Process Improvement Solutions

Warehouse Rack Module

Create Internal Database –

This model requires the use an Internal Database to store information in different tables

- **Model Extensions –**
  - Define Database Connections
  - New Connection – Internal
  - Save
  - Select Internal and check – Connect Database to Data Import Tables

Tables Needed –

These tables will keep track of different information as well as be able to look up information to guide model logic

- **SKUInformation –** This table holds the information for the all of the SKUs that are going to be palletized, imported from "Palletizing for Warehouse Import.xlsx" from the SKUInformation sheet
  - SKU – String
    - Item identifier
  - Item Description – String
  - Container Count – Integer/Double
    - Amount of items held in container
  - Container Weight – Double
  - Container Height – Double
  - Container Width – Double
  - Container Length – Double
  - PercentofFullPallet – Integer
  - FullPalletCount – Integer
  - IsFullPallet – Integer
- **PalletsBuilt –** This table records all of the pallets built from the model, for Mixed pallets it lists each SKU and the number of containers placed within the pallet, for Full pallets it lists the total containers of the SKU put in the pallet
  - SKU – String
  - PalletID – String
    - Identification value given to the Pallet being built
  - DestinationID – String
    - Identification value given to the destination of the pallet

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

    - PalletContainerCount – Integer
        - Number of containers placed on the pallet
    - CapacityVolume – Double
        - Total volume available for loading the pallet
    - VolumeRemaining – Double
        - Volume remaining for loading the pallet
    - PalletType – String
        - This can be used to describe the type of pallet that was built
- **SKURequests –** This table records all of the current orders for SKUs within the past 24 hrs within the model. Once a day this table is updated with the new SKUs from the SKUOrders table. This table also records when the SKUs are requested vs when they are fulfilled
    - SKU – String
    - ItemsNeeded – Integer
        - Quantity of the items of a specific SKU
    - DeliveryID – String
    - DateRequested – String
        - Records the date that the items were requested
    - DateFullfilled – String
        - Records the date that the order of the items was palletized
    - Picking – Integer
        - Records that the palletizing of the items are in progress
- **SKUOrders –** This table is where the compilation of all SKUs that are being requested for fulfillment, filled in by the Creating Item Requests section of the model
    - SKU – String
    - NumberofItems – Integer/Double
        - Total quantity of the SKU for the destination order
    - Destination – String
        - Record which destination is requesting the items

Additional Properties Definition –

- **Model Parameters –**
    - M_MixedPalletVolume – Double
        - 92160
    - M_MixedPalletID – String
    - M_MixedDestinationID – String
    - M_FullCurrentCount – Integer
    - M_CheckRequests – Event
    - M_FullPalletVolume – Double

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - 92160
    - M_FullContainerCount – Integer
    - M_FullDestinationID – String
    - M_FullPalletCount – Integer
        - 1
    - M_TotalContainers – Integer
    - M_MakeContainers – Integer
    - M_MixedPalletCount – Integer
        - 1
    - M_TempSKU – String
    - M_TempQuantity – Integer
    - M_TempDestination – String
    - M_PercentToMakeFullPallet –
        - 50
        - Check: Ask
        - Min – 0
        - Max - 100
    - M_CurrentRequests - Integer
- **Process Parameters –**
    - P_TempInt – Integer
    - P_TempSKU – String
    - P_Router – String
- **Object Parameters –**
    - O_Destination – String
    - O_SKU – String
    - O_ItemCount – Integer
    - O_ContainerItemCount – Integer
    - O_ContainerWeight – Double
    - O_ContainerWidth – Double
    - O_ContainerHeight – Double
    - O_ContainerLength – Double
    - O_ContainersNeeded – Integer
    - O_ContainerVolume – Double

Model Extensions –

- **SimulationStart –**
    - Line 1 – DB Interface Action – This is to delete the entries in the SKURequests table to reset the data collection for a fresh run

