**Master Design Document**

**Omnipress**

**Document v1.0**

August 2025

**Deposco, Inc.**1-877-770-1110 | www.deposco.com  
© Copyright 2025. All Rights Reserved.

# Document Revision History

| **Date** | **Version** | **Notes** |
| --- | --- | --- |
| 08/08/2025 | 1.0 | Initial Version |

# 

# Table Of Contents

[***Document Revision History 2***](#_heading=h.lkz0lmqbhugu)

[***Table Of Contents 2***](#_heading=h.hvcfuyt37khh)

[***1. Overview 7***](#_heading=h.3t7ocuq9d6a3)

[***1.1. Introduction 7***](#_heading=h.xm7zvpozngth)

[***1.2. Scope 7***](#_heading=h.do6k9vv5kdrc)

[***1.3. Success Criteria 8***](#_heading=h.fnq4evb5j9x4)

[***1.4. Terminology 8***](#_heading=h.c18pn7bilv2g)

[***1.5. Acronyms 10***](#_heading=h.8wfpgkclnc6w)

[***2. Master Data 12***](#_heading=h.xpm3s3hc0sxz)

[***2.1. Facility 12***](#_heading=h.r9qp8oj3j2lh)

[***2.2. Company 12***](#_heading=h.upsu1gzaedut)

[***2.3. Locations 12***](#_heading=h.kwmvt7750axo)

[***2.4. Zones 14***](#_heading=h.543myktktapt)

[***2.5. Storages 15***](#_heading=h.thls6u8xl9xb)

[***2.6. Items 15***](#_heading=h.7nj830qbpoug)

[***2.6.1. Item Assumptions 16***](#_heading=h.356osjd9g8hj)

[***2.7. Packs 16***](#_heading=h.affgb1xm85la)

[***2.7.1. Pack Assumptions 17***](#_heading=h.7ozsbmbrgp9p)

[***2.8. Trading Partners 17***](#_heading=h.ezdnnnl628ju)

[***2.9. Users 17***](#_heading=h.pw66pjvstiu1)

[***2.9.1. Company Assignments 18***](#_heading=h.55xlgeewxmdf)

[***2.9.2. User Assumptions 18***](#_heading=h.w5lv0lrd08x8)

[***2.10. Carton Types 19***](#_heading=h.ys4dqzcf31h8)

[***2.10.1. Carton Type Assumptions 19***](#_heading=h.pl3ea0czclde)

[***2.11. Containers 19***](#_heading=h.u8z9s7d08n9l)

[***2.11.1. Types 19***](#_heading=h.1u0bpz4j7alb)

[***2.11.2. Container Assumptions 20***](#_heading=h.f81np87htiiq)

[***3. Integrations 21***](#_heading=h.gwt77a8uljyb)

[***3.1. Integration Diagram 21***](#_heading=h.a2n5rw8zy2ep)

[***3.2. Sockets 21***](#_heading=h.u97xseoonwd0)

[***3.2.1. Socket Interfaces 21***](#_heading=h.b9do9r2wrpen)

[***3.2.1.1. Item 21***](#_heading=h.5g1gcaqe4jin)

[***3.2.1.2. Customer Order 22***](#_heading=h.dz5todcvrqmd)

[***3.2.1.3. Ship Confirm 22***](#_heading=h.pz1tcgohmrdr)

[***3.2.1.4. Inventory Sync 23***](#_heading=h.8oynywry7wu)

[***3.3. Shipping Carriers 23***](#_heading=h.i76em0p4p017)

[***3.3.1. Shipping Carrier Assumptions 23***](#_heading=h.kyf8ok5ifqx6)

[***3.4. Data Exchange 24***](#_heading=h.ff3mrh1k2i0b)

[***3.4.1. Import Templates 24***](#_heading=h.byjlojysbott)

[***3.4.2. Data Exchange Assumptions 25***](#_heading=h.g6xaz8r7331s)

[***4. Inbound 26***](#_heading=h.wvec9dv99n10)

[***4.1. Receiving 26***](#_heading=h.genvumcviv30)

[***4.1.1. Strategy 26***](#_heading=h.wr3cz58jft67)

[***4.1.2. Process Flow: Receiving 27***](#_heading=h.ny9tnbe2cml)

[***4.1.3. Receiving Assumptions 29***](#_heading=h.sxrzffoss2la)

[***4.2. UnReceiving 30***](#_heading=h.i29t3cy9wj85)

[***4.2.1. Unreceiving Assumptions 30***](#_heading=h.ksf5jx2lpjhh)

[***4.3. RMA/Returns Receiving 30***](#_heading=h.j7zg2mo1cjbd)

[***4.3.1. Strategy 30***](#_heading=h.v1rqjhsg5orf)

[***4.3.2. Process Flow: CreateRMA 30***](#_heading=h.mvskotuwbpaj)

[***4.3.3. Process Flow: Receive Customer Returns 34***](#_heading=h.b0cln0vl7y0j)

[***4.3.4. Return Receiving Assumptions 34***](#_heading=h.megna3982hzl)

[***4.4. Putaway 35***](#_heading=h.w3np8li4xxp7)

[***4.4.1. Strategy 35***](#_heading=h.dzjlo33zxbi6)

[***4.4.2. Process Flow: Putaway Process 35***](#_heading=h.2oxl2foqq3dx)

[***4.4.3. Assumptions 37***](#_heading=h.5vswz8skyxnj)

[***5. Inventory Management 39***](#_heading=h.l3z8rez8livf)

[***5.1. Inventory Transactions 39***](#_heading=h.ccs5anryw0ec)

[***5.2. Inventory Movement 40***](#_heading=h.vbz9vdmoyznm)

[***5.2.1. Move Item 40***](#_heading=h.a2ubipde8slo)

[***5.2.2. Move Stock 40***](#_heading=h.nhpdhsrgignl)

[***5.2.3. Process Flows 41***](#_heading=h.305isoh3tc3c)

[***5.2.4. Inventory Movement Assumptions 42***](#_heading=h.iiau90xz39s4)

[***5.3. Replenishments 42***](#_heading=h.5jkfmzw70le5)

[***5.3.1. Manual Replenishments 42***](#_heading=h.b8ku4rtmvzdf)

[***5.3.1.1. Strategy 42***](#_heading=h.e884ueu8h5ek)

[***5.3.1.2. Manual Replenishments Assumptions 42***](#_heading=h.g51835nawyfu)

[***5.3.2. Storage-Based Replenishments 43***](#_heading=h.77nkdoue8evj)

[***5.3.2.1. Strategy 43***](#_heading=h.jiuiflb1sgoy)

[***5.3.2.2. Replenishment Assumptions 44***](#_heading=h.zifgb1zbepc8)

[***5.3.3. Replenishment Picking Exception 44***](#_heading=h.79sif29u32en)

[***5.3.3.1. Screen Flow 44***](#_heading=h.9mhscgcfrlx4)

[***5.3.3.2. System Updates 45***](#_heading=h.oihrmbi9r9o3)

[***5.3.4. Replenishment Putaway Exception 45***](#_heading=h.9lx6cbpbjz2m)

[***5.3.4.1. Screen Flow 46***](#_heading=h.iqbk46lip49n)

[***5.3.4.2. System Updates 46***](#_heading=h.83ih38bucl3o)

[***5.3.5. Process Flow 47***](#_heading=h.ci8zid827rde)

[***5.4. Cycle Counts 47***](#_heading=h.jqsaj084n0jr)

[***5.4.1. Physical Counts 47***](#_heading=h.i3kt9wz4afta)

[***5.4.1.1. Strategy 48***](#_heading=h.wee1g12zlmue)

[***5.4.2. Cycle Counts 48***](#_heading=h.60mx88xd8cok)

[***5.4.3. Process Flow 49***](#_heading=h.wmzaoc7b35pc)

[***5.4.4. Cycle Count Task Generation 49***](#_heading=h.7e3jfz1ljy6v)

[***5.4.5. Cycle Count Execution 52***](#_heading=h.69a0f8bqmgak)

[***5.4.6. Cycle Count Assumptions 52***](#_heading=h.worme1zb366w)

[5.4.7. Managing Cycle Count Variances 53](#_heading=h.3nn0vbqdusov)

[***5.4.7.1. Strategy 53***](#_heading=h.7i0qrnex4u5f)

[***5.4.7.2. Managing Cycle Count Variances Assumptions 53***](#_heading=h.9ne7d5drvo0t)

[***5.5. Inventory Adjustments 54***](#_heading=h.b99lurmy87tp)

[***5.5.1. Strategy 54***](#_heading=h.44knwyc5rmp)

[***5.5.2. Process Flow 55***](#_heading=h.q5z0tkc773e4)

[***5.5.3. Key Assumptions 57***](#_heading=h.vsmkjt2jei3l)

[***5.6. Kitting 57***](#_heading=h.3k3mh2p8cpqa)

[***5.6.1. Process Flow 57***](#_heading=h.yi06aglv8mal)

[***5.6.2. Process Flow 60***](#_heading=h.strpkzirkgxf)

[***5.7. Miscellaneous Inventory Processes 60***](#_heading=h.empy7bnsm5c5)

[***5.7.1. End of Day Inventory 60***](#_heading=h.shvyojqtne5y)

[***5.7.2. Warehouse Lookup 61***](#_heading=h.i6kq7liw4wri)

[***5.7.3. Inventory Calculations 61***](#_heading=h.lkh60ef2rz9d)

[***6. Outbound 63***](#_heading=h.e65k0hxqqjl3)

[***6.1. Customer Order Management 63***](#_heading=h.apzqg4ejlfkl)

[***6.1.1. Pre-Processing 64***](#_heading=h.cse72gmqnhle)

[***6.1.2. Customer Order Statuses 64***](#_heading=h.dk57i2u1kgd6)

[***6.1.3. Process Flow 65***](#_heading=h.i6yv70qf7ouy)

[***6.1.4. Exception Management 66***](#_heading=h.4uwrl950rvy7)

[***6.1.5. Cancellations and Updates 67***](#_heading=h.cswypolgt4ae)

[***Key Assumptions 67***](#_heading=h.fc1u3sccpfmm)

[***6.2. Customer Order Workbench 67***](#_heading=h.h61v4ug4h0ei)

[***6.2.1. Key Assumptions 68***](#_heading=h.cv6wqzjuex0v)

[***6.3. Sales Order Management 69***](#_heading=h.6m3kcbzeze53)

[***6.3.1. Strategy 69***](#_heading=h.uphucwvmgbbs)

[***6.3.2. Sales Order Statuses 71***](#_heading=h.20g0kcd5bon5)

[***6.3.2.1. Order Status 71***](#_heading=h.d0dbfu1yf8au)

[***6.3.2.2. Current Status 72***](#_heading=h.aralk6xtq3at)

[***6.3.2.3. Shipping Status 72***](#_heading=h.bqoaj1szi8uk)

[***6.3.3. Order Cancelations 73***](#_heading=h.d78sn4roukw9)

[***6.3.4. Backorder Management 73***](#_heading=h.9e80yvfc1nlw)

[***6.3.5. Order Classification 73***](#_heading=h.mjujd7jf69ub)

[***6.3.6. Data Cleansing And Order Profiling 73***](#_heading=h.u976idi3lfmt)

[***6.4. Waving 74***](#_heading=h.ynkjquh38o7x)

[***6.4.1. Wave Profiles 74***](#_heading=h.k339dkdp4h6n)

[***6.4.1.1. Single line Single Unit Orders 75***](#_heading=h.v6geuyc9szk9)

[***6.4.1.2. Multi Line Orders 77***](#_heading=h.596syhbv2azu)

[***6.4.1.3. Partial Allocation Orders 78***](#_heading=h.2oou07l7rapm)

[***6.4.1.4. Freight Orders 79***](#_heading=h.vnqm7sjopd3i)

[6.4.2. Allocation/Wave Release 81](#_heading=h.kegvc7ftg4a0)

[***6.5. Picking 81***](#_heading=h.7je3jrcto15w)

[***6.5.1. Single Scan Picking 81***](#_heading=h.gv7ns5wijvnu)

[***6.5.1.1. Process Flow 82***](#_heading=h.jes8795rwrqr)

[***6.5.1.2. Single Scan Picking Assumptions 82***](#_heading=h.3cl8otllzf2m)

[***6.5.2. Batch Picking (multis parcel picking) 83***](#_heading=h.krfgyclfnv4l)

[***6.5.2.1. Process Flow 84***](#_heading=h.7u4gcm47sllu)

[***6.5.2.2. Screen Flow 85***](#_heading=h.k83u0ux2sfj)

[***6.5.2.3. Batch Picking Assumptions 85***](#_heading=h.8zgbbpbtch65)

[***6.5.3. Exception Handling 85***](#_heading=h.dmbf5oxkhcns)

[***6.5.3.1. Screen Flow 87***](#_heading=h.x274d8vdig1q)

[***6.6. Packing 88***](#_heading=h.w3ptdf1dchwu)

[***6.6.1. Pack Order By Item 88***](#_heading=h.dy11zi8fe0ds)

[***6.6.1.1. Process Flow 89***](#_heading=h.j5o1iet73ab4)

[***6.6.1.2. Screen Flow 89***](#_heading=h.3dpfsdkn5t2x)

[***6.6.1.3. Pack Order By Item Assumptions 90***](#_heading=h.hgrfrexz51ag)

[***6.6.2. Single Scan Packing 90***](#_heading=h.o0l35kotfc2u)

[***6.6.2.1. Process Flow 91***](#_heading=h.2kclbiszj8o8)

[***6.6.2.2. Screen Flow 92***](#_heading=h.7evqm7ir1g4y)

[***6.6.2.3. Single Scan Packing Assumptions 92***](#_heading=h.9rxgs5v762v1)

[***6.6.3. Bulk Packing 92***](#_heading=h.u1x2cgwokfkn)

[***6.6.3.1. Screen Flow 93***](#_heading=h.kwccululuyhu)

[***6.6.3.2. Bulk Packing Assumptions 94***](#_heading=h.8pypax1fe4hu)

[***6.6.4. Freight Packing 94***](#_heading=h.oouo6lm1ud9w)

[***6.6.4.1. Repalletize 94***](#_heading=h.dko8sfxxmx7b)

[***6.6.4.2. AssignTripToContainer 97***](#_heading=h.gr8vqhwf69iz)

[***6.6.4.2.1. Handheld Process 98***](#_heading=h.qaowjjrugyd3)

[***6.7. Outbound Flows Summary 100***](#_heading=h.w2wrzmugf2w)

[***6.8. Outbound Miscellaneous 100***](#_heading=h.a6rm1jjk69ve)

[***6.8.1. Void And Reprocess Shipment 100***](#_heading=h.g3xfimy8nudq)

[***6.8.2. Process Flow 100***](#_heading=h.i867kocirfbc)

[***6.8.3. Void and Reprocess Assumptions 101***](#_heading=h.muzg9b7lq4wm)

[***6.8.4. Roll Back Order After Picking 101***](#_heading=h.md0arreq985v)

[***6.8.5. Process Flow 102***](#_heading=h.mdlf1kr9whmb)

[***6.8.6. Screen Flow 103***](#_heading=h.h0epq2ktcn9h)

[***6.8.7. Rollback After Picking Assumptions 103***](#_heading=h.jm9ijrudgpqx)

[***7. Billing 104***](#_heading=h.lqb93xlydswz)

[***7.1. Scheduler-Based Billing Configs 105***](#_heading=h.81b683v5w6e2)

[***7.1.1. Order Processing Fees 105***](#_heading=h.9r5n9z9swxfy)

[***7.1.2. Shipping Fees 107***](#_heading=h.eh3ccr4ikisq)

[***7.1.3. Rush Order Fee 109***](#_heading=h.8mljlu18oz7p)

[***7.1.4. Customer Return fee 110***](#_heading=h.nfrxpmiqayr8)

[***7.1.5. Kitting 110***](#_heading=h.d6l4sk4ikv57)

[***7.1.6. Inventory Storage 111***](#_heading=h.2pbue0a99qni)

[***7.1.7. Administrative Fees 113***](#_heading=h.fmod7i4xv8em)

[***7.1.8. New Item Creation Fee 113***](#_heading=h.bwy2giii4v5)

[***7.1.9. Cycle Count Credit 114***](#_heading=h.pfb60gb82x3t)

[***7.2. AdHoc Billing Billing Configs 114***](#_heading=h.4q8hoksa5jwk)

[***7.2.1. Customer Setup Fee 114***](#_heading=h.u7pda3np0kdg)

[***7.2.2. Credits 114***](#_heading=h.bke1j4dym19a)

[***7.2.3. AdHoc Billing Process Flow 115***](#_heading=h.rtoyldxxf4s3)

[***7.3. Billing Transaction Export to Salesforce 116***](#_heading=h.isum5tfnvd68)

[***7.3.1. Assumptions 116***](#_heading=h.gb4rgukj2g0r)

[***8. Reporting 116***](#_heading=h.1ahyz9lz0jlo)

[***8.1. Inbound Reports 117***](#_heading=h.c2sikkt9evx0)

[***8.2. Inventory Management Reports 117***](#_heading=h.5j8k7nwuzrt2)

[***8.3. Outbound Reports 117***](#_heading=h.b7dog12zei7)

[***8.4. Customer Portal 117***](#_heading=h.ov4nxrb62wh)

[***8.5. Operations/Warehouse 118***](#_heading=h.tz0r3nqdxp0h)

[***8.6. Reporting Miscellaneous 118***](#_heading=h.qm86ku55atnt)

[***Customer Agreement 119***](#_heading=h.spssmwgabg43)

# Overview

## Introduction

The purpose of this document is to identify all requirements and processes necessary to implement Deposco WMS and OMS for the Omnipress Phase 1 Deposco Implementation.

## Scope

The full scope of this project is outlined in the Master SaaS Agreement and Statement of Work (SOW001). This document is used to align both Deposco and Omnipress on functionality to be implemented and is not considered a new Scope or Change Request (CR). A separate task list is provided with estimates of effort and duration to implement functionality outlined in this document. Estimates are not fixed, nor to be considered maximum/ ‘Not To Exceed’ (NTE) estimates and may change based on new information, design gaps, functionality gaps, or other. If estimates change before or during a task’s execution, Deposco will attempt to notify Omnipress to review options.

This document is considered a living document and will be maintained throughout the project by Deposco. Changes in functionality requested by Omnipress after Design Document sign off that require significant rework may lead to overages beyond original estimate; therefore, a CR is required to execute so all parties are aware of the change. The purpose of a CR is to define all requirements of change to mitigate future rework.

## Success Criteria

The success criteria of this project are as follows:

* Enable system-directed warehouse processes and remove paper
* Improve order accuracy
* Integrate shipping and enable a direct connection with shipping carriers
* Automate 3PL billing
* Provide a self service Client Portal
* Eliminate manual data entry

## Terminology

| **Terminology** | **Definition** |
| --- | --- |
| **Batch/Lot** | Group of items manufactured together |
| **BOL** | Bill of Lading.  A legal document used for the transportation of goods (non-parcel). |
| **CO (Customer Order)** | An order placed by a customer including header level information like destination address and line level information like item and quantity. |
| **SO (Sales Order)** | Sales orders are used for fulfillment processing through waving, picking, packing, and shipping. As a result, sales orders are also referred to as fulfillment orders. Customer orders are routed to facilities as sales orders. |
| **Business Unit** | A division or segment of an organization generally treated as a separate profit-and-loss center. Business units are defined in Deposco through the company entity. |
| **Error Message** | An error message that does NOT allow the user to continue processing after viewing the error. |
| **ATP** | Available to Promise. Inventory calculation used during allocation and when communicating inventory to an external system. |
| **Item/SKU** | Sellable product. Usually item name, item description, and item barcode are three unique values. |
| **LPN-Controlled Location** | Locations that do require an LPN to be associated with all inventory |
| **Pallet** | A physical surface (skid) to stack inventory on top of. Can be tracked systemically as an LPN or assists in tracking product movement where the group of LPNs move together. Can have one barcode for all inventory on the pallet or unique LPNs for each case (nested LPNs). |
| **Putaway** | The systematic process of locating products to a permanent location. |
| **Reserve Location** | Storage locations that are marked as an eligible replenishment source location. Generally not marked pickable. |
| **Ship Notice** | Notice that is sent to an external system through a socket integration to notify the external system that part or all of an order was fulfilled. Ship notices are sent on a real-time basis as each shipment is created. |
| **Shipment** | For outbound processes, a shipment contains the shipping information for a sales order (SO), including the dimensions of the shipping container and the tracking number. A shipment may include part of an order, one order, or more than one order, depending on the business scenario. |
| **Work Group** | A portion of the pick tasks in a pick wave. For example, if a pick wave includes 100 orders, and pickers use a cart that can hold between one and eight bins, then the 100 orders can be separated into work groups that contain between one and eight orders each so that each picker's cart is not overfilled by the end of picking. Large orders can be separated into a work group of two orders per cart, while small orders can be separated into a work group of eight orders per cart. |
| **Pick Tasks** | Within a work group, the details of the item and quantity that must be picked from a pick location. |
| **Scheduler Task** | Specifies when to run a background process in the Deposco application, such as the creation of a list of replenishment tasks by priority. A scheduler task can run at any time and can update data sets for entities. |
| **Screen Message** | A screen message is an informational message that does not prevent the user from continuing to process their task. |
| **Pick Wave** | Distribution Order selection process to release work to the warehouse. A Pick wave follows selection, allocation, and task creation. |
| **PAL** | Process Action Link. Used in the Deposco UI to initiate processes ad hoc. |
| **View** | A defined set of filters for an entity list. Admin users can create and edit views for any entity that they have permission to modify. Non-admin users can create their own views and use the standard shared views, but cannot edit the shared views created by admin users. |
| **Layout** | A predefined set of fields that appear on an entity detail page. Admin users can create and edit the standard layouts for entities. Users can create personal layouts for entities. |
| **Order Release Profile** | Wave Processing Type.  Determines the allocation strategy for a Pick Wave (e.g. Bulk Item Pick) |

## Acronyms

| **Acronym** | **Definition** |
| --- | --- |
| 3PL | Third-party Logistics |
| DC | Distribution Center |
| DEP | Deposco |
| CO | Customer Order |
| BOL | Bill of Lading |
| FTL | Full truckload. Also, referred to as TL |
| ATP | Available To Promise |
| UPC | Universal Product Code |
| LPN | License Plate Number |
| VAS | Value Added Service |
| LTL | Less Than Truckload |
| PO | Purchase Order |
| SO | Sales Order. Also, referred to as Fulfillment Order |
| UOM | Unit of Measure |
| B2B | Business-to-Business |
| B2C | Business-to-Consumer |
| SCAC | Standard Carrier Alpha Code |
| COO | Country of Origin |
| UI | User Interface |
| PAL | Process Action Link |
| EL | Event Log |
| ESL | Event Subscription Log |
| SKU | Stock Keeping Unit |
| UA | Test/Sandbox environment |
| UAT | User Acceptance Testing |
| PROD | Production environment |

# Master Data

## Facility

A facility is a physical location for a company, typically a building such as a warehouse/distribution center or a retail store. A facility entity is created in Deposco for every facility that is involved in warehouse and order management activities.

Facilities are added manually in the Deposco User Interface. The following facilities are in scope for the initial implementation:

| **Facility** | **Description** |
| --- | --- |
| OMNI01 | Omnipress Warehouse - Madison, WI |

## Company

A company is the root entity in Deposco. All other entities in Deposco are assigned to a company. For example, each facility, zone, product category, item, order, and so on is assigned to a company. In a 3PL setting, a Company is also referred to as a “Business Unit” or a “Brand”.

Companies are added manually in the Deposco User Interface or uploaded via Data Exchange. The Business Units in scope for the initial implementation are outlined in the 3PL Matrix.

## Locations

Locations are specific areas in a facility that are used for tracking movement of physical objects such as inventory, containers, and shipments. A location can be a specific inventory storage shelf, or a location might represent an area like the receiving or shipping area of the facility.

The following standard locations are defined:

| Location | Description |
| --- | --- |
| Receiving | Standard location where inventory is staged after receiving and before putaway. |
| Returns | Standard location where inventory is staged after receiving a customer return. |
| Damages/Quarantine | Standard location where inventory is staged if identified as Damaged during receiving. |
| WHSE-Floor | Standard location where inventory is staged during the picking process. |
| Shipping | Standard location where inventory is staged after picking and before packing. |

### 

* A Location can be added to the system by being uploaded via Data Exchange or created manually in the Deposco User Interface.
* It is not recommended that Location Master Data is removed from the system once in use, as there is historical process data tied to the specific location. Instead, the Location can be repurpose/updated or the status can be adjusted to an inactive status, such as “Disabled” or “On Hold”.
* In order to make the warehouse location pick sequencing flexible, it is recommended that a few trailing zeros are added to each numerical value. For example, if a location needs to be added in between two existing locations and the current pick sequence is 10001000, 10002000, it is much easier to simply upload a location with a pick sequence of 10001500 than it would be if the original pick sequence did not have trailing zeros (e.g. 10001 10002).
* All Locations are barcoded and can be scanned using an RF device.
* Locations are defined as single SKU or mixed SKU based on the Mixed Item Threshold field: 1=Multi SKU; 0=Single SKU.
* Location pick sequence and putaway sequence are included as part of the Location Master Data and are maintained by the Omnipress team. These fields, along with the zones to which each location is assigned, play a key role in how stock is allocated and pick tasks are defined and sorted for work groups on a pick wave.
* All Locations have a predefined, alphanumeric, mask. This is to be determined by the Omnipress team.

## Zones

A zone is a systematic area in a facility that is dedicated to a specific functional task, such as picking, receiving, or shipping.

The following zones are defined:

| **Zone** | **Description** |
| --- | --- |
| Receiving | Standard zone to control all receiving locations. |
| Picking | Zone for locations that are used for picking, typically floor level. |
| Shipping | Standard zone to control all shipping locations, including packing stations. |
| Staging | Standard zone to control all Damage, WHSE-Floor, or temporary locations |
| Reserve | Zone for non-pickable locations that contain reserve inventory |

## Storages

Storage entities are used to assign an item (more specifically, the pack for an item) to a picking location. If multiple items are stored in a single location, a storage entity must be created for each item in the location.

As part of the definition of a storage entity, the minimum and maximum inventory levels for the item in that picking location must be specified. When the stock in the location falls below the minimum, a warehouse replenishment task can be generated to move inventory from a reserve location so that the quantity of inventory at the picking location equals but does not exceed the specified maximum.

Locations, Zones, and Storages are uploaded via Data Exchange or created manually in the Deposco User Interface. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document. Storages will be leveraged by Omnipress in initial rollout.

## Items

An item, or stock keeping unit (SKU), is a distinct type of item for sale, including all attributes that distinguish the item type from other item types, such as manufacturer, description, material, size, color, and packaging.

The following inventory attribute tracking is available as a part of the initial implementation:

| **Inventory Attribute** | **Global Setting** |
| --- | --- |
| UPC(s) | Yes |
| Lot Tracking | No |
| Expiration Date Tracking | No |
| Serial Tracking | No |
| Born on Date Tracking | No |
| Hazmat | No |
| Harmonized Tracking Codes | Yes |

### Item Assumptions

* Deposco will autogenerate item identifiers to use as UPCs for any new items that are currently not managed in Veracore.
* Omnipress will use Veracore item identifiers as UPCs to facilitate item scanning during fulfillment processes.
* UPC labels can be printed in Deposco in a as needed basis for new items that are received
* Items that will be shipped internationally require HS codes and Unit Price
* Items may require Item Class Type for international shipping

Items are uploaded via Data Exchange, created manually in the Deposco User Interface, or via running an Integration Touchpoint. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document.

## Packs

Packs enable you to manage the stock of an item in multiple units of measure. For example, you may stock individual beverage cans (which are called *Each* packs or *Eaches*), and also cases of 12 beverage cans. Although uncommon, a single item can have multiple packs of a given type (e.g. case–12, case–24, pallet–100, pallet–150). The weights and dimensions of each item are stored in the packs for the item.

Omnipress will use Eaches as the only pack type for all items. Other pack types can be used after the phase 1 implementation if needed such as cases and pallets.

### Pack Assumptions

* Every item must have at least one pack with a quantity value of 1, which represents the most granular unit by which the item is managed.
* Pack dimensions and weights are required for international shipments

Packs are uploaded via Data Exchange, created manually in the Deposco User Interface, or via running an Integration Touchpoint. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document.

