***Introduction to the Salesforce Industries Communications Cloud***

The Communications Cloud Solution

Salesforce's Communications Cloud is the industry’s most comprehensive, cloud-native BSS solution, giving you intelligent digital customer engagement and a next-generation user experience to walk customers and agents through complex ordering and fulfillment processes.

The Communications Cloud solves several industry challenges.

1. Siloed systems

**Solution:**  A Communications Cloud platform with a shared enterprise product catalog

1. Slow CSR on-boarding

**Solution:** Rapid on-boarding of CSRs with Order Capture and Guided Selling

1. Slow time to market

**Solution:** Fast time to market with the Enterprise Product Catalog (EPC)

1. Low customer satisfaction

**Solution:**Satisfy customers with a fast, personalized shopping experience

1. Disjointed customer experience

**Solution:**Give consumers a unified experience by positioning sales agents to be in-the-know about each customer

**Key Capabilities of the Communications Cloud**

**Intelligent Digital Customer Engagement**

The Communications Cloud works with Salesforce Einstein to provide intelligent digital customer engagement through a next-generation user experience to walk customers and agents through complex ordering processes using a conversational omnichannel user interface, whether on mobile, the web, or from the office desk.

The end-to-end, concept-to-care process begins with a product manager or marketing manager defining new products and offers for launch. With the Communications Cloud, you can enable operators to create quotes, capture and fulfill orders, and synchronize with third-party billing systems to process orders, start billing, and provide effortless care to customers through their channel and device of choice.

**Customer Lifecycle Management**

Supporting the full customer lifecycle, the Communications Cloud orchestrates quoting, order capture, billing, and service inquiry resolution across channels and devices. Users can view customer account relationships, active devices and services, billing, usage, and interaction histories for full awareness of the customer account and preferences.

**Shared Product Catalog-Driven Order Management and Fulfillment**

The Communications Cloud enables product-catalog-driven order capture and fulfillment, so that you can:

* Quickly design and launch new products and promotions.
* Change product prices dynamically, across sales channels, to respond to changes in the market.
* Create and manage customer contracts throughout the full contract lifecycle.
* Capture accurate quotes and orders for B2B and B2C accounts.
* Orchestrate order fulfillment and ensure timely service delivery using Industries Order Management.

**Industry-Specific Data Model**

The Communications Cloud industry-specific data model is compliant with TM-Forum standards and enables flexible integration with BSS/OSS. By extending the market-leading marketing, sales, and service applications of Salesforce with a communications industry specific data model, the Communications Cloud gives you powerful, digital-process-automation technology called OmniScript. You also benefit from a library of pre-built business processes and productized integrations, all of which are built in close conformance with TM Forum Frameworx standards.

**The Digital Transformation**

What is the Communications Cloud?

A comprehensive suite of omnichannel CRM, retail, enterprise product catalog, configure-price-quote (CPQ), contract-management, and order-management applications.

What does it do?

Embeds processes built specifically for communications and media companies with guided-selling, service, and analytics functionality.

What else?

Enhances the B2B and B2C customer experience, enables revenue growth, and reduces cost to serve.

There's more?

Accelerates time to value, maximizes business agility, improves net promoter scores, and reduces the overall total cost of ownership for CRM deployments.

And with Salesforce...

100% native and 100% additive to Salesforce.

**Types of Transformation**

Engagement Layer Transformation

Many customers begin their transformation programs by deploying the Communications Cloud as an agile customer engagement layer to improve lead and opportunity management.

Line of Business Transformation

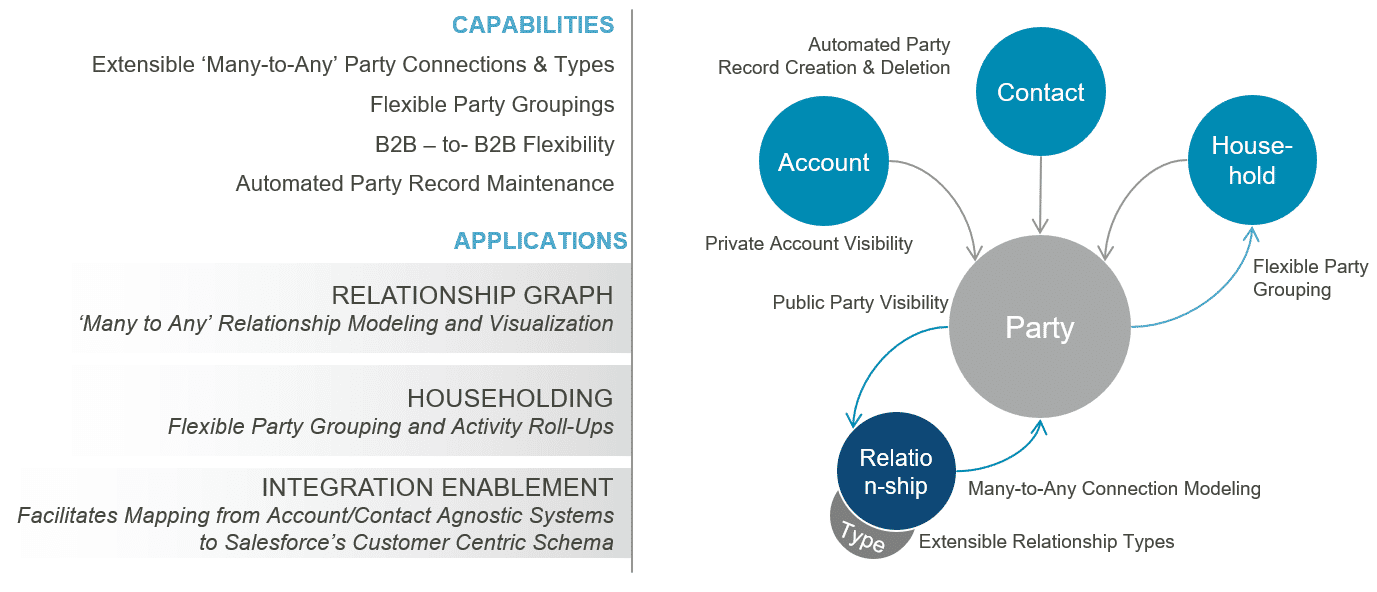
The most common deployment mode of the Communications Cloud, this method often complements existing Salesforce B2B SFA (lead and opportunity management) with guided selling and service, CPQ, contract management, and order management.

