

Automated Solution For Mass Allocation Of Inventory

Shekhar Tyagi, CPIM
Vertex Systems, Inc.

James Medeiros
Arthur Andersen

Simon Berriochoa
Havas Interactive

Introduction

Oracle's Open Inventory Interface has many powerful features, which can be utilized creatively to accomplish complex business requirements. One such example is utilizing the open interface to Mass Reserve and Unreserve Inventory for Sales Orders based on flexible priorities. This allowed *Havas Interactive* to maximize their fill rates and shorten their order cycle times.

Overview

It is a normal business practice to ship a "Complete Order" in an effort to minimize shipping and material handling costs while maximizing revenue. It is also a common business practice to fulfill a complete order in the minimum amount of time. However, these business practices are sometimes difficult to achieve because there are times when large orders cannot be shipped complete based on the first inventory allocation attempt. Orders with large quantities frequently run into the problem of only being able to partially fill order lines based on the current available on-hand inventory. Further, while larger orders are waiting for additional inventory to become available, orders with smaller inventory requirements consume the current inventory on hand making the overall task of matching available inventory with open order demand more difficult.

Standard Oracle can address inventory allocation requirements through the use of its standard reservation capabilities. This requires manually going into individual orders and using the scheduling capabilities to reserve the inventory to a single sales order. This manual practice becomes more labor intensive when you consider the process of allocating partial line quantities. Within Oracle, this requires going into individual lines on a sales order and splitting detail lines into the quantity available to reserve and the quantity to be allocated at a future time. The quantity currently available is reserved via the scheduling button. In order fulfillment environments characterized by high volume orders that contain large quantity order lines, a manual approach to inventory allocation becomes unfeasible while trying to accomplish the goal of shortening order cycle times, which is common in today's business environment.

In an effort to meet the high volume and short cycle time order fulfillment requirements at *Havas Interactive*, we automated the allocation process by developing a parameter driven custom batch processes to perform inventory allocation and un-allocation based on the features of Oracle's Open Inventory Interface. We developed custom forms and tables to store prioritization rules that drive the allocation process. Lastly, we utilized the standard Pick Release, but made sure it is run with the "Reservation Only Flag" checked to ensure that only allocated order lines are picked.

Requirements For The Allocation Process

Our client, *Havas Interactive*, is a worldwide leader in interactive entertainment and e-learning. *Havas Interactive* has a daily volume of 10,000 – 20,000 orders, with an order processing lead-time that averages one day. *Havas*

Interactive required a way to allocate and unallocate inventory based upon specific priorities such as Ship Method and Postal Zone. They also required an un-allocation process to provide control over situations where allocated inventory cannot be shipped, or when partially allocated inventory was being held for too long therefore preventing other orders from getting fulfilled. Based on their business needs, the following system objectives were defined:

- Ability to manually run on demand or schedule the Allocation Concurrent Process for a user-defined timeframe.
- The process must Allocate inventory to the eligible orders-lines, which can be completely filled based on allocation priorities.
- The process should Hold inventory for the orders lines, which can only be partially filled based on allocation priorities.
- Parameterize the allocation process to control which orders/lines should be selected for processing.
- The process must recognize orders, which are designated as “Ship Complete”. It should “Hold” these orders until they can be completely fulfilled. When enough inventory becomes available, the process must allocate inventory to these orders.
- When an order is placed on credit hold during booking, the allocation process should disregard this event and allocate the inventory regardless of the status.
- Ability to Enter/Change allocation priorities of Ship Method and Postal Zone.
- If the system date exceeds the “Cancel After Date” for an Order, the inventory should not be allocated.
- Ability to Un-Allocate/Un-Hold the inventory based on user entered parameters.
- Ability to run a custom report by summary or detail, which retrieves deficient items and their respective quantities by customer.

Components of the Solution

1. Custom Forms To Capture Allocation Priorities:
Allocation Priorities specific to *Havas Interactive* are Ship Method (Carrier) and Postal Zone. These priorities are captured using Custom Forms (Exhibit 1 & 2). The lower priority number implies higher allocation priority. If two order have same allocation priority based on ship method, the precedence is given to the order with higher priority ship-address.
2. Custom Form To Capture Customer / Item Exceptions:
A custom form (Exhibit 3) , is used to capture Customers / Items, which are to be temporarily excluded by the allocation process.
3. Custom Transaction Source Type:
A custom transaction source type, “Havas Inventory Hold” is entered on the Transaction Source Type form (Exhibit 4). This transaction source type is used by the allocation process to do partial allocation.
4. Demand Interface:
Standard *Demand Interface* from Oracle is used to inform the Inventory Module about the Open Demands in OE for different items. This step is necessary because the “Allocation Process” by design, needs a parent demand row against which it can allocate.
5. Custom Process to Allocate Inventory:
A custom PL/SQL stored procedure writes the allocation records to the Oracle Demand Interface Table. After writing records to this table, the process submits *Process Demand Interface*, which processes the written records and creates hard reservations. The OE tables are then updated with the information about the new reservations.
6. Process Demand Interface:

Oracle's standard *Process Demand Interface* processes the records written by the allocation process. The inventory gets allocated only after the completion of this process.

7. Pick Release Process:

Standard *Pick Release* process is used to Release Order Lines for picking (Exhibit 5). *CUSTOM* Library provided by Oracle is used to default the values of "Allow Partial Release Flag" as "Unchecked" and "Reservations Only Flag" as "Checked". This makes sure that only allocated order lines are released. The users still have the option of changing the values when they want to deviate from the process.

8. Custom Report:

A custom report, "Havas Mass Allocation Deficient Quantity Report" gives the Operation's user the ability to review the outstanding demands against various items. This report can be run in two modes; "Detail" and "Summary". In Detail mode the report provides the breakup of demand for each item, whereas in Summary mode it aggregates the demand per item.

9. Custom Process to Un-Allocate Inventory:

A custom PL/SQL stored procedure writes records to Oracle's Demand Interface Table, which are used to unallocate inventory to the selected Order/Lines. This process also submits "Process Demand Interface" to process the written records and updates OE tables with the information about the allocation

How Does The Process Work

1. The program looks at all eligible orders based on the following filter criteria entered by the user:

- Warehouse
- Subinventory
- Customer
- Order Number Range
- Order Type
- Item
- Order Date Range
- Schedule Date Range

2. It excludes the order lines based on User-Entered Exclusion Criteria of Customers and/or Items (Exhibit 3). This is used in situations where the user knows that a particular customer/item has outstanding orders which cannot be fulfilled currently and that may unnecessarily tie up the available inventory.

3. It sorts all the remaining order lines based on the allocation priorities and starts with the order line having the highest priority.

4. It calculates the "Available Quantity" as follows:

$$\begin{aligned} \text{Available Quantity} &= \text{Quantity On Hand} \\ &\quad - \text{Quantity Reserved In MTL_DEMAND} \\ &\quad - \text{Quantity Reserved In MTL_DEMAND_INTERFACE} \end{aligned}$$

5. If all the order lines for this particular order have "Order Quantity" <= "Available Quantity", the program reserves the items against the Sales Order Lines (Allocates) by writing a record to *MTL_DEMAND_INTERFACE*, as follows:

$$\begin{aligned} \text{Action_Code} &= 130 \text{ (Reservation Add)} \\ \text{Reservation_Type} &= 2 \text{ (Hard Reservation)} \\ \text{Demand_Source_Type} &= 2 \text{ (Sales Order)} \\ \text{Demand_Source_Header_Id} &= \text{Order Header Id} \end{aligned}$$

Demand_Source_Line = Order Line Id
Demand_Source_Delivery = Order Line Detail Id

6. If one of the lines of any order has “Order Quantity” > “Available Quantity”, the procedure Holds all the lines for this order. “Hold” is a reservation against a custom “Demand Source Type” (Exhibit 4) . This is done by writing a record to *MTL_DEMAND_INTERFACE* as follows:

Action_Code = 130 (Reservation Add)
Reservation_Type = 2 (Hard Reservation)
Demand_Source_Type = Havas Inventory Hold
Attribute1 = Order Header Id
Attribute2 = Order Line Id
Attribute3 = Order Line Detail Id

7. After going through steps 4-6 for all the selected orders, the process submits “Process Demand Interface” and waits for its completion. Then, it updates the OE tables with the allocation information.
8. The Un-Allocation Process goes through steps 1-7. The only difference being that it inserts line_item_quantity as 0 (Zero) to un-reserve the inventory.