## Trading Partners

A trading partner in Deposco represents an external party to the company that is typically associated with an order. The primary types of partners include:

* Customer – Used for partner companies that are purchasing products from the company and are therefore associated with sales orders. Also used for general shipping purposes.
* Supplier – Used for partner companies that a company is buying products from and are therefore associated with purchase orders. Also used for partners that provide drop shipping services.

Omnipress may leverage Trading Partners to store account information to ship Third Party using UPS.

Trading Partners are uploaded via Data Exchange or created manually in the Deposco User Interface. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document.

## Users

Deposco provides two different features for creating and managing users and passwords: *User Management*, which provides full user management functionality and requires membership in the Admin group, and *Associate User*, which can be configured to provide more limited user management functionality and is available on an application menu.

The Associate User feature is intended for users such as warehouse department managers or assistant store managers, who may need the ability to create and manage users for specific facilities but who should not have full administrative access to Deposco.

### Company Assignments

Company assignments limit users to viewing data and performing tasks for only the companies to which they are assigned. For example, entity lists are automatically filtered to display only entities for companies that the user has permissions for, and picking workflow processes can be configured to require the user to select one of their assigned tenant companies and then display only orders for the selected tenant.

Use the following strategies to set up company assignments for a user account:

* If a user should be assigned to a single tenant company, then select the company in the User of Tenant field on the User Information tab for the user account.
* If a user should be assigned to multiple companies, then enable company assignments in the user management options. Then select the primary company in the User of Tenant field on the User Information tab for the user account, and assign additional companies on the Company Assignments tab.
* If a user should be able to access data for all tenant companies, then leave the User of Tenant field blank on the User Information tab for the user account.

### User Assumptions

* Users, Groups, and Permissions are set up through the Deposco User Interface to ensure the right users have access to the right functionality/reports.
* Associate Users can be created manually in the User Interface.
* Omnipress can manage and update any group permissions, as well as group assignments to users.

## Carton Types

Carton types typically represent the different sizes of shipping boxes, and are used to facilitate the process of entering box dimensions during packing.

### Carton Type Assumptions

* Carton Types have accurate dimensions defined.
* Carton Types are uploaded via Data Exchange or created manually in the Deposco User Interface. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document.
* Carton Type information can be used during the packing process to enable packers to select the carton to use from a drop down list
* A Business unit can be assigned to a Carton Type to limit the number of options that packers have to select from

## Containers

A container is any object that holds items, such as a cart, bin, pallet, or shipping box. A container is also called an LPN (license plate number) since it has a preprinted, sequential barcode label for tracking.

### Types

* **Cart** - typically used for picking carts or for carts that are used for moving inventory. Carts are reusable containers.
* **Pallet** - typically used for inventory containers. Inventory may be received, put away, moved, and shipped while on a pallet container.
* **Transient** - typically used for reusable containers such as totes or bins.
* **Shipping** - typically created automatically by the system for containers that are shipped, including both parcel shipping containers and pallets that are used for shipping. Shipping pallets are typically identified with a secondary container type of Pallet.

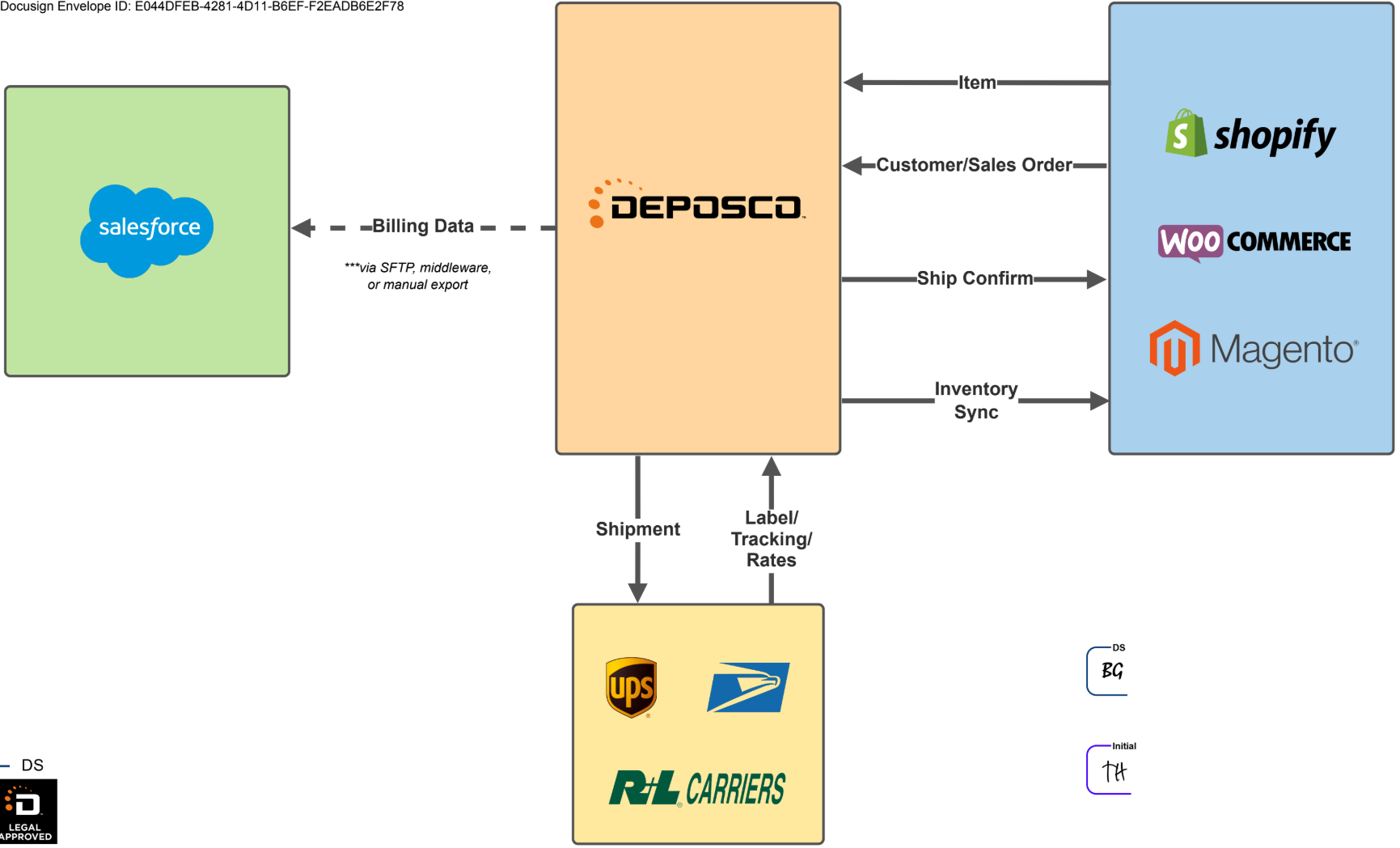
### Container Assumptions

* A container is associated with a single location at a time.
* Multiple stock units can be assigned to a single container.
* During shipping, the container is associated with a shipment before the shipment is completed. After the shipment is complete, the container becomes a container history (ContainerHist) entity, which is a separate entity from a container.
* All bins and carts are uploaded to the system. This allows the system to perform validations on user scans during handheld processes.
* All totes, bins, and carts are barcoded and can be scanned using an RF device.

Containers are uploaded via Data Exchange or created manually in the Deposco User Interface. Specific mapping requirements are outlined in more detail in the [Data Exchange](#_heading=h.ff3mrh1k2i0b) section of this document.

# Integrations

## Integration Diagram



## Sockets

Omnipress will have socket integrations to Shopify, WooCommerce, and Magento.

A default set of field mappings are available for each socket. Details on the mappings are provided in the content for the socket in the Deposco Help Center. Omnipress will inform Deposco of any mappings that should be configured differently.

### Socket Interfaces

### Item

Integration of items (SKUs) is a critical component to enable successful communication between Deposco and external systems such as marketplaces, web stores, and enterprise resource planning (ERP) systems.

Item data is typically pulled from an external system on a scheduled basis. Omnipress can configure schedules for how often item integration occurs, as well as options to filter the data that's pulled or sent and control how it is stored in Deposco depending on the socket type.

When an item in Deposco is integrated with an external system through a socket integration, a channel listing is created for the item and socket. The channel listing provides information about the item in the external system, such as the item number and internal IDs in the external system, and provides a link between the item in Deposco and the socket that enables integration with the external system.

* Omnipress does not wish to create new items through the item interface. Only channel listings will be created. This will require the Omipress team to manually create all new items to prevent pulling Customer Orders with empty order lines.
* Digital items will be created in the system, and their related channel listings will be marked as *non-saleable* by setting the ‘Saleable?’ flag to false. This will prevent customer order lines from being created.

### Customer Order

Integration of orders through a socket enables you to pull order information from an external system such as marketplaces, web stores, and enterprise resource planning (ERP) systems, and to create an order in Deposco.

To manage information for an order that comes from an external system, channel cross-references (channelXrefs) are created for the order header and each order line. The channelXrefs contain data from the external system, such as the order number and its internal ID, and provide a link between the order in Deposco and the socket that enables integration with the external system.

### Ship Confirm

After a shipment is sent to fulfill an order that was pulled from an external system, a ship notice can be sent to notify the external system that part or all of the order was fulfilled. A ship notice typically includes the quantity of each item that was shipped, as well as the tracking number, shipping carrier, and shipping service for the shipment.

Ship notices are sent on a real-time basis based on system events that are generated when a shipment is created. However, some sockets use a schedule-based shipment confirmation interface to send order and shipment updates.

### Inventory Sync

During the inventory synchronization (inventory sync) process, the current quantity of inventory for an item in Deposco is sent to an external system to synchronize the quantity of inventory between the two systems. Inventory sync is an event-based process that occurs only for items that are associated with the selected socket.

## Shipping Carriers

Omnipress will use the following shipping carriers.

* UPS
* USPS (STAMPS)
* R+L Carriers

### Shipping Carrier Assumptions

* Two separate UPS accounts will be used, one to ship internationally and another domestically.
* Omnipress may configure any BU specific shipping carrier accounts if needed.
* Omnipress will use UPS address validation for domestic orders.
* Omnipress will set up all the shipping services that will be used.
* Omnipress will occasionally ship Third Party through UPS, which will require providing the Bill To Account details required by the shipping carrier.
* Trading Partners can be used for customers that ship Third Party frequently.
* Omnipress wants to limit their customer’s access to the negotiated rate, and show the published rate with the applicable customer discount.
* The item Unit Price will be used to send as the item value for international shipments.

## Data Exchange

A user can import data to and export data from Deposco by using Data Exchange, which provides a rich feature set, including a wizard-style user interface, manual or scheduled imports and exports, and configurable notifications.

Imports can be configured to occur either manually or on a scheduled, ongoing basis (hourly, daily, weekly, or monthly):

* With manual imports, users upload the file with the data they want to import.
* With scheduled imports, the import data file is pulled automatically from an FTP or SFTP location.

Exports can be configured to occur either manually or on a scheduled, ongoing basis (hourly, daily, weekly, or monthly):

* With manual exports, you select the base entity and the date range for the data to export. You can then download the export data file from the Transmissions list.
* With scheduled exports, the export occurs automatically on a scheduled basis. The export data file is saved automatically to an FTP or SFTP location. You can also manually download the export data file from the Transmissions list.

The configuration steps for imports and exports can be found in the [Deposco Help Site](https://docs.deposco.com/docs/html/Content/Administration/Data_Exchange/Import_and_export_data.htm?tocpath=Administration%7CImport%20and%20export%20data%7C_____0).

### Import Templates

| **Name** | **Type** |
| --- | --- |
| **Item, Pack, UPC** | Master Data |
| **Location, Zone, Storage** | Master Data |
| **Company/Business Unit** | Master Data |
| **Purchase Order** | Master Data |
| **Container** | Master Data |
| **Trading Partner** | Master Data |
| **Component** | Master Data |
| **Carton Type** | Transactional Data |
| **Customer Order** | Transactional Data |
| **Storage Details** | Master Data |

### 

### Data Exchange Assumptions

* The default Data Exchange templates are configured underneath the parent Business Unit. For company-specific import templates, a new template needs to be created underneath the specific Business Unit. The templates underneath the parent Business Unit are copyable.
* All import failures are viewable within the Transmissions UI.

# Inbound

This section outlines all inbound related processes. This includes receiving at the warehouse, returns processing, and putaway processes.

## Receiving

### Strategy

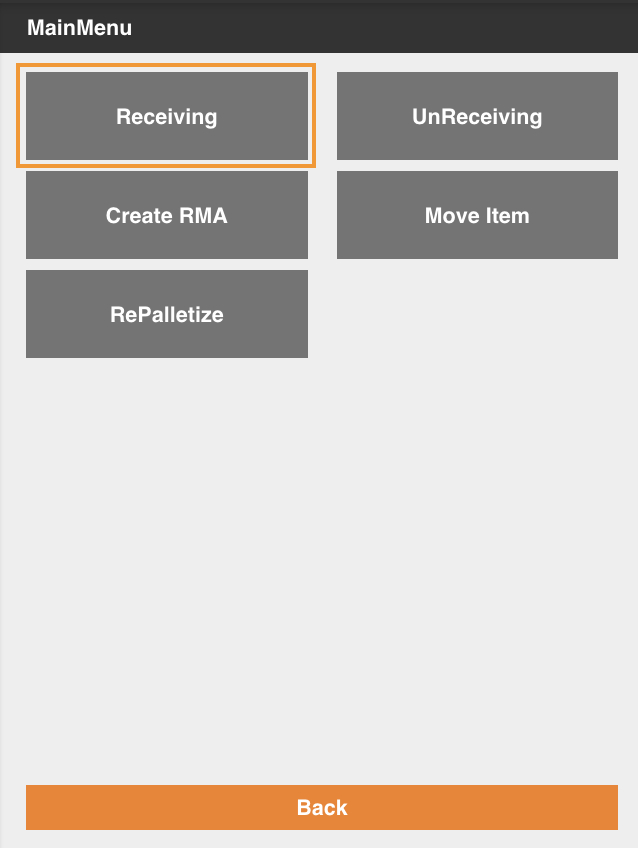
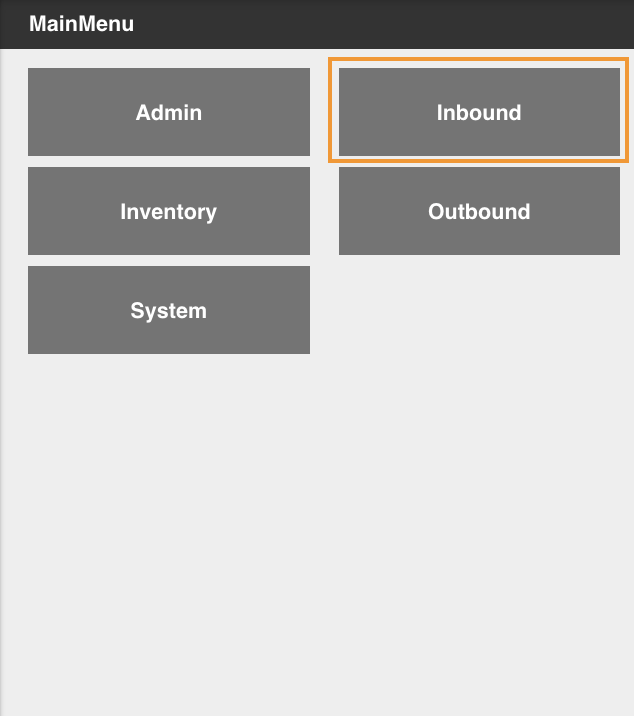
Receipts into Deposco will be performed using the Receiving process. Specifically, Omipress will use the Blind Receiving workflow to create stock units in the receiving location. After the phase 1 implementation, PO creation may be leveraged to confirm received quantities.

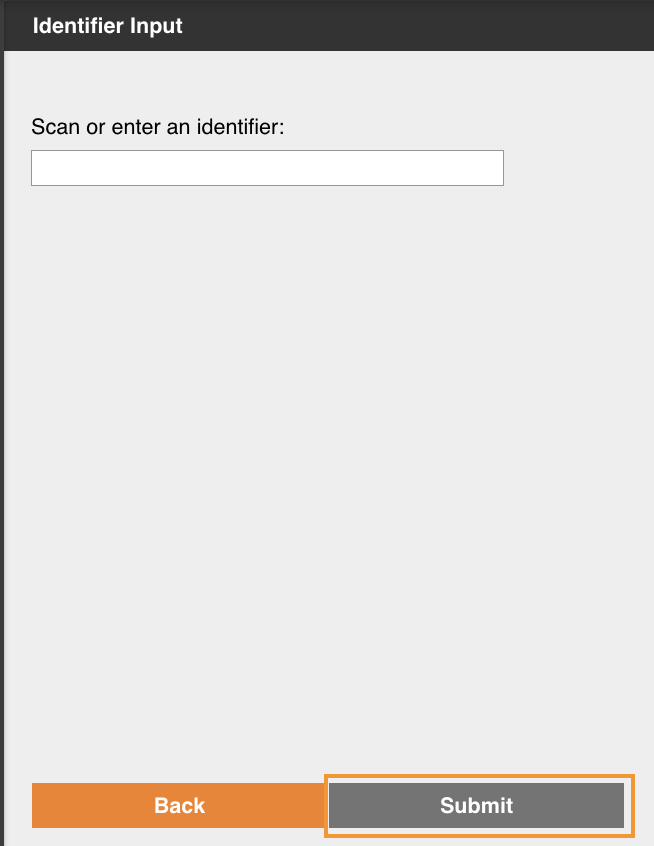
*PO Statuses:*

| **Status** | **Definition** |
| --- | --- |
| **New** | The order is a new order that was either received from an external system or manually created and submitted. |
| **Receiving** | A user has started using the Receiving process to receive the first shipment of stock for the order, but the process has not yet been completed. |
| **Partial Receipt** | Some stock but not all of the stock for the order has been received in Deposco. |
| **Received** | All of the stock for the order has been received in Deposco. An order may also move to this status if one or more order lines were canceled but the rest were successfully received. |
| **Closed** | The purchase order was manually closed to prevent additional receipts against the order. |
| **Canceled** | A process or a user canceled the order, and the order will not be received. |

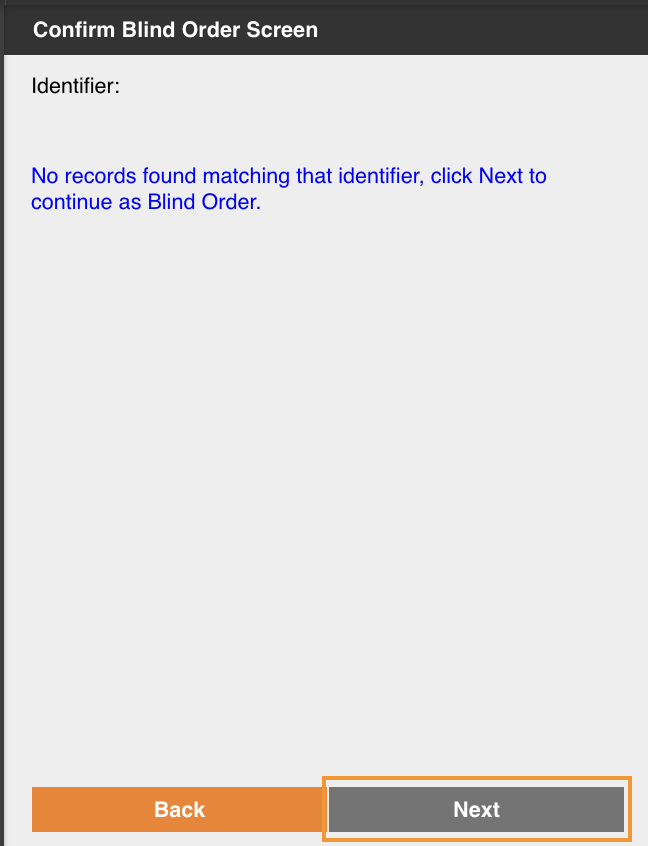
### Process Flow: Receiving

A user initiates the Receiving process by navigating to the Inbound menu in the Handheld and selecting Receiving. The Identifier input screen appears and the user selects Next.

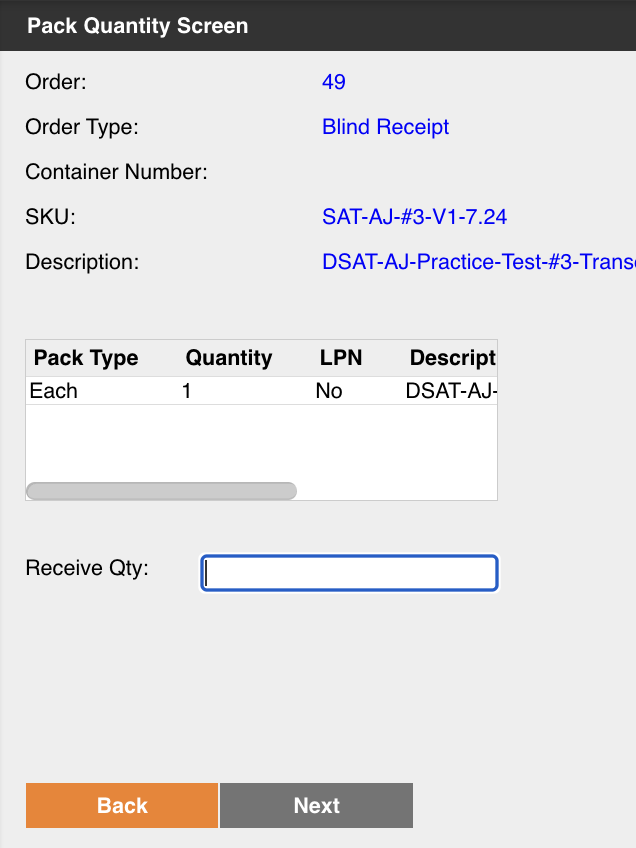




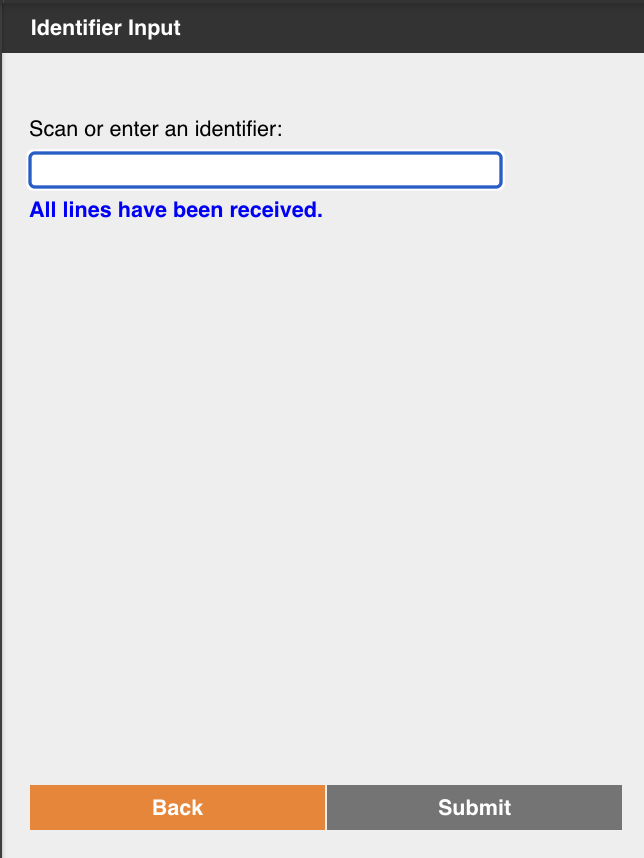
The user confirms the Blind Receipt in the Confirm Blind Order screen by selecting Next, which creates a new Blind Order in system to which items will be added as they are received.



The Receive Item Screen displays. The user scans the barcode on the label of the item to receive. In the next screen the user enters the quantity to receive.



Once all items have been received, the user clicks on Finish to close the Blind Order and mark it as fully received in the system. A confirmation message appears on the screen indicating that all lines have been received.



Items are received in the ‘Each’ pack type, a receiptLine is created, and the inventory is created in the ‘Receiving’ location.

*Damage Handling:*

Omnipress will not receive any damaged inventory but can leverage a quarantine location to move items that are not available to fulfill orders.

### Receiving Assumptions

* Items will be labeled prior to receiving.
* Putaway to storage will be handled separately via the Move Item process.
* Inventory is received and staged into the Receiving location in Deposco.
* Omnipress will have the option to print item labels on an as needed basis using the Print Report process.
* All screens with location information will display the location number, not the locator.

## UnReceiving

This process is used to undo the receipt of receiptLines already received in Deposco. The UnReceiving process creates receiptLines with negative quantity which decrements the quantity received in Deposco.

### Unreceiving Assumptions

* The inventory to un-receive must be located in the Receiving location.
* The UnReceiving process is used to fix errors during the receiving process. There is a separate process to adjust inventory in any location.

## RMA/Returns Receiving

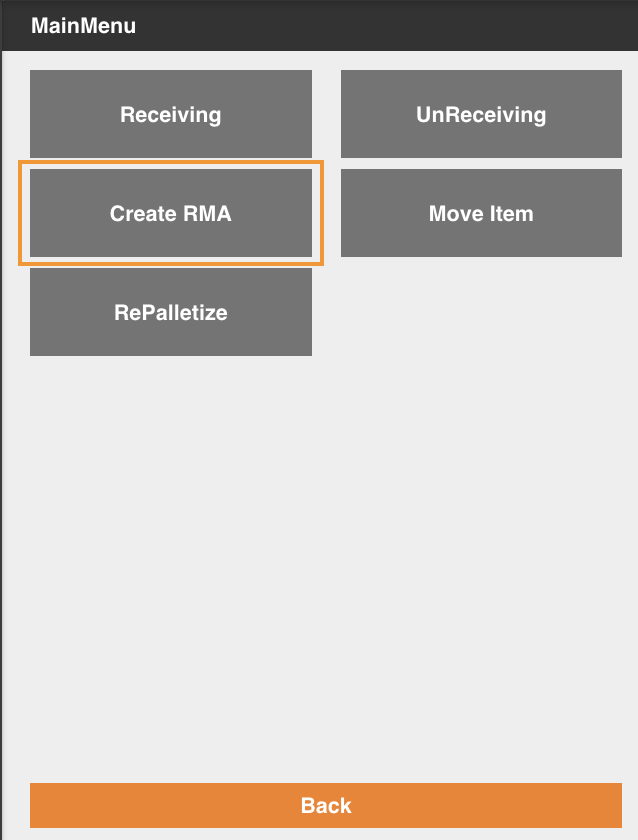
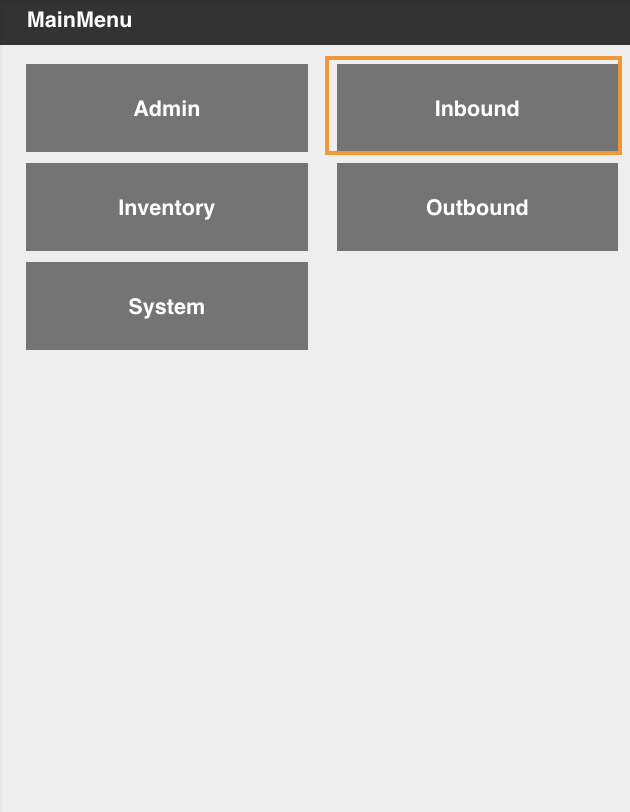
A Customer Return, or RMA, consists of shipped inventory that a customer is sending back to the warehouse.

