

Cait Rahm
Peter Faiman
Chase Ricketts
Jessica Cosio

CreateFulfiller

```
<wsdl:operation name="createFulfiller">
<wsdl:documentation>Request to create a single fulfiller which will later contain one or
more locations</wsdl:documentation>
<wsdl:input name="createFulfillerRequest" message="impl:createFulfillerRequest"/>
<wsdl:output name="createFulfillerResponse" message="impl:createFulfillerResponse"/>
```

Input

```
<wsdl:message name="createFulfillerRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:createFulfiller"/>
</wsdl:message>
```

```
<element name="createFulfiller">
<complexType>
<sequence>
<element name="request" type="impl:FulfillerRequest"/>
</sequence>
</complexType>
</element>
```

```
<complexType name="FulfillerRequest">
<sequence>
<element name="Name" type="xsd:string" nillable="true"/>
<element name="ManufacturerID" type="xsd:positiveInteger" nillable="true"/>
<element name="RetailerID" type="xsd:positiveInteger" nillable="true"/>
</sequence>
</complexType>
```

Output

```
<element name="createFulfillerResponse">
<complexType>
<sequence>
<element name="createFulfillerReturn" type="xsd:int"/>
</sequence>
</complexType>
</element>
```

SQL

```
--Must ask customer about relevance of ManufacturerID
INSERT INTO Fulfillers
VALUES(RetailerID, Name);
```

GetFulfillerStatus

```
<wsdl:operation name="getFulfillerStatus">
<wsdl:documentation>Data retrieval to check a fulfiller's status</wsdl:documentation>
<wsdl:input name="getFulfillerStatusRequest" message="impl:getFulfillerStatusRequest"/>
<wsdl:output name="getFulfillerStatusResponse" message="impl:getFulfillerStatusResponse"/>
```

```
</wsdl:operation>
```

Input

```
<wsdl:message name="getFulfillerStatusRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getFulfillerStatus"/>

<element name="getFulfillerStatus">
<complexType>
<sequence>
<element name="fulfillerID" type="xsd:int"/>
</sequence>
</complexType>
</element>
```

Output

```
<element name="getFulfillerStatusResponse">
<complexType>
<sequence>
<element name="getFulfillerStatusReturn" type="xsd:int"/>
</sequence>
</complexType>
</element>
```

SQL

```
--Status being present in table?
--Assuming status means present in table
--return 0 if not and 1 if so
SELECT COUNT(*)
FROM Fulfillers
WHERE fulfillerid = fulfillerID;
```

CreateFulfillmentLocation

```
<wsdl:operation name="createFulfillmentLocation">
<wsdl:input name="createFulfillmentLocationRequest"
message="impl:createFulfillmentLocationRequest"/>
<wsdl:output name="createFulfillmentLocationResponse"
message="impl:createFulfillmentLocationResponse"/>
</wsdl:operation>
```

Input

```
<wsdl:message name="createFulfillmentLocationRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:createFulfillmentLocation"/>
</wsdl:message>

<element name="createFulfillmentLocation">
<complexType>
<sequence>
<element name="request" type="impl:FulfillmentLocation"/>
</sequence>
</complexType>
</element>
```

```

<complexType name="FulfillmentLocation">
  <sequence>
    <element name="FulfillerID" type="xsd:positiveInteger"/>
    <element name="ManufacturerLocationID" type="xsd:positiveInteger" nillable="true"/>
    <element name="RetailerLocationID" type="xsd:positiveInteger" nillable="true"/>
    <element name="ExternalLocationID" type="xsd:string" nillable="true"/>
    <element name="LocationName" type="xsd:string" nillable="true"/>
    <element name="TypeID" type="xsd:positiveInteger"/>
    <element name="Latitude" type="xsd:double"/>
    <element name="Longitude" type="xsd:double"/>
    <element name="Status">
      <simpleType>
        <restriction base="xsd:int">
          <enumeration value="1">
            <annotation>
              <documentation>Location Active</documentation>
            </annotation>
          </enumeration>
          <enumeration value="2">
            <annotation>
              <documentation>Location Not Active</documentation>
            </annotation>
          </enumeration>
        </restriction>
      </simpleType>
    </element>
    <element name="CountryCode" type="xsd:string" nillable="true"/>
  </sequence>
</complexType>

```

Output

```

<element name="createFulfillmentLocationResponse">
  <complexType>
    <sequence>
      <element name="createFulfillmentLocationReturn" type="xsd:int"/>
    </sequence>
  </complexType>
</element>

```

SQL

```

--Type in WSDL is int but type in data is string (assume data is correct)
INSERT INTO Locations(name, fulfillerid, extfulfillerid, locationid, type, coordinates,
status)
VALUES(LocationName, FulfillerID, ExternalLocationID, RetailerLocationID, TypeID,
ST_SetSRID(ST_MakePoint(Latitude, Longitude)), Status);

```

CreateManufacturerCatalog

```

<wsdl:operation name="createManufacturerCatalog">
  <wsdl:input name="createManufacturerCatalogRequest"
message="impl:createManufacturerCatalogRequest"/>
  <wsdl:output name="createManufacturerCatalogResponse"
message="impl:createManufacturerCatalogResponse"/>

