TRAILHEAD

Industries Order Management Overview

Exercise Guide

Version 1.0

TABLE OF CONTENTS

Preface

These training exercises are based on the Fall '20 release of Salesforce Industries Communications, Media and Energy & Utilities Clouds. For additional information about the topics covered in this module, see the documentation available in the Success Community at https://success.vlocity.com.

Overview

This training covers fundamental features and functionality of the Salesforce Industries Order Management (OM) application. This module has been designed to review basic Order Management concepts, the UI (user interface), as well as introduce technical products in the Shared Catalog.

This is a hands-on course with practical lab exercises. The lab exercises are designed to be used with the provided training playground.

What You Will Learn

When you complete this training, you will be able to:

- Navigate the UI for Order Management in general, and more specifically the XOM Administration and Vlocity Product Console
- Create *technical products*, or the services and resources that are used to fulfill orders
- Understand how commercial products, technical products, and product attributes are represented in the shared catalog

Prerequisites

The prerequisites for this training include a solid understanding of basic Salesforce concepts and functionality. You should also have a good understanding of the principles of order management and a working knowledge of the telecommunications, media, or energy and utilities industry business objectives.

Overview of Order Management

Salesforce Industries Order Management

Salesforce Industries Order Management (OM) provides the first catalog-driven order management system that is fully integrated with a best-in-class CRM and order capture system. It brings business agility to the cloud, using seamless design-time and runtime navigation and order operations, abstracting users away from the underlying technologies and platform components that comprise the system.

What benefits does Salesforce Industries Order Management bring to its customers?

- **Drives revenue by reducing time-to-value (TTV) for new commercial offers**

  The Salesforce Industries Order Management solution provides rich capabilities that support the full commercial offer life cycle, from product design and fulfillment configuration through order capture, fulfillment, and fallout handling. Order Management extends Salesforce with industry-specific capabilities that significantly reduce IT and maintenance costs. The Shared Catalog provides Order Management with a unified view of commercial offers, as well as product, service, and resource specifications for ease of configuration when defining fulfillment rules.

- **Significantly reduced configuration costs**

  Order Management focuses on the principle of reusing existing technical configurations to create new commercial offerings rather than having to create custom configurations each time. To achieve these goals, Order Management supports the concept of reusable technical product types. Fulfillment workflows are dynamically generated at runtime based on these rules, thus avoiding the need to create a large set of statically-defined workflows to handle the complex combinations of commercial offerings that can be present on orders. As a result, introducing new commercial offers requires minimal configuration and testing effort in Order Management.

- **Reduced fulfillment costs**

  Order Management supports a variety of orchestration constructs, including both manual and automated tasks that allow you to achieve the correct level of automation in workflows to meet present needs, as well as into the future, as more automation becomes possible. Order Management also enables any user to visualize end-to-end product configurations and fulfillment processes and to quickly identify root causes during order fallout investigation. All Order Management related design-time and runtime screens reside within Salesforce, providing a seamless user experience across all roles.

- **Improved customer experience**

  Order Management uses a next-generation catalog-driven architecture that fulfills orders faster and more accurately, using dynamically-generated workflows. Any user at any time can track order fulfillment progress without having to wait for orders to complete.

Order Management offers comprehensive functionality across two primary areas:

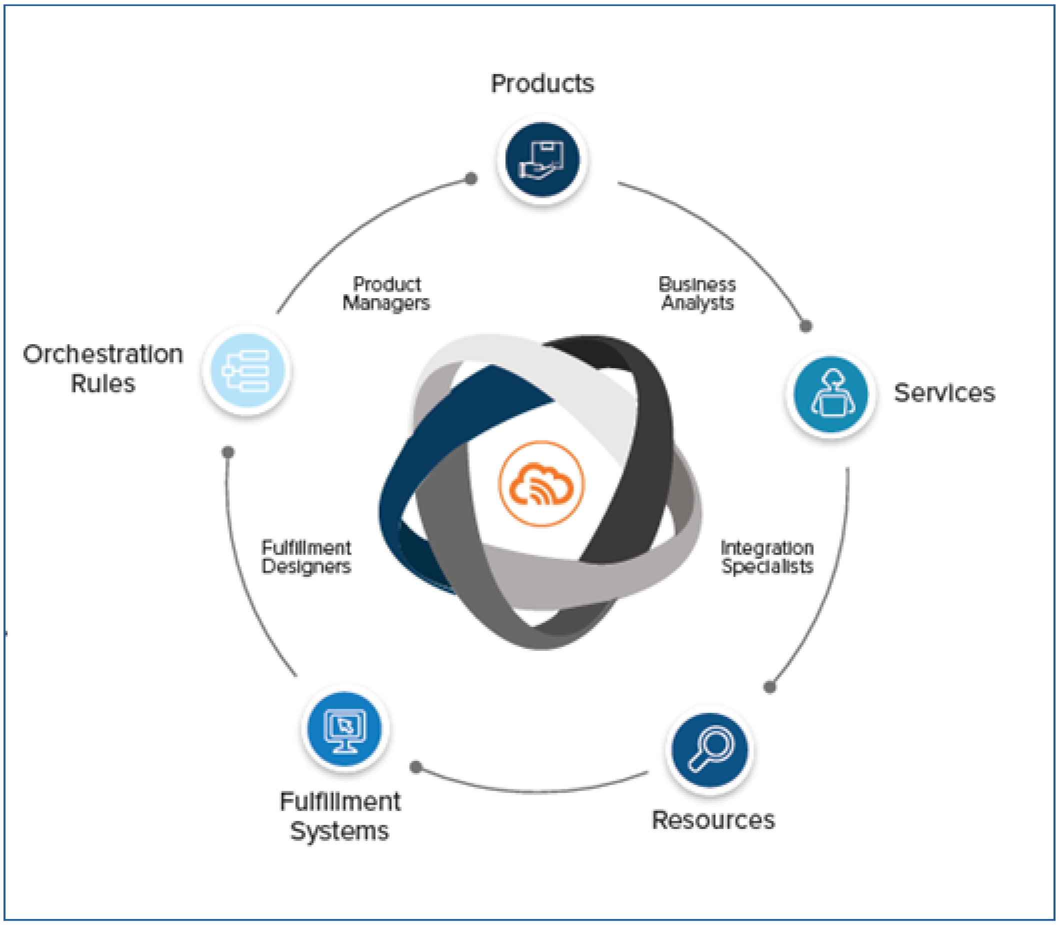
1. Order Decomposition
2. Order Orchestration and Management

What is Order Decomposition?

After the capture of a customer order, the first step in fulfillment processing is to decompose the order into sub-orders tailored for specific downstream systems. The decomposition process includes enriching the original order by executing rules that derive the technical attributes required to fulfill the order.

What is Order Orchestration and Management?

Once an order has been successfully decomposed, sub-orders are then fulfilled using an orchestration plan. The plan takes into account any dependencies that may exist between orchestration items in various flows. The orchestration execution engine evaluates the orchestration plan based upon pre-defined rules. It coordinates, sequences, and monitors all interactions with external fulfillment systems to ensure that they are ultimately successful.



NOTE:
For brevity sometimes **Salesforce Industries Order Management** is referred to as **Industries Order Management**, or even **Order Management** when the context provided makes it clear what is meant.
As a more generic industry term **order management** is used (all lower case).

Exercise 4-1: Salesforce Industries Order Management Overview

Scenario

At Infiwave, Greg has been riding high ever since his recent successful implementation of Industries CPQ. With that under his belt, he feels like he can take on, well, just about anything! Greg knows that his company has recently identified several key objectives, including increasing Time-to-Value margins for new product introduction and reducing fulfillment costs by modernizing their order management solution. Greg decides to take a look at Industries Order Management solution to determine if it will meet their requirements–and give him another opportunity to play the IT hero. Living the dream here!

Goal

- Tour the Salesforce Industries Order Management application
- Identify the difference between orchestration instances vs. orchestration definitions
- Define order management roles using *Vlocity XOM Administration*

Tasks

1. Review Order Management application tabs
2. Review the Vlocity XOM Administration tool

Time: 15 mins

⚠	**ALERT:** If you've just received your training playground, add your email address to the system administrator profile to ensure you receive all system notifications. In the upper-right, click on the **Avatar** and select **Settings**. Enter your email address in the **Email** field on the Personal Information page and click **Save**.

Task 1: Review Order Management application tabs

1. Using the **Lightning App Launcher** ⋮⋮⋮ , click **Order Management (OM)**.
2. Notice the **Orders** and **Products** tabs. These are standard Salesforce page layouts that have been enhanced with Order Management functionality.
3. Notice the **Vlocity Product Console** tab. This houses Industries Shared Catalog that is the foundation of Order Management.
4. Notice the **Orchestration Plans** and **Orchestration Plan Definitions** tabs.