### Strategy

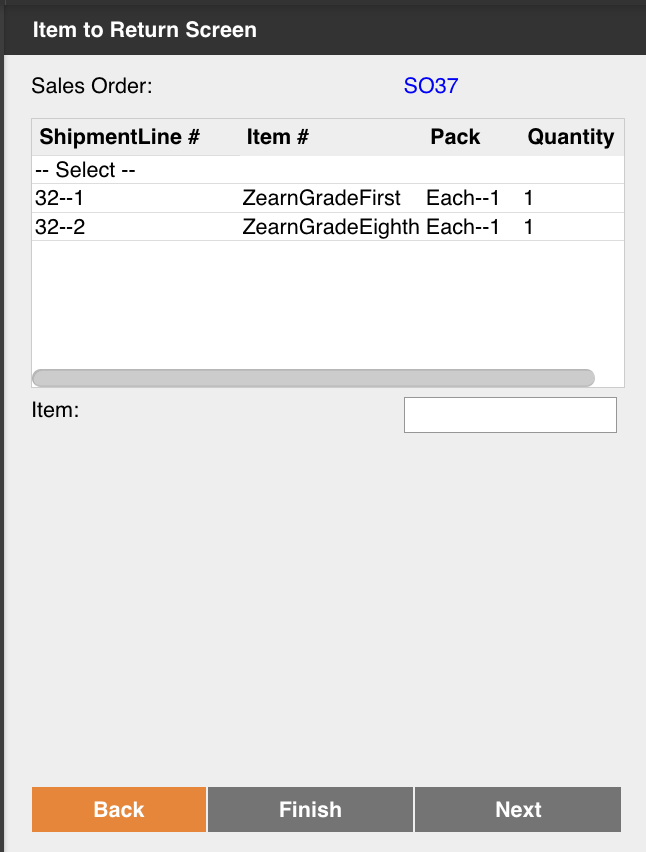
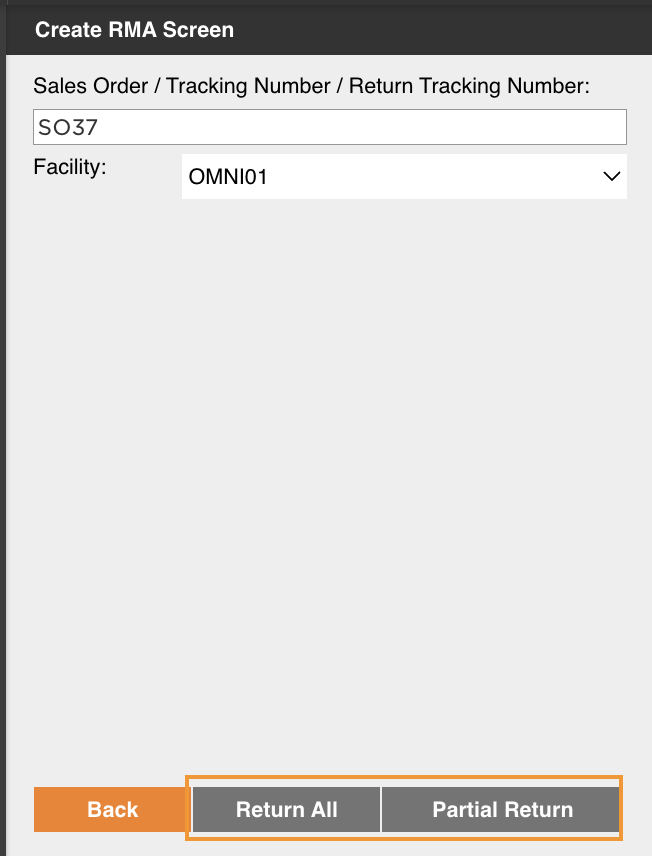
Omnipress will receive customer returns in Deposco using the Receiving process used for regular inventory receiving. Omnipress will have the option to use the CreateRMA process, which allows users to associate the customer return to an existing Sales Order.

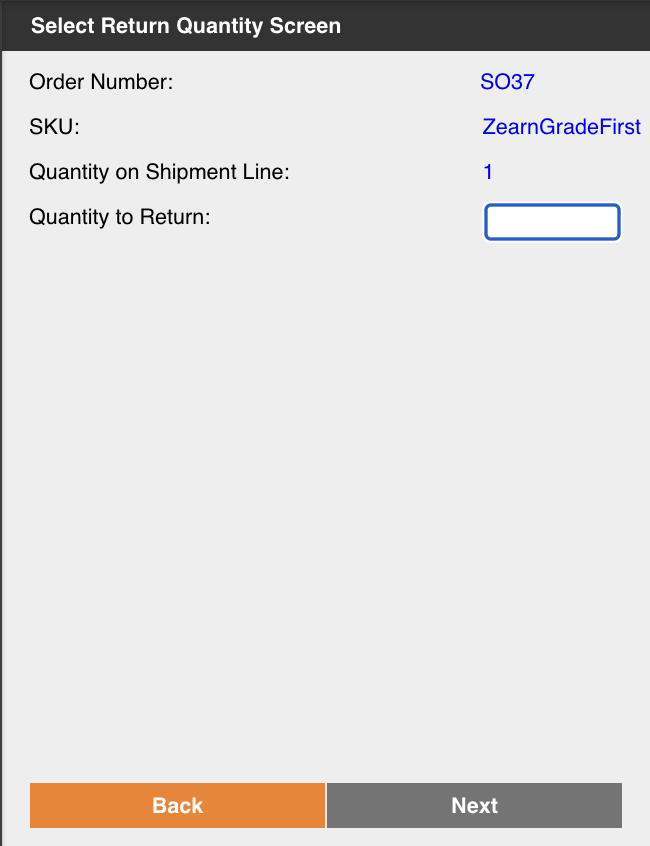
### Process Flow: CreateRMA

A user initiated the CreateRMA process by selecting Inbound from the handheld Main Menu, and then selecting the CreateRMA button.

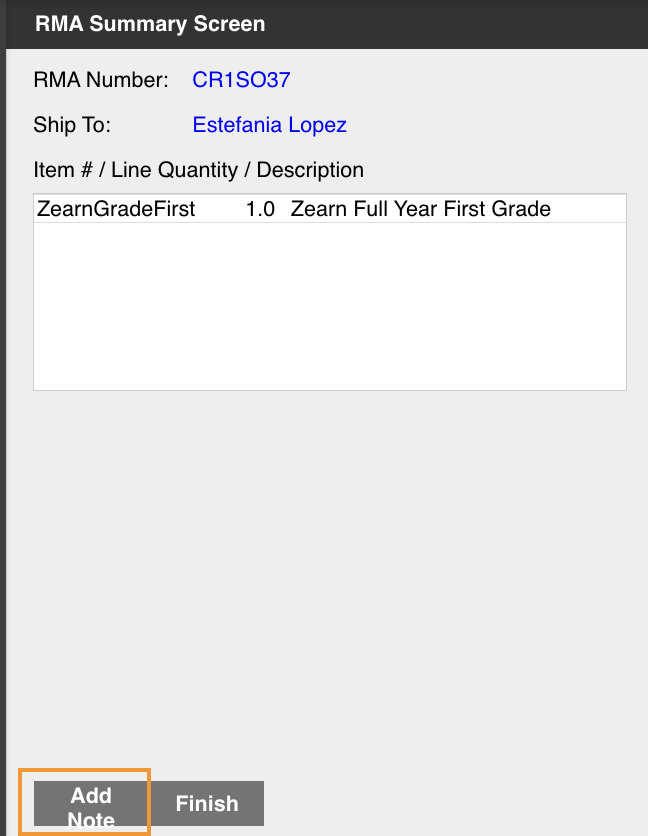
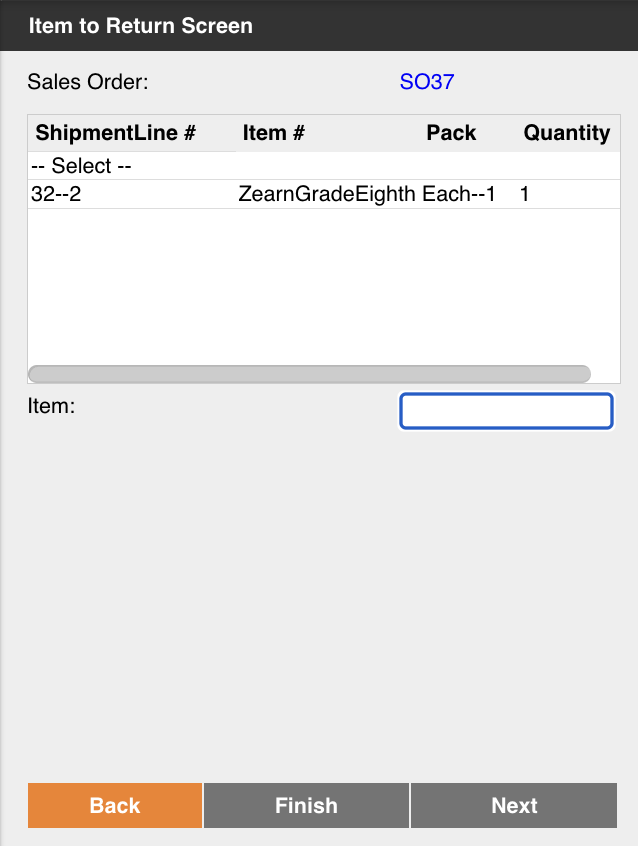


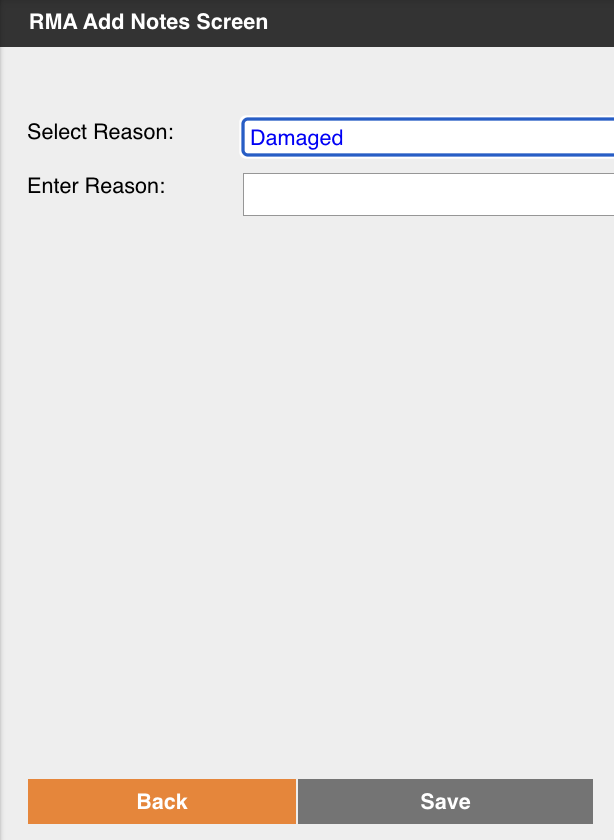
The Create RMA screen appears. The user enters the Sales Order number related to the customer return and selects Return All to create an RMA for all items in the order or Partial Return to indicate specific items and quantities that are being returned.



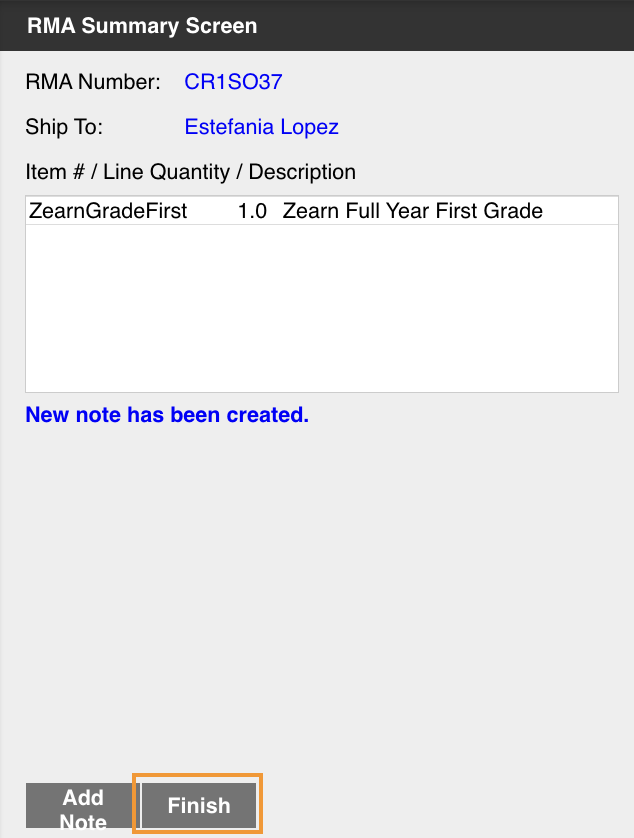


After selecting all the items to include in the RMA, the user can optionally add a note to indicate the return reason, which will create a Note record that will be linked to the Customer Return for future reference.



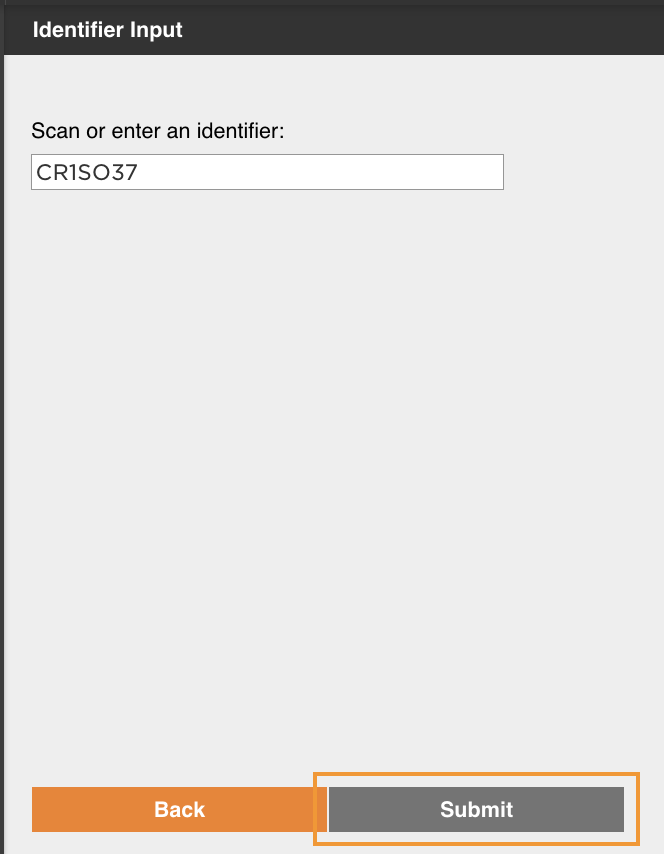


To finalize the process, the user clicks on Finish.



### Process Flow: Receive Customer Returns

To receive an RMA generated through the CreateRMA, the user will enter the identifier in the Receiving screen, and will proceed to receive the returned stock as normal.



To receive a blind customer return, the user will follow the normal Blind Receiving process flow and there will be no association to any shipped sales order.

### Create RMA And Return Receiving Assumptions

* The CreateRMA process will show the following list of reasons for a customer return.
  + Buyer no longer wants/needs
  + Incorrect item was ordered
  + Item did not meet buyer expectations
  + Incorrect Item delivered
  + Item arrived damaged
  + Refused Package
  + Package Undeliverable Address
  + Test Order
  + Customer Approved Return
  + Defective Product
  + Return from Conference/Class
* Omnipress will leverage the Receiving process to receive returns as a RMAs or Blind Orders.
* There will be a separate button to receive customer returns that will follow the same process flow as regular receiving, however, this process will have additional logic to identify the Blind Receipt as a customer return. This information will be used to generate customer return billable transactions.
* Returns will be staged to a systematic Returns location where received stock can either be returned to stock or move to the Quarantine location.

## Putaway

Receiving and Putaway are two separate steps and processes. Once inventory is created through the Receiving workflow process, it is systematically staged to Receiving staging location. At that point, the inventory is ready to be put away.

### Strategy

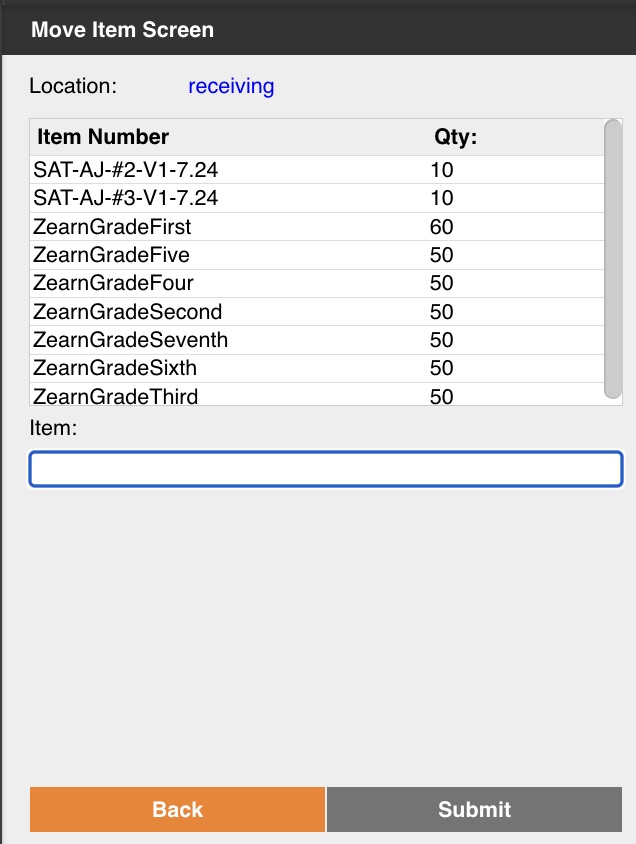
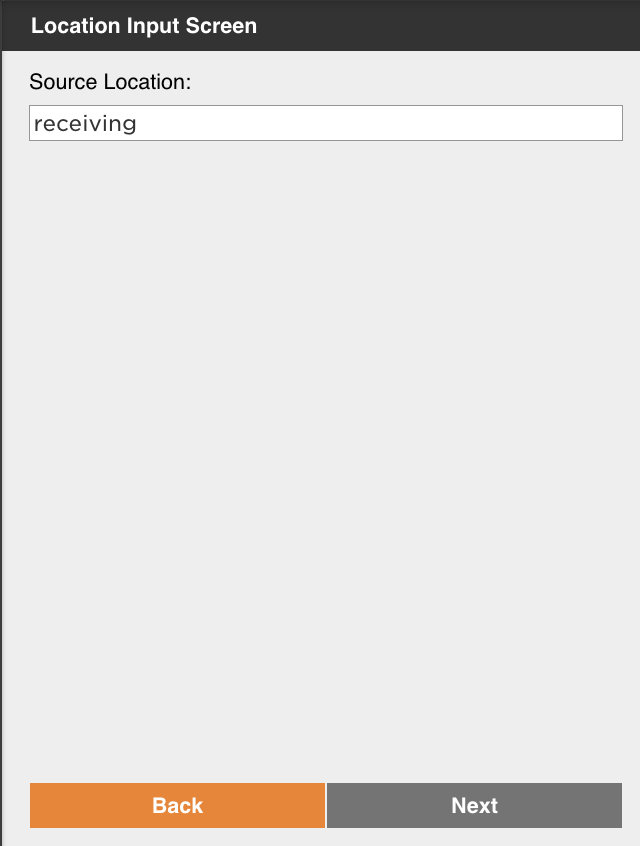
The putaway logic is detailed in the Putaway Logic section of this document.

Omnipress will be using the Move Item workflow process as the primary process for Putaway.

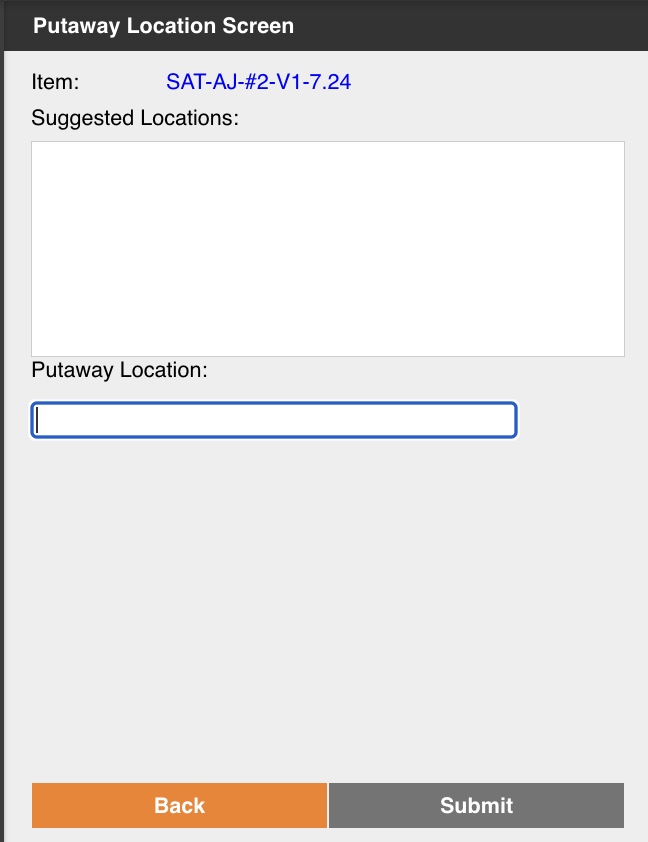
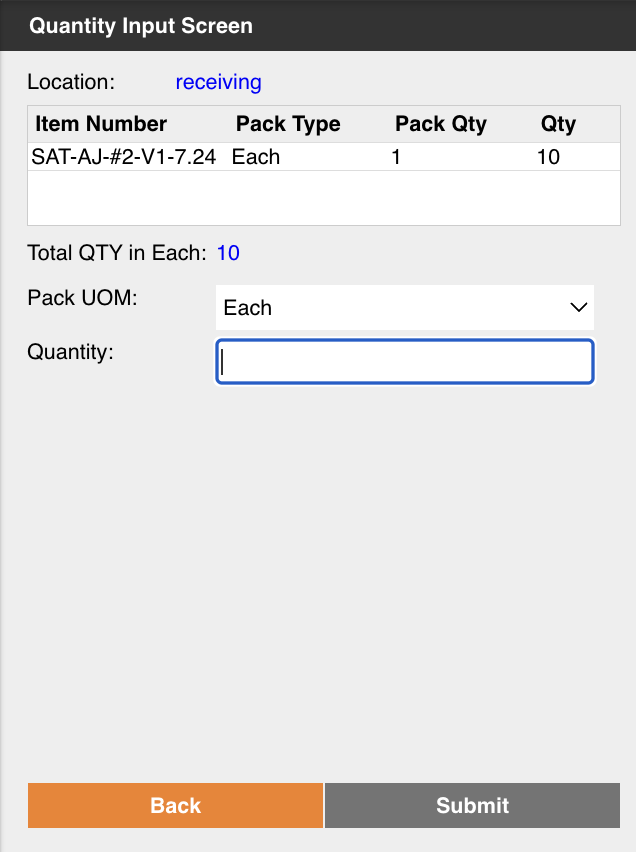
### Process Flow: Putaway Process

The user who is performing a putaway selects the Move Item button from the Inbound menu using a handheld device. The Location Input Screen appears and the user enters or scans the receiving location.

The Move Item Screen appears, showing a list of all the items that currently live in the Receiving location. The user selects the SKU to put away.



The Quantity Input Screen displays and the user enters the quantity to move. The Putaway Location Screen appears showing a list of suggested locations for the selected item. The user can use these suggestions or override them if needed.



After scanning the destination location, the user is returned to the Move Item screen, which allows them to select another item in the Receiving location to putaway to a new location.



### Assumptions

* The Move Item process allows the user to move any quantity of a single SKU number at a time.
* If Omnipress would like to use a process to allow movement of multiple SKU numbers, the Move Stock process can be used.
* The Move Item process will not force users to select one of the suggested locations, users can override the destination location if needed.
* The suggested location logic is described below and will be displayed in the same order
  1. Locations which have an associated Storage record (min/max) for that item
  2. Locations where the item currently resides in the warehouse.
* All screens with location information will display the location number, not the locator.

# Inventory Management

This section outlines all inventory management related processes. This includes cycle counts, physical counts, kitting, inventory adjustments, and replenishments. Deposco is the system of record for inventory tracking.

## Inventory Transactions

The Inventory Ledger provides a detailed list of increases and decreases in inventory quantities for each item in each location of a facility. For example, a credit (increase) for an item is recorded in the Inventory Ledger when stock for the item is received from a purchase order, and a debit (decrease) for an item is recorded when stock for the item is shipped to a customer as an order is fulfilled.

The records in the Inventory Ledger enable you to track changes to stock quantities as they occur throughout the day. Each record in the Inventory Ledger has a positive quantity value. A code of CR indicates that the quantity change is an increase or credit, while a code of DR indicates that the quantity change is a decrease or debit.