The deployment typically starts by creating a 360-degree customer view, then strips sales, ordering, and service processes away from legacy applications. During this time, multiple legacy CRM, order fulfillment, sales and service portals are retired.

Digital Operator Transformation

Many operators, particularly mobile operators, are now launching new, mobile-first digital brands, supported by a new end-to-end, native cloud digital stack based on Salesforce, and Matrix for online-charging and digital commerce.

**Salesforce Industries Party Model**



**Introduction to Industries CPQ**

**Introducing Industries CPQ**

Industries Configure, Price, Quote (CPQ) is a cloud-based, quote-and-order-capture system that is built natively and additively on the Salesforce platform.  It enables the configuration of product offers that are just right for the customer. Driven by underlying EPC product-service-resource (PSR) data, Industries CPQ gives sales teams advanced order-capture and guided-selling capabilities so that they can offer best-fit products and services to customers across different channels.

**How Industries CPQ Solves CPQ Challenges**

**Challenge:**  Produce accurate quotes

**Solution:**  Ensure availability and serviceability of products in addition to customer eligibility.

**Challenge:**  Accelerate order capture

**Solution:** Maximize customer orders with guided quote configuration and order capture.

**Challenge:** Reduce order fallout

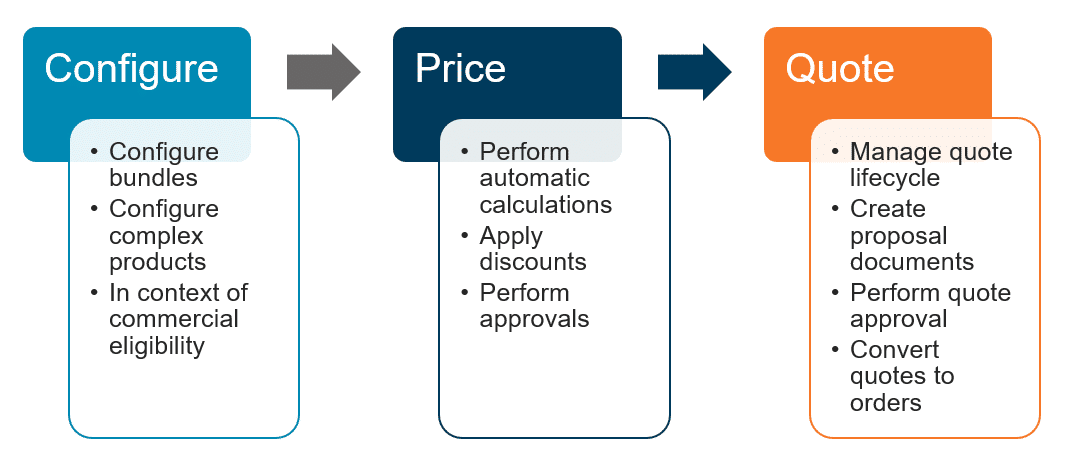
**Solution:** Enforce business rules to validate the configuration specified during order capture, minimizing the possibility of invalid orders.