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - Database Name – Internal
        - Table Name – SKURequests
        - Command Type – Update Table
        - Check Execute AQL (Advanced Mode) – DELETE From SKU/ItemRequests
        - Insert New Definition
    - Line 2 – DB Interface Action – This is to delete the entries in the PalletsBuilt table to reset the data collection for a fresh run
        - Database Name – Internal
        - Table Name – PalletsBuilt
        - Command Type – Update Table
        - Check Execute AQL (Advanced Mode) – DELETE From PalletsBuilt
        - Insert New Definition
    - Line 3 – DB Interface Action – This is to delete the entries in the SKUOrders table to reset the data collection for a fresh run
        - Database Name – Internal
        - Table Name – SKUOrders
        - Command Type – Update Table
        - Check Execute AQL (Advanced Mode) – DELETE From SKU/ItemOrders
        - Insert New Definition
    - Line 4 – DB Interface Action – This updates the SKUInformation table with the selected M_PercentToMakeFullPallet at the beginning of each model run
        - Database Name – Internal
        - Table Name – SKUInformation
        - Command Type – Update Table
        - PercentofFullPallet – M_PercentToMakeFullPallet
- **M_CheckRequests –** This section is executed when the event M_CheckRequests is triggered, here the SKURequests will be updated with the none requested entries from the SKUOrders table
    - Line 1 – DB Interface Action – Grab the information from the SKUOrders table
        - Database Name – Internal
        - Table name – SKUOrders
        - Command Type – Select From
        - Condition – Where Requested = 0
        - SKU – M_TempSKU
        - NumberofItems – M_TempQuantity
        - Destination – M_TempDestination
        - Insert New Definition – Select the definition – Edit Action Definition –
            - Line 1 – DB Interface Action – Insert this information into the SKURequests table

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - Database Name – Internal
        - Table Name – SKURequests
        - Command Type – Insert Into
        - SKU – M_TempSKU
        - ItemsNeeded – M_TempQuantity
        - DestinationID – M_TempDestination
        - DateRequested – '<%EXPS GetSystemDateMMDDYYY(ModelID)%>'
        - DateFullfilled – '0'
        - Picking – 0
        - Insert New Definition
    - Line 2 – Expression Action –
        - M_CurrentRequests = M_CurrentRequests + 1
    - Line 3 – DB Interface Action – Check that the SKU has been transitioned into the SKURequests table to be processed by the Palletizing section of the model
        - Database Name – Internal
        - Table Name – SKUOrders
        - Command Type – Update
        - Condition – Where SKU = '<%M_TempSKU%>' AND NumberofItems = <%M_TempQuantity%> AND Destination = '<%M_TempDestination%>' AND Requested = 0
        - Requested – 1
- Line 2 – Expression Action –
    - M_PickRequests = 1

Objects and Containment –

- **Item**
- **Boxes**
- **Pallet**
    - Components –
        - Boxes – Min 500/Max 500

Creating Item Requests –

This section of the model takes all requests from a Start process and Work Order/External Database and processes the requests to be inserted into a table

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

Start Process –

- **ItemRequests –** This process creates the necessary requests for all SKUs to be palletized
  - Define Object Creation Rules – Create Objects Based on a Work Order – This is an imported list of items that are requested to be palletized, this can be built from scratch or from the a slotting/picking model.
    - Object Type
    - Min
    - Max
    - O_Destination
    - O_SKU
    - O_ItemCount
    - O_ContainerItemCount
    - O_ContainerWeight
    - O_ContainerWidth
    - O_ContainerHeight
    - O_ContainerLength
    - O_ContainerVolume
    - O_ContainersNeeded

Generic Process –

- **UpdateOrders –** This process takes those requests and processes them to be inserted into the SKUOrders table
  - Process Extensions/Event Handling – ObjectActivated – Insert five lines –
    - Line 1 – Expression Action –
      - P_TempInt = 0
    - Line 2 – DB Interface Action – Check the SKUOrders table to see if the SKU is already being requested
      - Database Name – Internal
      - Table Name – SKUOrders
      - Command Type – Select From
      - Condition – Where SKU = '<%O_SKU%>' AND Destination = '<%O_Destination%>'
      - SKU – P_TempSKU
      - NumberofItems – P_TempInt
      - Destination – P_Router
      - Insert New Definition – Select the definition – Edit Action Definition –
        - Line 1 – Expression Action – If so then add the total and the new amount together