The changes that are tracked in the Inventory Ledger are categorized into the following action types:

| **Action Type** | **Description** |
| --- | --- |
| REC | The Receiving process creates a **CR** record when stock is received for a purchase order, inbound shipment, customer return, or inbound transfer order. The UnReceiving process creates a **DR** record with this action type when negative receipt lines are created for stock that was received. |
| SHP | A **DR** record with this action type is created when stock ships for a fulfilled order. |
| ADJ | A record with this action type is created when the Adjust Inventory workflow process is used to create **(CR)** or consume **(DR)** inventory, when the stock quantity for an item is manually adjusted, when a cycle count variance is approved, or during the kitting process when a kit is created. |
| STS | A **DR** record with this action type is created when the inventory status changes from Available to Blocked. A **CR** record with this action type is created when the inventory status changes from Blocked to Available. Inventory has a status of Blocked when any of the following conditions are true:  The storage type of the assigned location is Damaged, Quarantine, or Inspection.  The status of the assigned location is On Hold.  The status of the stock unit is On Hold, Damaged, Inspection, or Quarantine. |

## Inventory Movement

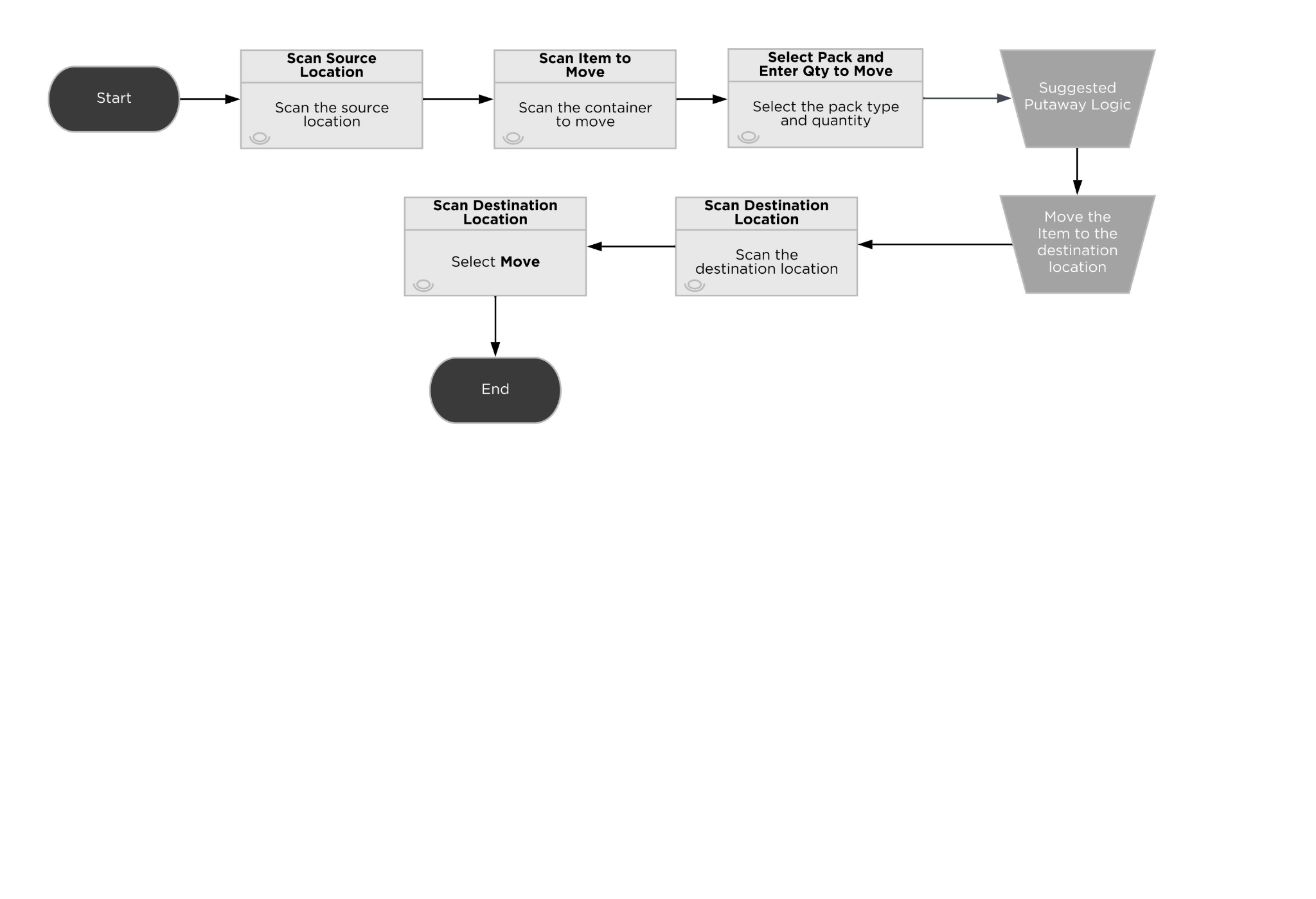
### Move Item

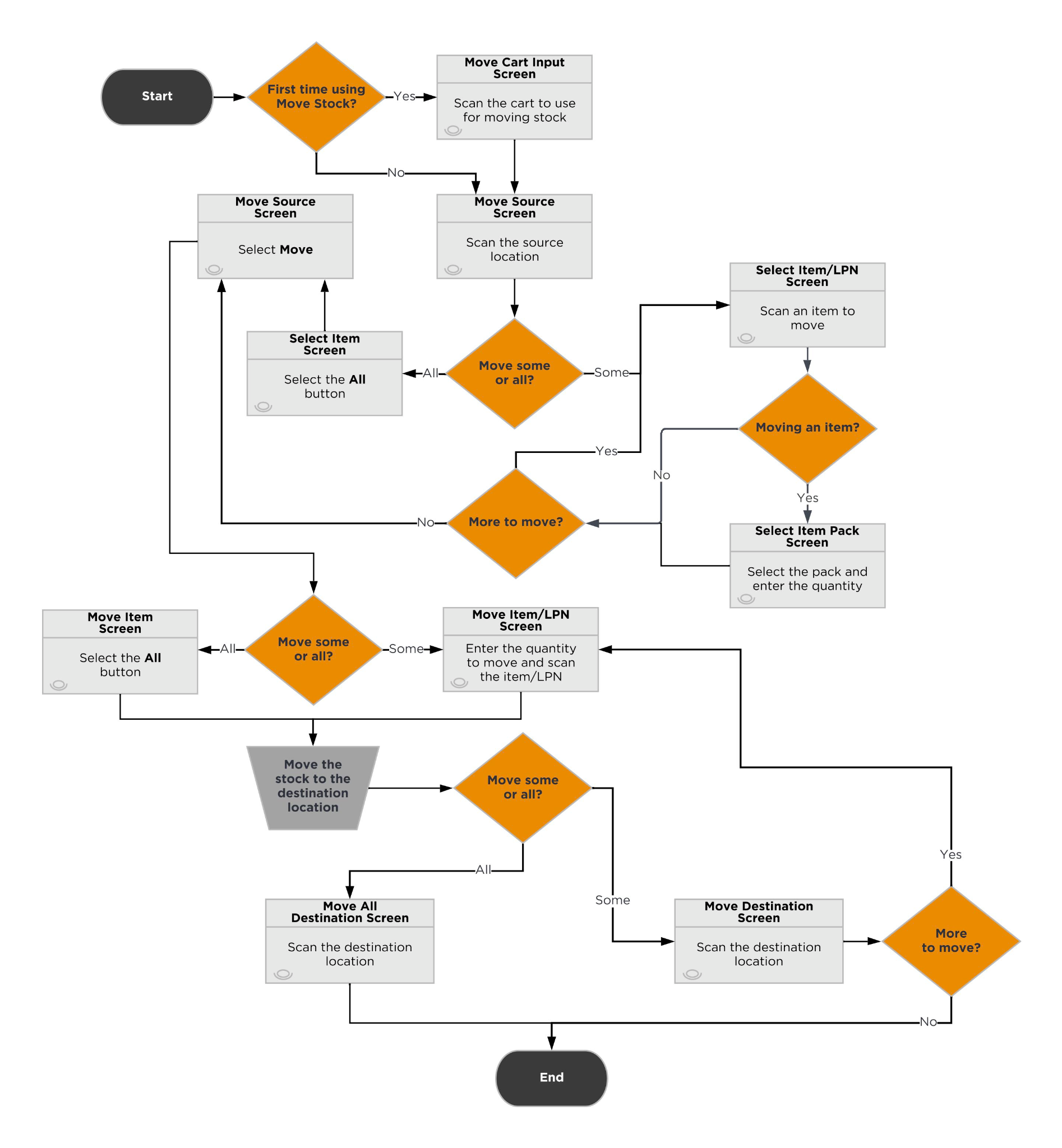
The Move Item process will be used for general inventory movement. The process flow is the same described in the putaway section of this document.

### Move Stock

The Move Stock may be used to move inventory. This process enables the user to load a cart or other equipment to move stock from one or more source locations to one or more destination locations.

### Process Flows

*Move Item*Move Stock

**

### Inventory Movement Assumptions

* The Move Item process will display suggested locations based on the logic described in the [Putaway section](#_heading=h.w3np8li4xxp7) of this document.
* A cart containing loose stock/LPNs can be reused in the Move Stock process. However, a cart containing picked stock/LPNs cannot be used in the Move Stock process.
* Move Stock does not contain Suggested Putaway Logic.
* All screens with location information will display the location number, not the locator.

## Replenishments

Warehouse replenishment is the process of moving inventory from reserve locations to locations used for fulfillment, such as split case picking locations. In general, the warehouse replenishment process results in movement of inventory from source locations to destination locations in the warehouse.

The following options are available for managing the warehouse replenishment process:

### Manual Replenishments

### Strategy

Users who understand when and where to replenish inventory and can perform the tasks without direction from the system can use processes such as Move Item, Move Stock and Move Container to manually perform replenishments or consolidate inventory by moving stock and containers from one location to another.

### Manual Replenishments Assumptions

* Users can put stock away at any location.
* Manual replenishments are performed ad hoc, using the Move processes.
* A validation is in place to prevent mixing multiple SKUs in a location when the location.mixedItemThreshold = “Single Item”.

### Storage-Based Replenishments

In general, the warehouse replenishment process directs the movement of inventory from source locations to destination locations in the warehouse.

A storage record in Deposco assigns an item (specifically the pack record for an item) to a picking location and records the minimum (min) and maximum (max) inventory levels for the item in that picking location. When the quantity of stock falls below the min, a warehouse replenishment task can be created to move inventory from a reserve location to the location identified in the storage record. The task is created only if a replenishment task does not already exist for the item-location record, and only for full pack quantities such that at least one pack is pulled and the max at the destination location is not exceeded.

There are two ways to generate a replenishment task. The first is through a scheduler that defines what set of locations need to be replenished and evaluates eligible locations that can be used to source a replenishment. The second option is to enter the Replenishment process and scan any location with a storage record to determine if a replenishment task needs to be created.

Replenishment tasks are executed in order by task priority.

### Strategy

| **Condition** | **Eligible Locations** | **Result** |
| --- | --- | --- |
| Total Stock Quantity  <  Storage Minimum | Destination locations: locations with a storage detail  Source Locations: locations with the reserved flag checked | Picking locations are replenished by the reserve locations. |

### Replenishment Assumptions

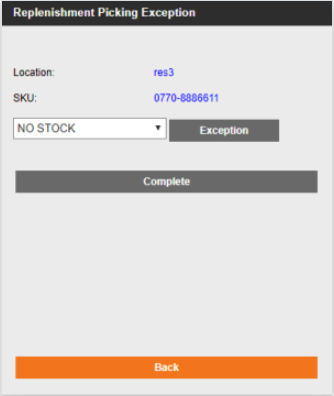
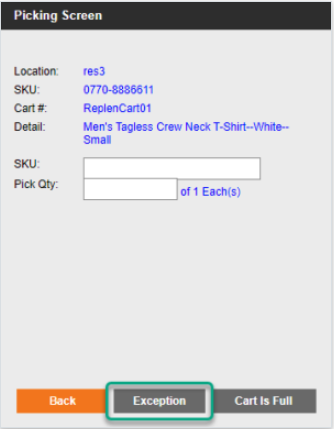
* Omnipress can create scheduler tasks to generate Replenishment tasks at any required cadence.
* The Shippable flag must be checked for the items to replenish
* For the source locations from which stock is picked for replenishment:
  + The Reserve checkbox must be selected.
  + Set the type of location to Permanent.
  + The status of the location must be Ready.
  + The storage type for the location must be Reserve Storage (more storage types can be configured if needed).
* For the destination locations that must be replenished:
  + The Pickable checkbox must be selected.
  + The status of the location must be Ready.
  + The storage type for the location must be Split Case Picking (more storage types can be configured if needed).
  + Omnipress must create a [storage](#_heading=h.thls6u8xl9xb) record for each item and location to replenish, and configure a minimum and maximum threshold.
* All screens with location information will display the location number, not the locator.

### Replenishment Picking Exception

If the requested quantity of items cannot be pulled from the reserve location during replenishment, then the user selects Exception on the Picking Screen.

### Screen Flow

After selecting Exception the user is prompted to select a reason code from the following list: No Stock, Stock Damaged, Stock Unreachable, Suspend Task.



### System Updates

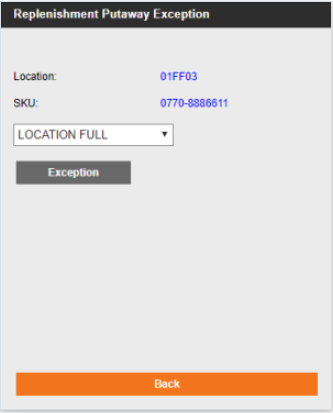
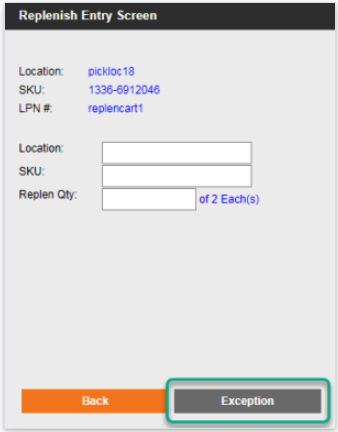
* The replenishment task is set to a status of Exception.
* If available stock is found in other reserve locations, then a new replenishment task is generated, and the user is prompted to pick the stock from the alternate location. (If other replenishment tasks have been assigned to the user, then the user may be directed to pick stock for those tasks before picking stock for the newly generated task.)
* The StockUnit is placed On Hold and a cycle count task is generated to validate the quantity on the location.
* The On Hold status is removed from the location and from the stock unit once the cycle count task has been completed.

### Replenishment Putaway Exception

If the user cannot put away (replenish) the specified quantity of items to the destination location during replenishment, then select Exception on the Replenish Entry Screen.

### Screen Flow

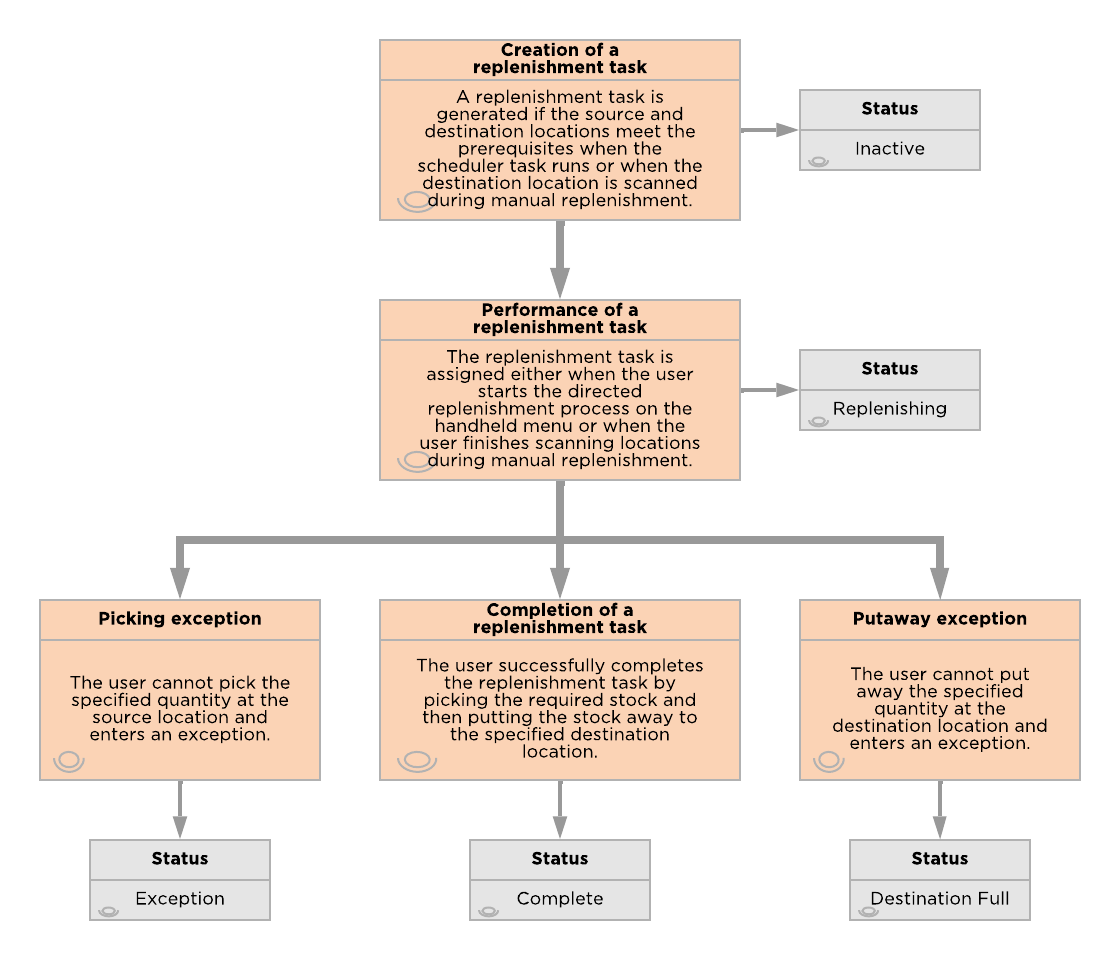
After selecting Exception the user is prompted to select a reason code from the following list: Location Full.



### System Updates

* If Location Full is selected as the exception reason, the replenishment task is set to a status of Destination Full.
* A new (reverse) replenishment task is created to return the stock to the source reserve location. The user is returned to the Replenish Entry Screen and is prompted to scan the source reserve location, the item, and the quantity that is being returned. (If other replenishment tasks have been assigned to the user, then the user may be directed to put away stock for those tasks before returning stock to the source reserve location for the newly generated task.)

### Process Flow



## Cycle Counts

Cycle counts are used to audit the inventory in the warehouse by going to a location and verifying the contents of the location. These can be system directed (Cycle Counts) or user directed (Physical Counts). Changes to inventory made during a count get recorded as Cycle Count or Physical Count variances and can be transmitted to external systems.

Please note that the process of updating stock due to an inventory count consists of two parts: associate users counting stock and admin users [accepting any variances](#_heading=h.3nn0vbqdusov).

### Physical Counts

In general warehousing terminology, a physical count refers to a count in which the entire facility or a zone is put on hold and counted all at once to ensure a complete picture of actual inventory. Despite its name in Deposco, this process does not require a full facility count and is performed on individual locations without having to put fulfillment on hold; this process is simply a count that is user directed (as opposed to the Cycle Counts).

### Strategy

Regular use of Cycle Counting avoids the need for a full shutdown for a physical count and enables counts to be performed by a smaller team of operations. Full physical counts are expensive and require a larger group of users to perform counts to minimize the shutdown time, which tends to make many physical counts inaccurate.

### Cycle Counts

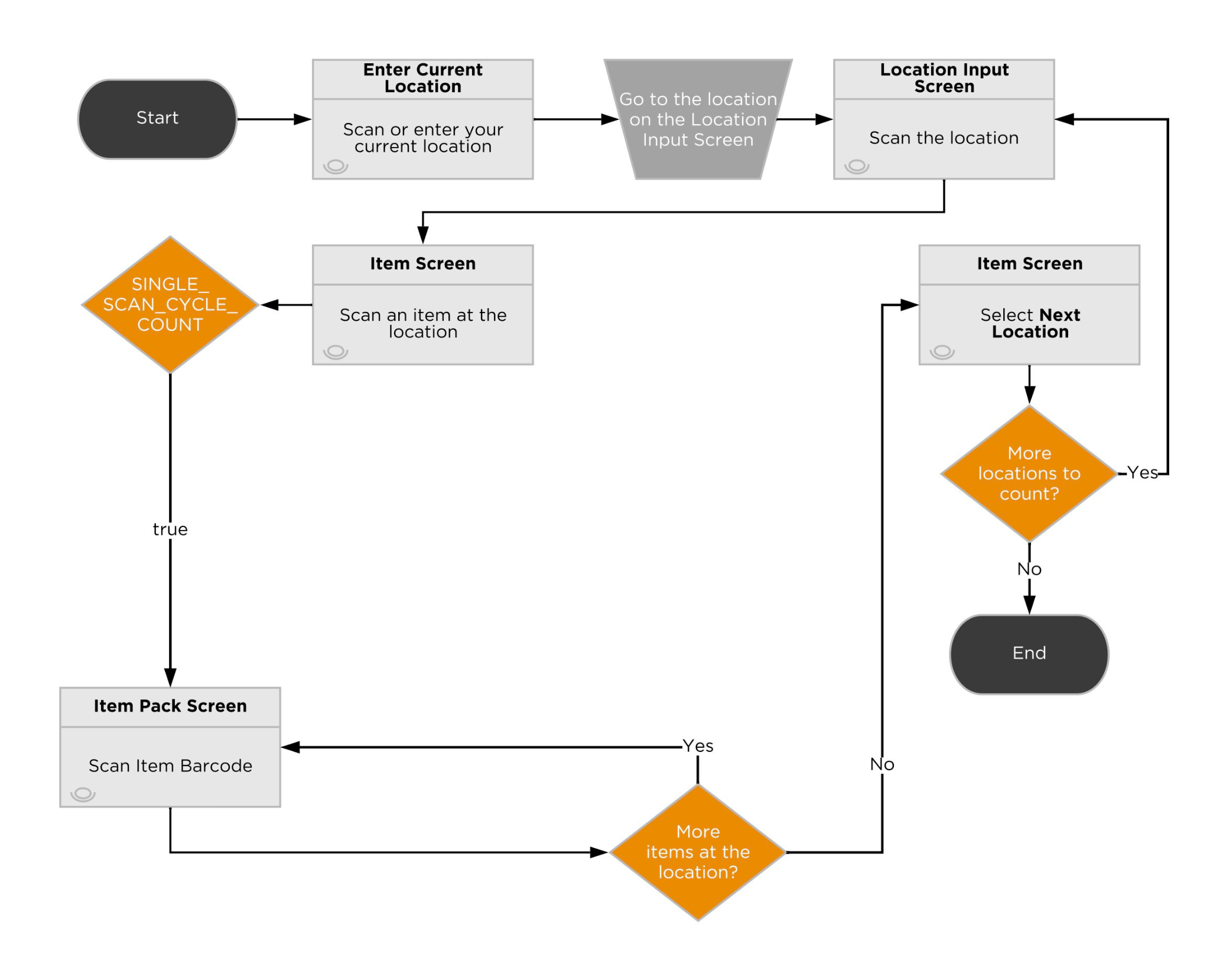
Regular cycle counts help to ensure that the inventory quantity in Deposco is as accurate as possible. Omnipress can create cycle counts by location or by item. During a cycle count, the quantity of inventory at a specific location is counted and then compared against the inventory quantity that is stored in Deposco. If the quantity that is scanned is different from the system quantity (a cycle count variance), then a supervisor needs to review the variance and either accept the quantity entered during the count, reject the quantity entered during the count and use the system quantity, or initiate a recount.

A cycle count is generated automatically for a location if a user enters a picking exception when picking stock at the location. Cycle counts can occur on a scheduled basis for items or locations, be manually created for a location or item, or performed on-demand.

The Cycle Count Queue enables the user to view details for active and completed cycle counts, including the item and location that are counted, the quantity of stock at the location, and the user who performed the count.

Any inventory changes that result from cycle counts are recorded in the Deposco Inventory Ledger.

### Process Flow



### Cycle Count Task Generation

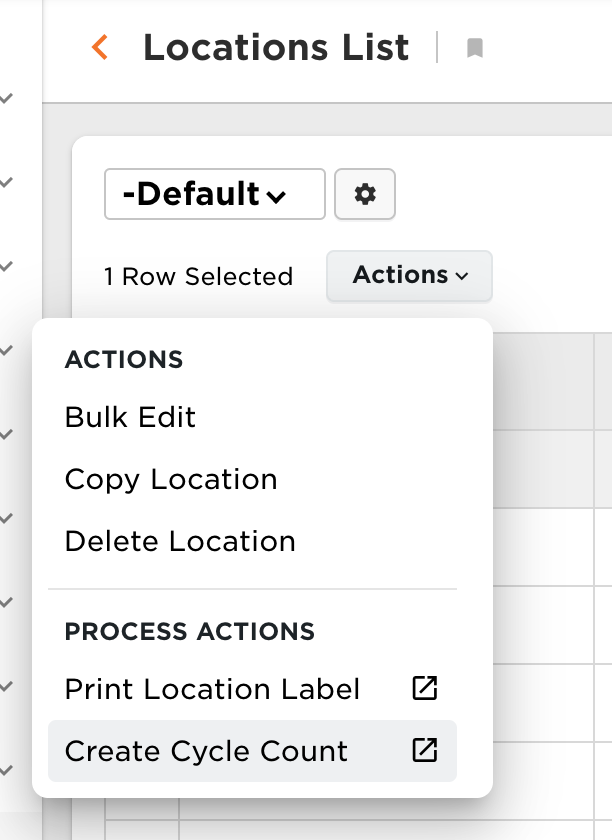
Omnipress will use the following methods to create cycle count tasks:

*Exceptions*

When an exception is created at a location during any picking process or replenishment process, the stockUnit is put on hold and a cycle count task is generated with priority = 0. This Cycle Count Queue task is created with Type = Picking Audit.

*Manual Creation*

* Location or Item (UI): At any time, locations are selected and cycle counts are created using Actions > Create Cycle Count (Shown Below).



These Cycle Count Queue tasks are created with Type = Count by Location (or Item).

* Location or Item (Handheld): At any time, a location can be scheduled for a cycle count using the Create Location Cycle Count process. Users are prompted to scan or enter a location number and a priority to assign to the count. A task is added to the cycle count queue for the scanned location with the given priority. In addition, an item can be scheduled for a cycle count using the Create Item Cycle Count process. A task is added to the cycle count queue for the scanned item with the given priority.
* On Demand Count (Handheld/Location): Users are able to create and execute a cycle count task for a particular location on an as needed basis using the “On Demand cycle Count” button in the Handheld menu. Through the process, a location and priority are entered and the user is immediately afterwards prompted to complete the count of the scanned location.

*System Generated Tasks*

* Cycle Count Profiles and Scheduler tasks are used to schedule the generation of cycle count tasks. Omnipress users own the configuration of Cycle Count Profiles. See below table for details on configuration options for a Cycle Count Profile. More information can be found in the Deposco [help site](https://docs.deposco.com/docs/html/Content/Warehouse_management/Inventory_management/Cycle_counts/Set_up_cycle_counts.htm?tocpath=Warehouse%20management%7CInventory%20management%7CCycle%20counts%7C_____2).

| **Field** | **Value** |
| --- | --- |
| Business Unit | Select the company, tenant company or business unit that is associated with the profile. |
| Type | Select By item to create cycle count tasks for items, or By location to create cycle count tasks for locations. |
| Cycle Count Profile Name | Enter a descriptive name for the profile. |
| Scheduler Task | Select the scheduler task that defines the schedule for when cycle count tasks are generated. |
| Record Selector | If you created a record selector to define a subset of items or locations, then select the record selector. |
| Only Reserve | Select the checkbox to create cycle count tasks only for locations where the Reserved flag is set to true. |
| Only Pickable | Select the checkbox to create cycle count tasks only for locations where the Pickable flag is set to true. |
| Cycle Count Queue Priority | Enter an integer for the priority to assign to cycle count tasks that are generated. You can use the PRIORITY\_LIST SI parameter to define the list of priorities that can be assigned to cycle count tasks. |
| Cycle Count Frequency | Enter the interval in days between cycle counts for an item or location. If the number of days that have passed since the Last Counted Date for the item or location exceeds the specified frequency, then a cycle count task is generated for the item or location. Cycle Count Frequency of 0 disables this functionality. |
| Cycle Count Limit | Enter the maximum number of cycle count tasks to generate each day. The default value is 50. |
| Cycle Count Item Limit | Enter the maximum number of items for which cycle count tasks are generated during a scheduler task run. A task is generated for each item and location. For example, if you set the maximum to 20 and each item is available in 3 different locations, then 60 tasks may be created.  This setting applies only when the Type field is set to By item, and overrides the value set in the Cycle Count Limit field. |

### Cycle Count Execution

Through this process, users are guided throughout the warehouse to complete tasks in the CycleCountQueue based on task priority and the location’s pick sequence. Users blindly count stock units in a given location when they arrive and are prompted to confirm a counted quantity twice only if the initial quantity entered varies from the quantity recorded in the system.

### Cycle Count Assumptions

* All cycle counts will be blind counts. The Cycle Count will not show the quantities associated with the items in a location.
* Users are prompted to recount the location one additional time if they record a variance from the first count until they enter the same value twice in a row.
* Omnipress will primarily perform cycle counts by item.
* Omnipress can utilize the Record Selector functionality to create cycle count tasks for specific business units. The record selector will be tied to a cycle count profile and a scheduler with the option to run manually.
  + A [record selector](https://docs.deposco.com/docs/html/Content/Administration/Record_selectors/Record_selectors.htm?Highlight=record%20selector) enables you to define a data set in Deposco that can be used to create cycle counts for items that belong to a specific business unit.
* Item cycle counts will be scheduled to be created automatically based on inventory volume and movement.
  + High velocity items (Top 100 sku’s) will be counted weekly.
  + All other items will be counted monthly and only if they have been shipped since the last count.
* The maximum number of cycle counts to create is 10. This number can be adjusted as needed.
* All screens with location information will display the location number, not the locator.

## Managing Cycle Count Variances

After cycle counts are completed, any variances are stored in the Manage Cycle Count Variance application. No variance record is recorded for a zero variance count.

### Strategy

From the cycle count variance list, an Admin user must reconcile each variance.

*Accept Variance*

* Adjusts the inventory to the value of the variance of the cycle count.
* Any adjustments that are made in this process are written to the Inventory General Ledger with an Action Type of ADJ

*Reject Variance*

* Leaves inventory levels as they currently stand.

*Recount Variance*

* Triggers the creation of a new cycle count task and leaves inventory levels as they currently stand.

### Managing Cycle Count Variances Assumptions

* If there is a difference between the inventory attributes counted at a location, two variance records are created – an increment for the inventory attribute counted and a decrement for the inventory attribute originally in the location.
* All Cycle Count Variances must be approved by an admin group member.

## Inventory Adjustments

The AdjustInventory process is used to add or remove inventory from a location. This differs from a count in that it is incremental instead of a complete location verification and used primarily for adding found inventory to a location or removing damaged inventory.