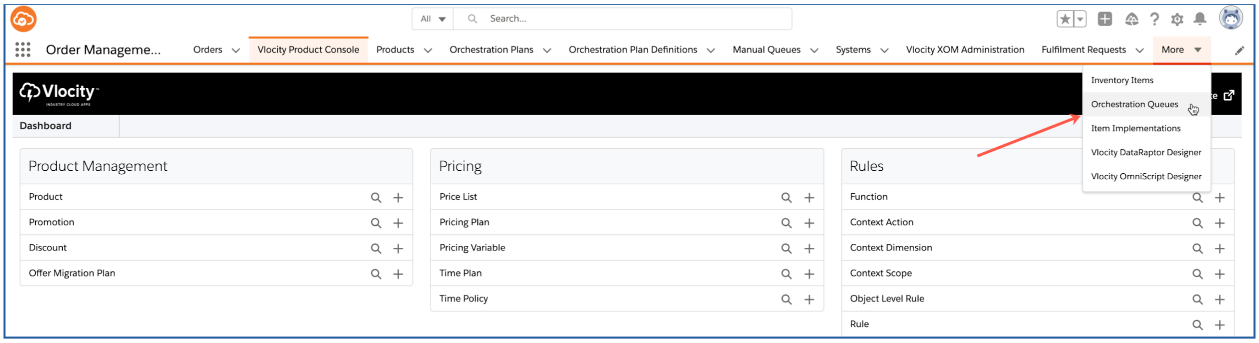
   These tabs illustrate a key organizing principle of Order Management: run-time orchestration objects in most cases have a corresponding design-time definition object. The run-time object is an "instance of" the definition object.

   Fulfillment designers create and test orchestration definitions during design-time, and then, at run-time when an order is submitted that triggers the definition, the orchestration execution engine spawns an "instance of" the definition that contains all the details of that orchestration object as it relates to the specific order and the results of its decomposition.

   We will review what each of these objects does during the order management process, but for now, just realize the link between them.

Run-time Instance	**Design-time Definition**
Orchestration Plan	Orchestration Plan Definition
Orchestration Item	Orchestration Item Definition
Orchestration Dependency	Orchestration Dependency Definition

5. In the Lightning tab navigation bar, click **More** to expand the list of overflow tabs.

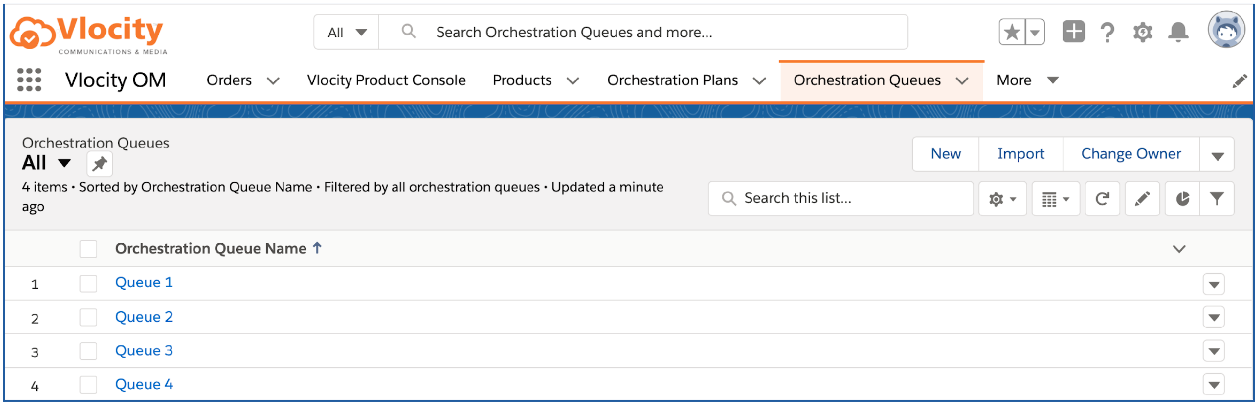


6. Click **Orchestration Queues**.

7. Click **Recently Viewed**, and then select **All** from drop-down menu of the list views.

8. Notice the four orchestration queues.

NOTE:
The orchestration execution engine automatically and dynamically load-balances order processing tasks (Orchestration Items) across the four orchestration queues when run on the Salesforce platform.



Orchestration Queues
All

4 items • Sorted by Orchestration Queue Name • Filtered by all orchestration queues • Updated a minute ago

		Orchestration Queue Name ↑
1	□	Queue 1
2	□	Queue 2
3	□	Queue 3
4	□	Queue 4

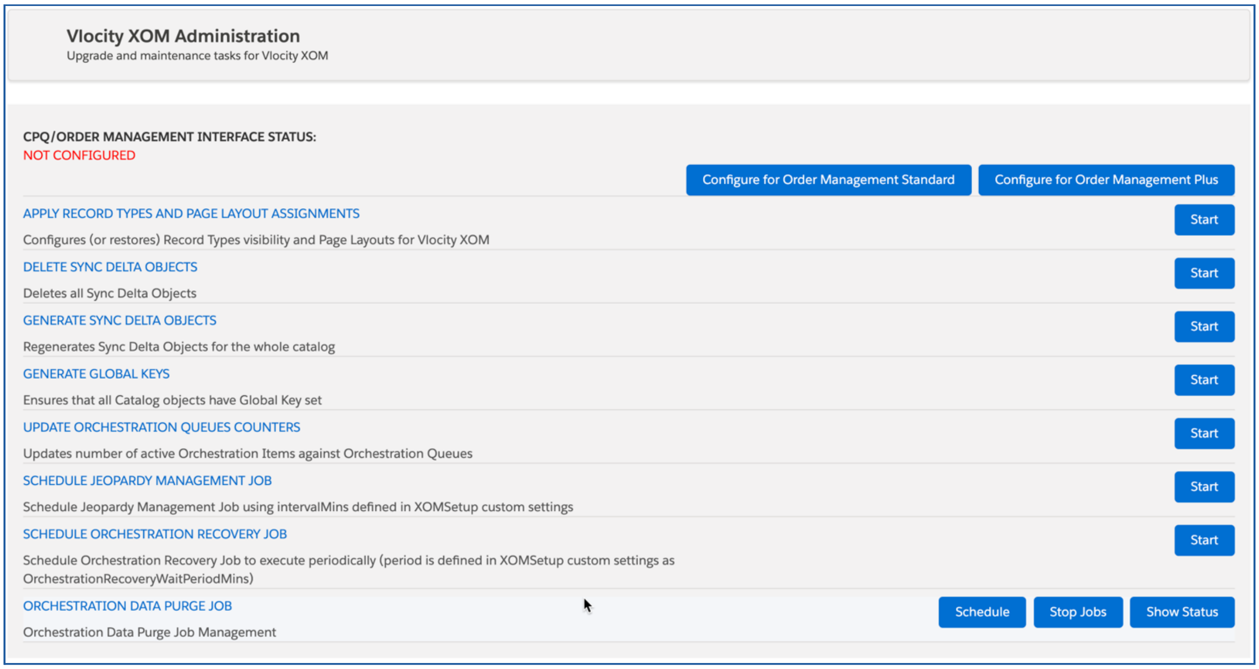
Task 2: Review the Vlocity XOM Administration tool

What is Vlocity XOM Administration?

Industries Order Management has been designed with an open and shared architecture to allow the entire user ecosystem to collaborate from offer to product to service and resource modeling through to fulfillment configuration.

The Vlocity XOM Administration tool allows administrators to specify access to Industries Order Management record types and layouts through standard Salesforce profiles. This allows product managers, fulfillment designers, technology specialists, quality assurance specialists, and other interested users the appropriate access to Order Management's powerful capabilities.

1. From the Order Management application, in the **Lightning tab navigation bar**, click **Vlocity XOM Administration**. (If you don't see it, click **More** to expand the overflow dropdown menu.)



Vlocity XOM Administration
Upgrade and maintenance tasks for Vlocity XOM

CPQ/ORDER MANAGEMENT INTERFACE STATUS:
NOT CONFIGURED

APPLY RECORD TYPES AND PAGE LAYOUT ASSIGNMENTS
Configures (or restores) Record Types visibility and Page Layouts for Vlocity XOM

DELETE SYNC DELTA OBJECTS
Deletes all Sync Delta Objects

GENERATE SYNC DELTA OBJECTS
Regenerates Sync Delta Objects for the whole catalog

GENERATE GLOBAL KEYS
Ensures that all Catalog objects have Global Key set

UPDATE ORCHESTRATION QUEUES COUNTERS
Updates number of active Orchestration Items against Orchestration Queues

SCHEDULE JEOPARDY MANAGEMENT JOB
Schedule Jeopardy Management Job using intervalMins defined in XOMSetup custom settings

SCHEDULE ORCHESTRATION RECOVERY JOB
Schedule Orchestration Recovery Job to execute periodically (period is defined in XOMSetup custom settings as OrchestrationRecoveryWaitPeriodMins)

ORCHESTRATION DATA PURGE JOB
Orchestration Data Purge Job Management

Configure for Order Management Standard
Configure for Order Management Plus
Start
Start
Start
Start
Start
Start
Start
Schedule
Stop Jobs
Show Status

2. Spend a few minutes reading the name and high level description of each job on the administration panel.

⚠ **ALERT:**
Do not click Start on any jobs! Your training playground has already been configured for you. Running several administrative jobs could hinder your ability to complete labs.