Operational Procedure

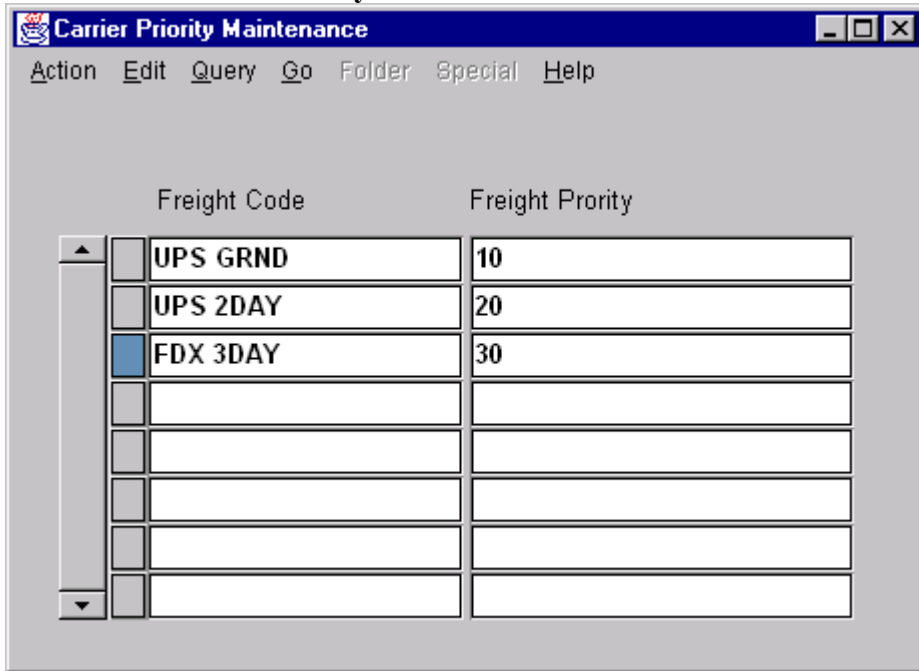
1. Run Allocation Program. Allocate all available inventory based on Havas Parameters.
2. Pick Release Orders that have all required inventory available. In the Release Sales Orders For Picking window, uncheck “*Allow Partial Release*” and check “*Reservations Only Flag*” (Exhibit 5).
3. Run “Havas Mass Allocation Deficient Quantity Report” to review” Ship Complete Customer Orders” that are cannot be fulfilled Attempt to fill in one of the following ways:
 - A. Manually unallocate/allocate inventory from other orders using standard Oracle functionality in OE.
 - B. Review current production schedule to determine whether the inventory is being built.
 - C. Choose to ship short by enabling partial shipments for the respective orders.
4. Review “Non-Ship Complete Orders”. Attempt to fill in one of the following ways:
 - A. Manually unallocate/allocate inventory from other orders using standard Oracle functionality in OE.
 - B. Review current production schedule to determine whether the inventory is being built.
 - C. Choose to ship short by enabling partial shipments for the respective orders.
5. Re-run Pick Release.
6. (If Required) Run Un-Allocation program to unallocate inventory.

Conclusion

In order environments characterized by high volumes with large quantity order lines, a Mass Allocation process can be utilized in place of Oracle’s manual inventory reservation process. At *Havas Interactive*, this modification equipped our client with an efficient tool to address their Mass Allocation functional requirements while accomplishing their business objectives of keeping fill rates high, order cycle times low, and material handling costs low.

From a technology perspective, the custom processes described in this paper utilize Oracle’s standard Open Inventory Interface feature for reserving inventory. Using the Open Interface as part of the solution, minimizes the risk of the modification being affected by future patches or upgrades.