**Challenge:** Improve sales processes

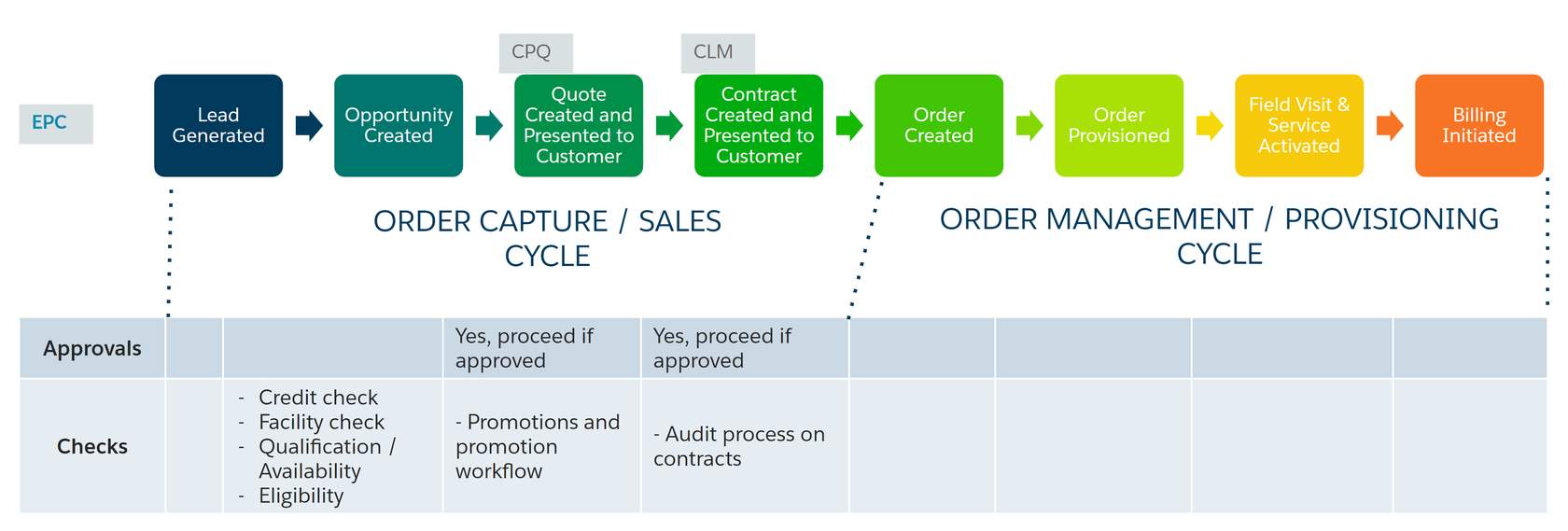
**Solution:**Provide flexible, configurable sales guidance based on your sales and approval processes and industry best practice