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - O_ItemCount = O_ItemCount + P_TempInt
    - Line 1 – DB Interface Action – Update the SKU in the table with the new amount
        - Database Name – Internal
        - Table Name – SKUOrders
        - Command Type – Update Table
        - NumberofItems – O_ItemCount
        - Insert New Definition – Close – Close and Save – Close
- Line 3 – Condition Action –
    - IF –
        - P_TempInt < 1
- Line 4 – DB Interface Action – If not then insert the SKU into the table
    - Database Name – Internal
    - Table Name – SKUOrders
    - Command Type – Insert Into
    - SKU – O_SKU
    - NumberofItems – O_ItemCount
    - Destination – O_Destination
    - Insert New Definition
- Line 5 – Condition Action –
    - END IF

Palletizing System –

This section of the model takes all requested SKUs and sorts them

Start Process –

- **Start Items –**
    - Define Object Creation Rules – Create Single Object Type –
        - Object type to Create –
            - Item

Connection Line leaving Start Items –

- Behavior: Conveyors, Polling, and Carriers – Check the following –
    - Block if next destination is busy
    - Enable Group Polling
- Process Extensions/Event Handling – ObjectActivated – Insert one line –
    - Line 1 – Expression Action –

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeFullPallet') 'Pallet', 1, 0)

Generic Process –

- **GrabItemInformation –**
  - Process/Object/Condition Routing – Condition Based Routing –
    - M_TotalContainers < P_TempInt –
      - Next Process – GrabContainers
    - M_TotalContainers >= P_TempInt –
      - Next Process – SplitContainers
    - O_ContainerVolume < 1 –
      - Next Process – NoRequests
  - Process Extensions/Event Handling – ObjectActivated – Insert five line –
    - Line 1 – DB Interface Action –
      - Database Name – Internal
      - Table Name – SKURequests
      - Command Type – Select From
      - Condition – Where Picking = 0 AND DestinationID = '<%M_MixedDestinationID%>' LIMIT 1
      - SKU – O_SKU
      - ItemsNeeded – O_ItemCount
      - DestinationID – O_Destination
      - Insert New Definition – Select the definition – Edit Action Definition –
        - Line 1 – Data Interface Action –
          - Section Name – SKU/ItemRequests
          - Action Type – Update
          - Condition – SKU = '<%O_SKU%>' AND ItemsNeeded = <%O_ItemCount%> AND DestinationID = '<%O_DestinationID%>
          - Field Name – Picking
          - Update with Value – 1
        - Close and Save – Close
    - Line 2 – Condition Action –
      - IF –
        - DBQueryResult <= 0
    - Line 3 – DB Interface Action –
      - Database Name – Internal
      - Table Name – SKURequests
      - Command Type – Select From

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - Condition – Where Picking = 0 LIMIT 1
        - SKU – O_SKU
        - ItemsNeeded – O_ItemCount
        - DestinationID – O_Destination
        - Insert New Definition – Select the definition – Edit Action Definition –
            - Line 1 – Data Interface Action –
                - Section Name – SKU/ItemRequests
                - Action Type – Update
                - Condition – SKU = '<%O_SKU%>' AND ItemsNeeded = <%O_ItemCount%> AND DestinationID = '<%O_DestinationID%>
                - Field Name – Picking
                - Update with Value – 1
            - Close and Save – Close
    - Line 4 – Condition Action –
        - END IF
    - Line 5 – DB Interface Action –
        - Database Name – Internal
        - Table Name – SKUInformation
        - Command Type – Select From
        - Condition – Where SKU = '<%O_SKU%>'
        - ContainerCount – O_ContainerItemCount
        - ContainerWeight – O_ContainerWeight
        - ContainerHeight – O_ContainerHeight
        - ContainerWidth – O_ContainerWidth
        - ContainerLength – O_ContainerLength
        - PercentofFullRack – P_TempInt
        - FullPalletCount – M_FullContainerCount
        - Insert New Definition – Select the definition – Edit Action Definition –
            - Line 1 – Expression Action –
                - P_TempInt = (P_TempInt / 100) * M_FullContainerCount
    - Line 6 – Expression Action –
        - O_ContainersNeeded = (O_ItemCount / O_ContainerItemCount) + 1
    - Line 7 – Expression Action –
        - M_TotalContainers = O_ContainersNeeded
    - Line 8 – Expression Action –
        - O_ContainerVolume = O_ContainerLength * O_ContainerWidth * O_ContainerHeight
- **SplitContainers –**