```

```
</wsdl:operation>
```

Input

```
<wsdl:message name="createManufacturerCatalogRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:createManufacturerCatalog"/>
</wsdl:message>
```

```
<element name="createManufacturerCatalog">
<complexType>
<sequence>
<element name="request" type="impl:ManufacturerCatalogRequest"/>
</sequence>
</complexType>
</element>
```

```
<complexType name="ManufacturerCatalogRequest">
<sequence>
<element name="ManufacturerID" type="xsd:positiveInteger"/>
<element name="CatalogID" type="xsd:positiveInteger"/>
<element name="FulfillerLocationID" type="xsd:positiveInteger"/>
<element name="Status" type="xsd:int"/>
</sequence>
</complexType>
```

Output

```
<element name="createManufacturerCatalogResponse">
<complexType/>
</element>
```

SQL

--CatalogID is string in data but int in WSDL (assume data is correct)

```
INSERT INTO Catalogs(catalogid, manufacturerid)
VALUES(CatalogID, ManufacturerID);
```

```
INSERT INTO LocationsManCats
```

```
VALUES(FulfillerLocationID, (SELECT id FROM Catalogs WHERE catalogid = CatalogID AND
manufacturerid = ManufacturerID));
```

GetFulfillmentLocations

```
<wsdl:operation name="getFulfillmentLocations">
<wsdl:input name="getFulfillmentLocationsRequest"
message="impl:getFulfillmentLocationsRequest"/>
<wsdl:output name="getFulfillmentLocationsResponse"
message="impl:getFulfillmentLocationsResponse"/>
</wsdl:operation>
```

Input

```
<wsdl:message name="getFulfillmentLocationsRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getFulfillmentLocations"/>
</wsdl:message>
```

```

<element name="getFulfillmentLocations">
  <complexType>
    <sequence>
      <element name="request" type="impl:OrderRequest"/>
    </sequence>
  </complexType>
</element>

<complexType name="OrderRequest">
  <sequence>
    <element name="Catalog" type="impl:ManufacturerCatalog"/>
    <element name="Location" type="impl:RequestLocation" nillable="true"/>
    <element name="Items" type="impl:ArrayOf_impl_ItemQuantity" nillable="true"/>
    <element name="MaxLocations" type="xsd:positiveInteger" nillable="true"/>
    <element name="RejectedFulfillmentLocations" type="impl:ArrayOf_xsd_FulfillerLocationID"
nillable="true"/>
  </sequence>
</complexType>

<complexType name="ManufacturerCatalog">
  <sequence>
    <element name="ManufacturerID" type="xsd:positiveInteger"/>
    <element name="CatalogID" type="xsd:positiveInteger"/>
  </sequence>
</complexType>

<complexType name="RequestLocation">
  <sequence>
    <element name="Unit" type="xsd:string" default="MILES" nillable="true"/>
    <element name="Radius" type="xsd:positiveInteger" nillable="true"/>
    <element name="PostalCode" type="xsd:string" nillable="true"/>
    <element name="Latitude" type="xsd:double" nillable="true"/>
    <element name="Longitude" type="xsd:double" nillable="true"/>
    <element name="CountryCode" type="xsd:string" nillable="true"/>
  </sequence>
</complexType>

<complexType name="ArrayOf_impl_ItemQuantity">
  <sequence>
    <element name="items" type="impl:ItemQuantity" minOccurs="1" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="ItemQuantity">
  <sequence>
    <element name="PartNumber" type="xsd:string" nillable="true"/>
    <element name="UPC" type="xsd:string" nillable="true"/>
    <element name="LocationUPC" type="xsd:string" nillable="true"/>
    <element name="Quantity" type="xsd:int"/>
  </sequence>
</complexType>

```

```

<complexType name="ArrayOf_xsd_FulfillerLocationID">
  <sequence>
    <element name="items" type="xsd:positiveInteger" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

```

Output

```

<element name="getFulfillmentLocationsResponse">
  <complexType>
    <sequence>
      <element name="getFulfillmentLocationsReturn" type="impl:AssignmentResponse"
maxOccurs="unbounded"/>
    </sequence>
  </complexType>
</element>

```

```

<complexType name="AssignmentResponse">
  <sequence>
    <element name="FulfillerID" type="xsd:positiveInteger"/>
    <element name="FulfillerLocationID" type="xsd:positiveInteger"/>
    <element name="Items" type="impl:ArrayOf_impl_ItemQuantity" nillable="true"/>
  </sequence>
</complexType>

```

SQL

```

SELECT fp.fulfillerid, lp.locationid, p.upc
FROM FulfillersProducts fp, Products p, Catalogs c, LocationsProducts lp
WHERE fp.fulfillerid = lp.fulfillerid AND p.upc = fp.upc AND p.mancatid = c.id AND
c.catalogid = CatalogID;

```

GetFulfillmentLocationTypes

```

<wsdl:operation name="getFulfillmentLocationTypes">
  <wsdl:input name="getFulfillmentLocationTypesRequest"
    message="impl:getFulfillmentLocationTypesRequest"/>
  <wsdl:output name="getFulfillmentLocationTypesResponse"
    message="impl:getFulfillmentLocationTypesResponse"/>
</wsdl:operation>