### Strategy

The following adjustment processes are available:

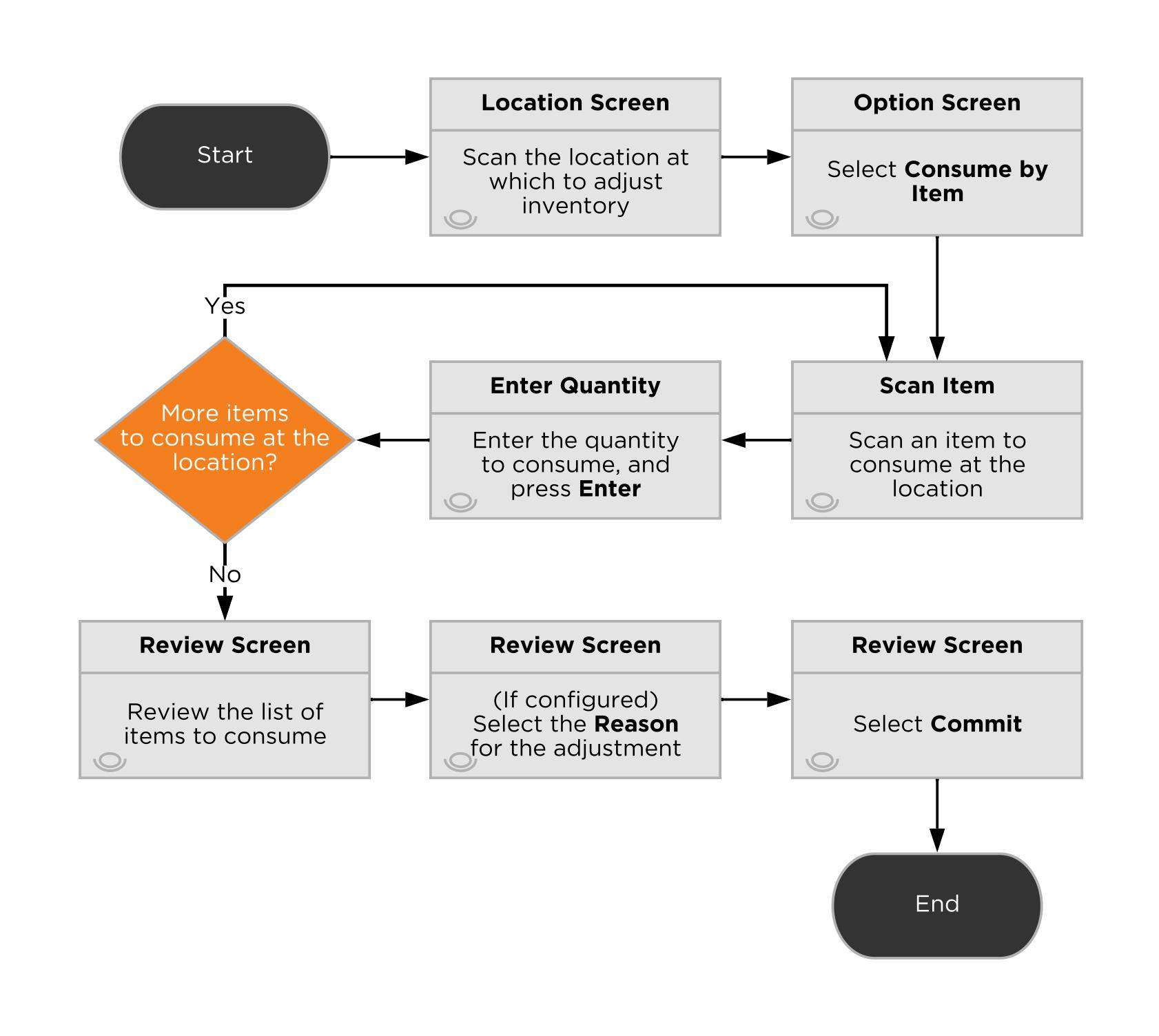
* Consume by Item
* Consume by LPN
* Consume by Location
* Create Inventory

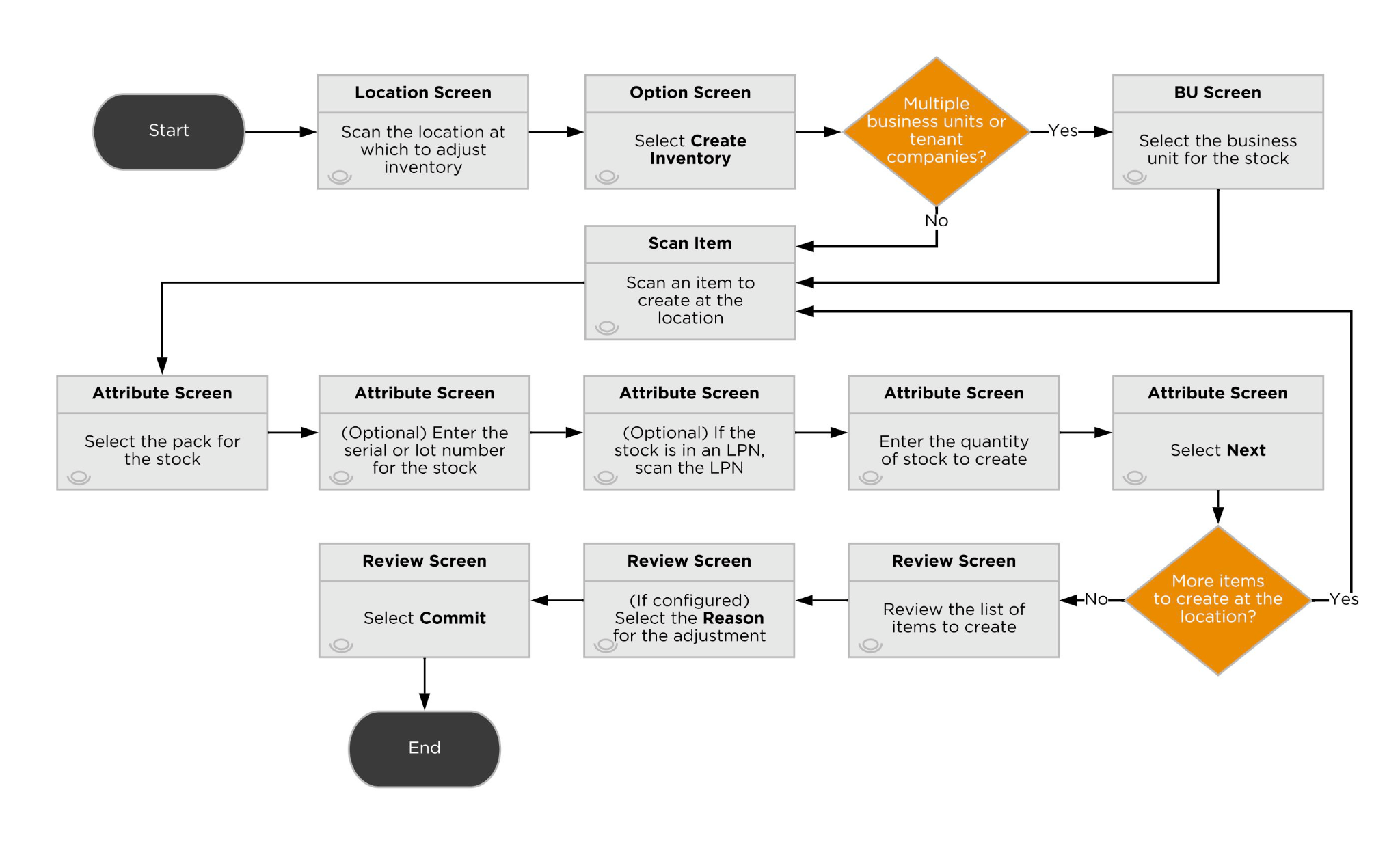
During the adjustment, the user is prompted for an adjustment reason code, which is stored against the transaction so that it can be extracted or transmitted to upstream systems. The following adjustment codes have been identified:

| **Adj. Code** | **Description** |
| --- | --- |
| CCA | Cycle Count Adjustment |
| PIA | Physical Inventory Adjustment |
| DMG | Damaged |
| RET | Returns |
| LST | Lost |
| FND | Found |
| HDL | Handling Error |
| REC | Receiving error |
| SAM | Samples |

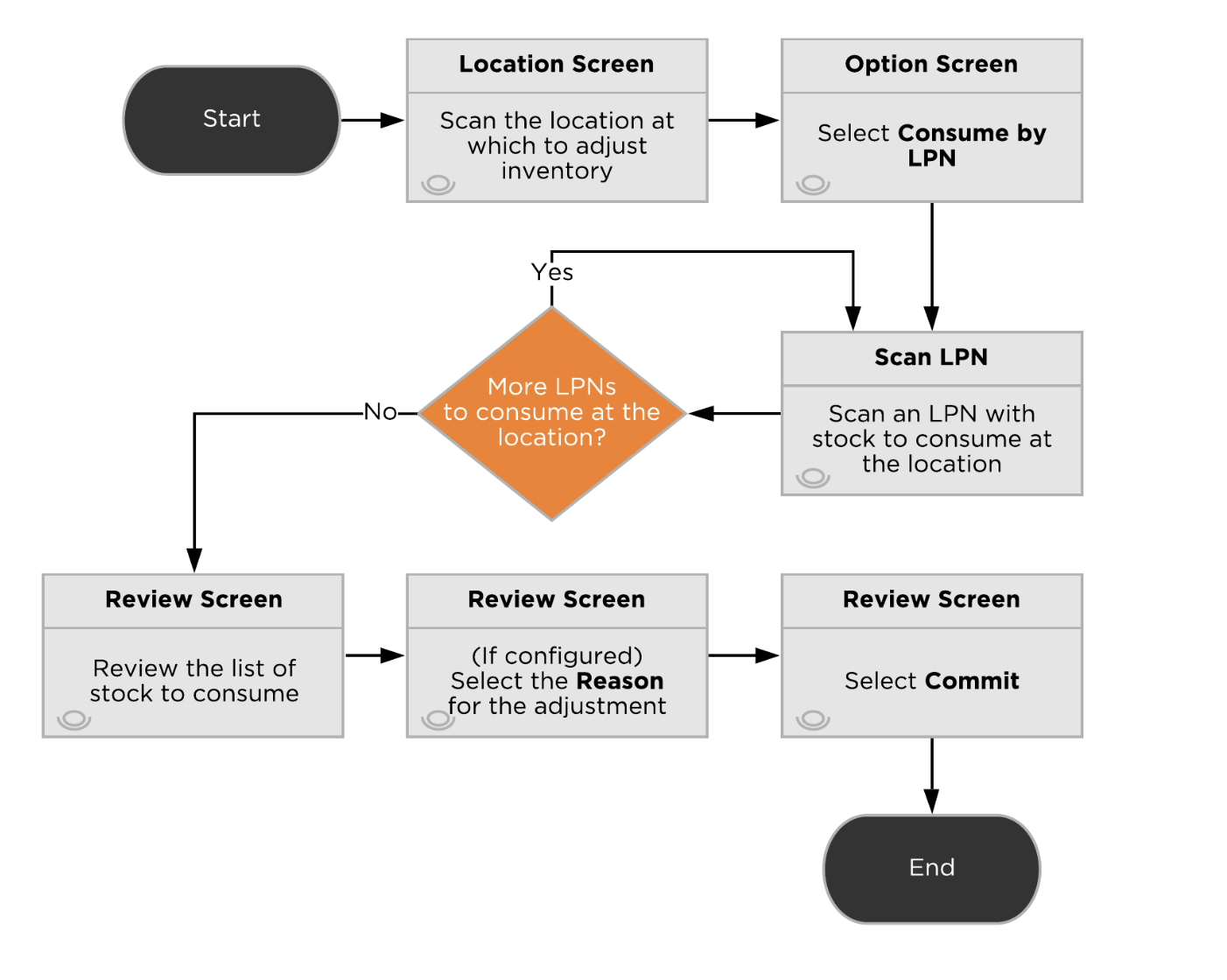
### 

### Process Flow

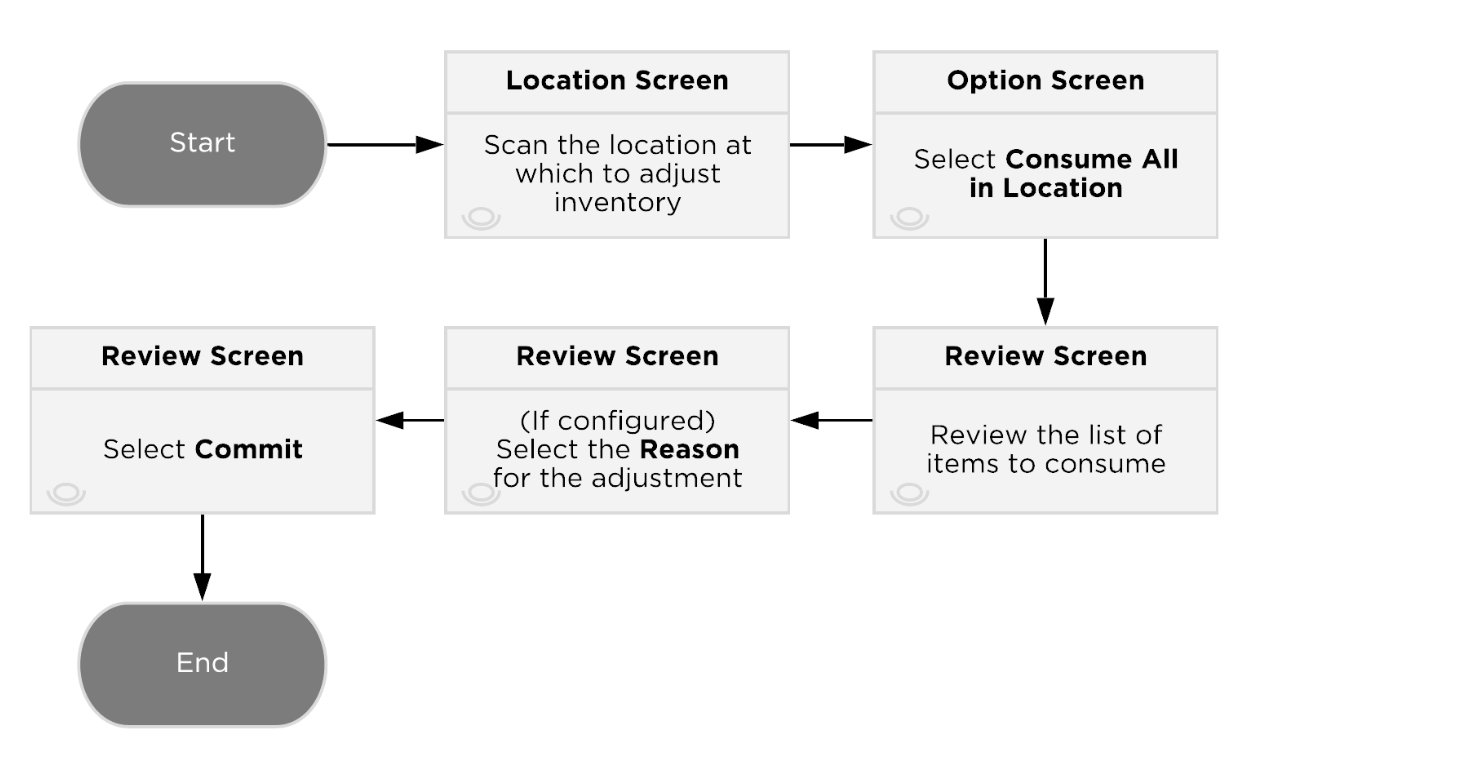
Adjust Inventory: *Consume by Item*Adjust Inventory: *Create Inventory*



Adjust Inventory: *Consume by LPN*



Adjust Inventory: *Consume by Location*



### Key Assumptions

* Inventory Adjustments are for Admin level users only.
* Reason codes are global, across Business Units. There are not any customer specific reason codes.

## Kitting

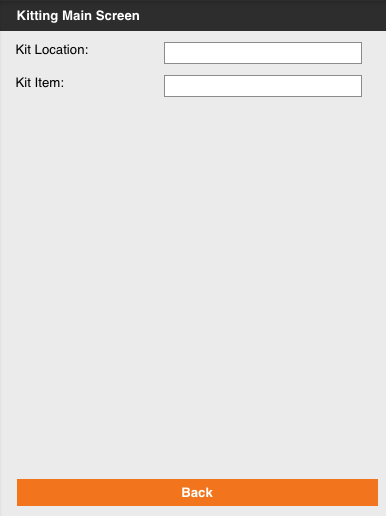
Omnipress will assemble kit items using the Kit To Stock process, which is a kitting functionality that allows for a quick and easy way to build kits and offers some basic tracking methods.

Setup for a Kit occurs at the Item Entity Level. The Output Items and Input Items are all set up as their own item and Component records are created to define the parent item, its components, and the quantity required of each component.

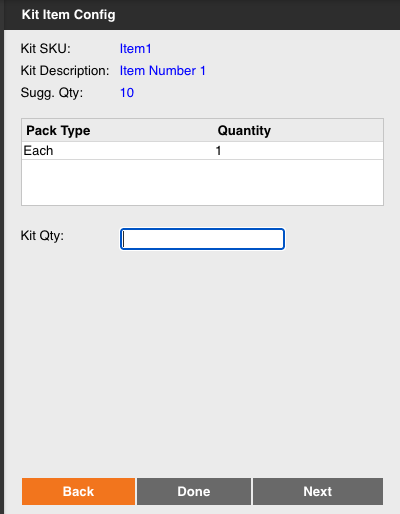
Before starting the process, all the kit components must be in the kitting location.

### Process Flow

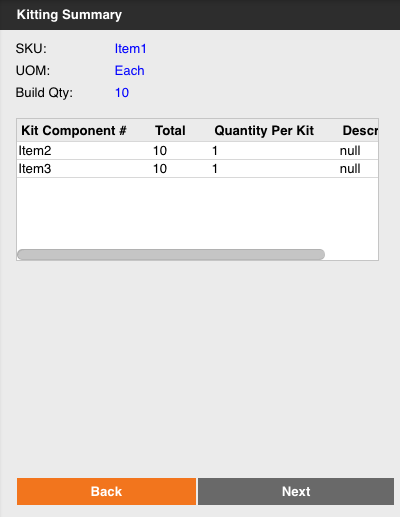
Users navigate to the Kit to Stock process in the handheld menu and enter the location where kitting will take place and the Kit item number.

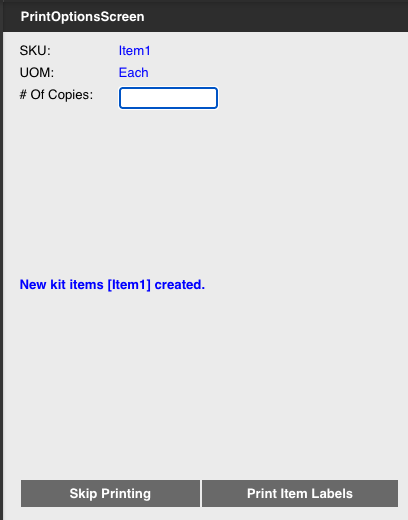


In the Kit Item Config screen, a suggested quantity is displayed based on the total quantity of item components at the kitting location. In the Kit Qty field, the total number of kits to build is entered.



The Kitting Summary screen shows the previously entered Build Quantity and details for the kit components at the kitting location.

  
Users can optionally print item labels and enter the number of copies required.



### Process Flow

* Component records must be created to define the item to build and the component quantity required.
* Parent and component items must have their own item record in the system.
* A kit location exists in the system to perform kitting activity.
* Kit components must exist at the kitting location before initiating the Kit To Stock process.
* No more than the suggested number of kits in the Kit Item Config screen can be built, which is based on components availability.
* Users can optionally print labels for the kitted items.

## Miscellaneous Inventory Processes

### End of Day Inventory

The Daily Item Facility Inventory History provides a snapshot of daily inventory levels for each item, facility, and business unit.

This information is useful for inventory analysts in the following situations:

* To review inventory changes for each item in each facility on a daily basis to detect any unusual activity.
* To review inventory changes over the course of a month or quarter.

The process for generating the Daily Item Facility Inventory runs automatically on a nightly scheduled basis after the copying of data from the core transactional database to the historical data store.

When the process runs, it generates an entry based on each item ID, facility ID, and company ID where stock exists for the item as either a stock unit record for Deposco-managed facilities or an inventory notice record for virtual facilities.

The process also generates an entry if Inventory Ledger activity occurred for the item and facility or if an open purchase order is available for the item and facility.

The process includes only items with inventory tracking of physical stock. In other words, it excludes digital items.

Calculations of inventory quantities are in terms of Each pack types; the process converts quantities for other pack types to the individual unit (Each pack) quantity and then adds them to the Each pack quantity.

Omnipress can enable alerts to notify you if there is a failure during the process of generating the Daily Item Facility Inventory History:

* Inventory History Discrepancies notifies you when there is a system error during calculations and a discrepancy occurs. When you select the link for the notification, a report with a list of discrepancies appears.
* Inventory History Failure notifies you if the Daily Item Facility Inventory History fails to generate.

### Warehouse Lookup

The Warehouse Lookup process allows users to enter warehouse identifiers, such as item barcodes or location barcodes, to display more detailed information about the selected entity. Scan or enter an identifier for the entity, and then key information for the entity appears on the lookup screen, such as the locations, packs, and quantities for an item. This option is purely informational. The following entities are searchable via Warehouse Lookup: item, location, shipment, container, sales order, or purchase order.

### Inventory Calculations

Deposco calculates different types of inventory classifications. Starting with On Hand Inventory, Deposco always displays the total on hand inventory for items. Other calculations like Available to Promise and Available to Release are done in Deposco; adjustments to these calculation queries can be done and require specific approval. These inventory calculations are typically viewed on the Item UI.

| **Field** | **Description** | **Calculation** |
| --- | --- | --- |
| *Total On Hand Qty AP (totalOnHandQty)* | The Total On Hand calculates all inventory that is physically in the warehouse. | Sum of stock unit quantity times pack quantity across all locations |
| *Total ATP Qty API*  *(totalAtp)* | Total Available to Promise (ATP)  This value is passed to any integrated system.  This value is also what is used when releasing orders to the WMS from the OMS. | ATP = On-Hand(Including Receiving locations) - Allocated - Open Order Lines - Damaged |
| *Total ATR Qty API  (totalAtr)* | Available to Release (ATR) is the inventory that is available to release and fulfill Sales Orders.  This value gives insight into what items may need to be replenished to pickable locations. | ATR = On-Hand - Allocated - Unpickable - Damaged |
| *Whse On Back Order API  (qtyOnBO)* | Total quantity on Sales Orders which are backordered due to lack of availability (non-pickable) or another reason. | Orderline quantity where line status is ‘Back Ordered’ |
| *Reserve Loc Qty API (totalReservedQty)* | This is the total inventory of an item residing in Reserve Locations. | Sum of stock unit quantity times pack quantity where reserved=true or storage type = ‘reserve storage’ |
| *Pickable Loc Qty API*  *(pickableStockQty)* | This is the total inventory of an item residing in Pickable Locations. | Sum of stock unit quantity times pack quantity where pickable=true and stock unit is not ‘on hold’ |
| *Open Pick Task Qty API (pickTaskQty)* | Total quantity of an item that is allocated to open/active pick tasks and therefore unavailable to allocate to Sales Orders. | Quantity to be picked for all open/active pick tasks |
| *Open Order Line Qty API (openOrderLineQty)* | Total quantity of an item that is required to fulfill New/Back Ordered order lines and does not include already fulfilled/canceled order line quantities. | Orderline quantity where line status is ‘New’ or ‘Back Ordered’ |
| *Qty On Purchase Order API (qtyOnPO)* | Total quantity of an item that is on Purchase Orders and yet to be received and does not include already received/canceled order line quantities. | Orderline quantity where line status not equal to Closed, ‘Received’, or Canceled’ |

# 

# Outbound

This section outlines all outbound related processes. This includes Customer Order and Sales Order management, wave planning, inventory allocation, picking, packing and shipping.

Customer Orders can be created by either Bright Sockets, Data Exchange, or Customer Order Workbench. Any other method to create customer orders can be added after the Phase 1 implementation.

## Customer Order Management

Order management features provide visibility and life cycle management capabilities for orders from customers to purchase goods from the time they are created through the picking, packing, and shipping process. Sales orders are used for fulfillment processing through waving, picking, packing, and shipping. As a result, sales orders are also referred to as *fulfillment orders*.

Several validations occur during pre-processing of COs:

* External order number must be unique within a single Business Unit and/or Trading Partner.
* Must be at least one CoLine with an item and each pack specified.
* Ship To contact name and Ship To address must be populated for the order
* Shipping Service (ShipVia) is not blank.
* Additional validations for required attributes can be added via business rules.

These validations are to ensure that all necessary information is included for each order. If any of the validations fail, then the order moves to the Review status, and an order management user or client user must manually review the exceptions to determine if the order should be either fixed and reprocessed or canceled.

Reviewing Customer Orders is designated using the Order Management group permissions. An order management profile (CoManagementProfile) is assigned to each CO.

A CO Activity (CoActivity) record is a key activity or important milestone during order processing that is tracked so that you can view the steps that were taken for the CO and when they were taken. These activity records are helpful when inquiring on the status of a CO and troubleshooting an order exception scenario.

Once a CO is released, a sales (fulfillment) order is sent to that source in the warehouse management system (WMS).

### Pre-Processing

Additional actions may also occur to prepare the order for processing and fulfillment. These actions are configured as business rules by the Deposco team. If any of the validations fail, then the order moves to the Review status, and a user must review the exceptions to determine if the order should be either fixed and reprocessed, or canceled.

At this time, no actions are needed by Omnipress to prepare orders for fulfillment.

### Customer Order Statuses

Customer Orders are updated throughout the duration of order fulfillment. The list below provides the possible statuses that can be applied to a Customer Order.

* Draft – A draft order that was created but has not yet been submitted. Draft orders aren't processed through fulfillment until they are submitted and moved to a status of New.
* New – The order is either a new order that was received from outside sales channels or that was manually created and submitted.
* Review – An error or other issue that requires manual intervention occurred during order pre-processing. A user must review the order, and then either fix and reprocess the order or cancel the order.
* Future – An order is placed by a trading partner, but with the expectation that the order is to be released for fulfillment sometime in the future, based on an agreed-upon planned release date. The order remains in Future status until its planned release date.
* Open – The order is ready for fulfillment by pass-through directly to release processing.
* Allocated – The order was processed through routing, and specific fulfillment sources were found to fulfill the order, which resulted in soft allocation of inventory.
* Released – One or more sales (fulfillment) orders were created for the customer order and released to fulfillment locations.
* In Fulfillment – The sales (fulfillment) orders for the customer order have started processing in the fulfillment locations. For example, pick tasks to pick the stock for the sales order were created in Deposco when a pick wave that includes the order was released.
* Canceled – A process or a user canceled the order, and the order won't be processed for fulfillment.
* Complete – Shipping of the order is complete, and the customer order is now complete.
* Archived – The order has reached the archive data retention period and has been moved to long-term storage.

### Process Flow

*Customer Order Life Cycle for Phase 1*

### 

### Exception Management

Bright Order directs COs through automated steps to fulfill COs with the least amount of manual processing. However, validations and actions that are performed during the CO life cycle might direct an order to Review status so that it can be evaluated.

When an order is directed to Review status during order processing, where a pre-processing validation could have failed, a user can review the order data to determine if the order should be either fixed and reprocessed or canceled.

Order management analysts or customers monitor the system for order exceptions that need attention. For each order in review status, CoNotice records provide information about the validation issues that set the order to be reviewed.

After the user clears up any issues with the order data, click the Reprocess Customer Order process action link (from the CoHeader Entity) to have the order pass through the same previous steps. Reprocessing COs in review can be automated as well via the OMS Retry Scheduler.

If the order is in a bad state and should not be processed, the cancel CO process action can be used. All process actions are driven by group permissions to ensure only trained users can cancel and reprocess COs.

### Cancellations and Updates

Order Management users have the ability to reprocess or cancel COs directly from the Deposco User Interface. CO Cancellations are allowed until the Fulfillment Orders are created. After Fulfillment Orders are created, users must cancel from the Fulfillment Order as opposed to the CO.

### Key Assumptions

* Customer orders (COs) are created by Customer Order Workbench, Data Exchange, Shopify, WooCommerce, and Magento sockets.
* The Customer Order External Order Number must be unique within a single Business Unit and/or Trading Partner.
* A Customer Order must have at least one CoLine (order line) with an item and each pack specified.
* A Ship To Contact name and Ship To address must be populated for the order.
* The Shipping Service (Ship Via) field must be populated.
* Bill To address fields are populated.
* Ship via translations are configurable for orders created via Sockets using the Cross References Application.
* Orders can only be updated if the fulfillment order (sales order) has yet to be waved in the WMS.
* Cancellations will be handled at the Sales Order level if the CO is released/fulfillment object has been created.

## Customer Order Workbench

Customer Order Workbench is an application within the Depsoco UI that allows users to manually create customer orders by selecting the customer, the item and quantity for each order line, and delivery settings. Users can also enter other optional information such as notes, dates, or other reference values.

When users finish entering information for the order, they can either save it as a draft so they can continue working on it later, or submit the order for fulfillment.