Review

Confirm your understanding by answering these questions.

1. When are orchestration definitions created?
2. What is an "instance of" an orchestration definition?
3. What is the purpose of Orchestration Queues?
4. What does the Vlocity XOM Administration tool do? Where can you find more information about it?

Exercise 4-2: Creating Technical Products in the Enterprise Product Catalog

Scenario

This may be love at first install. Greg has a really good feeling about Order Management! He wants to get started using the application to design some of Infiwave's order processes, but before he can do that, he needs to create some technical products in the system using the Shared Catalog. He decides to model a new product offering called the "Back to School Student Internet Offer," and then builds the commercial and technical products required to create the offer.

Goal

- Navigate Vlocity Product Console
- Understand product models
- Create *technical products*, or the services and resources that are used to fulfill orders
- Define *decomposition attributes* to be used in order processing
- Use *object types* to allow attribute inheritance

Tasks

1. Review a decomposition attribute for a technical product
2. Create an object type for a technical product
3. Create technical products

Time: 25 mins

What are technical products?

While you can find many nuanced and complex definitions of technical products throughout the industry, at Salesforce Industries we have decided on a simple way to think about it: if you consider that a commercial product is any product that is visible to the customer either on the shelf or in their cart, then a technical product is any product entity on the other side of the curtain that is not visible to the customer.

Technical products are used to fulfill orders and often are defined by the requirements of the downstream system, such as billing, scheduling, logistics, and other systems required to complete the order fulfillment process.

However, it is important to note that technical products are *not* required for Industries Order Management.

Commercial products that are defined sufficiently can be used interchangeably to submit fulfillment information to downstream systems. In that case, you are collapsing the commercial and technical product model for simplicity, which from a long-term maintenance perspective is a good thing to do.

Technical products should be created in the following situations:

- The commercial product entity cannot fully describe the product entity in terms that can be understood by the downstream systems.
- The details of the product entity should be masked from the customer.
- The product entity maps to a multitude of underlying products, services, or resources.
- To consolidate the number of callouts to a downstream system.

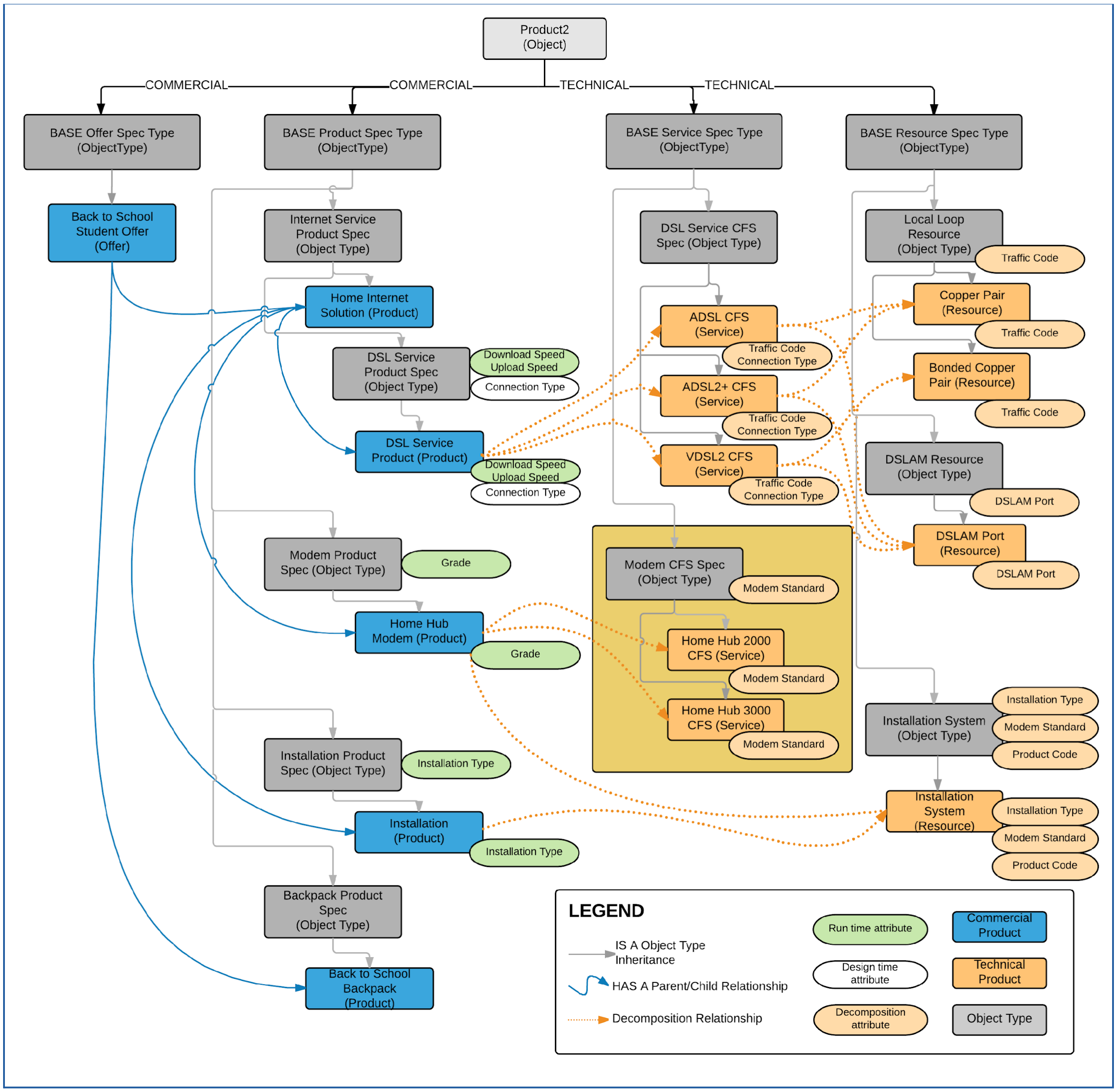
Order Management gives you the tools to translate commercial products to their corresponding technical products. It does this using Decomposition Relationships that map commercial products' fields and attributes to decomposition attributes.

ALERT:
Technical product specifications should be defined using product attributes *only*. Industries Order Management functionality relies on product attributes to communicate product and fulfillment information with external systems.

Designing product models requires careful consideration. But today is your lucky day--for the purposes of this training, we have designed an entire product model for you that describes Infiwave's new "Back to School Student Internet Offer." You can review this product model on the diagram on the next page.

This product model includes commercial and technical product entities and has been prebuilt into your training playground *with the exception of the entities in the gold box*. In this lab exercise, you will create these entities, and later in this training module, you will also create the Decomposition Relationships that are represented by the orange dotted lines in the diagram.

“Back to School Student Internet Offer” Product Model


Product2 (Object)
COMMERCIAL
COMMERCIAL
TECHNICAL
TECHNICAL
BASE Offer Spec Type (ObjectType)
BASE Product Spec Type (ObjectType)
BASE Service Spec Type (ObjectType)
BASE Resource Spec Type (ObjectType)
Back to School Student Offer (Offer)
Internet Service Product Spec (Object Type)
Home Internet Solution (Product)
DSL Service Product Spec (Object Type)
Download Speed Upload Speed
Connection Type
DSL Service Product (Product)
Download Speed Upload Speed
Connection Type
Modem Product Spec (Object Type)
Grade
Home Hub Modem (Product)
Grade
Installation Product Spec (Object Type)
Installation Type
Installation (Product)
Installation Type
Backpack Product Spec (Object Type)
Back to School Backpack (Product)
DSL Service CFS Spec (Object Type)
ADSL CFS (Service)
Traffic Code Connection Type
ADSL2+ CFS (Service)
Traffic Code Connection Type
VDSL2 CFS (Service)
Traffic Code Connection Type
Modem CFS Spec (Object Type)
Modem Standard
Home Hub 2000 CFS (Service)
Modem Standard
Home Hub 3000 CFS (Service)
Modem Standard
Local Loop Resource (Object Type)
Traffic Code
Copper Pair (Resource)
Traffic Code
Bonded Copper Pair (Resource)
Traffic Code
DSLAM Resource (Object Type)
DSLAM Port
DSLAM Port (Resource)
DSLAM Port
Installation System (Object Type)
Installation Type
Modem Standard
Product Code
Installation System (Resource)
Installation Type
Modem Standard
Product Code
LEGEND
IS A Object Type Inheritance
HAS A Parent/Child Relationship
Decomposition Relationship
Run time attribute
Design time attribute
Decomposition attribute
Commercial Product
Technical Product
Object Type

Task 1: Review a decomposition attribute for a technical product

What is a decomposition attribute?