```

Input

```

<wsdl:message name="getFulfillmentLocationTypesRequest">
  <wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
  <wsdl:part name="parameters" element="impl:getFulfillmentLocationTypes"/>
</wsdl:message>

```

```

<element name="getFulfillmentLocationTypes">
  <complexType/>
</element>

```

Output

```

<element name="getFulfillmentLocationTypesResponse">
  <complexType>
    <sequence>
      <element name="getFulfillmentLocationTypesReturn"

```

```

        type="impl:FulfillmentLocationType" maxOccurs="unbounded"/>
    </sequence>
</complexType>
</element>

<complexType name="FulfillmentLocationType">
    <sequence>
        <element name="FulfillmentLocationTypeID" type="xsd:positiveInteger"/>
        <element name="Description" type="xsd:string" nillable="true"/>
    </sequence>
</complexType>

```

SQL

```

-- Data has no typeID and WSDL does (assume data correct)
-- should consider normalizing types or creating an enum
SELECT DISTINCT type
FROM Locations;

```

AllocateInventory

Increase allocation, decrease on-hand

```

<wsdl:operation name="allocateInventory">
    <wsdl:input name="allocateInventoryRequest"
        message="impl:allocateInventoryRequest"/>
    <wsdl:output name="allocateInventoryResponse"
        message="impl:allocateInventoryResponse"/>
</wsdl:operation>

```

Input

```

<wsdl:message name="allocateInventoryRequest">
    <wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
    <wsdl:part name="parameters" element="impl:allocateInventory"/>
</wsdl:message>

<element name="allocateInventory">
    <complexType>
        <sequence>
            <element name="request" type="impl:UpdateRequest"/>
        </sequence>
    </complexType>
</element>

<complexType name="UpdateRequest">
    <sequence>
        <element name="FulfillerLocationCatalog" type="impl:FulfillmentLocationCatalog"/>
        <element name="Items" type="impl:ArrayOf_impl_UpdateItem"/>
    </sequence>
</complexType>

<complexType name="FulfillmentLocationCatalog">
    <sequence>
        <element name="ManufacturerCatalog" type="impl:ManufacturerCatalog" nillable="true"/>
        <element name="FulfillerLocationID" type="xsd:positiveInteger"/>
    </sequence>
</complexType>

```

```

        <element name="ManufacturerLocation" type="impl:ManufacturerLocation"
            nillable="true"/>
        <element name="RetailerLocation" type="impl:RetailerLocation" nillable="true"/>
    </sequence>
</complexType>

<complexType name="ArrayOf_impl_UpdateItem">
    <sequence>
        <element name="items" type="impl:UpdateItem" minOccurs="0" maxOccurs="unbounded"/>
    </sequence>
</complexType>

<complexType name="UpdateItem">
    <sequence>
        <element name="PartNumber" type="xsd:string" nillable="true"/>
        <element name="UPC" type="xsd:string" nillable="true"/>
        <element name="LocationUPC" type="xsd:string" nillable="true"/>
        <element name="Quantity" type="xsd:int"/>
        <element name="OrderID" type="xsd:positiveInteger"/>
        <element name="OrderItemID" type="xsd:positiveInteger"/>
        <element name="ShipmentID" type="xsd:positiveInteger"/>
        <element name="FulfillerLocationID" type="xsd:positiveInteger" nillable="true"/>
    </sequence>
</complexType>

```

Output

```

<wsdl:message name="allocateInventoryResponse">
    <wsdl:part name="parameters" element="impl:allocateInventoryResponse"/>
</wsdl:message>

<element name="reallocateInventoryResponse">
    <complexType/>
</element>

```

SQL

```

-- Increase allocation, decrease on-hand
SELECT BinsProducts.fulfillerid, BinsProducts.sku, BinsProducts.binid, availableinventory
FROM BinsProducts, Bins, Locations
WHERE Bins.locationid = Locations.locationid
AND Bins.id = BinsProducts.binid
AND BinsProducts.sku = ?
AND Locations.locationid = ?
FOR UPDATE

```

```

-- With parameters: PartNumber, FulfillerLocationID

```

```

-- Verify available inventory first
UPDATE BinsProducts
SET onhandinventory = onhandinventory - ?,
    allocatedinventory = allocatedinventory + ?
WHERE fulfillerid = ?
AND sku = ?
AND binid = ?

```


-- With Parameters: Quantity, Quantity, fulfillerid, sku, binid

DeallocateInventory

Decrease allocation, increase on-hand

```
<wsdl:operation name="deallocateInventory">
  <wsdl:input name="deallocateInventoryRequest"
    message="impl:deallocateInventoryRequest"/>
  <wsdl:output name="deallocateInventoryResponse"
    message="impl:deallocateInventoryResponse"/>
</wsdl:operation>
```