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

- Process Extensions/Event Handling – ObjectActivated – Insert six lines –
    - Line 1 – Expression Action –
        - M_MakeContainers = 0
    - Line 2 – DB Interface Action –
        - Database Name – Internal
        - Table Name – SKUInformation
        - Command Type – Select From
        - Condition – Where SKU = '<%O_SKU%>'
        - PercentofFullRack – P_TempInt
        - FullPalletCount – M_FullContainerCount
        - Insert New Definition
    - Line 3 – Expression Action –
        - P_TempInt = (P_TempInt / 100) * M_FullContainerCount
    - Line 4 – Condition Action –
        - IF –
            - O_ContainersNeeded >= P_TempInt
    - Line 5 – Expression Action –
        - O_ContainersNeeded = O_ContainersNeeded – M_FullContainerCount
    - Line 6 – Expression Action –
        - M_MakeContainers = M_FullPalletCount
    - Line 7 – Condition Action –
        - END IF

- **CheckContainersNeeded –**
    - Process/Object/Condition Routing – Condition Based Routing –
        - M_MakeContainers < 1 –
            - Next Process – GrabContainers
        - M_MakeContainers > 0 –
            - Next Process – MakeContainersForFullPallet
- **Connection Line Leading to MakeContainersForFullPallet –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy
- **MakeContainersForFullPallet –**
    - Process/Object/Condition Routing – Object Based Routing –
        - Containers –
            - Next Process – NumberofContainerstoSend
        - Item –
            - Next Process – SplitContainers
    - Modify/Split Object Type –

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - Select Object To Split – Item
            - Expand – Item
                - Select – New Object Definition
                - Object Type – Containers
                - Select – Containers
                - #of Objects To Create – M_MakeContainers
- Process Extensions/Event Handling – ObjectActivated – Insert one line –
    - Line 1 – Expression Action –
        - M_TotalContainers = M_TotalContainers – M_FullContainerCount

- **Connection Line Leading to NumberofContainerstoSend –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy
        - Enable Group Polling
    - Process Extensions/Event Handling – ObjectActivated – Insert three lines –
        - Line 1 – Condition Action –
            - IF –
                - O_ContainersNeeded < 1
        - Line 2 – Expression Action –
            - O_ContainersNeeded = 0
        - Line 3 – Condition Action –
            - END IF
- **NumberofContainerstoSend –**
    - Process/Object/Condition Routing – Condition Based Routing –
        - O_ContainersNeeded == P_TempInt –
            - Next Process – FullPallet
        - P_TempInt > O_ContainersNeeded –
            - Next Process – CheckPalletForRoom
    - Process Extensions/Event Handling – ObjectActivated – Insert one line –
        - Line 1 – DB Interface Action –
            - Database Name – Internal
            - Table Name – SKUInformation
            - Command Type – Select From
            - Condition – Where SKU = '<%O_SKU%>'
            - FullPalletCount – P_TempInt
            - Insert New Definition
- **Connection Line Leading to CheckPalletForRoom –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

- **CheckPalletForRoom –**
  - Process Extensions/Event Handling – ObjectActivated – Insert three line –
    - Line 1 – Condition Action –
      - IF –
        - (O_ContainerVolume > M_MixedPalletVolume) OR (M_MixedDestinationID <> O_Destination)
    - Line 2 – Expression Action –
      - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeMixedPallet'), 'Pallet', 1, 1)
    - Line 3 – Condition Action –
      - END IF
- **Connection Line Leading to MakeMixedPallet –**
  - Behavior: Conveyors, Polling, and Carriers – Check the following –
    - Block if next destination is busy
    - Ignore Active Objects Based on Assembly
    - Allow Objects Based on Assembly
    - Based on Code Condition
      - Edit Condition –
        - M_MixedPalletVolume >= O_ContainerVolume
  - Process Extensions/Event Handling – ObjectActivated – Insert six lines –
    - Line 1 – Expression Action –
      - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeMixedPallet'), 'Pallet', 0, 0)
    - Line 2 – Condition Action –
      - IF –
        - M_MixedDestinationID == 0
    - Line 3 – Expression Action –
      - M_MixedDestinationID = O_Destination
    - Line 4 – Condition Action –
      - ELSE IF –
        - (M_MixedPalletVolume < O_ContainerVolume) OR (M_MixedDestinationID <> O_Destination)
    - Line 5 – Expression Action –
      - M_MixedDestinationID = O_Destination
    - Line 6 – Condition Action –
      - END IF
- **MakeMixedPallet –**
  - Process Assembly/Disassembly – Enable object assembly –