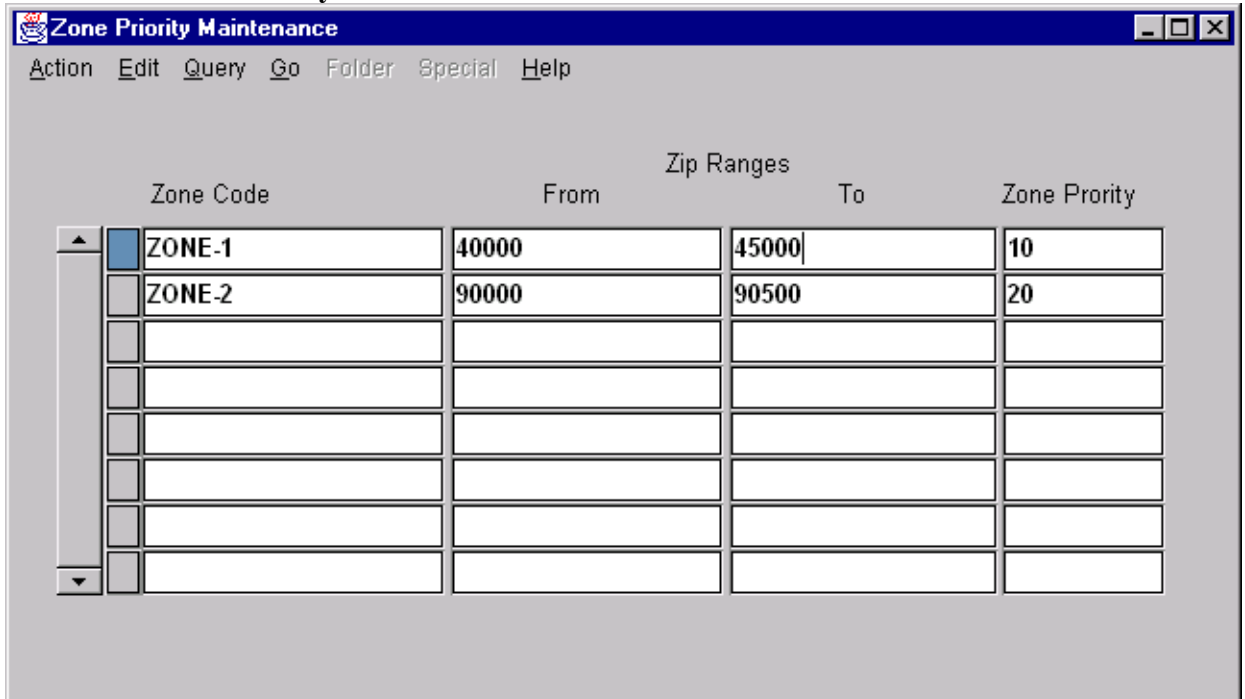
Exhibit 1: Carrier Priority Maintenance Form



The screenshot shows a software window titled "Carrier Priority Maintenance" with a menu bar containing "Action", "Edit", "Query", "Go", "Folder", "Special", and "Help". Below the menu is a table with two columns: "Freight Code" and "Freight Priority". The table contains several rows, with the first three rows populated: "UPS GRND" with priority 10, "UPS 2DAY" with priority 20, and "FDX 3DAY" with priority 30. The "FDX 3DAY" row is highlighted in blue. There are also several empty rows below.

Freight Code	Freight Priority
UPS GRND	10
UPS 2DAY	20
FDX 3DAY	30

Exhibit 2: Zone Priority Maintenance Form



The screenshot shows a software window titled "Zone Priority Maintenance" with a menu bar containing "Action", "Edit", "Query", "Go", "Folder", "Special", and "Help". Below the menu is a table with four columns: "Zone Code", "From", "To", and "Zone Priority". The table contains several rows, with the first two rows populated: "ZONE-1" with "From" 40000 and "To" 45000 and priority 10, and "ZONE-2" with "From" 90000 and "To" 90500 and priority 20. There are also several empty rows below.

Zone Code	From	To	Zone Priority
ZONE-1	40000	45000	10
ZONE-2	90000	90500	20

Exhibit 3 Customer/Item Exceptions

Customer/Item Exception Entry

Action Edit Query Go Folder Special Help

Exceptions

Customer Name	Item
<input type="checkbox"/> Customer1	<input type="checkbox"/> 70729
<input type="checkbox"/> Customer2	<input type="checkbox"/> 70874
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>

Exhibit 4: Inventory -> Setup -> Transactions->Source Types

Transaction Source Types

Action Edit Query Go Folder Special Help

User Defined

—Validation—

	Name	Description	Type	Context	[]
<input type="checkbox"/>	Admin Hold	Admin hold	None		<input type="checkbox"/>
<input checked="" type="checkbox"/>	Havas Inventory Hold	Havas Inventory Hold	None		<input type="checkbox"/>
<input type="checkbox"/>	Intransit	Inter Org Transfers	None		<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>
<input type="checkbox"/>					<input type="checkbox"/>

Transaction Types

Exhibit 5

Order Entry->Shipping->Release Sales Orders->Release Sales Orders

Check This Flag

Uncheck This Flag

Release Sales Orders For Picking (INV)

Action Edit Query Go Folder Special Help

Batch

Document Set

Based On Rule

Release Criteria

Orders

Shipment Priority

Subinventory

Carrier

Autocreate Deliveries

Scheduled Ship Dates

From

To

Requested Dates

From

To

Departure

Departure

Delivery

Allow Partial Release

Include Planned Lines

Order Type

Prior Reservations Only

Order Number

Ship Set

Item

Customer

Release Sequence Rule

Ship To

Pick Slip Grouping Rule []