table/View Name: V4311B Purchase Order Detail Update - POE

Filter: [Type to filter]

Description	Data Dictionary	Options
Order Number	DOCO	[Icons]
Or Ty	DCTO	[Icons]
Order Co	KCOO	[Icons]
Ord Suf	SFXO	[Icons]

Return Fields and Variable Names

Include Count: ☒ Count

Description	Variable	Associated Description
Address Number	Supplier	<input checked="" type="checkbox"/>
Quantity Open	Quantity Open	<input checked="" type="checkbox"/>
Order Number	Blanket Order Number	<input checked="" type="checkbox"/>

Make sure Include Count is selected

In Transformation, link inputs manually:

Transformations

Check for Blanket Order

Auto Map: ☒ Iterate Over: [Dropdown]

Return Raw Output: ☐ Add Inputs to Orchestration: ☒

Input	Available Values	Default Value
Branch	Branch	[Dropdown]
Item Number	Item Number	[Dropdown]

Step 9: Add rule to check blanket orders

Rule

Name: **Blanket Order Exists**

Description: Blanket Order Exists

Product Code: 55 - Reserved for Clients

Category: Demo

Match Type: Match All

Rule Type	Value 1	Operator	Literal	Value 2	Literal Value Type
Numeric	Count	>=	<input checked="" type="checkbox"/>	1	Numeric
		=	<input type="checkbox"/>		

Transformations		Blanket Order Exists
		Rule
1 Auto Map	Iterate Over	
Input	Available Values	Default Value
Count	Count	
	Count	

Save Orchestrations

Step 10: Create form request to call off blanket order

From EnterpriseOne Client, P43060, Record Process

Supplier = 4343, Branch Plant = 30, Item = 1001

Buyer = 6001

Call off qty = 100

Output = Generated order

Step 11: Add form request to Orchestration, at True branch

Remove default values from recording

Transformations: map inputs as shown

Transformations		Release from Blanket
		Form Request
1 Auto Map	Iterate Over	
		Return Raw Output <input type="radio"/>
		<input checked="" type="radio"/> Add Inputs to Orchestration
Input	Available Values	Default Value
Blanket Order Number	Blanket Order Number	
Item Number	Item Number	
Buyer	User's Address Book Number	
Quantity		2
P43060_Version		

Step 12: Add Notification to Buyer

After Form Request, add new Notification Step

ORACLE JD Edwards EnterpriseOne Orchestrator Studio

Notification

Name: **Notify Buyer of Automated PO**

Description: **Notify Buyer of Automated PO**

Product Code: **55 - Reserved for Clients**

Category: **Demo**

Type: **Simple**

Run As Subscriber: ☐ Allow Subscriber Overrides: ☐

Test

Notification Inputs

Rule

Message

Schedule

Expand Notification Inputs and add the rows as shown:

ORACLE JD Edwards EnterpriseOne Orchestrator Studio

Notification

Name: **Notify Buyer of Automated PO**

Description: **Notify Buyer of Automated PO**

Product Code: **55 - Reserved for Clients**

Category: **Demo**

Name	Value Type	Default Value	Required	
Purchase Order Number	String		<input type="checkbox"/>	X
Item	String		<input type="checkbox"/>	X
Branch	String		<input type="checkbox"/>	X
Available	String		<input type="checkbox"/>	X
Inventory Threshold	String		<input type="checkbox"/>	X
Current Price	String		<input type="checkbox"/>	X
Supplier	String		<input type="checkbox"/>	X
Price Threshold	String		<input type="checkbox"/>	X
periodName	String		<input type="checkbox"/>	X

You can skip some of these rows and simplify the notification message

Expand Message and enter as per below:

ORACLE JD Edwards EnterpriseOne Orchestrator Studio

Notification

Name: **Notify Buyer of Automated PO**

Description: **Notify Buyer of Automated PO**

Product Code: **55 - Reserved for Clients**

Category: **Demo**

Rule

Message

Output: **Watchlist**

Subject: **Purchase Order \${Purchase Order Number} Automatically Generated**

Body: **Plain Text**

Purchase Order \${Purchase Order Number} was automatically generated for Item \${Item} at Branch \${Branch}.
Inventory availability of \${Available} is below the inventory threshold of \${Inventory Threshold}.
As of \${periodName}, the market price of \${Current Price} is below the price threshold of \${Price Threshold}.
Blanket Order \${Blanket Order Number} exists with supplier \${Supplier}.

If you disable Plain Text, you can format the message

Select Actions and add as shown:

Notification

Name: Notify Buyer of Automated PO

Description: Notify Buyer of Automated PO

Product Code: 55 - Reserved for Clients

Category: Demo

Share

Actions

Application Links - 1

Work With Order Details

Application: P4310

Form: W4310G - Work With Order Details

Version:

Pre Text: Click here to

Link Text: Work With Order Details

Post Text: or to cancel it.