Input

```
<wsdl:message name="deallocateInventoryRequest">
  <wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
  <wsdl:part name="parameters" element="impl:deallocateInventory"/>
</wsdl:message>

<complexType name="UpdateRequest">
  <sequence>
    <element name="FulfillerLocationCatalog" type="impl:FulfillmentLocationCatalog"/>
    <element name="Items" type="impl:ArrayOf_impl_UpdateItem"/>
  </sequence>
</complexType>

<complexType name="FulfillmentLocationCatalog">
  <sequence>
    <element name="ManufacturerCatalog" type="impl:ManufacturerCatalog" nillable="true"/>
    <element name="FulfillerLocationID" type="xsd:positiveInteger"/>
    <element name="ManufacturerLocation" type="impl:ManufacturerLocation"
      nillable="true"/>
    <element name="RetailerLocation" type="impl:RetailerLocation" nillable="true"/>
  </sequence>
</complexType>

<complexType name="ArrayOf_impl_UpdateItem">
  <sequence>
    <element name="items" type="impl:UpdateItem" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="UpdateItem">
  <sequence>
    <element name="PartNumber" type="xsd:string" nillable="true"/>
    <element name="UPC" type="xsd:string" nillable="true"/>
    <element name="LocationUPC" type="xsd:string" nillable="true"/>
    <element name="Quantity" type="xsd:int"/>
    <element name="OrderID" type="xsd:positiveInteger"/>
    <element name="OrderItemID" type="xsd:positiveInteger"/>
    <element name="ShipmentID" type="xsd:positiveInteger"/>
    <element name="FulfillerLocationID" type="xsd:positiveInteger" nillable="true"/>
  </sequence>
</complexType>
```

```
    </sequence>
</complexType>
```

Output

```
<wsdl:message name="deallocateInventoryResponse">
  <wsdl:part name="parameters" element="impl:deallocateInventoryResponse"/>
</wsdl:message>

<element name="deallocateInventoryResponse">
  <complexType/>
</element>
```

SQL

```
-- Decrease allocation, increase on-hand
SELECT BinsProducts.fulfillerid, BinsProducts.sku, BinsProducts.binid,
FROM BinsProducts, Bins, Locations
WHERE Bins.locationid = Locations.locationid
AND Bins.id = BinsProducts.binid
AND BinsProducts.sku = ?
AND Locations.locationid = ?
FOR UPDATE
```

```
-- With parameters: PartNumber, FulfillerLocationID
```

```
UPDATE BinsProducts
SET onhandinventory = onhandinventory + ?,
allocatedinventory = allocatedinventory - ?
WHERE fulfillerid = ?
AND sku = ?
AND binid = ?
```

```
-- With Parameters: Quantity, Quantity, fulfillerid, sku, binid
```

FulfillInventory

Decrease allocation and on-hand.

```
<wsdl:operation name="fulfillInventory">
  <wsdl:input name="fulfillInventoryRequest" message="impl:fulfillInventoryRequest"/>
  <wsdl:output name="fulfillInventoryResponse" message="impl:fulfillInventoryResponse"/>
</wsdl:operation>
```

Input

```
<wsdl:message name="fulfillInventoryRequest">
  <wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
  <wsdl:part name="parameters" element="impl:fulfillInventory"/>
</wsdl:message>

<element name="fulfillInventory">
  <complexType>
    <sequence>
      <element name="request" type="impl:UpdateRequest"/>
    </sequence>
```

```

    </complexType>
</element>

<complexType name="UpdateRequest">
  <sequence>
    <element name="FulfillerLocationCatalog" type="impl:FulfillmentLocationCatalog"/>
    <element name="Items" type="impl:ArrayOf_impl_UpdateItem"/>
  </sequence>
</complexType>

<complexType name="ArrayOf_impl_UpdateItem">
  <sequence>
    <element name="items" type="impl:UpdateItem" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="FulfillmentLocationCatalog">
  <sequence>
    <element name="ManufacturerCatalog" type="impl:ManufacturerCatalog" nillable="true"/>
    <element name="FulfillerLocationID" type="xsd:positiveInteger"/>
    <element name="ManufacturerLocation" type="impl:ManufacturerLocation"
      nillable="true"/>
    <element name="RetailerLocation" type="impl:RetailerLocation" nillable="true"/>
  </sequence>
</complexType>

<complexType name="UpdateItem">
  <sequence>
    <element name="PartNumber" type="xsd:string" nillable="true"/>
    <element name="UPC" type="xsd:string" nillable="true"/>
    <element name="LocationUPC" type="xsd:string" nillable="true"/>
    <element name="Quantity" type="xsd:int"/>
    <element name="OrderID" type="xsd:positiveInteger"/>
    <element name="OrderItemID" type="xsd:positiveInteger"/>
    <element name="ShipmentID" type="xsd:positiveInteger"/>
    <element name="FulfillerLocationID" type="xsd:positiveInteger" nillable="true"/>
  </sequence>
</complexType>

```

Output

```

<wsdl:message name="fulfillInventoryResponse">
  <wsdl:part name="parameters" element="impl:fulfillInventoryResponse"/>
</wsdl:message>

<element name="fulfillInventoryResponse">
  <complexType/>
</element>

```

SQL

```

-- Decrease allocation and on-hand
SELECT BinsProducts.fulfillerid, BinsProducts.sku, BinsProducts.binid,
FROM BinsProducts, Bins, Locations
WHERE Bins.locationid = Locations.locationid