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

    - Add Object Definition –
        - Pallet
        - Check –
            - Assemble
            - Create Object
- Process Extensions/Event Handling – ObjectProcessingCompleted – Insert two line –
    - Line 1 – Expression Action –
        - O_Destination = P_Router
    - Line 2 – Expression Action –
        - O_SKU = P_TempSKU
- Process Extensions/Event Handling – ObjectActivatedForAssembly – Insert ten lines –
    - Line 1 – Expression Action –
        - P_TempInt = 0
    - Line 2 – Expression Action –
        - P_Router = O_Destination
    - Line 3 – Expression Action –
        - P_TempSKU = O_SKU
    - Line 4 – Expression Action –
        - M_MixedPalletVolume = M_MixedPalletVolume – O_ContainerVolume
    - Line 5 – Expression Action –
        - M_MixedDestinationID = O_Destination
    - Line 6 – Expression Action –
        - M_MixedPalletID = 'M_' & IntToStr(M_MixedPalletCount)
    - Line 7 – DB Interface Action –
        - Database Name – Internal
        - Table Name – PalletsBuilt
        - Command Type – Select From
        - Condition – Where SKU = '<%O_SKU%>' AND PalletID = '<%M_MixedPalletID%>'
        - PalletCapacity – P_TempInt
        - Insert New Definition – Select the definition – Edit Action Definition –
            - Line 1 – Expression Action –
                - P_TempInt = P_TempInt + 1
            - Line 2 – DB Interface Action –
                - Database Name – Internal
                - Table Name – PalletsBuilt
                - Command Type – Update Table
                - Condition – Where SKU = '<%O_SKU%>' AND PalletID = '<%M_MixedPalletID%>'

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - PalletCapacity – P_TempInt
        - VolumeRemaining – M_MixedPalletVolume
        - Insert New Definition – Close – Close and Save – Close
    - Line 8 – Condition Action –
        - IF –
            - P_TempInt < 1
    - Line 9 – DB Interface Action –
        - Database Name – Internal
        - Table Name – PalletsBuilt
        - Command Type – Insert Into
        - SKU – O_SKU
        - PalletID – M_MixedPalletID
        - DestinationID – M_MixedDestinationID
        - ContainerCount – 1
        - CapableVolume – 92160
        - VolumeRemaining – M_MixedPalletVolume
        - PalletType – 'MixedPallet'
        - Insert New Definition
    - Line 10 – Condition Action –
        - END IF
- **Connection Line Leading to FullPallet –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy
- **FullPallet –**
    - Process Extensions/Event Handling – ObjectActivated – Insert three line –
        - Line 1 – Expression Action –
            - IF –
                - (M_FullPalletVolume < O_ContainerVolume) OR (M_FullDestinationID <> O_Destination) OR (M_FullCurrentCount >= M_FullContainerCount)
            - Line 2 – Expression Action –
                - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeFullPallet'), 'Pallet', 1, 1)
            - Line 3 – Condition Action –
                - END IF
- **Connection Line Leading to MakeFullPallet –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - Ignore Active Objects at Destination
        - Allow Objects Based on Assembly
        - Based on Code Condition
            - Edit Condition –
                - M_FullPalletVolume >= O_ContainerVolume AND M_FullCurrentCount < M_FullContainerCount
- Process Extensions/Event Handling – ObjectProcessingCompleted – Insert six lines –
    - Line 1 – Expression Action –
        - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeFullPallet'), 'Pallet', 0, 0)
    - Line 2 – Condition Action –
        - IF –
            - M_FullDestinationID == 0
    - Line 3 – Expression Action –
        - M_FullDestinationID = O_Destination
    - Line 4 – Condition Action –
        - ELSE IF –
            - (M_FullPalletVolume < O_ContainerVolume) OR (M_FullDestinationID <> O_Destination)
    - Line 5 – Expression Action –
        - M_FullDestinationID = O_Destination
    - Line 6 – Condition Action –
        - END IF