Technically speaking, all product attributes are created and stored the same way. However, it may help to think about attributes differently based upon their purpose. For example, any attribute used by technical products during order decomposition is thought to be a *decomposition attribute*.

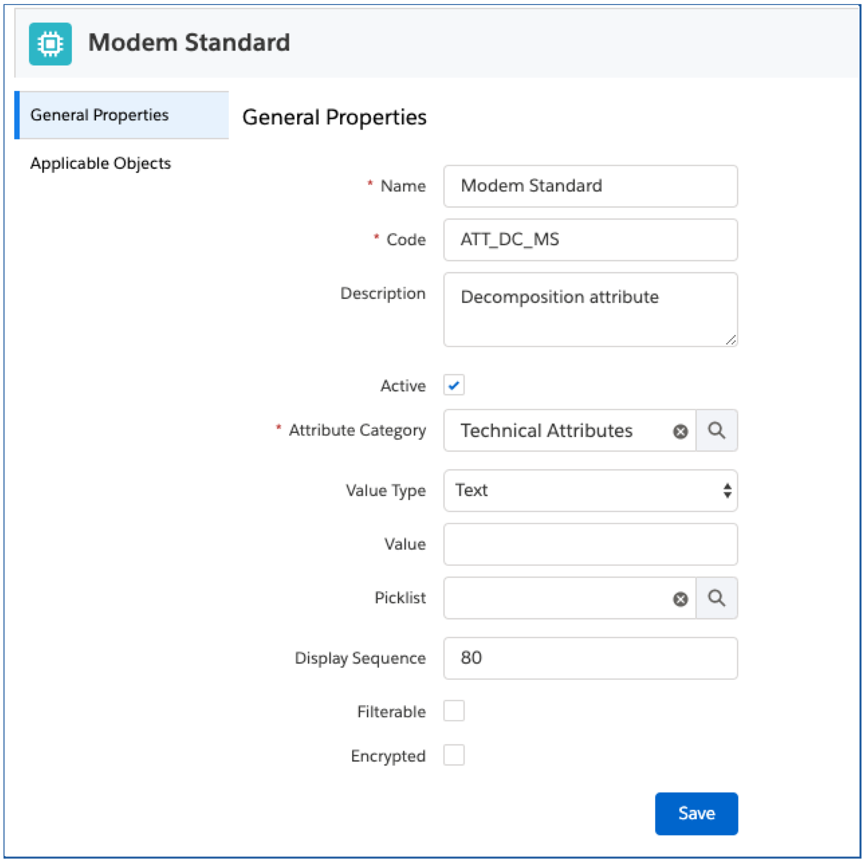
Conversely, an attribute that is defined by a product manager or architect at the time of product creation is called a design-time attribute. Lastly, an attribute that is specified by a customer during order capture is called a *run-time attribute*.

1. In the **Lightning tab navigation bar**, click **Vlocity Product Console**.

Vlocity Product Console provides a unified user experience to manage the Shared Catalog and Industries CPQ elements in a single, contextual interface. Using this interface, you will perform product management by defining and maintaining product entities. You can also build foundational constructs like attributes and picklists, as well as metadata constructs like product object types. You will also see many elements of Industries CPQ co-located in the Product Console, to provide seamless integration between the shared product catalog and the ability to configure, price, and quote the products in the catalog.

2. Under **Foundation** next to **Attribute**, click the search icon 🔍.
3. In the **Search** dialog, enter `Modem` and press **<ENTER>**.
4. Click **Modem Standard**.
5. Review the prebuilt **Modem Standard** attribute.

NOTE:
Salesforce Industries recommends but does not require adhering to a naming convention for attributes. For example, an attribute code that contains the prefix `ATT_` for ease of use in identifying it as an attribute across the Industries Shared Catalog. For those who *really* enjoy a good classification schema and pencils arranged in tidy rows, you can use the `ATT_DC` code to identify it as a decomposition attribute and then use `ATT_DT` (design-time) and `ATT_RT` (run-time) as appropriate for the other types of attributes.



Modem Standard

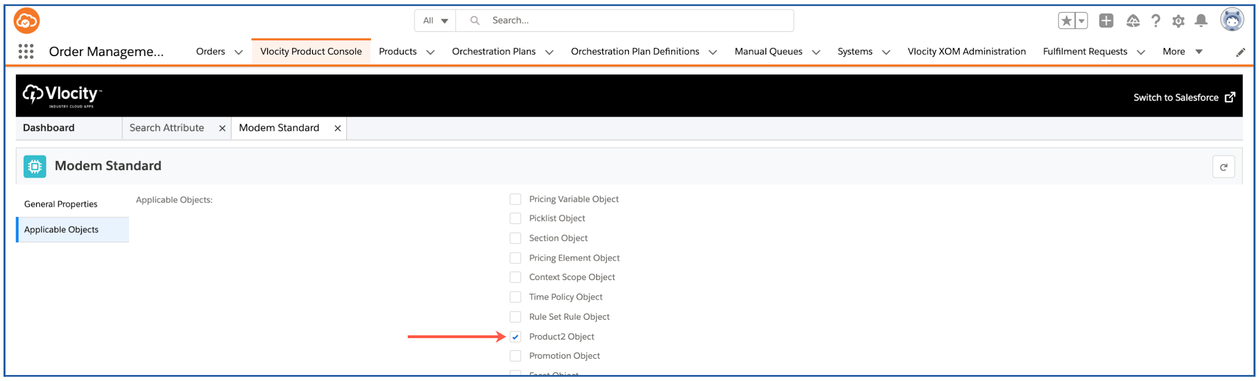
General Properties

* Name	Modem Standard
* Code	ATT_DC_MS
Description	Decomposition attribute
Active	☑
* Attribute Category	Technical Attributes
Value Type	Text
Value	
Picklist	
Display Sequence	80
Filterable	☐
Encrypted	☐

6. In the left sidebar, click the **Applicable Objects** facet.

7. Notice the checkbox next to **Product2 Object**.

ALERT:
If the Product2 Object is not set as the applicable object, the attribute will not be available to any object types or products.



Modem Standard

Applicable Objects:

- ☐ Pricing Variable Object
- ☐ Picklist Object
- ☐ Section Object
- ☐ Pricing Element Object
- ☐ Context Scope Object
- ☐ Time Policy Object
- ☐ Rule Set Rule Object
- ☑ Product2 Object
- ☐ Promotion Object

8. Click the **X** on the **Modem Standard** and **Search Attribute** tabs, which will return you to the **Product Console Dashboard**.

Task 2: Create an object type for a technical product

What is an object type in the Enterprise Product Catalog?

An object type in the Shared Catalog is a reusable entity that defines properties such as fields and attributes as well as the layout for all linked products. Object types can be used to group products with similar characteristics, to ensure consistent behavior and application of rules.

Object types are designed as an IS-A inheritance architecture. For those of you familiar with object-oriented programming and design, you'll recognize that object types include relationships between abstractions where one object type is a subtype of another object type. The first level of abstraction is called simply an object type, and the subsequent levels are called subtypes. Then, when this relationship is applied to a product, a product instance is created (also known as an asset).

Salesforce Industries recommends that all products be created with object types, and each type of product entity (offer, product spec, service, and resource) has a corresponding base object type at the top of the hierarchy.

In the steps below, you will create a new object type called Modem CFS Spec that will be a subtype of the BASE Service Spec Type object type, which houses CFS and RFS technical products in the training playground's product model.