**Challenge:** Expedite time-to-market

**Solution:**Enable quick update of products and promotions in response to market changes.

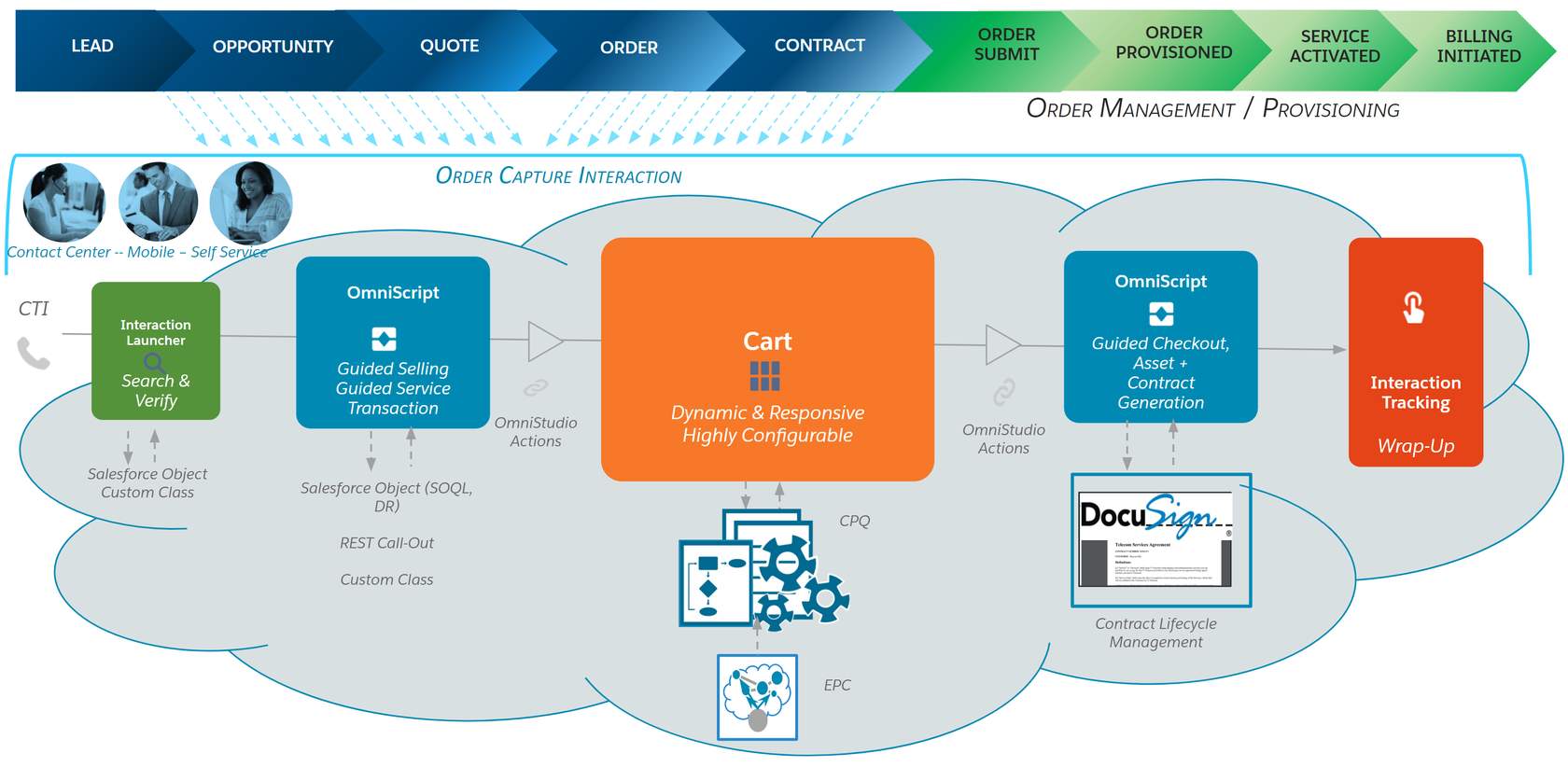


**Lead-to-Cash: Typical Process Flow**



**Enhanced Configure Price Quote**

Industries CPQ works in partnership with several components in the Digital Interaction Platform to provide enhanced order capture, including EPC and the Cart.



OmniScript provides guided selling and guided service transitions as well as guided checkout, asset generation, and contract generation, which you learn about in the Contract Lifecycle Management (CLM) course.

**Order Capture**

With our order-capture capabilities, you can ensure accurate orders on the way to submitting the perfect order for the customer. Industries CPQ applies business rules to ensure that products and services presented to the customer for purchase are:

* Available to the consumer or business account.
* Items for which the customer is eligible.
* Priced accurately.
* Compatible with any existing products and services held by the customer account.

**The Cart**

The Cart is the shopping-cart UI provided with Industries CPQ. Developed using the Vlocity Card Framework, you can modify the appearance and behavior of the cart to customize it according to your specific business requirements. The Vlocity Cards framework provides configurable cards, layouts, and templates, which are UI building blocks included out-of-the-box with the Industries CPQ managed package.

**Pricing**

Industries CPQ gives you a component-based pricing system with reusable items, such as charges, discounts, adjustments, and penalties. Pricing components are independent of products and allow seamless transitioning to new pricing structures to minimize expense and disruption.

Daily, your company faces many pricing challenges that include:

* Grouping products together, yet also keeping them separate.
* Discounting individual products in a bundle without changing the base price of individual products when purchased separately.
* Limiting product discounts to a specific time frame.
* Pricing products differently for different customer groups.
* Assigning costs as well as charges to products.
* Applying penalties when subscription commitments are broken.
* Providing customers with the option of paying for some products with loyalty points.

**Promotions**

You can use the Product Console to create and maintain promotions and discounts. Promotions is a separate module that provides greater flexibility and automated functionality for product promotions and discounts. With Promotions:

* Each bundled product you have created can remain intact. Any promotion you create can apply to the product without altering it.
* You can apply multiple promotions to one bundle.
* You can apply discounted prices in a promotion to existing assets.
* You can create contractual and transactional discounts.
* Any cardinality changes made to child products within a promotion do not require a refresh of the product hierarchy

**Rules**

**Rules ensure that:**

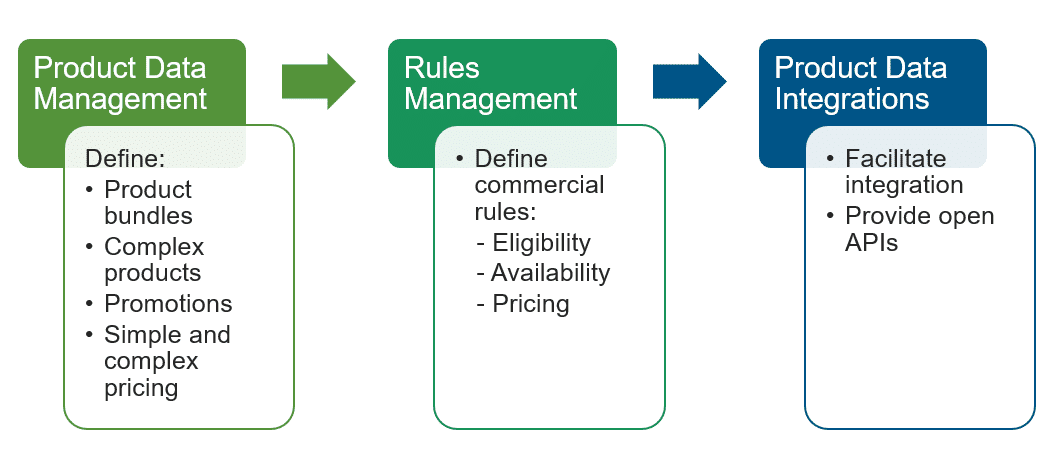
* Every order is a perfect order.
* The right products and promotions are available to customers at the right price.
* The appropriate penalties for product or service changes and cancellations are applied.