### Key Assumptions

* Customer Order Workbench settings can be configured by Business Unit/family using [Application Settings.](https://docs.deposco.com/docs/html/Content/Order_management/Enterprise_order_management/Set_up_enterprise_order_management/Set_up_the_Customer_Order_Workbench.htm?Highlight=workbench#ConfigureCustomerOrderWorkbenchsettings)
* Users that are assigned to multiple Business Units/families will need to select the applicable company code before CO creation.
* New customer records can be created by clicking on Create New under the Add Customer to Order section.
* During new customer record creation users have the ability to validate the Ship To address by clicking on the Validate Address button under the Ship To section.
* Customer information is stored in the CxCustomer entity. Existing customer records can be selected during order creation by using the Customer Lookup search bar. Customer data is filtered by the selected Business Unit
* Users can add items to the order by using the Item search bar or the Browse Items button. Item data is filtered by the selected Business Unit.
* Item images will be displayed if available. This feature cannot be removed.
* Omnipress will configure BU specific shipping rates to display the shipping services that are available for a specific Business Unit. During order creation, the user will select the shipping service to use from the Shipping Rates options. After order creation, Deposco will use the value selected as the order shipping service.
* Displayed fields in the Order Detail section can be adjusted by the Omnipress team by editing the Customer Order Order Management layout.
* A Trading Partner can be selected for customers that receive frequent shipments and that require third party shipping for shipping costs.
* Users can specify a Bill To Account for third party shipping.
* A flag field is available for users to indicate if the order is priority/rush.
* Users can save Customer Orders as drafts if needed. This feature cannot be removed.
* The Submit as Paid button in the Payment Method section is used to finalize order creation. This will release the order to the WMS for fulfillment.
* Users will occasionally create orders in advance and select a Planned Release date for the order. A scheduler will run daily to release the orders that have reached the defined date.

## Sales Order Management

Sales orders are used for fulfillment processing through waving, picking, packing, and shipping. As a result, sales orders are also referred to as *fulfillment orders (FO)*. Sales Order (SO) management outlines how SOs are defined, managed, and monitored in Deposco.

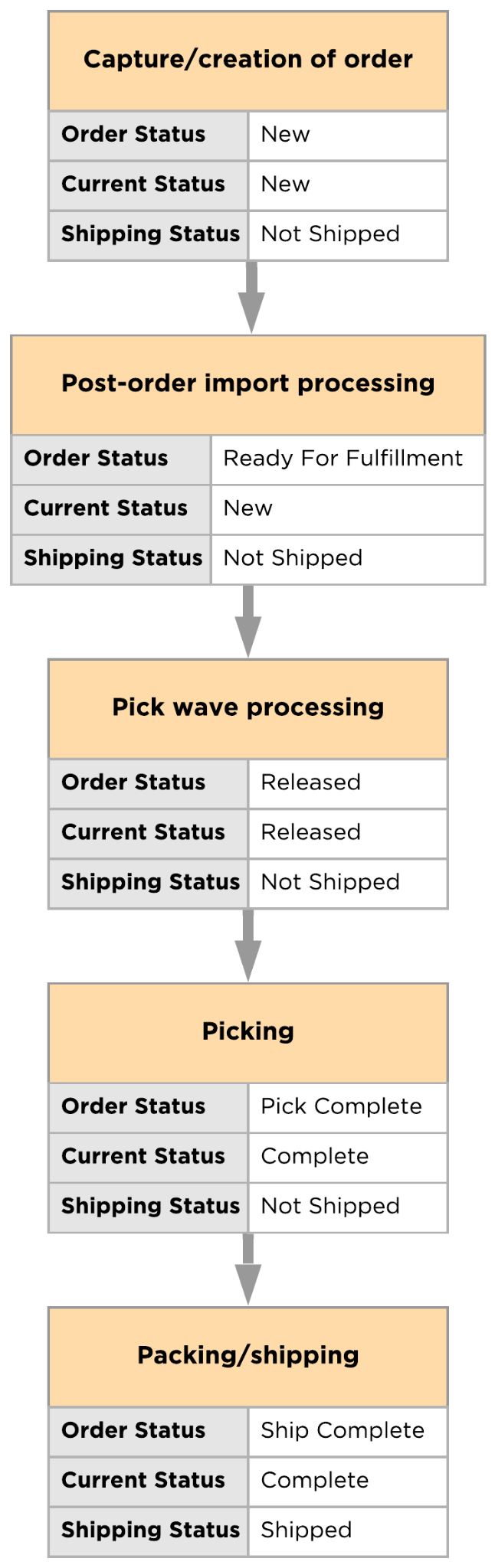
### Strategy

As orders are released from the OMS and created in the WMS, they are classified to determine how they should be processed. This is handled using Automations. Automation activities are outlined in the Outbound [Order Classification section](#_heading=h.mjujd7jf69ub) of this document. Multiple orders (or sometimes individual orders) are grouped into a Pick Wave to determine which orders are processed together; purpose being to increase the efficiency when orders are processed.

Pick Waves are released/allocated, and orders are directly linked to Pick Tasks that specify the item, quantity, and location of the allocated stock for the order. At this point the inventory is hard allocated and cannot be used for other orders. Pick Waves are completed by users using a handheld device, where they scan the inventory to pick and place it into pick bins/carts/pallets, then stage to a location or packing area.

Packing is performed as users scan the pick bins/carts/pallets, and packing documentation and shipping labels are automatically printed. The packed boxes are closed automatically. For freight packing and shipping, users will assign picked containers or LPNs onto a trip/load.

*Life Cycle of a Sales Order*

**

### 

### Sales Order Statuses

There are three fields used on the Sales Order Header to provide information on the status of the order. Each status field is listed below with a table providing the different status values and their definition.

### Order Status

The Order status is the overall status of the order based on order activity tracking as the order progresses through the fulfillment process. It provides a more granular view than the Current Status field.

| **Status** | **Description** |
| --- | --- |
| New | The order is a new order that was either received from an external system or manually created and submitted. |
| Hold | The order has been processed through post-order import, failed one or more validation checks, and requires review. |
| Ready For Fulfillment | The order has been processed through post-order import and is ready for fulfillment. |
| Planned | The order has been added to a pick wave as the pick wave was built. |
| Partially Released | When the pick wave for the order was released, stock was allocated to partially or completely fulfill at least one but not all order lines for the order. |
| Released | When the pick wave for the order was released, stock was allocated to fulfill the order. |
| Back Ordered | When the pick wave for the order was released, there was not enough stock to fulfill one or more of the order lines for the order. |
| Picking | A work group that includes pick tasks that are associated with the order has been assigned to a user, and the user has picked at least one stock unit to fulfill the order. |
| Pick Complete | All pick tasks to fulfill the order have been completed. |
| Canceled | A process or a user canceled the order, and the order will not be fulfilled. |
| Partially Staged | Some of the stock to fulfill the order has been picked and packed, but it has not yet shipped. |
| Staged | All of the stock to fulfill the order has been picked and packed, but it has not yet shipped. |
| Partially Loaded | For freight shipping scenarios, some of the stock to fulfill the order has been picked, packed, and loaded onto a truck, but the trip has not yet been closed. |
| Loaded | For freight shipping scenarios, all of the stock to fulfill the order has been picked, packed, and loaded onto a truck, but the trip has not yet been closed. |
| Partially Shipped | Some of the stock to fulfill the order has shipped, and the shipment (or the trip, for freight shipping scenarios) has been closed. Other parts of the order are either being picked or are back-ordered. |
| Ship Complete | All order lines have been fulfilled and shipped, and the shipment (or the trip, for freight shipping scenarios) has been closed. An order may also move to this status if one or more order lines were canceled but the rest were successfully fulfilled. |

### 

### Current Status

The Current status is the overall status of the order as the order progresses through the fulfillment process.

| **Status** | **Description** |
| --- | --- |
| New | The order is a new order that was either received from an external system or manually created and submitted. |
| Hold | The order has been processed through post-order import, failed one or more validation checks, and requires review. |
| Released | The order was added to a pick wave, and stock was allocated to fulfill the order when the pick wave was released. |
| Back Ordered | The order was added to a pick wave when the pick wave was built. However, when the wave was released, there was not enough stock to fulfill one or more of the order lines for the order. |
| Picking | A work group that includes pick tasks that are associated with the order has been assigned to a user so that the user can pick stock to fulfill the order. |
| Canceled | A process or a user canceled the order, and the order will not be fulfilled. |
| Complete | All pick tasks to fulfill the order have been completed. |
| Closed | All pick tasks to fulfill the order have been completed, but some or all of the order was returned. |

### Shipping Status

The Shipping status indicates whether the order has not yet shipped, part of the order has shipped, or all of the order has shipped.

| **Status** | **Description** |
| --- | --- |
| Not Shipped | No shipments have been created for fulfillment of the order. |
| Partially Shipped | At least one shipment has been created for fulfillment of the order, but one or more order lines have not yet been fulfilled. |
| Shipped | All order lines have either been fulfilled and shipped or canceled. |

### Order Cancelations

Sales Orders can be cancelled either by using a process action link (PAL) on the Order Header List or by using an option on the handheld menu.

If a sales order is associated with a customer order, then a Cancel Confirmation page appears with the associated customer order number, and notifies you that the order will be rerouted for fulfillment if possible. Click Confirm to continue.

### Backorder Management

Back orders are managed in the WMS. A Sales Order is back ordered depending on the [Order Fulfillment Profile configuration](#_heading=h.k339dkdp4h6n).

If an exception is created during picking and no additional pickable stock is available for allocation, then the order status is updated to “Back Ordered”. At this point, the Omnipress team would need to Replenish picking locations or determine if the order should be cancelled.

### Order Classification

Order classification encompasses the pre-fulfillment activities that occur when a Sales Order is created during Automations. These actions are performed automatically, by the system, as orders are created in the WMS. This includes data cleansing, and order profiling, which are all detailed in the below sections.

### Data Cleansing And Order Profiling

* UPS address validation for domestic orders
* Orders with a USPS shipping service will be assigned an order priority of 0, to ensure that they are processed first and are ready before the USPS pickup time.
* A Trading Partner is assigned to orders that are shipped to frequent customers (e.g. book stores) that use Third Party shipping.
* The Omnipress team can use Sales Order Automations to set values for attributes required for order processing such as Freight Terms Type, Order Priority, and Rush Order flag.
* When Automations is completed, the field Automations Complete will be set to True to indicate that the order was processed successfully.

## Waving

Pick waves organize the daily flow of outbound order fulfillment work within a warehouse or distribution center. Waving is an application of short-interval scheduling that selects a set of orders based on properties like planned release or ship time, then releases the work needed for those orders in an organized fashion, so that the work can be performed efficiently by a set of users within a work shift.

Waving enables management to monitor and manage performance throughput throughout the day so that problems can be addressed quickly and staffing can be more effectively utilized throughout the shift.

### Wave Profiles

Waves are grouped as follows:

* Priority Orders
* Identical Orders
* Singles Orders
* Multis Orders
* Partial Allocation
* Freight Orders

Example wave numbers:

| **Wave Type** | **Name Format** | **Example** |
| --- | --- | --- |
| Singles | Singles-[facility]–[date]-[build time]-[counter] | Singles*-OMNI01–250807\_08:33:35-1* |
| Identicals | ID-[Batch-ID]-[facility]–[date]-[build time]-[counter] | ID-123*-OMNI01–250807\_08:33:35-1* |
| Multis | Multis-[facility]–[date]-[build time]-[counter] | Multis*-OMNI01–250807\_08:33:35-1* |
| Partials | Partial-[facility]–[date]-[build time]-[counter] | *Partial-OMNI01–250807\_08:33:35-1* |
| Priority | -Priority-[facility]-[date]-[build time]-[counter] | *-Priority-OMNI01-250807\_08:33:35-1* |
| Freight | Freight-[facility]–[date]-[build time]-[counter] | *Freight-OMNI01–250807\_08:33:35-1* |

Each wave type is described in detail below.

### Priority Orders

Priority orders are processed together on a Priority wave and will be picked into individual bins to separate orders and facilitate the packing process.

Conditions for a Sales Order to classify for the Priority Parcel profile

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* Order has the Priority flag checked or the Priority field is set to “0” (highest priority).
* Order is in a New or Back Ordered Status.
* Order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by the created date.
* The maximum number of orders to allocate is the number of bins that can fit on a cart (4).
* Orders will be back ordered if anything is out of stock i.e. there is not enough stock in pickable locations.
* Both Kit to Order and Kit to Stock items will be considered for picking.
* The Omnipress team can further break down priority order waves into the groups below to leverage other picking and packing processes.
  + Priority identical orders
  + Priority Singles orders
  + Priority Multis orders

Picking process: [Batch Picking](#_heading=h.krfgyclfnv4l)

Packing process: [Single Scan Packing](#_heading=h.o0l35kotfc2u)

### Singles Orders

Single line single unit orders are processed together on a Singles wave; this allows for increased picking efficiency since pickers can pick multiple orders/items into one container while traveling, without the need to separate the inventory by order.

Conditions for a Sales Order to classify for this profile:

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* The order has a single order line and the ordered quantity is 1.
* Order is in a New or Back Ordered Status.
* The order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by priority first and then by the created date.
* The maximum number of orders to allocate is 20.
* Orders will be back ordered if anything is out of stock i.e. there is not enough stock in pickable locations.
* Only already assembled kits will be considered for picking. Orders containing kit to order items will be excluded.

Picking process: [Single Scan Picking](#_heading=h.gv7ns5wijvnu)

Packing process: [Pack Order By Item](#_heading=h.dy11zi8fe0ds)

### Identical Orders

Identical orders are processed together on an ID wave; this allows for increased picking efficiency since pickers can pick multiple orders/items into one container while traveling, without the need to separate the inventory by order.

Conditions for a Sales Order to classify for this profile:

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* The order has a batch ID that matches many others currently in the system.
* Order is in a New or Back Ordered Status.
* The order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by priority first and then by the created date.
* The maximum number of orders to allocate is 100, the minimum is 10.
* Orders will be back ordered if anything is out of stock i.e. there is not enough stock in pickable locations.
* Both Kit to Order and Kit to Stock items will be considered for picking.

Picking process: [Single Scan Picking](#_heading=h.gv7ns5wijvnu)

Packing process: [Bulk Packing](#_heading=h.u1x2cgwokfkn)

### Multi Line Orders

Multi line orders are processed together on a Multis wave and will be picked into individual bins to separate orders and facilitate the packing process.

Conditions for a Sales Order to classify for the Multis Parcel profile

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* Order has more than one order line.
* Order is in a New or Back Ordered Status.
* Order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by priority first and then by the created date.
* The maximum number of orders to allocate is the number of picking bins that fit in a cart (4).
* Orders will be back ordered if anything is out of stock i.e. there is not enough stock in pickable locations.
* Both Kit to Order and Kit to Stock items will be considered for picking.

Picking process: [Batch Picking](#_heading=h.krfgyclfnv4l)

Packing process: [Single Scan Packing](#_heading=h.o0l35kotfc2u)

### Partial Allocation Orders

Occasionally, Omnipress will ship partial orders, that is, an order will be fulfilled in multiple installments, if stock for the entire order is not available. To accommodate for this need, a partial allocation profile will be available. Users will select the orders that need to be partially allocated and run the scheduler manually.

Conditions for a Sales Order to classify for the Partial Parcel profile

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* Order is in a New or Back Ordered Status.
* Order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by priority first and then by the created date.
* There is no minimum or maximum number of orders to allocate.
* Back order policy: allocate available stock, back order the remainder.
* Both Kit to Order and Kit to Stock items will be considered for picking.

Picking process: [Batch Picking](#_heading=h.krfgyclfnv4l)

Packing process: [Single Scan Packing](#_heading=h.o0l35kotfc2u)

### Freight Orders

Freight orders are processed individually on a Multi Parcel wave and will be picked using any picking container.

Conditions for a Sales Order to classify for the Freight profile:

* Automations Complete field is true, i.e. the order went through data cleansing and order profiling successfully.
* Order has a freight ship via.
* Order is in a New or Back Ordered Status.
* Order is not already associated with a Pickwave.

Inventory will be allocated as described below

* Orders will be sorted by priority first and then by the created date.
* The maximum number of orders to allocate is 1.
* Orders will be back ordered if anything is out of stock i.e. there is not enough stock in pickable locations.
* Both Kit to Order and Kit to Stock items will be considered.

Picking process: [Single Scan Picking](#_heading=h.gv7ns5wijvnu).

Packing process: [Repalletize](#_heading=h.dko8sfxxmx7b) and [Assign Trip to Container](#_heading=h.oouo6lm1ud9w).

### Allocation/Wave Release

Orders will be automatically waved and released via a scheduler configured by the Omnipress team. They can also be manually released, if needed.

The scheduler tasks can be run manually at any time throughout the day if additional waves need to be released (shown below).

Should the allocation of stock have to be undone before picking begins, a user can enter any wave in Released status and select Actions > Undo Release (Shown Below). After picking has begun, the allocation of stock cannot be rolled back.

## Picking

### Single Scan Picking

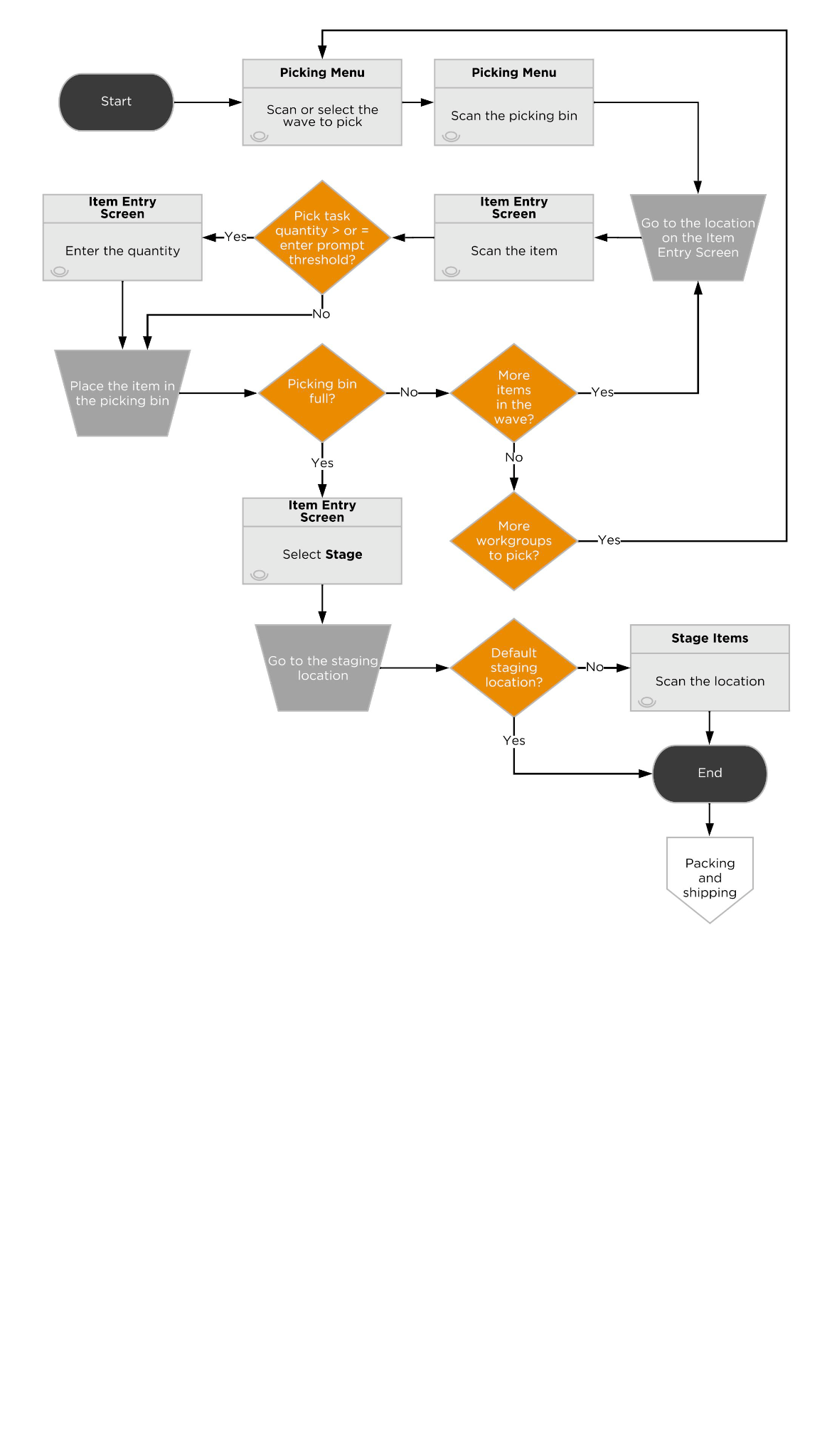
Single Scan Picking is used to pick:

* Singles orders
* Identical orders
* Freight orders

With Single Scan Picking, which is also called bulk picking, a group of orders is picked at the same time and placed into a single large bin.

Each item in an order, or order line, is considered a separate pick task. Pick tasks are grouped within a pick wave into work groups based on the zones in which the pick tasks must occur. Each picker is assigned a work group and is directed through the picking process along the most efficient pick path to complete all pick tasks for the work group.

### Process Flow



### Single Scan Picking Assumptions

* For Single Line Single Unit waves, pickers will use a new picking bin before picking a new item number. For example, if the wave contains pick tasks for ItemA and ItemB, the system will direct the user to complete tasks for itemA first and when the user is instructed to pick ItemB, a new picking bin will be entered to ensure ItemA and ItemB aren’t mixed in the same bin. This will facilitate the Bulk Packing process.
* For Freight waves, users will pick one order at a time.
* Picking bins must be empty to be used for picking.
* Waves will be sorted by priority and then by creation date.
* [Exception handling](#_heading=h.dmbf5oxkhcns) is defined in a separate section of this document.
* Users scan the picking location, item to pick, and quantity to pick.
* Grocery scanning is not required, users will enter the total quantity picked instead.
* Picking bins are automatically staged at the shipping location.
* All screens with location information will display the location number, not the locator.

### Batch Picking (multis parcel picking)

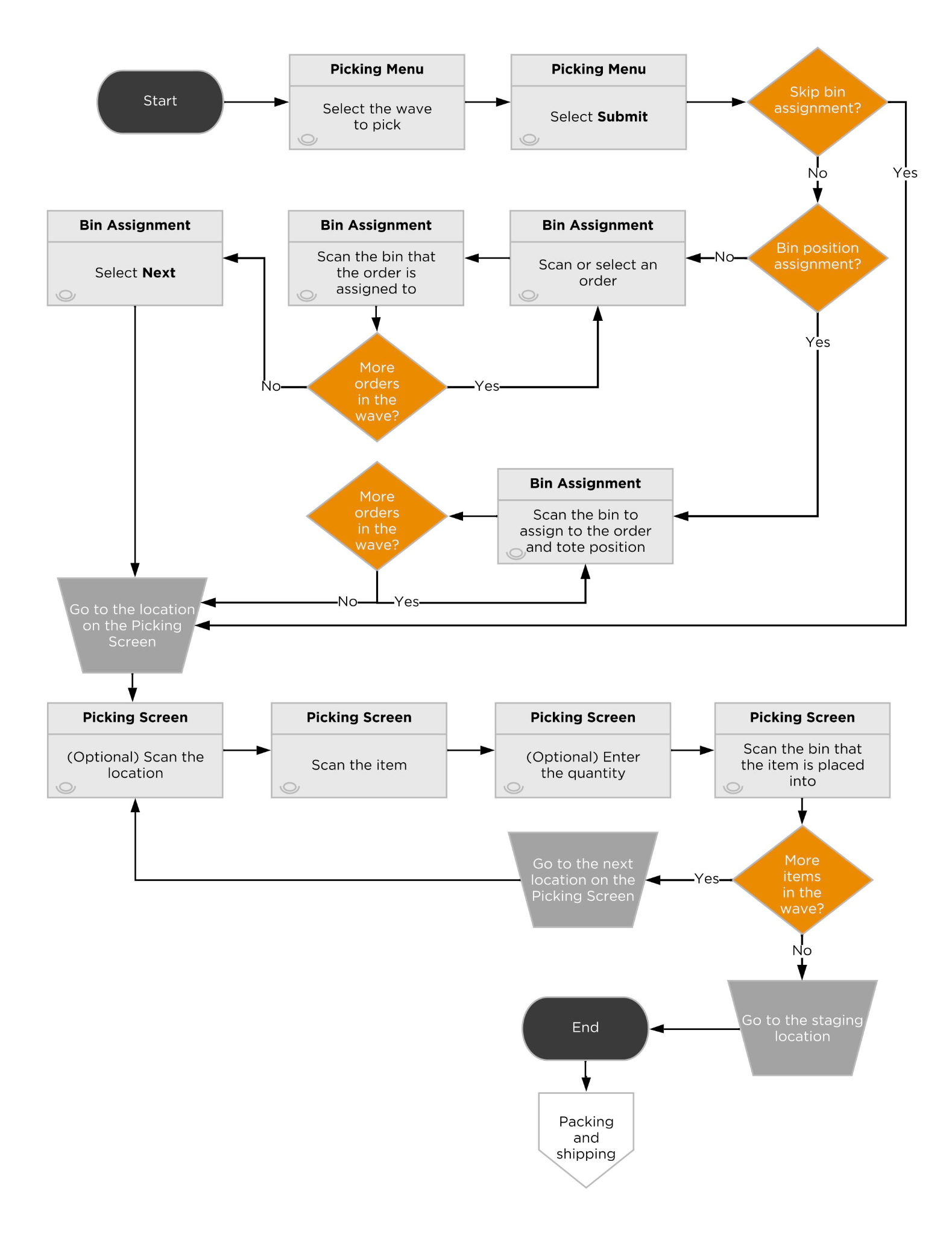
Batch Picking is used to pick:

* Multis orders
* Partial orders

With Batch Picking, a group of orders is picked at the same time and sorted into reusable or transient bins on carts or similar equipment, which provides preliminary staging for packing and shipping of each order.

Batch Picking is typically used for parcel orders and is most efficient when there are multiple orders with the same item or group of items that should ship at the same time. The order picker can then travel to a pick location once to pick items for multiple orders at the same time into different containers, which increases productivity by reducing the overall travel time for a picker.

### Process Flow



### 

### Screen Flow

### Batch Picking Assumptions

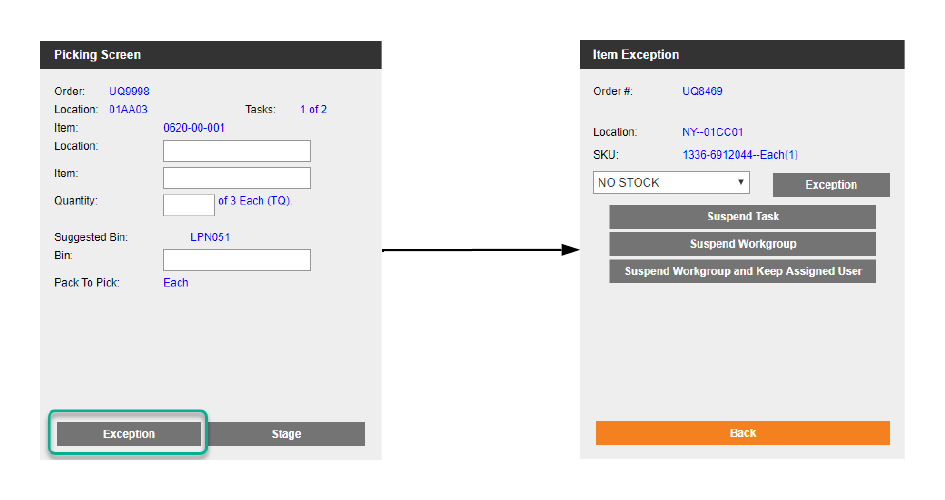
* Multis Picking is accomplished using the Batch Picking process.
* Pick waves are sorted based on order priority and then by the created date.
* Batch Picking will not require grocery scanning, users will enter the total picked quantity instead.
* Each picking screen that includes a bin scan will display “Suggested LPN”
* After picking is completed containers and inventory are staged to the Shipping staging location by the users.