1. In the **Product Console** Dashboard under **Metadata** next to **Object**, click the search icon 🔍.
2. In the **Search Object** dialog, enter `Product2` and press **<ENTER>**.
3. Click **Product2 Object**.
4. In the left sidebar, click the **Object Types** facet.
5. Review the list of object types that display. (Click **>** to expand collapsed sections.)
6. Notice the **Modem Product Spec** that is a subtype of the **BASE Product Spec Type**. This object type is for the commercial modem product entities, and your task is to

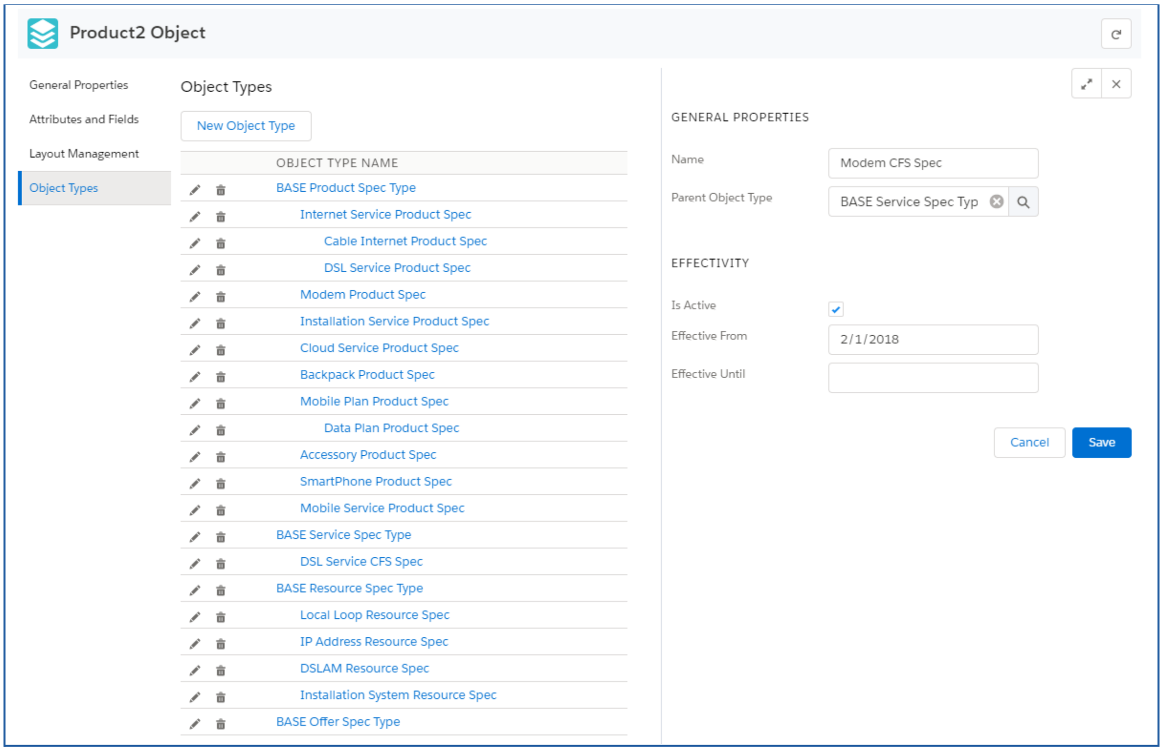
create the corresponding object type for the technical modem product entities. (Note that you may need to click the expand/collapse to see the correct fields.)

7. Click **New Object Type** (near the top of the page).

8. In the new object type dialog on the right, enter the following information. (For the **Parent Object Type**, use the search dialog to find and select the specified Object Type.)

Field / Property	Value
Name	`Modem CFS Spec`
Parent Object Type	**BASE Service Spec Type**
Is Active	✔
Effective From	**[today's date]**

9. Click **Save**.



Object Types

OBJECT TYPE NAME

- BASE Product Spec Type
  - Internet Service Product Spec
    - Cable Internet Product Spec
    - DSL Service Product Spec
  - Modem Product Spec
  - Installation Service Product Spec
  - Cloud Service Product Spec
  - Backpack Product Spec
  - Mobile Plan Product Spec
    - Data Plan Product Spec
  - Accessory Product Spec
  - SmartPhone Product Spec
  - Mobile Service Product Spec
- BASE Service Spec Type
  - DSL Service CFS Spec
- BASE Resource Spec Type
  - Local Loop Resource Spec
  - IP Address Resource Spec
  - DSLAM Resource Spec
  - Installation System Resource Spec
- BASE Offer Spec Type

GENERAL PROPERTIES

Name: Modem CFS Spec

Parent Object Type: BASE Service Spec Typ

EFFECTIVITY

Is Active: ☑

Effective From: 2/1/2018

Effective Until:

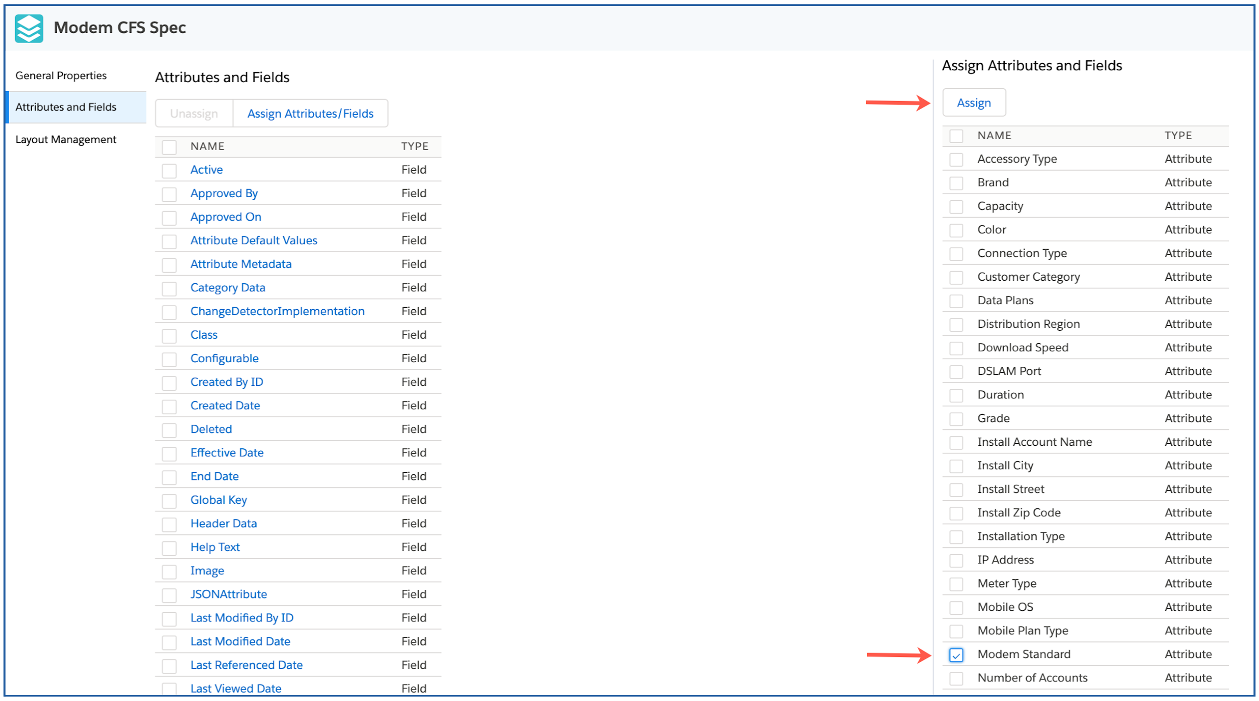
10. In the **Object Types** list, expand **BASE Service Spec Type** and click the new **Modem CFS Spec**.

11. In the left sidebar, click the **Attributes and Fields** facet.

    In the **Attributes and Fields** facet, notice the list of fields that appear. These fields are inherited from the BASE Service Spec Type object super type. If there were any attributes assigned to the BASE Service Spec Type, they would have been inherited as well. However, there were not, so you'll assign some attributes to the new Modem CFS Spec.

12. Click **Assign Attributes/Fields**. On the right, all of the attributes applicable to Product2 are displayed.

13. Click the checkbox next to **Modem Standard** and click **Assign** at the top of the **Assign Attributes and Fields** dialog.



Attributes and Fields

□	NAME	TYPE
□	Active	Field
□	Approved By	Field
□	Approved On	Field
□	Attribute Default Values	Field
□	Attribute Metadata	Field
□	Category Data	Field
□	ChangeDetectorImplementation	Field
□	Class	Field
□	Configurable	Field
□	Created By ID	Field
□	Created Date	Field
□	Deleted	Field
□	Effective Date	Field
□	End Date	Field
□	Global Key	Field
□	Header Data	Field
□	Help Text	Field
□	Image	Field
□	JSONAttribute	Field
□	Last Modified By ID	Field
□	Last Modified Date	Field
□	Last Referenced Date	Field
□	Last Viewed Date	Field

Assign Attributes and Fields

□	NAME	TYPE
□	Accessory Type	Attribute
□	Brand	Attribute
□	Capacity	Attribute
□	Color	Attribute
□	Connection Type	Attribute
□	Customer Category	Attribute
□	Data Plans	Attribute
□	Distribution Region	Attribute
□	Download Speed	Attribute
□	DSLAM Port	Attribute
□	Duration	Attribute
□	Grade	Attribute
□	Install Account Name	Attribute
□	Install City	Attribute
□	Install Street	Attribute
□	Install Zip Code	Attribute
□	Installation Type	Attribute
□	IP Address	Attribute
□	Meter Type	Attribute
□	Mobile OS	Attribute
□	Mobile Plan Type	Attribute
☑	Modem Standard	Attribute
□	Number of Accounts	Attribute

The **Modem Standard** attribute is added to the list of assigned attributes and fields.