**Introducing EPC**

EPC gives your business an industry-leading, catalog-driven platform that revolutionizes the CPQ and order-management space by:

* Using a metadata-driven approach to accommodate all the incarnations of a product as it travels from a twinkle in the eyes of the customer to the physical implementation in the customer’s hands.
* Integrating with your end-to-end ecosystem using a complete product-service-resource (PSR) model to drive sales, configure-price-quote (CPQ), order-management, and order-fulfillment capabilities.
* Using an evolutionary approach to transform both front-office and back-office systems into a master-catalog-driven architecture.

**EPC Core Capabilities**



**EPC Reuse and Shared Resources**

Through an enterprise scope, enables the definition of a single product catalog in one place, with multiple subscribers and systems that consume the product catalog data.

Enables the creation of products once, making them available for reuse across the enterprise by different systems.

Gives you a single UI with common tools, enabling teams to create, manage, and deliver product offers to the market collaboratively and rapidly.

**Product-Service-Resource Layers**

What's in the product layer?

Commercial view of what the business sells

What's in the service layer?

Can include service specs and service configurations, such as a broadband internet service

What's in the resource layer?

May include equipment, such as cable attributes and access ports, typically to support the network

**ORDER CAPTURE**

With Industries CPQ's **order-capture**capabilities, you can ensure **accurate orders** on the way to submitting the **perfect order** for the customer.  Industries CPQ enables you to design and apply business rules to ensure that products and services presented for customers to buy are:

* Available to the consumer or business account.
* Items for which the customer is eligible.
* Priced accurately.
* Compatible with any existing products and services held by the customer account.

**How Industries CPQ Solves the Challenges**

The order-capture capabilities of Industries CPQ let you to meet the business challenges by **controlling the selling process** using the Cart and an **asset-based** ordering system. Customer service representatives (CSRs) and sales teams can perform a variety of business tasks through **guided selling** to capture the **perfect customer order**. The product-service-resource (PSR) data from the Enterprise Product Catalog (EPC), the master product catalog, underpins the Industries CPQ solution with a shared, central single point of truth for all PSR data.

**Challenge:**  Order fallout

**Solution:**  Offer products across channels based on highly configurable, accurate, and consistent quotes.

**Challenge:**  Slow sales

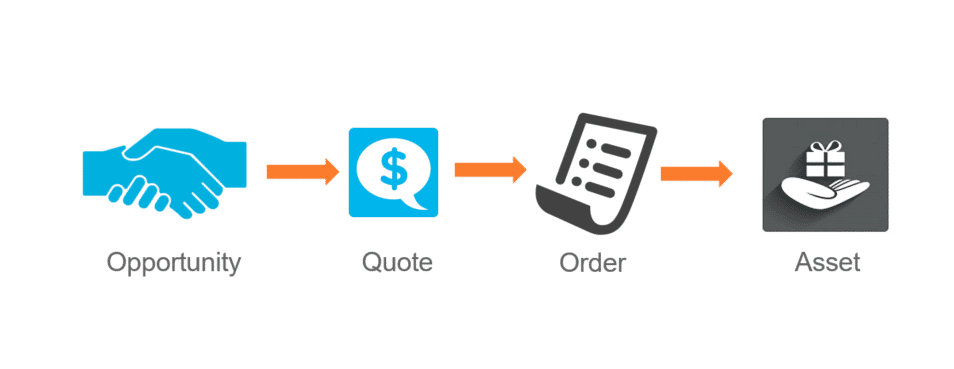
**Solution:** Equip and guide sales teams to sell the right products quickly to customers.

**Challenge:** Unsatisfied customers

**Solution:** Enable the easy and fast creation of accurate quotations based on best-fit products for customers.

**Challenge:** Product complexity

**Solution:**Ensure valid product offers based on business rules, such as availability, eligibility, and customer context.



Industries CPQ extends the native capabilities of Salesforce to provide comprehensive management of the order-capture flow. Opportunity, Quote, Order, and Asset are standard Salesforce objects. The two types of process flows include:

* **Business to business (B2B)**, which starts in a business account.
* **Business to consumer (B2C)**, which starts in a consumer account.

Use your Salesforce Industries Cloud applications to define and manage process flows to determine what you can offer to customers.

# What's an Asset?

* An asset is an item of value that an account or contact owns. You may know assets as products or services in a customer portfolio.
* In Salesforce, assets are products or subscribed services captured during the order-capture process.
* Asset-based ordering takes into account the existing products and services of a customer.
* Salesforce Industries extensions to the asset object support discounts, special pricing, and customer preferences.
* Whether an asset is created right away or later depends on the defined order-management process flow.