```

```
AND Bins.id = BinsProducts.binid
AND BinsProducts.sku = ?
AND Locations.locationid = ?
FOR UPDATE
```

```
-- With parameters: PartNumber, FulfillerLocationID
```

```
UPDATE BinsProducts
SET onhandinventory = onhandinventory - ?,
allocatedinventory = allocatedinventory - ?
WHERE fulfillerid = ?
AND sku = ?
AND binid = ?
```

```
-- With Parameters: Quantity, Quantity, fulfillerid, sku, binid
```

getItemLocationsByFulfiller

Use Case: gets a list of inventory for one item with regards to fulfiller and location

```
<wsdl:operation name="getItemLocationsByFulfiller">
<wsdl:input name="getItemLocationsByFulfillerRequest"
message="impl:getItemLocationsByFulfillerRequest"/>
<wsdl:output name="getItemLocationsByFulfillerResponse"
message="impl:getItemLocationsByFulfillerResponse"/>
```

Input

```
<wsdl:message name="getItemLocationsByFulfillerRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getItemLocationsByFulfiller"/>
```

```
<element name="getItemLocationsByFulfiller">
<complexType>
<sequence>
<element name="request" type="impl:InventorySearchRequest"/>
```

```
<complexType name="InventorySearchRequest">
<sequence>
<element name="FulfillerIDs" type="impl:ArrayOf_xsd_FulfillerID"/>
<element name="LocationID" type="xsd:positiveInteger" nillable="true"/>
<element name="PostalCode" type="xsd:string" nillable="true"/>
<element name="PartNumber" type="xsd:string" nillable="true"/>
<element name="UPC" type="xsd:string" nillable="true"/>
<element name="LocationUPC" type="xsd:string" nillable="true"/>
```

Output

```
<wsdl:message name="getItemLocationsByFulfillerResponse">
<wsdl:part name="parameters" element="impl:getItemLocationsByFulfillerResponse"/>
```

```
<element name="getItemLocationsByFulfillerResponse">
<complexType>
<sequence>
<element name="getItemLocationsByFulfillerReturn" type="impl:InventorySearchResponse"
minOccurs="0" maxOccurs="unbounded"/>
```

```

<complexType name="InventorySearchResponse">
<sequence>
<element name="FulfillerID" type="xsd:positiveInteger"/>
<element name="FulfillerLocationID" type="xsd:positiveInteger"/>
<element name="RetailerID" type="xsd:positiveInteger" nillable="true"/>
<element name="RetailerLocationID" type="xsd:positiveInteger" nillable="true"/>
<element name="ManufacturerID" type="xsd:positiveInteger" nillable="true"/>
<element name="ManufacturerLocationID" type="xsd:positiveInteger" nillable="true"/>
<element name="LocationName" type="xsd:string"/>
<element name="PartNumber" type="xsd:string" nillable="true"/>
<element name="UPC" type="xsd:string" nillable="true"/>
<element name="LocationUPC" type="xsd:string" nillable="true"/>
<element name="Available" type="xsd:int"/>
<element name="OnHand" type="xsd:int"/>
<element name="Allocated" type="xsd:int"/>
<element name="Distance" type="xsd:double"/>

```

SQL

```

-- Note, must iterate over array of fulfillers on backend and merge the results
-- Account for part number? Ask David.
-- Is RetailerId the same as FulfillerId? Ask David.
SELECT lp.FulfillerId, lp.LocationId, lp.FulfillerId, lp.LocationId,
       c.ManufacturerId, lp.LocationId, l.name, p.UPC,
       SUM(bp.AvailableInventory), SUM(bp.OnHandInventory), SUM(bp.AllocatedInventory),
       (l.Coordinants <@> PostalCodeCoordinants) as Distance
FROM LocationsProducts lp
JOIN FulfillersProducts fp ON (fp.SKU = lp.SKU)
JOIN Locations l on (l.Id = lp.LocationId)
JOIN Products p on (p.UPC = fp.UPC)
JOIN Catalogs c on (c.Id = d.ManCatId)
JOIN BinProducts bp on (bp.FulfillerId = lp.FulfillerId AND bp.SKU = fp.SKU)
WHERE lp.FulfillerId = FulfillerId
      AND lp.LocationId = LocationId
      AND fp.UPC = UPC
GROUP BY lp.FulfillerId, lp.LocationId, c.ManufacturerId, l.name, p.UPC;

```

createBin

Use Case: creates a new empty bin for a specific location

```

<wsdl:operation name="createBin">
<wsdl:input name="createBinRequest" message="impl:createBinRequest"/>
<wsdl:output name="createBinResponse" message="impl:createBinResponse"/>

```

Input

```

<wsdl:message name="createBinRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:createBin"/>

<element name="createBin">
<complexType>
<sequence>
<element name="request" type="impl:Bin"/>

```

```

<complexType name="Bin">
<sequence>
<element name="BinID" type="xsd:positiveInteger" nillable="true"/>
<element name="FulfillerLocationID" type="xsd:positiveInteger"/>
<element name="BinTypeID" type="xsd:positiveInteger"/>
<element name="BinStatusID" type="xsd:positiveInteger"/>
<element name="Name" type="xsd:string" nillable="true"/>

```

Output

```

<element name="createBinResponse">
<complexType>
<sequence>
<element name="createBinReturn" type="xsd:positiveInteger"/>