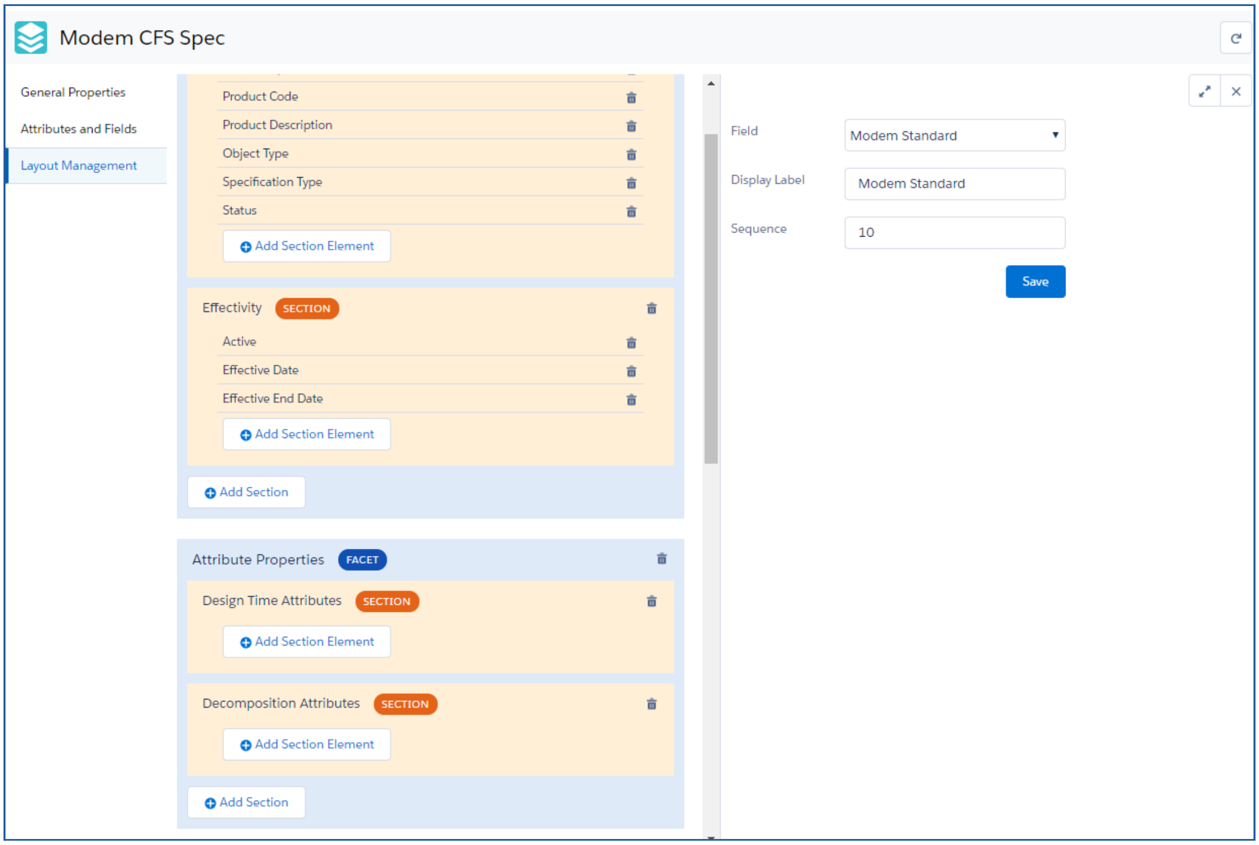
14. In the left sidebar, click the **Layout Management** facet. Review the layout that has been inherited from the BASE Service Spec Type.

15. In the layout, find the **Attribute Properties** facet, and click **Add Section Element** in the **Decomposition Attributes** section.

16. In the **Add Section Element** dialog on the right, enter the following information.

Field / Property	Value
Field	**Modem Standard**
Sequence	`10`

17. Click **Save**.

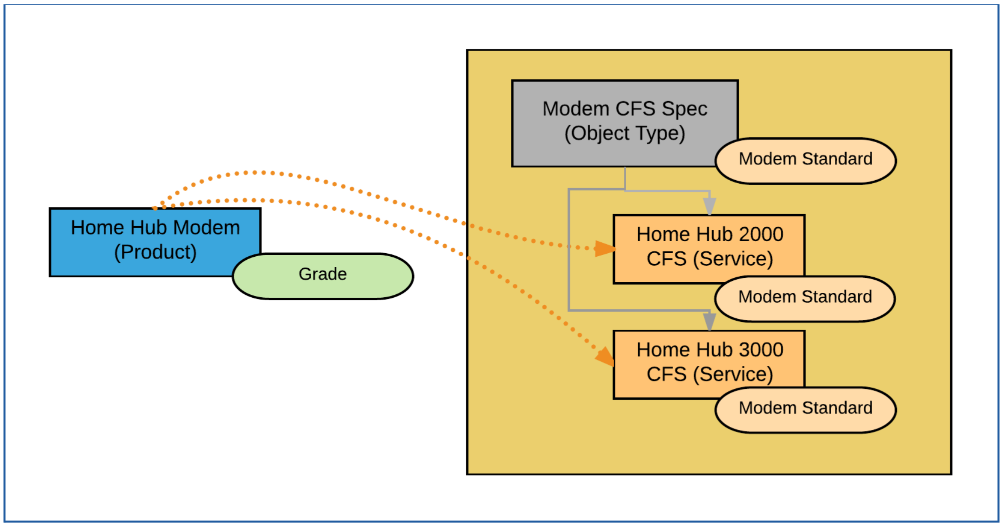


Field: Modem Standard
Display Label: Modem Standard
Sequence: 10

18. Click the **X** on the **Modem CFS Spec**, **Product2 Object** and **Search Object** tabs, which will dismiss those windows and return you to the Product Console Dashboard.

Task 3: Create technical products

For this task, you will create two new technical products, the **Home Hub 2000 CFS** and the **Home Hub 3000 CFS**. Both will be used in the following exercises to fulfill Home Hub Modem commercial product orders. Review the detailed diagram below.

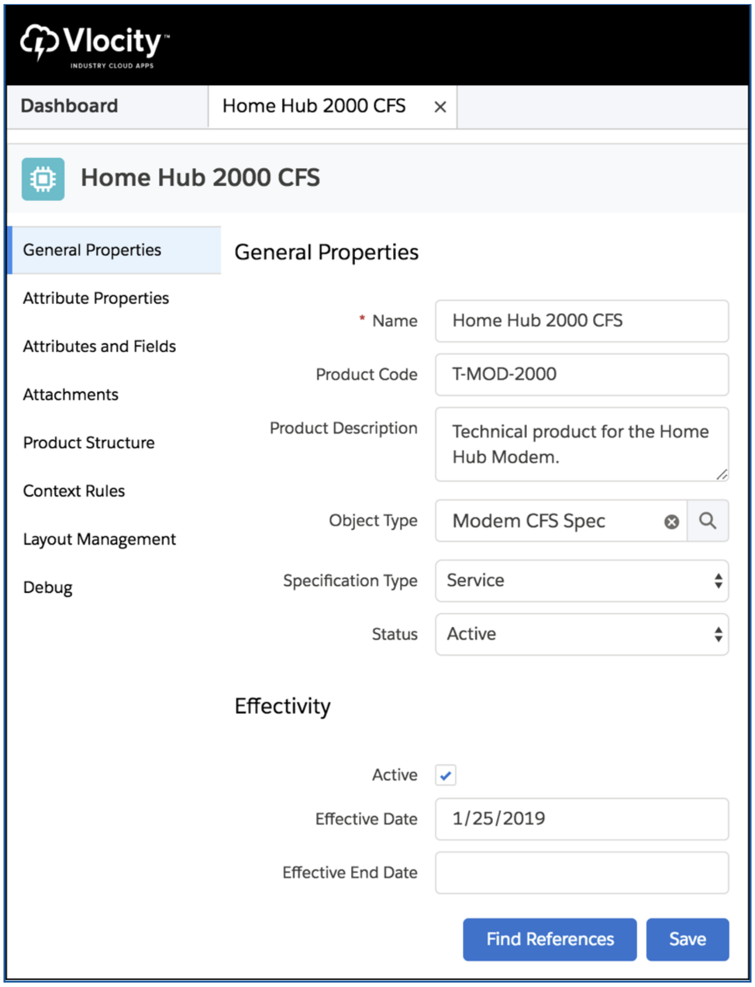


1. In the **Dashboard** under **Product Management**, click the **+** sign next to **Product** to launch the **New Product** dialog.