ID	Name	Description	Literal	Form Interconnect Value
1	OrderNumber	Document (Order No, Invoice, etc.) (DOCO)	<input type="radio"/>	Purchase Order Number
2	OrderType	Order Type (DCTO)	<input type="radio"/>	

You only need to do a form interconnect with the Order Number

Transformations: link the required input parameters (only the ones you need for the notification)

Input	Available Values	Default Value
Available	Available	
Supplier	Supplier	
Purchase Order Number	Purchase Order Number	
Item	Item Number	
Branch	Branch	
Inventory Threshold	Quantity Threshold	
Current Price	Market Price	
Price Threshold	Price Threshold	
periodName	Period	
year	Year	
Blanket Order Number		

In EnterpriseOne Client, add notification to Manage Notifications – My Subscriptions

Step 13: Add the Orchestration to a Schedule

Create a schedule. Use a cron expression. You can use www.cronmaker.com to construct a valid cron expression.

The screenshot shows the Oracle JD Edwards EnterpriseOne Orchestrator Studio interface. The top navigation bar includes 'Home > Schedules > Every Weekday Morning at 0612'. The main form has the following fields:

- Name:** Every Weekday Morning at 0612
- Description:** Every Weekday Morning at 0612
- Product Code:** 55 - Reserved for Clients
- Category:** Ex 5 - Intelligent Inventory

Below the form, there is a section for scheduling:

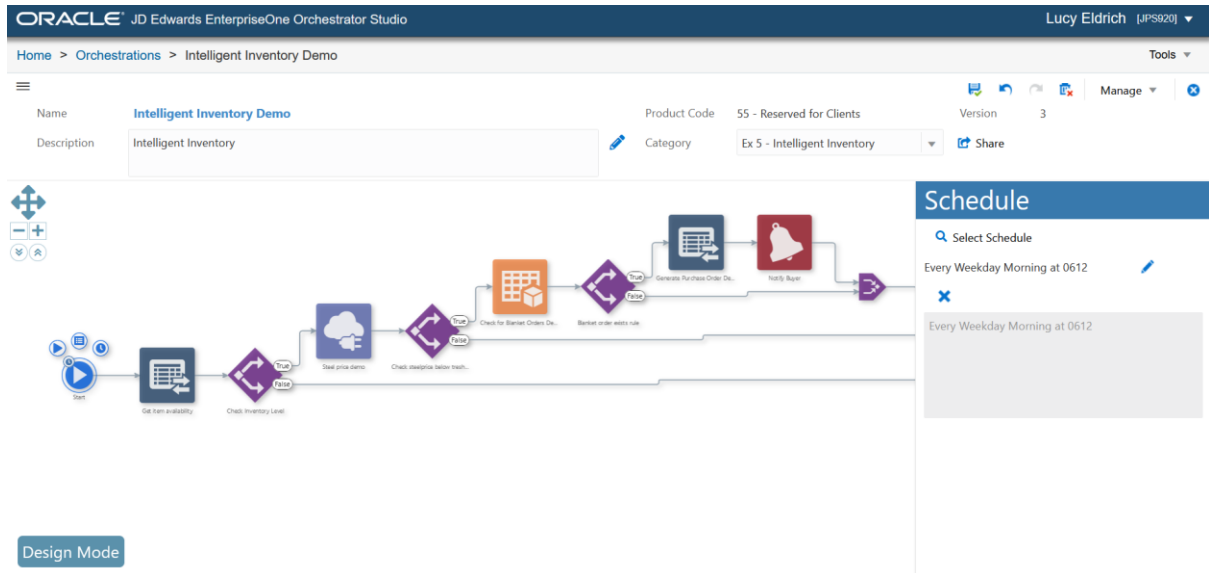
Scheduled to run every

0 Minutes

Or Enter a Cron String

0 12 6 ? * MON-FRI *

Attach the schedule to the orchestration start. Make sure to have default parameter values in the orchestration start.



The job will be visible in the Scheduler UI.

Scheduler not started

	Started	Type	Prod Cd	Notification/Orchestration Name	Autostart	Schedule Name	User	Environment	Role
<input type="checkbox"/>	<input type="checkbox"/>		55	Intelligent Inventory Demo	<input type="checkbox"/>	Every Weekday Morning at 0612	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_AmortizationSched...	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_AssetBalancesRequi...	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_CommencementJo...	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_CommencementIn JDE_NOTIFY_15_CommencementJournalNeeded	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_IncompleteLeaseTer...	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_LeasesSetToExpire	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_MovedInWithinLast...	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL
<input type="checkbox"/>	<input type="checkbox"/>		15	JDE_NOTIFY_15_NewLeasedAssets	<input type="checkbox"/>	JDE_Schedule_8_AM	JDE	JPS920	*ALL