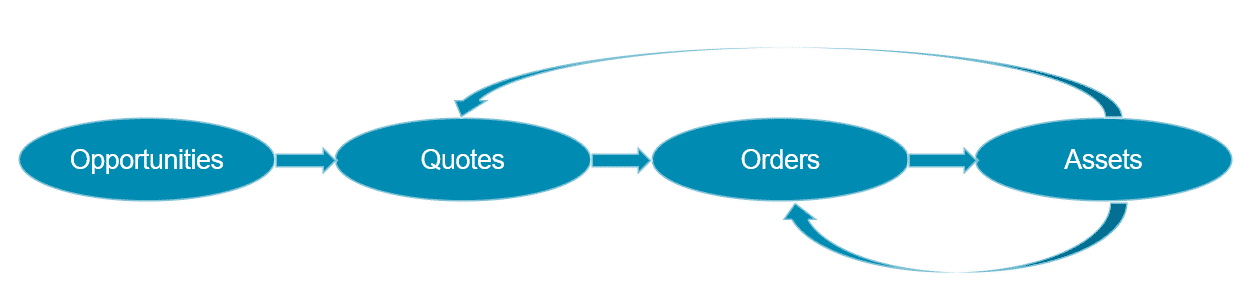
**Asset-Based Ordering**

Using asset-based ordering, Industries CPQ lets you manage products and services throughout the concept-to-care process. Standard ordering systems end the customer relationship once the order is complete. However, not all orders are about buying new products or services. For example, a customer may want to modify their assets by:

* Adding a service, for example, to enable international roaming from a smartphone.
* Adding a product, for example, to add another smartphone to a family phone plan.
* Updating a service, for example, to change the internet speed for their residential broadband service.
* Disconnect a service, for example, to cancel a cable service.

**Introducing Asset-Based Ordering**

Many customer orders impact products and services that the customer has already purchased. Salesforce Industries applications extend general order-processing capabilities to manage the products and services of customers throughout the order-capture lifecycle.



Customers may want to change their assets in a number of ways. Specifically, they may want to:

**Add a service**, for example, to enable international roaming with a wireless plan.

**Update an existing service**, for instance, to increase internet speed from 25Mbps to 50Mbps.

**Disconnect a service**, such as to cancel a cable subscription.

**Customer Lifecycle and Reverting Assets**

Customer lifecycle

* In the first stage of order capture, an opportunity moves to a quote and then to an order.
* In the next stage, with completion of the order, the order becomes an asset.
* At some point, you may need to revert the asset back to order status, namely if the customer wants to change the asset.

Reverting Assets

The option to revert assets is an out-of-the-box capability. You may need to revert an asset six months, one year, or two years after the customer bought the initial product or subscribed to the original service, for instance, if they want to change a service. This may include a location change if the customer moves residences.

In this case, you need to update the existing asset with the new details, as opposed to creating a new asset. Otherwise, the customer will have two of the same assets, which can cause data ambiguity.

Updating the existing asset enables retention of service continuity for the customer and avoids faults that may occur through unnecessary provisioning of additional assets.

Example

**Basic Example:**

1. A customer with a residential account wants to change a broadband internet service from 25 to 50 MB.
2. The customer calls their service provider, who moves the broadband internet service asset back to an order and changes the configuration.

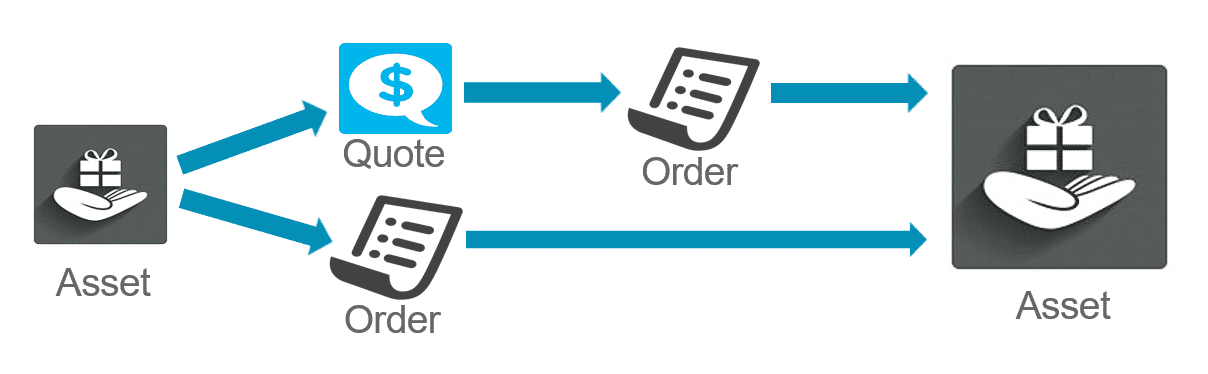
**Advanced Example:**

1. A customer with a commercial account wants to change a B2B service from Gold to Platinum level.
2. The customer calls the service provider to request the change.
3. However, the service provider finds that the service upgrade requires approval by someone else in the company.
4. In this case, the service provider reverts the asset to a quote in preparation for the change process.

**Asset-Based Ordering**

Industries CPQ supports the creation of quotes and orders against a set of customer assets. In all cases of asset-based ordering, you change an existing asset by moving it to either:

* **A quote**, and once the quote is complete, move the quote to an order, and then an asset, or
* **An order**, and once the order is complete, move the order to an asset.



**Order Line Item Status**

To help you work with assets, the Cart shows you the current lifecycle status of order line items.