2. In the **New Product** dialog, enter the following information.

Field / Property	Value
Object Type	**Modem CFS Spec**
Name	`Home Hub 2000 CFS`
Product Code	`T-MOD-2000`
Description	`Technical product for the Home Hub Modem`
Specification Type	**Service**
Status	**Active**
Active	✔
Effective Date	**[today's date]**

3. Click **Save**.



Home Hub 2000 CFS

General Properties

* Name: Home Hub 2000 CFS

Product Code: T-MOD-2000

Product Description: Technical product for the Home Hub Modem.

Object Type: Modem CFS Spec

Specification Type: Service

Status: Active

Effectivity

Active: ☑

Effective Date: 1/25/2019

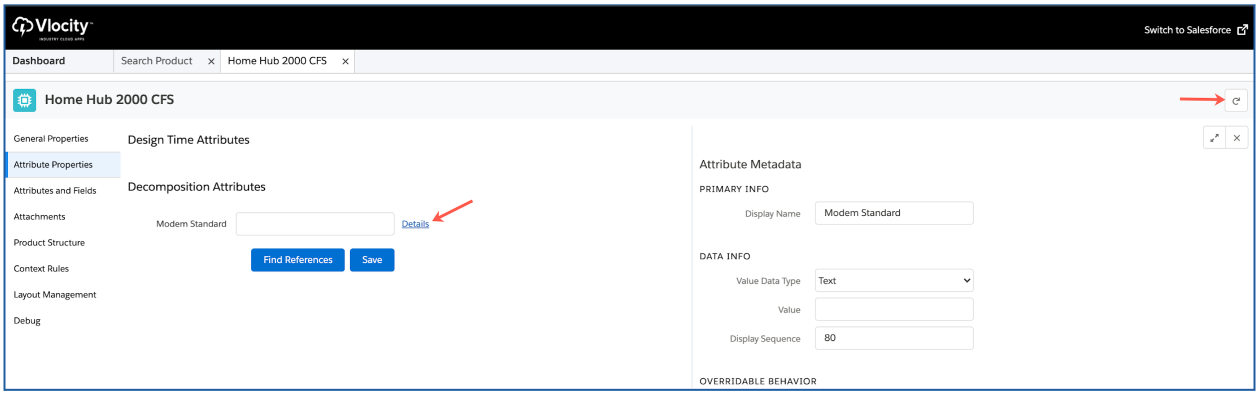
Effective End Date:

The new technical product is created, and you are left in the **General Properties** facet.

4. In the left sidebar, click the **Attribute Properties** facet.
5. Notice that the **Modem Standard** attribute that you placed in the Decomposition Attributes section of the Attribute Properties layout of the Modem CFS Service Spec object type displays here.
6. Under **Decomposition Attributes**, click the **Details** link next to the **Modem Standard** attribute.

7. Take a moment to look over the **Attribute Metadata**.

NOTE:
Prior to this step, the Modem Standard attribute was assigned and available to this product, but not yet instantiated and written to the Product2.JSONAttribute field. You will need this attribute instantiated before you can create mapping rules in our next task.



Home Hub 2000 CFS

Design Time Attributes

Decomposition Attributes

Modem Standard

Attribute Metadata

PRIMARY INFO

Display Name: Modem Standard

DATA INFO

Value Data Type: Text

Value:

Display Sequence: 80

OVERRIDABLE BEHAVIOR

8. In the upper right corner of the Product Console, click the **reload** button.

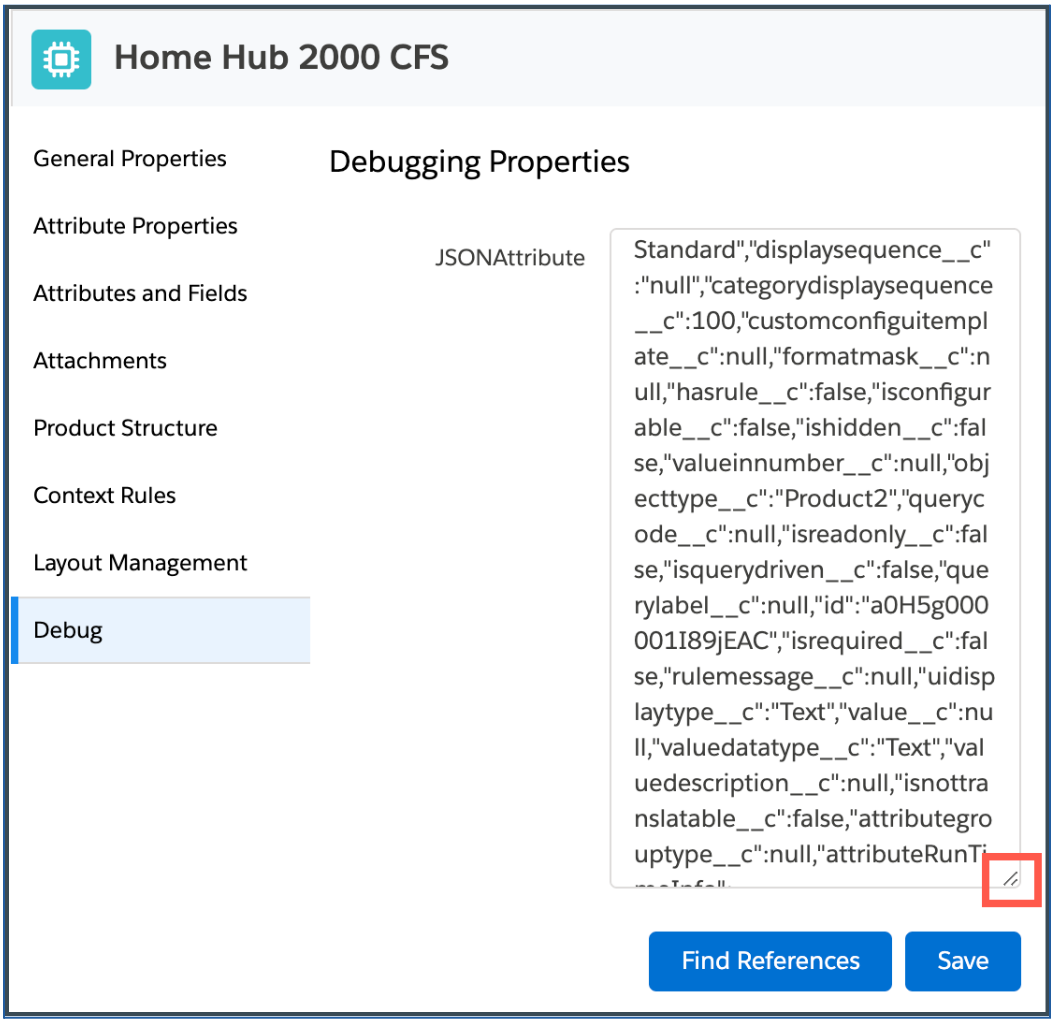
9. In the left sidebar, click the **Debug** facet.

NOTE:
Your training playground is configured to include the Debug facet. You can create custom facets in your own orgs, if desired.

ALERT:
Do not directly edit the **JSONAttribute** field, no matter how much you love writing in JSON notation! This data is intended for debugging purposes only.

10. Using the resizing tool on the **JSONAttribute** field, pull the field down so that more lines are visible.

11. Notice that the **JSONAttribute** field now displays the attribute metadata that we created earlier in this exercise.



JSONAttribute

Standard","displaysequence__c":"null","categorydisplaysequence__c":100,"customconfiguitemplate__c":null,"formatmask__c":null,"hasrule__c":false,"isconfigurable__c":false,"ishidden__c":false,"valueinnumber__c":null,"objecttype__c":"Product2","querycode__c":null,"isreadonly__c":false,"isquerydriven__c":false,"querylabel__c":null,"id":"a0H5g000001I89jEAC","isrequired__c":false,"rulemessage__c":null,"uidisplaytype__c":"Text","value__c":null,"valuedatatype__c":"Text","valuedescription__c":null,"isnottranslatable__c":false,"attributegrouptype__c":null,"attributeRunTi

TIPS:
Consider using a JSON validator website, such as www.jsonlint.com, if you need to view the JSON in a formatted, easier to read display. Simply copy and paste the **JSONAttribute** data shown here.

12. Click the **X** on the **Home Hub 2000 CFS** tab, which dismisses the window and returns you to the Product Console Dashboard.

13. In the **Dashboard** under **Product Management**, click the **+** sign next to **Product** to launch the **New Product** dialog.