### Exception Handling

If you cannot pick the requested number of items at the specified location during Picking, or if you need to skip the current pick task or suspend work on this work group (for example, if you need to go on a break), then select Exception on the Picking Screen. The different options available to the picking user within the picking exception process are detailed in the below table.

| **Action** | **Reason** | **Steps** | **Results** | **Next Steps** |
| --- | --- | --- | --- | --- |
| Skip the Pick Task | You want to skip the pick task. | Select Suspend Task. | The status of the pick task changes to Suspended. | You are returned to the Picking Screen and prompted to pick the next item in the work group. After the remaining pick tasks for the work group are complete, you are prompted again to pick the item that you skipped. |
| Temporarily stop work on the pick wave | You want to temporarily stop work on the pick wave, but plan to return and complete the pick wave later in the shift. | Select Suspend Workgroup and Keep Assigned User. | The status of the work group for the pick wave changes to Suspended, and remains assigned to the same user. A WorkGroup suspended message appears, and you are returned to the Picking Menu screen. Select the wave again to continue picking the wave. | When you are ready to resume work on the pick wave, select Outbound > Batch Picking from the main menu, select the pick wave to resume on the Picking Menu page, and then complete the remaining pick tasks. Only the assigned user can continue work on the work group. |
| Permanently stop work on the pick wave | You want to stop work on the pick wave and allow another user to complete the pick wave. | Select Suspend Workgroup. | The status of the work group for the pick wave changes to Suspended, and the assigned user is removed from the work group. | A WorkGroup suspended message appears, and you are returned to the Picking Menu screen. You or another user can select the pick wave and complete the remaining pick tasks. |
| Exception > Other, Damaged, No Stock | Stock is available at the specified location and is not damaged, but cannot be picked for some other reason. | Select OTHER, DAMAGED, or NO STOCK from the list, and then select Exception. Depending on the configuration (if VALIDATE\_EXCEPTION\_PRIVILEGE=true), you may be prompted to enter administrator credentials to proceed with the exception. | Selecting OTHER, DAMAGED, or NO STOCK directs the system to find all pick tasks for this pick wave that are pointed to the same location where the exception was requested, even if other pick tasks for this location are for different items, and moves those pick tasks to a status of Exception. After the picker completes this work group, the pick tasks with an Exception status are deleted, which enables the order line(s) to be included in a future pick wave so that the order can be completed.  If available stock was found in other locations, the system generates new pick tasks under the same work group, and the picking user is directed to those locations for picking.  A cycle count is generated for the location, if cycle counts are enabled for the location.  An EXCEPTION entry is written to the activity log. | You are returned to the Picking Screen and prompted to pick the next item in the work group. If the exception was for the last item in the work group, then you are returned to the Picking Menu to start work on the next work group. For new pick tasks generated from an exception where stock was found in other locations, you are prompted to pick from these locations. As with other pick tasks, you can create an exception for the new pick task if necessary. A cycle count should be performed for the location flagged with an exception. When the cycle count is completed (or when a supervisor reviews the cycle count results), the On Hold status for the location and/or stock unit is cleared or set to Ready. A supervisor can also manually clear or reset the status of the location or stock unit; however, resolution through a cycle count is recommended. |

### Screen Flow



## Packing

The packing processes in Deposco are used to confirm exactly what is being shipped, confirm items and quantities, and print documentation such as packing slips, shipping labels, etc.

### Pack Order By Item

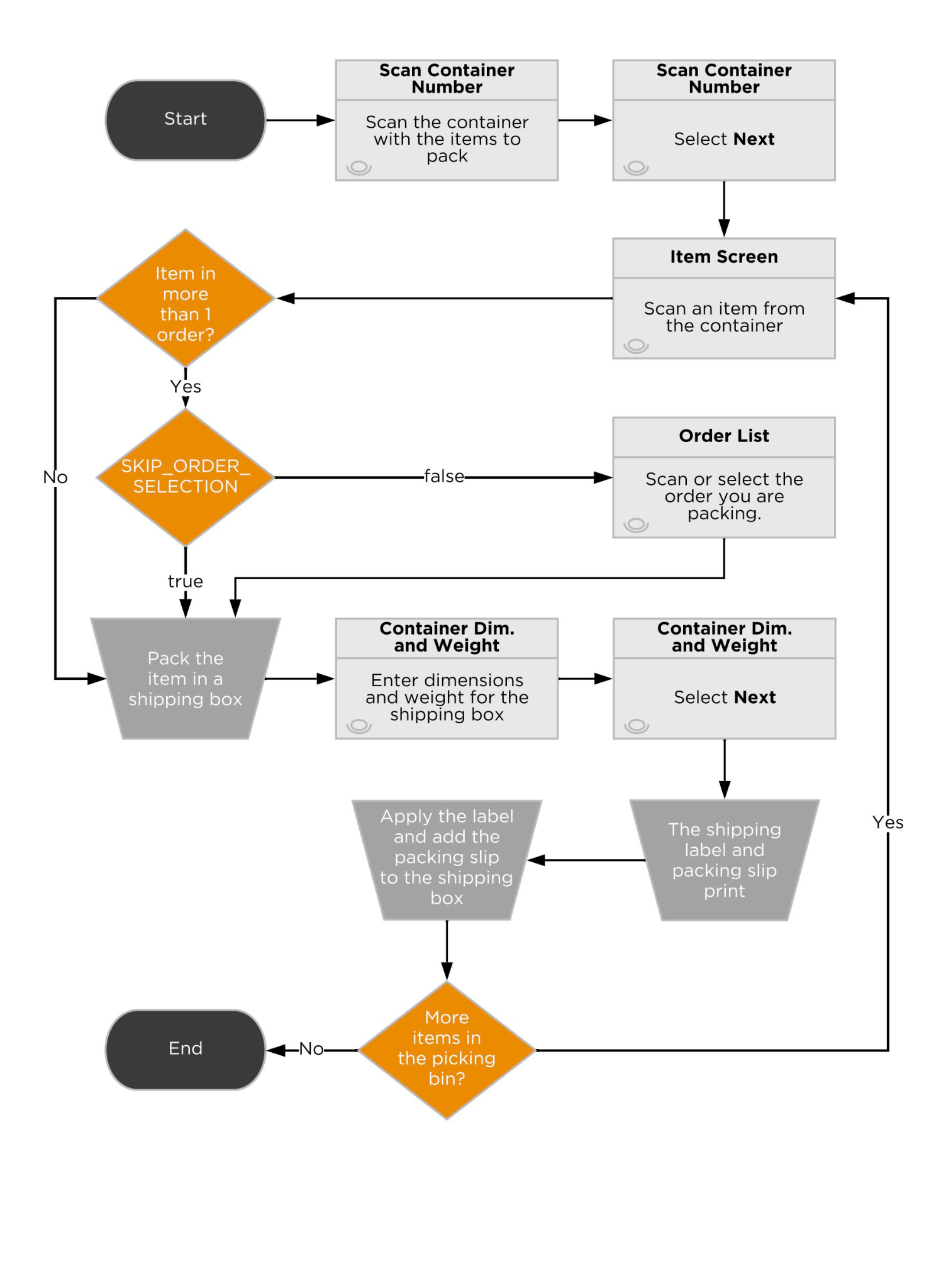
Single orderline, single quantity orders that were picked using **SingleScanPicking**  flow through the **PackOrderByItem** process. This process allows the bulk picked bin to be scanned and the user can pack out one order at a time by simply scanning an item.

Users will have 1 of 3 options of entering in the shipping box dimensions.

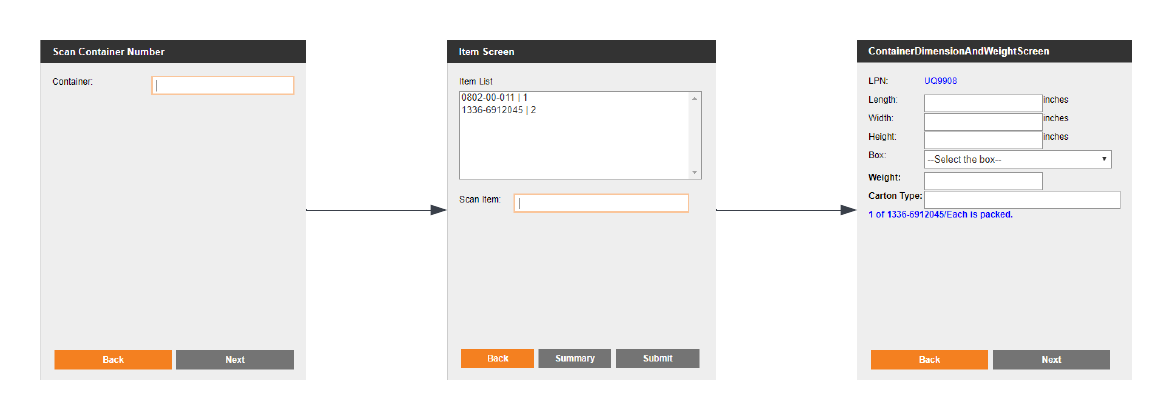
* CartonType drop down screen element
* Manual entry of Length, Width, and Height screen elements
* A CartonType screen element to enter in the predefined CartonType name

The weight screen element can be either manually entered or pre-populated by the pack weights if available.

### Process Flow



### Screen Flow



### Pack Order By Item Assumptions

* Carton dimensions and weight are required.
* After an order has been packed, Deposco will print a packing slip and a shipping label.

### Single Scan Packing

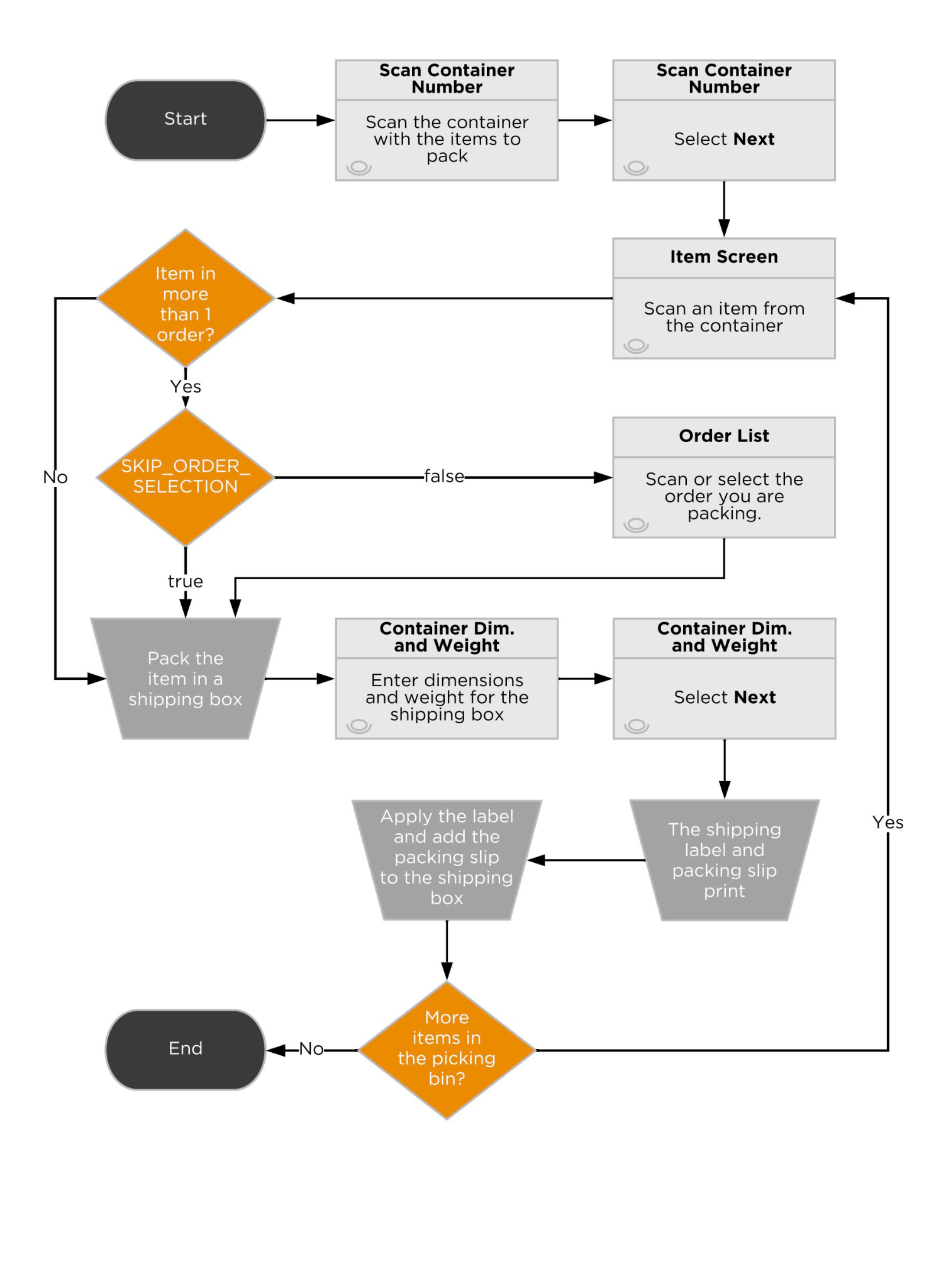
Multi quantity orders that were picked using **BatchPicking,**  flow through the **SingleScanPacking** process. This process allows the picked bin to be scanned and the user can pack out one order at a time by simply scanning the items to a shipping box.

Users will have 1 of 3 options of entering in the shipping box dimensions.

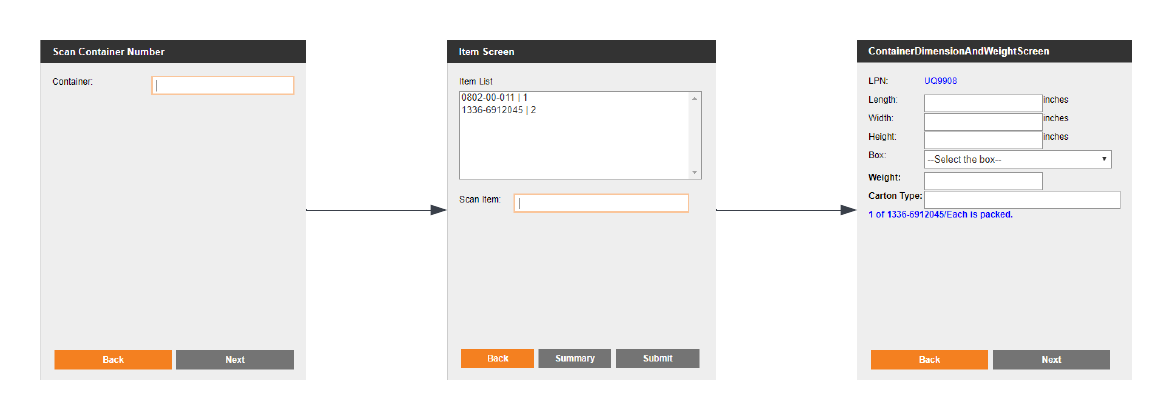
* CartonType drop down screen element
* Manual entry of Length, Width, and Height screen elements
* A CartonType screen element to enter in the predefined CartonType name

The weight screen element can be either manually entered or pre-populated by the pack weights if available.

### Process Flow



### Screen Flow



### Single Scan Packing Assumptions

* Only orders in current status Complete are eligible to be packed. Partially Picked orders will receive an error message, unless the order is marked as ‘ship partial order’.
* Grocery scanning is not required. Users will be allowed to enter the total packed quantity.
* All box dimensions will be required.
* Ready to Ship packs will use pack dimensions.
* A shipping label and a packing slip will be printed after an order is packed.

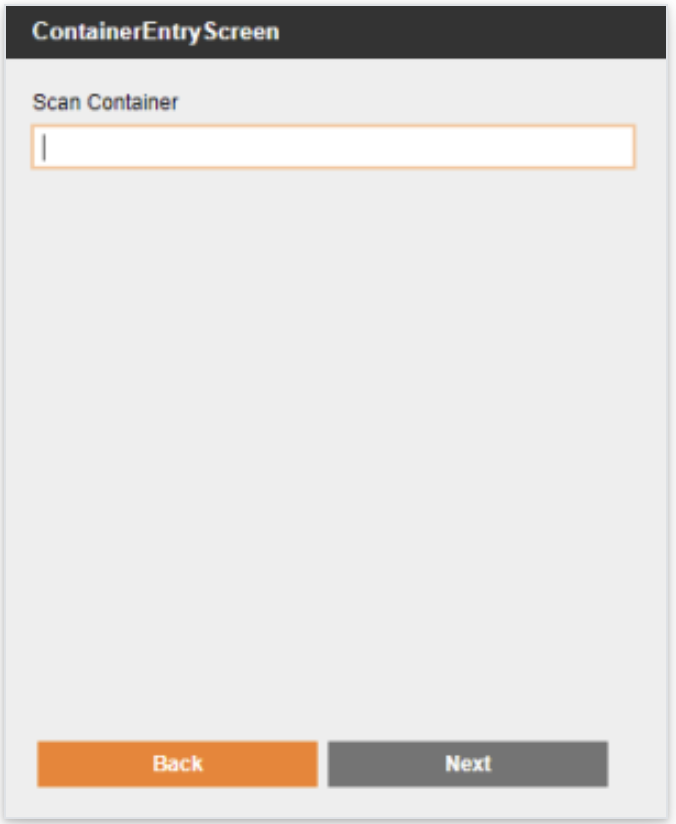
### Bulk Packing

The Bulk Packing process is used to pack and ship multiple identical orders that were picked together using Single Scan Picking, such as subscription orders, or multiple single quantity orders for the same item.

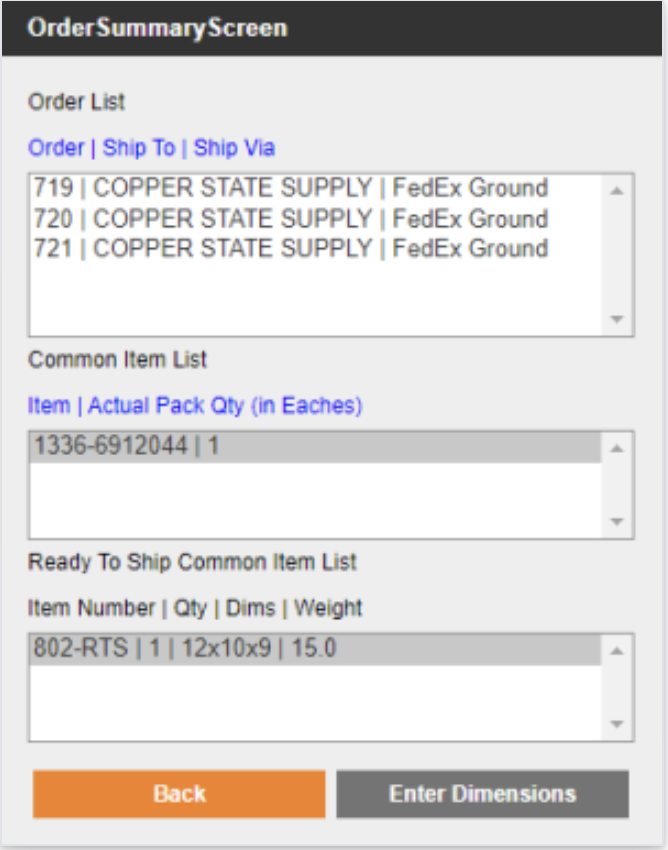
During Single Scan Picking, a group of identical orders is picked at the same time and placed into a single large bin. Then, during Bulk Packing, the orders are packed into identical shipping containers, the shipping labels are printed

### Screen Flow

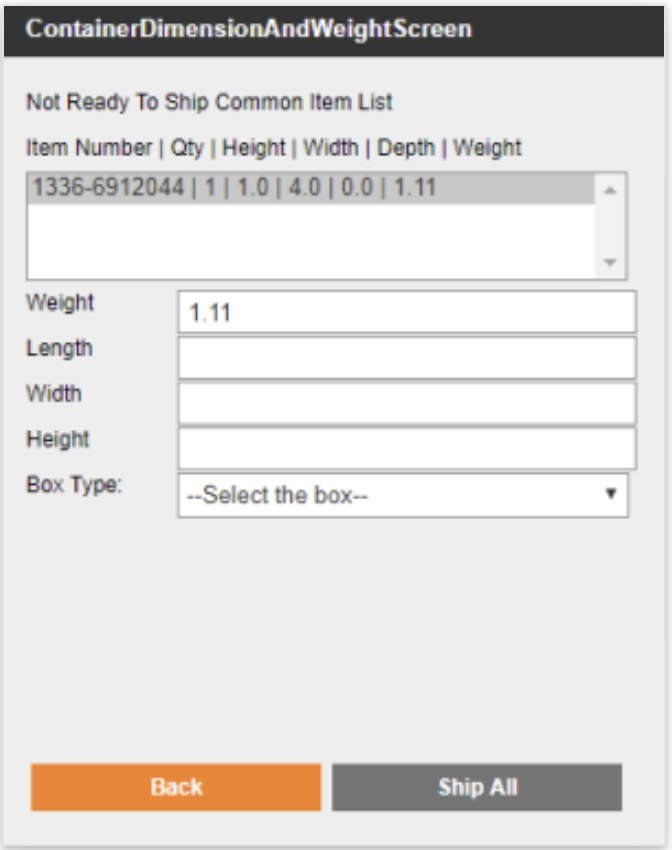
Scan the LPN label of the picking container.



The Order Summary Screen appears with a list of orders and the items for each order that are in the picking container. Pack each order in an identical shipping box.



Enter weight and dimensions. Select Ship All.



### Bulk Packing Assumptions

* Only orders in current status Complete are eligible to be packed. Partially Picked orders will receive an error message
* Omnipress will use the Bulk Packing process to pack multiple identical orders at the same time.
* The shipping labels for each shipping container will print after clicking the Ship All button on the screen
* Packing slips will print separately, which will require packers to match the shipping labels if there is order specific information on the packing slip.
* All box dimensions will be required.

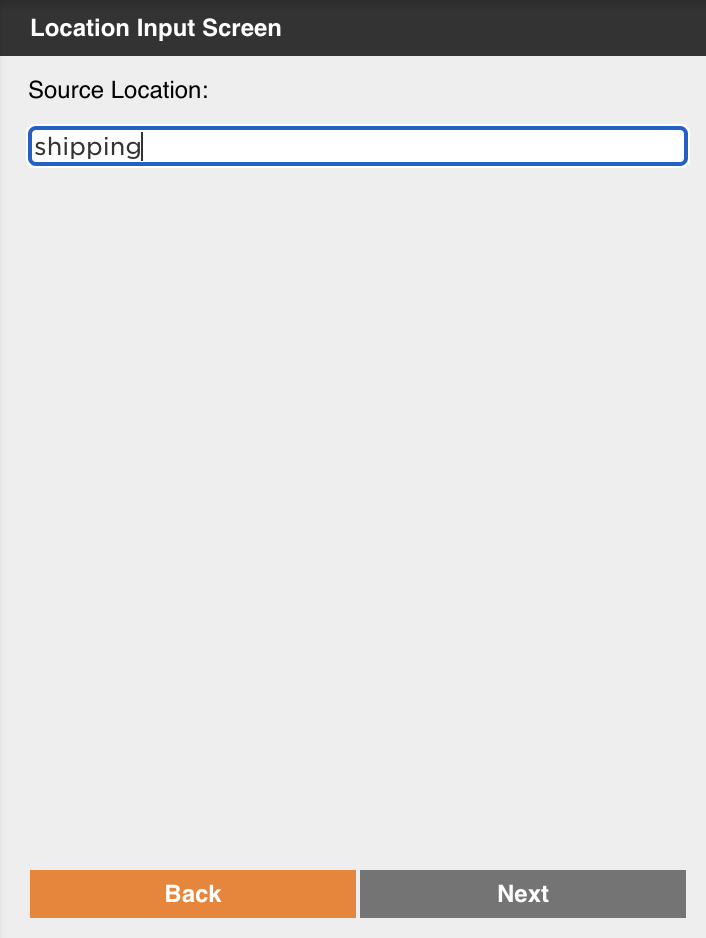
### Freight Packing

### Repalletize

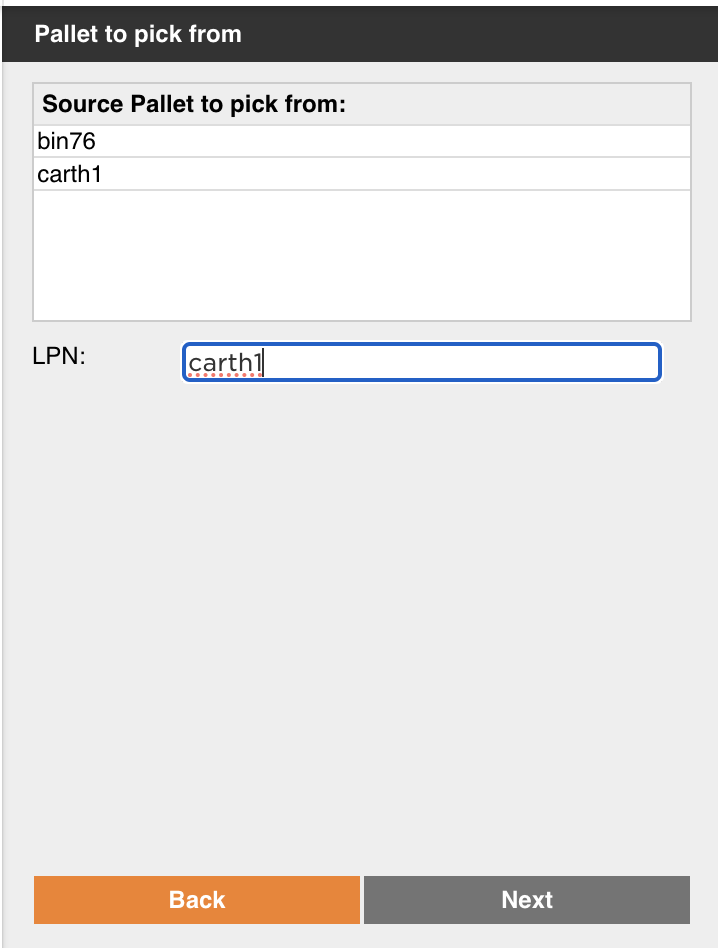
The RePalletize process will be used by the Omnipress team on an as needed basis to build outbound pallets. If this process is used, the user will scan the Shipping staging location as the source location. Next, the user will see the list of LPNs (pallets or other pieces of equipment) in the Location scanned. Once the user selects the LPN they wish to move cases/eaches off of to build their pallet, they are shown all items in eaches/cases on the Pallet to pick from screen.

Within the Items in Source Pallet screen, the user scans the item and the destination pallet to move the cases to the outbound pallet. The next screen, Packs In Source Pallet For Selected Item, the user selects the quantity of cases to move onto the pallet.

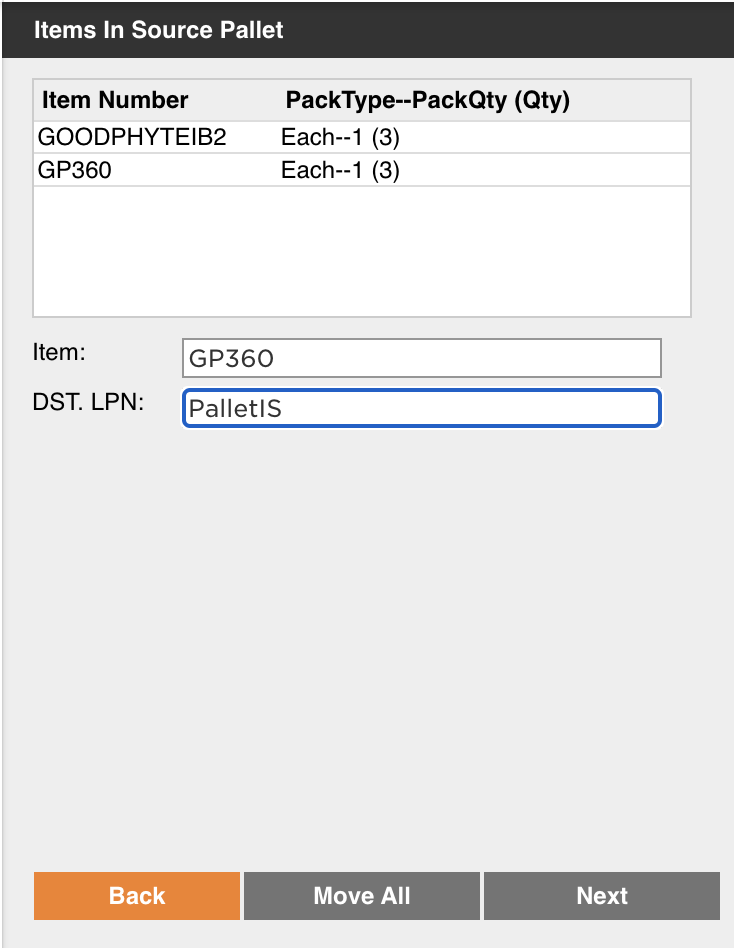
The user will be able to proceed building the next pallet(s) for the remaining orders by repeating the process above. After a pallet is completed, the user will shrink wrap the pallet. Please refer to process screenshots below:



The user scans the Shipping staging location where the pallet or other piece of equipment used to pick the inventory is currently located.