- **MakeFullPallet –**
    - Process Assembly/Disassembly – Enable object assembly –
        - Add Object Definition –
            - Pallet
            - Check –
                - Assemble
                - Create Object
    - Process Extensions/Event Handling – ObjectProcessingCompleted – Insert three lines –
        - Line 1 – Expression Action –
            - O_Destination = P_Router
        - Line 2 – Expression Action –
            - O_SKU = P_TempSKU
        - Line 3 – Expression Action –
            - O_ItemCount = P_TempCurrent
    - Process Extensions/Event Handling – ObjectActivatedForAssembly – Insert five lines –
        - Line 1 – Expression Action –

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

        - P_Router = O_Destination
    - Line 2 – Expression Action –
        - P_TempSKU = O_SKU
    - Line 3 – Expression Action –
        - M_FullPalletVolume = M_FullPalletVolume – O_ContainerVolume
    - Line 4 – Expression Action –
        - M_FullCurrentCount = M_FullCurrentCount + 1
    - Line 5 – Expression Action –
        - P_TempInt = O_ItemCount
- Process Extensions/Event Handling – ObjectAssemblyCompleted – Insert three lines –
    - Line 1 – Condition Action –
        - IF –
            - M_FullCurrentCount == M_FullContainerCount
    - Line 2 – Expression Action –
        - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeFullPallet'), 'Pallet', 1, 5)
    - Line 3 – Condition Action –
        - END IF

- **Connection Line Leading to Labeling from MakeMixedPallet –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy
        - Ignore Object in Transition
    - Process Extensions/Event Handling – ObjectActivated – Insert three lines –
        - Line 1 – Expression Action –
            - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeMixedPallet'), 'Pallet', 0, 0)
        - Line 2 – Expression Action –
            - M_MixedPalletVolume = 92160
        - Line 3 – Expression Action –
            - M_MixedPalletCount = M_MixedPalletCount + 1
- **Connection Line Leading to Labeling from MakeFullPallet –**
    - Behavior: Conveyors, Polling, and Carriers – Check the following –
        - Block if next destination is busy
    - Process Extensions/Event Handling – ObjectActivated – Insert five lines –
        - Line 1 – Expression Action –
            - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeFullPallet'), 'Pallet', 0, 0)
        - Line 2 – DB Interface Action –
            - Database name – Internal

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

            - Table Name – PalletsBuilt
            - Command Type – Insert Into
            - SKU – O_SKU
            - PalletID – '<%EXPS 'F_' & IntToStr(M_FullPalletCount)%>'
            - DestinationID – M_FullDestinationID
            - PalletContainerCount – M_FullContainerCount
            - CapacityVolume – 92160
            - VolumeRemaining – M_FullPalletVolume
            - PalletType – 'FullPallet'
        - Line 3 – Expression Action –
            - M_FullPalletVolume = 92160
        - Line 4 – Expression Action –
            - M_FullCurrentCount = 0
        - Line 5 – Expression Action –
            - M_FullPalletCount = M_FullPalletCount + 1
- **Labeling –**
    - Process Extensions/Event Handling – ObjectActivated – Insert one line –
        - Line 1 – DB Interface Action –
            - Database Name – Internal
            - Table Name – SKURequests
            - Command Type – Update Table
            - Condition – Where DestinationID = '<%O_Destination%>' AND SKU = '<%O_SKU%>' AND Picking = 1
            - DateFullfilled – '<%EXPS GetSystemDateMMDDYYY(ModelID)%>'
            - Insert New Definition

End Processes –

- **NoRequests –**
    - Process Extensions/Event Handling – ObjectActivated – Insert one line –
        - Line 1 – Expression Action –
            - SetAllowPartialAssembly(GetProcessID(ModelID, 'MakeMixedPallet'), 'Pallet', 1, 5)

Output –

- Tables –
    - PickSequence –
        - Lists all orders requested by the model and the record of whether or not the full order was picked or if items were left out of the order and how many of each

@Copyright 2017, CreateASoft, Inc. All Rights Reserved

item. This table lists the OrderNumber, each SKU, and quantity of each SKU for every order.

@Copyright 2017, CreateASoft, Inc. All Rights Reserved