14. In the **New Product** dialog, enter the following information.

Field / Property	**Value**
Object Type	**Modem CFS Spec**
Name	`Home Hub 3000 CFS`
Product Code	`T-MOD-3000`
Description	`Technical product for the Home Hub Modem.`
Specification Type	**Service**
Status	**Active**
Active	✔
Effective Date	**[today's date]**

15. Click **Save**.

16. In the left sidebar, click the **Attribute Properties** facet.

17. Confirm that the **Modem Standard** decomposition attribute is there, and the **Details** link reveals familiar metadata.

18. In the upper right, click **Switch to Salesforce**.

Since technical products are not orderable, nor do they have a price (as commercial products require), this is all that is necessary to create a technical product. In most implementations, you would create many more decomposition attributes to fully capture all aspects of this technical product, but for training purposes, this is sufficient.

Review

Confirm your understanding by answering these questions.

1. What is a technical product?
2. Can you use commercial products without a corresponding technical product in Order Management?
3. How are technical products used?
4. What is a decomposition attribute?
5. Should technical products have an object type? If so, why?
6. Are technical products orderable?

Exercise 4-3: Creating Technical Products Challenge

Scenario

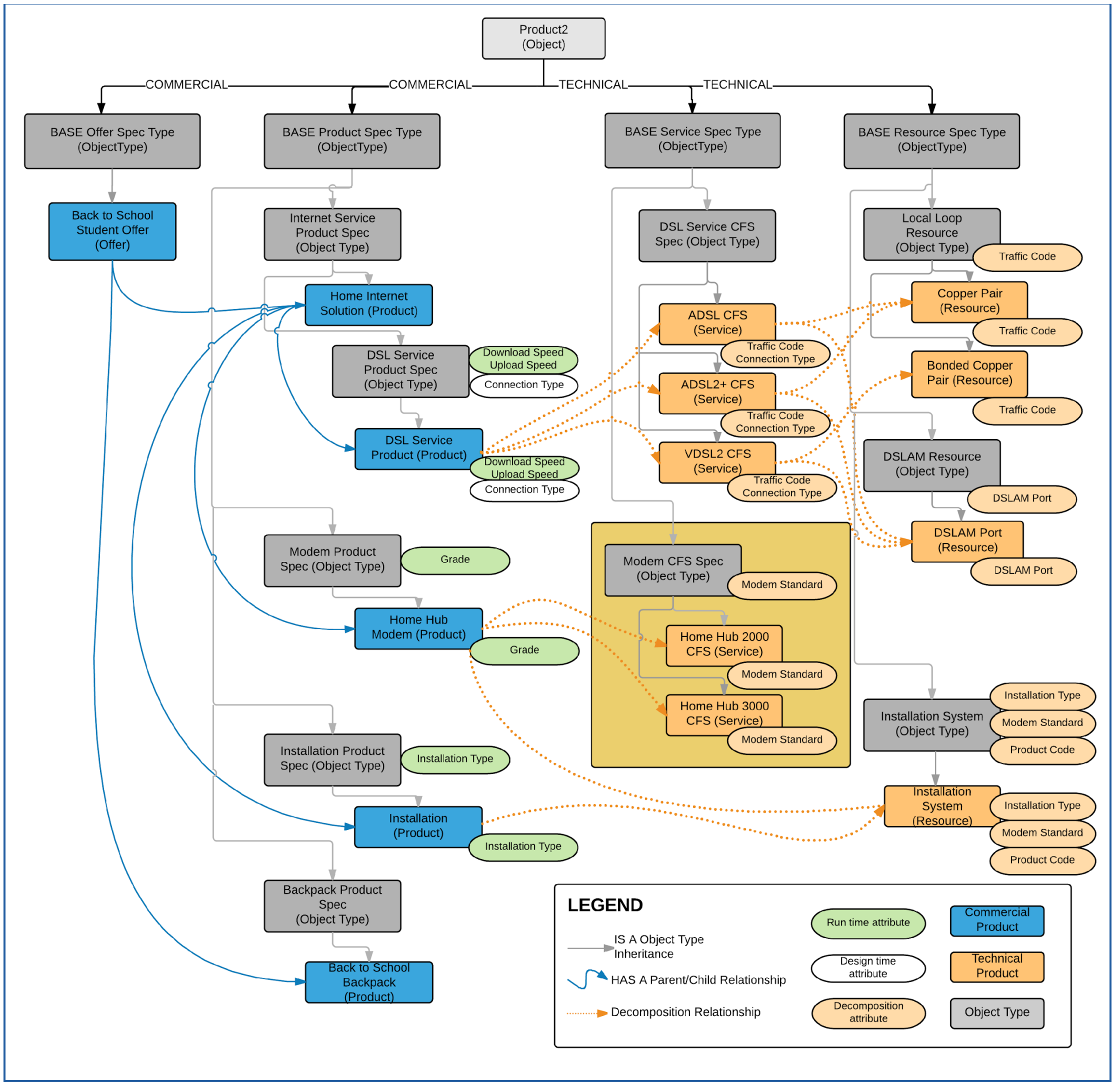
Newsflash! Technology changes… rapidly! In the old days of home internet broadband services, both a router and a modem were required. The router allows all your local computers and devices to join your network. However, you still needed a modem to access the internet via your ISP (Internet Service Provider). The trend is towards a single hybrid device, that combines the functionality of a router and modem. There is some confusion between customers, CSRs, and the current product data model. Marketing and Product Management at Infiwave think the best way to eliminate confusion is to market, sell, and model a new hybrid device typically known as an internet gateway (or simply "gateway").

Tasks

1. Review similar portions of the product model in the training playground
2. Use the Vlocity Product console to create technical products

Time: 20 mins

Task 1: Review similar portions of the product model in the training playground


Product2 (Object)
COMMERCIAL
COMMERCIAL
TECHNICAL
TECHNICAL
BASE Offer Spec Type (ObjectType)
BASE Product Spec Type (ObjectType)
BASE Service Spec Type (ObjectType)
BASE Resource Spec Type (ObjectType)
Back to School Student Offer (Offer)
Internet Service Product Spec (Object Type)
Home Internet Solution (Product)
DSL Service Product Spec (Object Type)
Download Speed Upload Speed
Connection Type
DSL Service Product (Product)
Download Speed Upload Speed
Connection Type
Modem Product Spec (Object Type)
Grade
Home Hub Modem (Product)
Grade
Installation Product Spec (Object Type)
Installation Type
Installation (Product)
Installation Type
Backpack Product Spec (Object Type)
Back to School Backpack (Product)
DSL Service CFS Spec (Object Type)
ADSL CFS (Service)
Traffic Code Connection Type
ADSL2+ CFS (Service)
Traffic Code Connection Type
VDSL2 CFS (Service)
Traffic Code Connection Type
Modem CFS Spec (Object Type)
Modem Standard
Home Hub 2000 CFS (Service)
Modem Standard
Home Hub 3000 CFS (Service)
Modem Standard
Local Loop Resource (Object Type)
Traffic Code
Copper Pair (Resource)
Traffic Code
Bonded Copper Pair (Resource)
Traffic Code
DSLAM Resource (Object Type)
DSLAM Port
DSLAM Port (Resource)
DSLAM Port
Installation System (Object Type)
Installation Type
Modem Standard
Product Code
Installation System (Resource)
Installation Type
Modem Standard
Product Code
LEGEND
IS A Object Type Inheritance
HAS A Parent/Child Relationship
Decomposition Relationship
Run time attribute
Design time attribute
Decomposition attribute
Commercial Product
Technical Product
Object Type

1. Look at the product model above, then use the Vlocity Product Console to look up the following:

   - Modem Standard attribute
   - Modem CFS Spec Object Type
   - Home Hub 2000 CFS technical product
   - Home Hub 3000 CFS technical product

Task 2: Use the Vlocity Product Console to create technical products

1. Create a new technical attribute. Name it `Gateway Standard` and model it after the **Modem Standard** attribute.

TIPS:
Make sure the **Product2 Object** is checked in the **Applicable Objects** facet.

2. Create a new object type for a technical product. Name it `Gateway CFS Spec`, and model it after the **Modem CFS Spec**.

TIPS:
Don't forget to assign the technical attribute previously created to the new Gateway CFS Spec.

In the **Layout Management** facet add the **Gateway Standard** technical attribute to the **Attribute Properties** facet (**Decomposition Attributes** section, *not* the **Design Time section**).

3. Create two internet gateway technical products, model it after the Home Hub Modems:
   a. Acme Internet Gateway Standard CFS (Similar to the Home Hub Modem 2000)
   b. Acme Internet Gateway Premier CFS (Similar to the Home Hub Modem 3000)

Yay! All done!