```

SQL

```

-- WSDL says BinTypeID is a number, but data says a string (ask David)
-- WSDL says BinStatusID is a number, but data says a string (ask David)
INSERT INTO Bins(Id, LocationId, Type, Status, Name)
VALUES (BinID, FulfillerLocationID, BinTypeID, BinStatusID, Name);

```

getBins

Use Case: loosely searches for a bin name among a fulfiller's bins

```

<wsdl:operation name="getBins">
<wsdl:input name="getBinsRequest" message="impl:getBinsRequest"/>
<wsdl:output name="getBinsResponse" message="impl:getBinsResponse"/>

```

Input

```

<wsdl:message name="getBinsRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getBins"/>

```

```

<element name="getBins">
<complexType>
<sequence>
<element name="request" type="impl:BinRequest"/>

```

```

<complexType name="BinRequest">
<sequence>
<element name="FulfillerLocationID" type="xsd:positiveInteger" nillable="false"/>
<element name="SearchTerm" type="xsd:string" nillable="true"/>
<element name="NumResults" type="xsd:positiveInteger" nillable="true"/>
<element name="ResultsStart" type="xsd:positiveInteger" nillable="true"/>

```

Output

```

<wsdl:message name="getBinsResponse">
<wsdl:part name="parameters" element="impl:getBinsResponse"/>

<element name="getBinsResponse">
<complexType>
<sequence>
<element name="getBinsReturn" type="impl:BinResponse" maxOccurs="unbounded"/>

```

```

<complexType name="BinResponse">
<sequence>
<element name="Bins" type="impl:ArrayOf_impl_Bin" nillable="false"/>
<element name="ResultCount" type="xsd:positiveInteger" nillable="false"/>

<complexType name="ArrayOf_impl_Bin">
<sequence>
<element name="items" type="impl:Bin" minOccurs="0" maxOccurs="unbounded"/>

<complexType name="Bin">
<sequence>
<element name="BinID" type="xsd:positiveInteger" nillable="true"/>
<element name="FulfillerLocationID" type="xsd:positiveInteger"/>
<element name="BinTypeID" type="xsd:positiveInteger"/>
<element name="BinStatusID" type="xsd:positiveInteger"/>
<element name="Name" type="xsd:string" nillable="true"/>

```

SQL

```

SELECT *
FROM Bins b
WHERE b.LocationId = FulfillerLocationID
      AND b.name LIKE '%SearchTerm%'
LIMIT NumResults OFFSET ResultsStart;

```

getBinTypes

Use Case: loosely searches for a bin type among a fulfiller's bins

```

<wsdl:operation name="getBins">
<wsdl:input name="getBinsRequest" message="impl:getBinsRequest"/>
<wsdl:output name="getBinsResponse" message="impl:getBinsResponse"/>

```

Input

```

<wsdl:message name="getBinsRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getBins"/>

```

```

<element name="getBins">
<complexType>
<sequence>
<element name="request" type="impl:BinRequest"/>

```

```

<complexType name="BinRequest">
<sequence>
<element name="FulfillerLocationID" type="xsd:positiveInteger" nillable="false"/>
<element name="SearchTerm" type="xsd:string" nillable="true"/>
<element name="NumResults" type="xsd:positiveInteger" nillable="true"/>
<element name="ResultsStart" type="xsd:positiveInteger" nillable="true"/>

```

Output

```

<wsdl:message name="getBinsResponse">
<wsdl:part name="parameters" element="impl:getBinsResponse"/>

```

```

<element name="getBinsResponse">

```

```

<complexType>
<sequence>
<element name="getBinsReturn" type="impl:BinResponse" maxOccurs="unbounded"/>

<complexType name="BinResponse">
<sequence>
<element name="Bins" type="impl:ArrayOf_impl_Bin" nillable="false"/>
<element name="ResultCount" type="xsd:positiveInteger" nillable="false"/>

<complexType name="ArrayOf_impl_Bin">
<sequence>
<element name="items" type="impl:Bin" minOccurs="0" maxOccurs="unbounded"/>

<complexType name="Bin">
<sequence>
<element name="BinID" type="xsd:positiveInteger" nillable="true"/>
<element name="FulfillerLocationID" type="xsd:positiveInteger"/>
<element name="BinTypeID" type="xsd:positiveInteger"/>
<element name="BinStatusID" type="xsd:positiveInteger"/>
<element name="Name" type="xsd:string" nillable="true"/>

```

SQL

```

SELECT *
FROM Bins b
WHERE b.LocationId = FulfillerLocationID
      AND b.type LIKE '%SearchTerm%'
LIMIT NumResults OFFSET ResultsStart;

```

getBinStatuses

```

<wsdl:operation name="getBinStatuses">
<wsdl:input name="getBinStatusesRequest" message="impl:getBinStatusesRequest"/>
<wsdl:output name="getBinStatusesResponse" message="impl:getBinStatusesResponse"/>
</wsdl:operation>

```

input

```

<wsdl:message name="getBinStatusesRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getBinStatuses"/>
</wsdl:message>