* **Add**: Order has been placed, but the product or service is yet provisioned/activated
* **Existing**: An existing asset. No changes have been requested.
* **Change**: A change has been requested.
* **Disconnect**: A request has been made to end the product or service.
* **Suspend**: A request has been made to temporarily halt the service.
* **Resume**: A request has been made to reinstate the temporarily halted service.

**Introducing Order Cancellation**

Change happens. Sometimes that "change" is more than a MACD (move-add-change-delete) tweak, and actually requires canceling an order that has been captured, but not completely fulfilled.

Industries CPQ and Industries Order Management (OM) work together to support:

**Canceling part of an order**

**Canceling the entire order**

Some basic orders complete immediately. Other orders are more complex, and some period of time elapses from the time the order is captured until it is fulfilled. For example, a home internet package that requires shipping out a router from a warehouse and scheduling an on-site technician to configure it for the customer after it arrives.

So, completing an order could take seconds, minutes, days or even weeks!

**Q.)** What if a customer places an order, but needs to cancel it before it has completed?

**A.)** That requires Industries Order Management (OM).

**Introducing Industries Order Management (OM)**

Generically speaking, order management brokers the data required to fulfill the products and services on an order.

Industries Order Management does this through two processes:

* 1**Decomposition** - Maps commercial products to technical products
* **Orchestration** - Communicates technical information with downstream fulfillment systems

**Terminology**

A basic understanding of the following new terms is important when learning order cancellation and the communications between Industries CPQ and Industries OM.

* **In-flight order -**An order that has been submitted (from CPQ to OM) but not completed.
* **Supplemental order -**An order created by Industries CPQ to revise an in-flight order. Supplemental orders supersede the original in-flight order.
* **Point of no return (PONR)**- Point in the order orchestration process that once passed, order cancellation is no longer possible or permitted.

**Status**

There are several different types of statuses worth noting for those needing to understand the order cancellation process.

**Salesforce Status**

New orders use the Salesforce Status field. It has two picklist values by default: **Draft**and **Activated**

**Industries Order Status**

Industries CPQ and Order Management orders require additional functionality to implement functions such as order cancellation. Enter the Order Status field. The default view of the Orders tab provides an excellent snapshot of key order status fields. Note the status fields are shown for both the original in-flight order *and* the supplemental order used to implement order cancellation. Lastly, note that the supplemental order is associated with the original in-flight order via the entry in the Superseded Order column.

**Orchestration Status**

Orchestration Plans and the individual tasks that comprise them have states as well. The following is a simple executing Orchestration Plan made up of three individual tasks. (OC2 Start, OC2 Pause, OC2 Stop)

Industries Order Management makes it simple to see the states at a glance.

* The **State** of the plan itself is "In Progress". Another common state is "Completed".
* The **Status**of the individual tasks are color coded:
  + **Completed** (OC2 Start)
  + **Ready** (OC2 Pause)
  + **Pending** (OC2 Stop) - The task is ready to run but has not started yet. (It is waiting on a task dependency to complete before it can start running.)
  + *Note*: When you hover over a task, the status is displayed.
* There are two other common states for order cancellations that are not shown above:
  + **Cancelled** - Task that was canceled.
  + **Discarded** - Task that has not been executed when a supplemental order required that it be canceled. The task will not be executed.

**Working with Quotes using Industries CPQ**

A**Quote** represents the proposed prices of your company's products and services. In standard Salesforce, you create a quote from an opportunity and its products, and they can be viewed in the Quotes tab. Using Industries CPQ, you can also create quotes from an asset as a part of the asset-based ordering process. This is typically done as part of a B2B sales process, but it can also be done in B2C.

**Change to Quote Action Button**

In the asset management layouts, you can select one or more assets, and then click **Change to Quote**. This action button launches the CPQ Create Quote OmniScript, which creates a new quote for the selected assets and opens the quote in the Cart.  Once the quote is opened in the Cart, all of the applicable Industries CPQ rules and pricing will be enforced. The line items that were created from the assets will have a status of "Existing".

# Field Mapper

Field Mapper is a declarative mapping tool that provides simple, extensible, and comprehensive capabilities for transforming objects in the order-capture lifecycle.

In the context of commercial assetization, you can use the Field Mapper to specify the conversion of the following objects:

* Opportunities to quotes
* Quotes to orders
* Orders to assets
* Assets back to orders
* Assets back to quotes

You can use the Field Mapper to create a filter that keeps not-assetizable products from being assetized. Namely, the filter specifies assetization only when the Not Assetizable option is un-selected. Filters allow you to limit source fields based on specified conditions. Filters are inclusive, and you can define the conditions in a formula expression. If a formula is not indicated, the Field Mapper joins the filters using an AND statement.

Without the use of the Field Mapper, you would need to write transformation code to perform the conversions.

**About Commercial Assetization**

**Commercial assetization** is the act of setting a product as assetizable, which means that the product is tracked as an asset in the customer account. Typically, only products of a certain value are considered assets. For example, a company may not want to keep track of promotional items, such as a promotional hat or pen with the company logo.