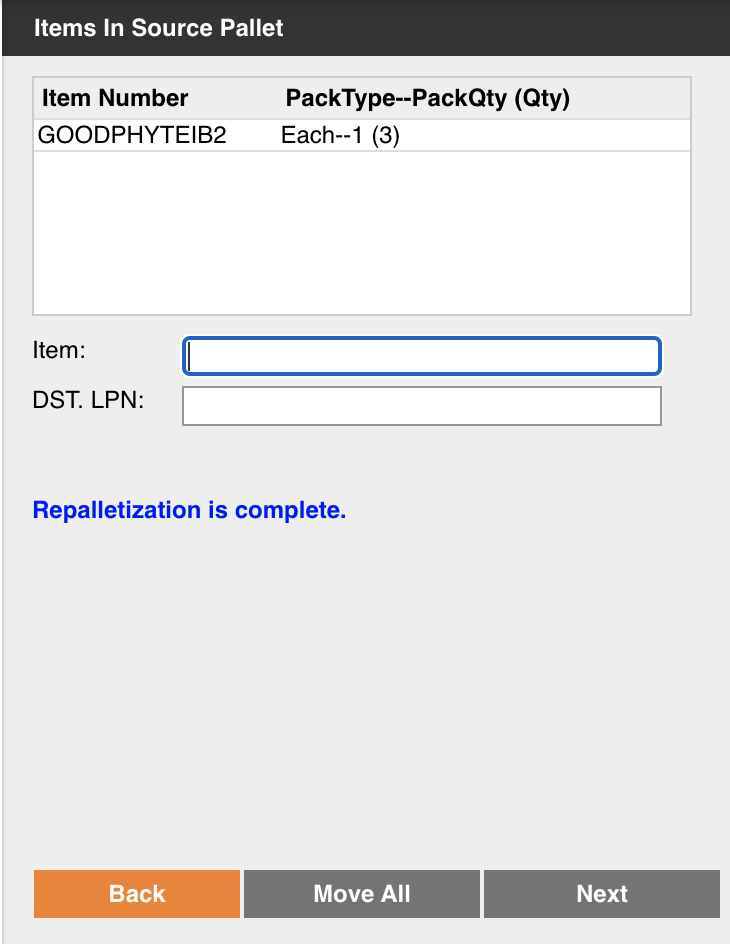
The user scans the container where the inventory they wish to use to build a pallet lives.



The user scans the sku they wish to put on the pallet followed by the barcode of the destination pallet.



The user specifies the quantity they are moving to the pallet.



Users are led back to the Items in Source Pallet screen where they repeat the same steps until their pallet is finished.

### **AssignTripToContainer**

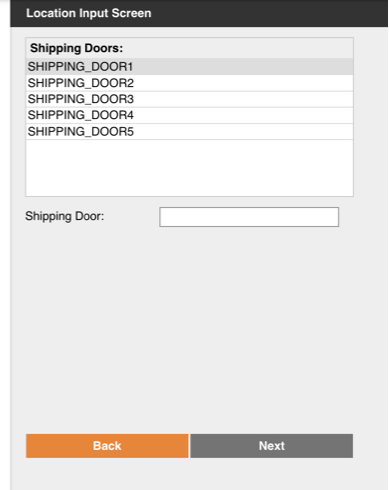
The AssignTrip process will be used to load final outbound containers onto a trip for freight orders (LTL/TL). Trips in Deposco represent a trailer load. Pallets are scanned and assigned to a dock door. After all pallets are loaded to the trip, the associated shipment and order are closed.

* All freight orders will be packed using the Assign Trip process.
* An API call will be made to R&L to retrieve a shipping label and BOL, which will be printed at the end of the process.
* Freight orders will print packing slips at the end of the packing process.

### **Handheld Process**

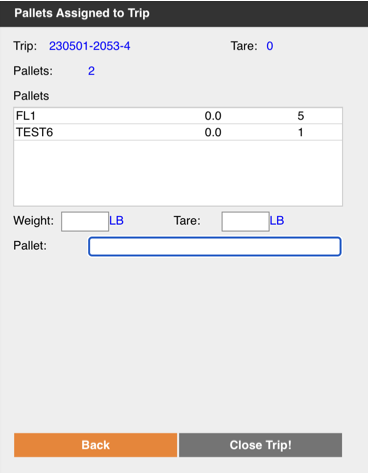
Users will first be prompted to scan a source staging location where the containers are located.

A Shipping Door then needs to be scanned to determine where the containers will be loaded.

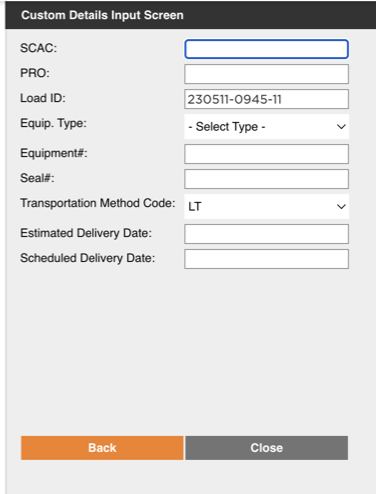


The pallets scanned and assigned to a shipping door must be in the source staging location that was scanned during the first step.

Users will not be required to enter in dimensions before loading.



Users will have the option to enter information (SCAC, PRO etc.) on the custom details input screen. r.



After assigning the trip, shipment lines will be created as the order will be closed.

## Outbound Flows Summary

Once orders are grouped into pick waves, they are allocated and ready to begin fulfillment. The following table details the process for which each wave type is picked and packed.

| **Wave Type** | **Picking Process** | **Packing Process** |
| --- | --- | --- |
| Priority | Batch Picking | Single Scan Packing |
| Identical | Single Scan Picking | Bulk Packing |
| Singles | Single Scan Picking | Bulk Packing |
| Multis | Batch Picking | Single Scan Packing |
| Partial | Batch Picking | Single scan Packing |
| Freight | Single Scan Picking | Repalletize and Assign Trip To Container |

## 

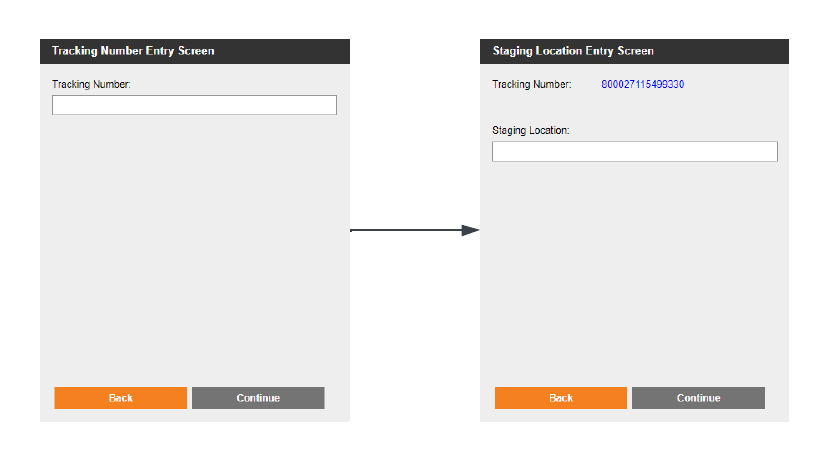
## Outbound Miscellaneous

### Void And Reprocess Shipment

The VoidAndReprocessShipmentprocess undoes the shipment of an order. Previously shipped stock is returned to a system-generated picking container. You may want to roll back a shipment if you need to change the Ship Via or the shipping address for the order.

### Process Flow

The user enters the Sales Order number that needs to be voided and in the next screen the user enters the staging location where the stock will be created, typically the Shipping location.



### Void and Reprocess Assumptions

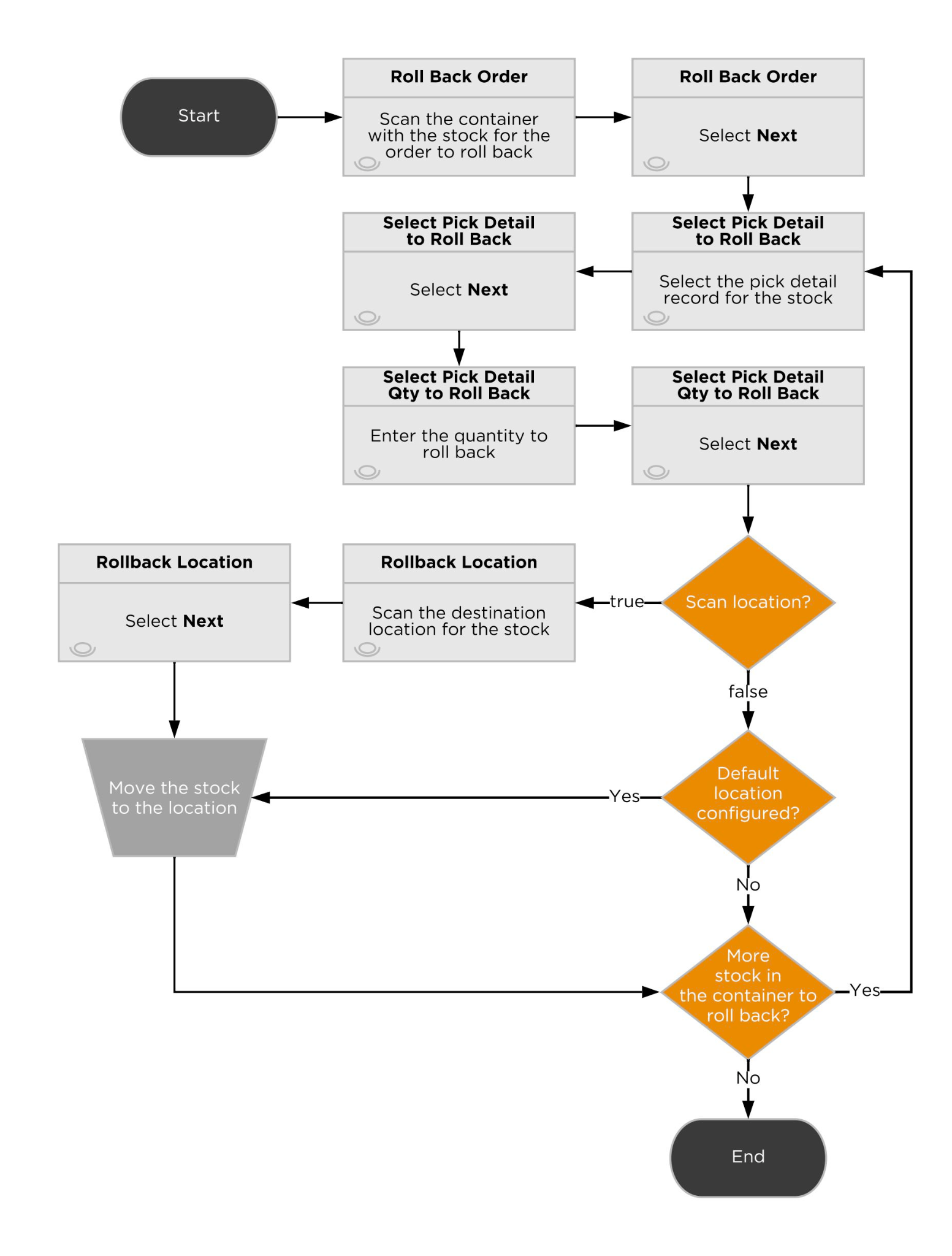
* All shipping containers, stock in the shipping containers, and pick details for the shipping containers are re-created.
* The status of the shipment and the ContainerHist entities for the shipment move to Rolled Back.
* The shipped pack quantity for each order line in the shipment is updated.
* Activity log and Inventory General Ledger (IGL) records are generated.
* An order activity record is generated for all orders that are associated with the shipment. The record notes that the shipment was rolled back.
* The shipment’s tracking number is voided specifically for parcel carriers UPS and USPS.

### Roll Back Order After Picking

You may decide to roll back an order if the order has been picked but hasn't yet shipped.

The **GPRollBackOrderAfterPicking** process allows stock to be staged to a location selected by the user. From this location, a Move processis used to return stock to a pickable location.

### Process Flow



### Screen Flow



### Rollback After Picking Assumptions

* The status of the sales order moves to Back Ordered. If all of the order lines for the order were rolled back, then the pick wave is removed from the order header.
* The Allocated Quantity and Picked Pack Quantity fields for the order line are set to zero (0), and the Shortage Pack Quantity field is set to the same value as the Order Pack Quantity field.
* The pick task, pick line, and pick detail are deleted for each order line that's rolled back.
* The location of the canceled stock moves from the picking container to the specified location by the user.
* The picking container is updated to remove the stock units and quantities that were rolled back (if the canceled stock was moved to a particular location). If the container was a shipping container and all of the stock was removed from the container and moved to a location, then the container is deleted.
* Activity log records are created so that reports can display the detailed or aggregated activity by order, user, or date range.
* After completing this process users can proceed to cancel the Sales Order or release it again for fulfillment.

# Billing

Third-party logistics (3PL) providers capture billable activities that are performed for tenant companies in the Deposco system, calculate the cost for those activities, and then generate invoices to send to the tenant companies.

Each tenant is defined as a company, or business unit (BU), in the Deposco system. To capture billing activity for a tenant, Omnipress creates a billing profile for each tenant. Within the billing profile multiple billing config records are added, each representing a unique billable activity. Billing Configs include the cost or charge for each activity and the frequency at which data is captured for that activity.

Billing schedulers run daily to gather data for activities that are charged to tenants based on the billing config records in the billing profiles for tenants. The data, which includes the quantity and price of the activities that are performed, is stored as a set of billable transactions in the Deposco system.

In some cases, Omnipress performs **AdHocBilling** to generate billable transactions for work that is manually tracked at completion so that it can be billed to a tenant.

After billable transactions are created, the transactions can be grouped and used to generate invoices that are sent to tenants. Omnipress will not use this functionality, as invoices will be generated in Salesforce

## Scheduler-Based Billing Configs

Omnipress will be utilizing the following billing configs.

### Order Processing Fees

**Shipped Order Fee**

Description: Flat fee per shipped order.

Logic: Deposco will generate a Billing Transaction per shipped order with the price specified on the Shipped Order Fee Billing Config.

This charge will be generated per order, regardless of the number of items or the number of shipping containers that were used to pack the items. For example, if one order shipped using 3 boxes, a single billing transaction will be generated, instead of 3.

Assumptions

* All billing transactions will specify the related order number.
* The total price will be equal to the price defined in the Billing Config.
* Billing transactions will be generated daily.

**Shipped Item Fee**

Description: Flat fee per shipped item.

Logic: Deposco will generate a Billing Transaction per shipped order, where the quantity is the total sum of the shipped items in the order, and the total price is the result of multiplying the quantity by the price on the Billing Config.

Three tier fields will be provided for Omnipress to define different prices based on the total item quantity shipped.

Assumptions:

* All Billing Transactions will specify the related order number.
* Omnipress has the ability to define and adjust the item quantity tiers.
* Pre-built kits will be considered as a single item.
* Components for Kit To Order items will be considered as separate items.
* Billing transactions will be generated daily.

**Production Fee for Shipped Items**

Description: Fee for production cost.

Logic: Deposco will generate a Billing Transaction per shipped item per order, where the quantity is the total quantity shipped of the item in the order, and the total price is the result of multiplying the quantity by the unit purchase cost defined in the Item master data record.

Assumptions:

* This Billing Transaction is generated only for items that contain a Unit Purchase Cost.
* The Billing Transaction will specify the related item and order number.
* Billing transactions will be generated daily.

### **Shipping Fees**

**USPS International Shipping Charges**

Description: Flat fee per shipped order.

Logic: Deposco will generate a Billing Transaction per order shipped through USPS to a country different from the US, where the quantity is 1 and the total price is the price on the USPS International Shipping Charges Billing Config.

Assumptions:

* The Billing Config price will be $15 and can be adjusted by the Omnipress team.
* Shipments to US territories and military bases will be considered international shipments.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

**USPS Domestic Shipping Charges**

Description: Flat fee per shipped order.

Logic: Deposco will generate a Billing Transaction per order shipped through USPS within the US, where the quantity is 1 and the total price is the price on the USPS Domestic Shipping Charges Billing Config.

Assumptions:

* The Billing Config price will be $2 and can be adjusted by the Omnipress team.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

**UPS International Shipping Charges**

Description: Flat fee per shipped order with discount on published rate, except for UPS Standard.

Logic: Deposco will generate a Billing Transaction per order shipped through UPS to a country different from the US, where the quantity is 1 and the total price is the result of multiplying the shipment published rate by (1 - the factor specified on the UPS International Shipping Charges Billing Config) to apply a discounted rate.

No discount will be applied to orders shipped with UPS Standard and the total price will be the shipment published rate.

Assumptions:

* A custom field will be used to indicate the percentage of the published rate to charge. For example, for a 20% discount, Omnipress will enter a value of 0.2 in the custom field.
* Shipments to US territories and military bases will be considered international shipments.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

**UPS Domestic Shipping Charges**

Description: Flat fee per shipment with discount on published rate.

Logic: Deposco will generate a Billing Transaction per order shipped through UPS within the US, where the quantity is 1 and the total price is the result of multiplying the shipment published rate by (1 - the factor specified on the UPS Domestic Shipping Charges Billing Config) to apply a discounted rate.

Assumptions:

* A custom field will be used to indicate the percentage of the published rate to charge. For example, for a 20% discount, Omnipress will enter a value of 0.2 in the custom field.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

**R+L International Shipping Charges**

Description: Flat fee per shipment with discount on published rate.

Logic: Deposco will generate a Billing Transaction per order shipped through R+L to a country different from the US, where the quantity is 1 and the total price is the result of multiplying the shipment published rate by (1 - the factor specified on the R+L International Shipping Charges Billing Config) to apply a discounted rate.

Assumptions:

* A custom field will be used to indicate the percentage of the published rate to charge. For example, for a 20% discount, Omnipress will enter a value of 0.2 in the custom field.
* Shipments to US territories and military bases will be considered international shipments.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

**R+L Domestic Shipping Charges**

Description: Flat fee per shipment with discount on published rate

Logic: Deposco will generate a Billing Transaction per order shipped through R+L within the US, where the quantity is 1 and the total price is the result of multiplying the shipment published rate by (1 - the factor specified on the R+L Domestic Shipping Charges Billing Config) to apply a discounted rate.

Assumptions:

* A custom field will be used to indicate the percentage of the published rate to charge. For example, for a 20% discount, Omnipress will enter a value of 0.2 in the custom field.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

### **Rush** Order Fee

Description: Flat fee for same-day orders..

Logic: Deposco will generate a Billing Transaction per rush order shipped, where the total quantity is 1 and the total price is the price specified on the Rush order fee Billing Config.

Assumptions:

* Orders that require same day shipping will be identified by using a Rush Order flag at the Sales Order level. Users placing an order will need to set the flag to true or Automations can be used to mark orders that meet certain conditions as Rush orders. The Rush Order flag can be updated ad hoc prior to shipping.
* The price on the Billing config for rush orders is $10 and can be adjusted by the Omnipress team.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

### Customer **Return** fee

Description: Flat fee per received customer return.

Logic: Deposco will generate a Billing Transaction per received return order, where the total quantity is 1 and the total price is the price specified on the Customer return fee Billing Config.

Assumptions:

* Deposco will consider the order types ‘Customer Return’ or ‘Blind Receipts’ that contain a note specifying that the received inventory is related to a customer return.
* Deposco will consider customer returns with a status of Received or Partial Receipt.
* Any Blind Receipts that are not identified as a customer return will not be considered.
* The Billing Transaction will specify the related order number.
* Billing transactions will be generated daily.

### **Kitting Fee**

Description: Flat fee per built kit.

Logic: Deposco will generate a Billing Transaction per kitted item, where the total quantity is the total number of built kits and the total price is the result of multiplying the quantity by the price specified on the Kitting fee Billing Config.

Assumptions:

* Deposo will use Activity Log records to compute the quantity of kitted items.
* The Billing Transaction will specify the related item number.
* Billing transactions will be generated daily.

### **Inventory Storage**

**SKU Storage Fee**

Description: Total item inventory fee.

Logic: Deposco will generate a Billing Transaction per item with physical inventory, where the quantity is the sum of all the inventory in the facility, regardless of location type or inventory condition/status, and the total price is the result of multiplying the quantity by the price specified on the Item storage fee Billing Config.

Assumptions:

* No stock units are excluded to calculate the total inventory, regardless of their current location or any other attributes.
* The Billing Transaction will specify the related item number.
* Billing transactions will be generated monthly and the quantity of the stock will be a snapshot of the previous month’s end inventory.

**Location Storage Fee**

Description: Flat fee per location holding inventory, excluding skid locations.

Logic: Deposco will generate a Billing Transaction per location that holds inventory, where the quantity is the count of locations that contain inventory for the Business Unit/family and the total price is the result of multiplying the quantity by the price specified on the Location storage fee Billing Config.

Assumptions:

* Skid locations are excluded from this logic.
* Locations to consider need to be assigned to the Picking and/or to the Reserve warehouse zones. Any locations that do not exist within these zones will not generate a location storage fee. This is to prevent generating location storage fees for staging locations such as Receiving and Shipping.
* The Billing Transaction will specify the related location number.
* Billing transactions will be generated monthly and the locations with stock will be a snapshot of the previous month’s end inventory.

**Skid Location Storage Fee**

Description: Flat fee per skid location holding inventory.

Logic: Deposco will generate a Billing Transaction per skid location that holds inventory, where the quantity is the count of skid locations that contain inventory for the Business Unit/family and the total price is the result of multiplying the quantity by the price specified on the Skid location storage fee Billing Config.

Assumptions:

* A Skid Location flag is available in the Location entity. The flag must be checked for the location to be considered for this logic.
* All Skid locations are considered regardless of their zone assignment, since the assumption is that there will be no staging or temporary skid locations.
* The Billing Transaction will specify the related location number.
* Billing transactions will be generated monthly and the locations with stock will be a snapshot of the previous month’s end inventory.

### **Admi**nistrative Fees

**Admin Fee**

Description: Flat fee associated with the Business Unit/family.

Logic: Deposco will generate a Billing Transaction where the quantity is 1 and the total price is the price specified on the Admin Fee Billing Config.

Assumptions:

* Billing transactions will be generated monthly.

**Integration Fee**

Description: Flat fee per integration point.

Logic: Deposco will generate a Billing Transaction where the quantity is the count of integration points associated with the Business Unit/family and the total price is the price specified on the Integration fee Billing Config.

Assumptions:

* Billing transactions will be generated monthly.

### **New Item Creation Fee**

Description: Flat fee per new item created.

Logic: Deposco will generate a Billing Transaction per newly created item, where the quantity is 1 and the total price is the price specified on the New item creation fee Billing Config.

Assumptions:

* The Billing Transaction will specify the related item number.
* Billing transactions will be generated daily.
* Items with the Inventory Tracking Enabled flag set to false will be excluded, under the assumption that these are digital items.

### Cycle Count Credit

Description: Credit for cycle count adjustments.

Logic: Deposco will generate a Billing Transaction for all approved cycle count variances that triggered a negative inventory adjustment, where the quantity is the adjusted inventory quantity and the total price is the result of multiplying the quantity by the item Unit Price.

Assumptions:

* The Billing Transaction will specify the related item number and the location number.
* Only accepted negative variances will be considered to generate credits.
* Items must have a Unit Price.
* The Unit Price is the same value used for customs purposes.
* Billing transactions will be generated daily.

## AdHoc Billing Billing Configs

The Omnipress team will manage the creation of ad hoc charges/credits in Deposco. Each ad hoc charge will have an individual billing config.

### Batch Order Fee

Description: Fee for batch orders processing.

A Billing Config exists for the Batch Order Fee, where the Transactions Generated By field is set to Ad Hoc, the quantity is 1 and the price is the fee to charge for the custom setup.

### Customer Setup Fee

Description: One-time fee for new customer setup.

A Billing Config exists for the Customer Setup Fee, where the Transactions Generated By field is set to Ad Hoc, the quantity is 1 and the price is the fee to charge for the custom setup.

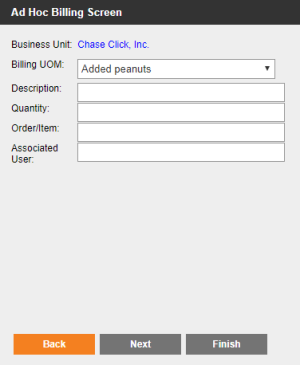
### Credits

Description: Ad Hoc credits

A Billing Config exists for Credits, where the Transactions Generated By field is set to Ad Hoc, the quantity is 1 and the price -1.

### AdHoc Billing Process Flow

Users select Ad Hoc Billing on the handheld menu and select the business unit/family. The Ad Hoc Billing Screen appears.



Enter or select values in the following fields:

* Billing UOM – Select the activity to generate billable transactions by selecting the ad hoc billing config.
* Description – Optional. Enter a description of the activity.
* Quantity – Enter the quantity for the activity that was performed. For example, if the charge is for applying stretch wrap to pallets, enter the quantity of pallets that were wrapped.
* Order/Item – If the selected value in the Billing UOM list indicates (Item Req) or (Order Req), then enter the item number or order number, respectively, that's associated with the activity.
* Associated User – Optional. Enter the username for the user who performed the work for the billable transaction.

## Billing Transaction Export to Salesforce

Omnipress will export all Billable Transactions generated in Deposco and import them to Salesforce. The method defined for the Phase 1 implementation is through Data Exchange.

Deposco will configure a standard template that will be used to export billing transactions on a monthly basis. The export data file is saved automatically to an SFTP location. The Omnipress team can also manually download the export data file from the Transmissions list.

### Assumptions

* Omnipress will own the creation and management of the SFTP location.
* Basic authentication or a public/private key pair for authentication is available for the SFTP protocol.
* Notifications can be sent based on the success or failure of the export.
* The export file type is Text and available delimiters are comma, pipe, semicolon, tab, and custom.

# Reporting

Two types of reports are available in Deposco:

Global reports are predefined reports that are developed and managed by Deposco using Pentaho business intelligence (BI) software. These reports are intended to help users with their day-to-day tasks and provide key performance metrics. They are also used as templates for many common documents and labels that are generated in Deposco, such as packing lists, invoices, item labels, location labels, and container labels.

Company reports are reports that you can create and manage in Report Builder and share with other users in Deposco.

In addition, dashboards enable you to create and manage pages with interactive data visualizations that provide insight into your Deposco data so that you can optimize workflow and performance and make informed strategic decisions.

Security permissions for users and groups enable you to control who has access to these features and to specific categories or groups of reports.

## Inbound Reports

* Item UPC label (4x8)

Label to print optionally before the receiving process.

## Inventory Management Reports

* Reorder point notification emails showing monthly and yearly usage. Deposco will evaluate the total inventory held in reserve and picking locations and compare it to the item’s defined reorder point.
* 12 Month Product usage report (SKUs shipped per month and then averaged per month)
* Aging inventory
* Understock report
* Back order by SKU report
* Receiving report

## Outbound Reports

* Generic Packing Slip

Generic Packing Slip provided by Deposco. Omnipress wants to define contact information for each Business Unit.

* Branded Shipping Label

Pentaho report template for 4x8 shipping label for UPS shipments. 6 customers currently have this. Two inches at the top of the report to display the company logo or other branding. Bottom 4x6 is an image of a standard shipping label returned from UPS.

* Shipping Report
* Shipping Charge Report

## Customer Portal

* Dashboard to show all orders and their status.
* Deposco will look into the possibility of creating notifications to inform customers of any orders that need attention.

## Operations/Warehouse

* Warehouse dashboard (open orders, orders being processed, shipped orders).
* Team productivity dashboard.

## Reporting Miscellaneous

* Orders From Previous Month
* Billing report
* FIFO with values
* Offer with price
* Monthly Invoice total report all BU's
* Monthly Invoice report per BU

# 

# Customer Agreement

Your signature is required as proof of understanding and acceptance of the design document as described above. Upon acknowledgement of your approval, Deposco will engage in development of the work stated in this document and the design specification. Please review this document in its entirety. Upon approval, please sign, date and return.

| **Deposco, Inc.** 11605 Haynes Bridge Road Suite 200 Alpharetta, GA 30009  By: | |
| --- | --- |
| Name: |  |
| Title: |  |
| Date: |  |

| **Omnipress** 2600 Anderson St.  Madison, WI 53704  By: | |
| --- | --- |
| Name: |  |
| Title: |  |
| Date: |  |