```

```

<element name="getBinStatuses">
<complexType/>
</element>

```

output

```

<wsdl:message name="getBinStatusesResponse">
<wsdl:part name="parameters" element="impl:getBinStatusesResponse"/>
</wsdl:message>

```

```

<element name="getBinStatusesResponse">
<complexType>

```



```

<sequence>
<element name="getBinStatusesReturn" type="impl:BinStatus" maxOccurs="unbounded"/>
</sequence>
</complexType>
</element>

<complexType name="BinStatus">
<sequence>
<element name="BinStatusID" type="xsd:positiveInteger"/>
<element name="Description" type="xsd:string" nillable="false"/>
</sequence>
</complexType>

```

SQL

```

SELECT binstatusid, description
FROM binstatus;

```

getInventory

```

<wsdl:operation name="getInventory">
<wsdl:input name="getInventoryRequest" message="impl:getInventoryRequest"/>
<wsdl:output name="getInventoryResponse" message="impl:getInventoryResponse"/>
</wsdl:operation>

```

input

```

<wsdl:message name="getInventoryRequest">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:getInventory"/>
</wsdl:message>

```

```

<element name="getInventory">
<complexType>
<sequence>
<element name="request" type="impl:InventoryRequest"/>
</sequence>
</complexType>
</element>

```

```

<complexType name="InventoryRequest">
<sequence>
<element name="Catalog" type="impl:ManufacturerCatalog" nillable="false"/>
<element name="Quantities" type="impl:ArrayOf_impl_ItemQuantity" nillable="false"/>
<element name="LocationNames" type="impl:ArrayOfLocationNames" nillable="true"/>
<element name="Location" type="impl:RequestLocation" nillable="true"/>
<element name="Type" type="impl:InventoryRequestType" nillable="false" minOccurs="1"/>
<element name="Limit" type="xsd:int" default="10000"/>
<element name="IgnoreSafetyStock" type="xsd:boolean" default="false" nillable="true"/>
<element name="IncludeNegativeInventory" type="xsd:boolean" default="false"
nillable="true"/>
<element name="OrderByLTD" type="boolean"/>
</sequence>
</complexType>

```

```

<complexType name="ManufacturerCatalog">
  <sequence>
    <element name="ManufacturerID" type="xsd:positiveInteger"/>
    <element name="CatalogID" type="xsd:positiveInteger"/>
  </sequence>
</complexType>

<complexType name="ArrayOf_impl_ItemQuantity">
  <sequence>
    <element name="items" type="impl:ItemQuantity" minOccurs="1" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="ItemQuantity">
  <sequence>
    <element name="PartNumber" type="xsd:string" nillable="true"/>
    <element name="UPC" type="xsd:string" nillable="true"/>
    <element name="LocationUPC" type="xsd:string" nillable="true"/>
    <element name="Quantity" type="xsd:int"/>
  </sequence>
</complexType>

<complexType name="ArrayOfLocationNames">
  <sequence>
    <element name="LocationNames" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="RequestLocation">
  <sequence>
    <element name="Unit" type="xsd:string" default="MILES" nillable="true"/>
    <element name="Radius" type="xsd:positiveInteger" nillable="true"/>
    <element name="PostalCode" type="xsd:string" nillable="true"/>
    <element name="Latitude" type="xsd:double" nillable="true"/>
    <element name="Longitude" type="xsd:double" nillable="true"/>
    <element name="CountryCode" type="xsd:string" nillable="true"/>
  </sequence>
</complexType>

<simpleType name="InventoryRequestType">
  <annotation>
    <documentation/>
  </annotation>
  <restriction base="xsd:string">
    <enumeration value="ALL"/>
    <enumeration value="PARTIAL"/>
    <enumeration value="ANY"/>
    <enumeration value="ALL_STORES"/>
  </restriction>
</simpleType>

```

output

```
<wsdl:message name="getInventoryResponse">
<wsdl:part name="parameters" element="impl:getInventoryResponse"/>
</wsdl:message>

<element name="getInventoryResponse">
<complexType>
<sequence>
<element name="getInventoryReturn" type="impl:InventoryResponse" maxOccurs="unbounded"/>
</sequence>
</complexType>
</element>

<complexType name="InventoryResponse">
<sequence>
<element name="LocationName" type="xsd:string" nillable="true"/>
<element name="CatalogID" type="xsd:int"/>
<element name="ManufacturerID" type="xsd:int"/>
<element name="OnHand" type="xsd:int" nillable="true"/>
<element name="Available" type="xsd:int" nillable="true"/>
<element name="PartNumber" type="xsd:string" nillable="true"/>
<element name="UPC" type="xsd:string" nillable="true"/>
<element name="LocationUPC" type="xsd:string" nillable="true"/>
<element name="LTD" type="xsd:double" nillable="true"/>
<element name="Floor" type="xsd:int" nillable="true"/>
<element name="SafetyStock" type="xsd:int" nillable="true"/>
<element name="STHEnabled" type="xsd:boolean" nillable="true"/>
<element name="RestockEnabled" type="xsd:boolean" nillable="true"/>
<element name="PickupEnabled" type="xsd:boolean" nillable="true"/>
<element name="CountryCode" type="xsd:string" nillable="true"/>
<element name="Distance" type="xsd:double" nillable="true"/>
</sequence>
</complexType>
```

SQL

```
SELECT l.name, c.catalogid, c.manufacturerid, bp.onhandinventory, bp.availableinventory,
l.upc, l.safetystocklimit, lp.ltd
FROM BinsProducts bp, FulfillersProducts fp, Products p, Catalogs c, Bins b, Locations l,
LocationsProducts lp
WHERE bp.fulfillerid = fp.fulfillerid AND fp.upc = p.upc AND p.mancatid = c.id AND
bp.binid = b.id AND b.locationid = l.locationid AND lp.locationid = l.locationid AND
l.name = LocationNames AND c.catalogid = CatalogID AND c.manufactuererid = ManufacturerID
ORDER BY lp.ltd;
```

adjustInventory

```
<wsdl:operation name="adjustInventory">
<wsdl:input message="impl:AdjustInventorySoapIn"/>
<wsdl:output message="impl:AdjustInventorySoapOut"/>
</wsdl:operation>
```

Input

```
<wsdl:message name="AdjustInventorySoapIn">
<wsdl:part name="AuthenticationHeader" element="impl:AuthenticationHeader"/>
```

```

<wsdl:part name="parameter" element="impl:AdjustRequest"/>
</wsdl:message>

<element name="AdjustRequest">
<complexType>
<sequence>
<element name="LocationName" type="xsd:string"/>
<element name="Items" type="impl:ArrayOf_impl_AdjustItem" nillable="true"/>
</sequence>
</complexType>

<complexType name="ArrayOf_impl_AdjustItem">
<sequence>
<element name="items" type="impl:AdjustItem" minOccurs="0" maxOccurs="unbounded"/>
</sequence>
</complexType>
<complexType name="AdjustItem">
<sequence>
<element name="PartNumber" type="xsd:string" nillable="true"/>
<element name="UPC" type="xsd:string" nillable="true"/>
<element name="LocationUPC" type="xsd:string" nillable="true"/>
<element name="BinID" type="xsd:int"/>
<element name="Quantity" type="xsd:int"/>
</sequence>
</complexType>

```

Output

```

<wsdl:message name="AdjustInventorySoapOut">
<wsdl:part name="parameter" element="impl:AdjustResponse"/>
</wsdl:message>

<element name="AdjustResponse" type="xsd:string"/>

```

SQL

```

UPDATE BinsProducts BP
SET onhandinventory = onhandinventory + quantity,
    availableinventory = availableinventory + quantity
WHERE upc = (SELECT upc FROM fulfillersproducts FP
WHERE BP.sku = FP.sku AND BP.fulfillerid = FP.fulfillerid)
AND locationname = (SELECT l.name FROM bins b, locations l
WHERE BP.binid = b.id AND b.locationid = l.id)
AND binid = BP.binid;

```

RefreshInventory

```

<wsdl:operation name="refreshInventory">
<wsdl:input name="refreshInventoryRequest2" message="impl:RefreshInventorySoapIn"/>
<wsdl:output name="refreshInventoryResponse2" message="impl:RefreshInventorySoapOut"/>
</wsdl:operation>

```

Input

```

<wsdl:message name="RefreshInventorySoapIn">
<wsdl:part name="parameter" element="impl:AuthenticationHeader"/>
<wsdl:part name="parameters" element="impl:RefreshRequest"/>
</wsdl:message>

<complexType name="RefreshRequest">
<sequence>
<element name="LocationName" type="xsd:string"/>
<element name="Items" type="impl:ArrayOf_impl_RefreshItem" nillable="true"/>
</sequence>
</complexType>

<complexType name="ArrayOf_impl_RefreshItem">
<sequence>
<element name="items" type="impl:RefreshItem" minOccurs="0" maxOccurs="unbounded"/>
</sequence>
</complexType>

<complexType name="RefreshItem">
<sequence>
<element name="PartNumber" type="xsd:string" nillable="true"/>
<element name="UPC" type="xsd:string" nillable="true"/>
<element name="LocationUPC" type="xsd:string" nillable="true"/>
<element name="BinID" type="xsd:int"/>
<element name="Quantity" type="xsd:int"/>
<element name="LTD" type="xsd:double"/>
<element name="Floor" type="xsd:int"/>
<element name="SafetyStock" type="xsd:int"/>
</sequence>
</complexType>

```

Output

```

<wsdl:message name="RefreshInventorySoapOut">
<wsdl:part name="parameter" element="impl:RefreshResponse"/>
</wsdl:message>

<element name="RefreshResponse" type="xsd:string"/>

```

SQL

```

UPDATE BinsProducts BP
SET onhandinventory = onhandinventory + quantity,
    availableinventory = availableinventory + quantity
WHERE upc = (SELECT upc FROM fulfillersproducts FP
WHERE BP.sku = FP.sku AND BP.fulfillerid = FP.fulfillerid)
AND binid = BP.binid;

```

```

UPDATE LocationsProducts LP
SET ltd = newLtd,
    safetystocklimit = newSafetyStock
WHERE locationid = (SELECT l.id FROM locations l, fulfillersproducts fp
WHERE l.id = LP.id AND fp.fulfillerid = LP.fulfillerid AND fp.sku = LP.sku);

```