* Within Salesforce Industries applications, you can designate commercial products as **not assetizable**.
* Commercial assetization stores instances of commercial products in Salesforce, creating a snapshot of the products or services the customer has purchased.
* Commercial assetization is different from **technical assetization**, which stores the output of decomposition in the technical inventory of Industries Order Management.

**Deciding Whether or Not to Assetize a Product**

**Why set a product as not assetizable?**In a nutshell, if you don't want to track the commercial value of a customer asset, you should configure the product as not assetizable.

As part of the managed package, Salesforce Industries includes a Not Assetizable field on products. However, the same field is absent from Order Products, which are order line item sObjects. Because of this, you need to create the Not Assetizable field manually on Order Products if required.

**Enterprise product Catalog**

# About Picklists

* Picklists are designed to be reusable across attributes, products and offers
* Name, Code and Data Type properties are required
* The name of the picklist is used only at design-time and not at run-time (visible to customers)
* We recommends (but do not enforce):
  + Use PKL\_[code] naming convention
  + Setting the Active flag and Effective From date

# About Product Attributes

* Provides a powerful and flexible way to extend product entities
* Stored in aggregate “blob” form in the JSONAttribute field on the Product object
* Used extensively throughout Industries CPQ and Industries Order Management (OM) to:
  + Filter products at run-time
  + Configure product specifications at run-time
  + Change product pricing using attribute-based pricing rules
  + In Industries OM, to map commercial products to technical products

**When to Use An Attribute Versus a Field**

**Field**

* If the data element is common for all products across the entire product catalog, create a new field on the Product object.
* Typically, adding new fields requires Salesforce Administrator privileges.

**Attribute**

* If the data element is specific to a product or a class of product, create a product attribute.
* OOTB, attributes are used at run-time (visible to customers) in the Cart to filter the product catalog and to provide product configuration capabilities.
* Adding attributes simply requires access to the Product Console.

**Product Attributes - General Properties**

* The **Attribute Name** is required and visible at run-time so it's important to pick a customer friendly name.
* The **Attribute Code** is required but not visible at run-time. We recommend the following naming code convention: ATT\_[code].  This makes it easy to identify the EPC element type during design-time.
* The **Attribute Category** is required and is created through Attribute Categories in the Product Console.
* You can optionally designate a **Picklist**to provide the values for your attribute.
* **Filterable** attributes can be used within the Cart to filter product lists and order items.
* You can set your attribute to be **encrypted**, if it contains sensitive information.
* Product Attributes can be assigned directly to a product or to a product Object Type, which allows it to be inherited.

# What is a Product Object Type?

* Reusable entity that defines properties (fields and attributes) and layout for all product instances
* Used to group products with similar characteristics and ensure consistent behavior and application of rules
* Supports hierarchy inheritance
* At this time, we do not recommend moving object types within the hierarchy. Careful planning is required during initial setup.
* Salesforce Industries Expert Services can assist with product catalog design and modeling

**Object Types "Is A" Inheritance Architecture**

* Object Type is the 1st level of abstraction or “Supertype”
* 2nd+ levels of abstractions are “Subtypes”
* The application of the Object Type is an “instance-of” relationship, or Product Instance

**EPC - Product Object Types**

* We recommend creating a “Base Product” during initial setup. This will be your product catalog’s “primordial object” and should include all of the fields in the product catalog.
* Object types for specific product classes can then be created to inherit the Base Product’s layout, fields and any attributes.
* The Base Product’s layout must be created manually.
* An object subtype will inherit the layout via deep copy, but any subsequent changes to the object super type’s layout will not be replicated down the hierarchy after initial creation.
* Inherited attributes and fields cannot be deleted, and instead must be unassigned from the object super type.
* New attributes and fields assigned to an object type will be inherited dynamically by its subtypes.

# What is a Product Life Cycle?

A product life cycle is the process that every product goes through from its introduction to its retirement. There are four defined stages for a product lifecycle:  introduction, growth, maturity and decline.

The shared product catalog supports **commercial** product lifecycles by allowing you to define current, future, and retired products based on product selling period dates.

**Current Product**

A product that is currently available for sale.

**Future Product**

A product that will be sold in the future. This product can be configured and ordered but will not be fulfilled until its selling period and fulfilment begin.

**Past Product**

A product that is no longer for sale but still can be fulfilled for orders that have already been placed and changed for orders that have already been assetized.

**Retired Product**

A product that is no longer supported, cannot be ordered or fulfilled, or changed for orders that have been assetized

# Setting Product Selling Periods on Products

Product selling period dates are set in the General Properties facet of the product. The Product Console validates all date and times entered using the rule:

**SellingStartDate  <  =  FulfilmentStartDate  <  =  SellingEndDate  <  =  EndOfLifeDate**

The Product Console will not allow you to save products until the dates are set correctly. If you choose not to enter product selling period dates, it implies that the product is available for all of time.

**Caution**

Product selling period data is stored in the platform cache. After you make changes to product selling period fields, you must update the product hierarchy data in the platform cache in this sequence:

1. Product Hierarchy Maintenance
2. Clear Managed Platform Cache
3. Refresh Platform Cache

When a product is used in bundles, it is the responsibility of the product administrator to ensure the selling periods are consistent.

**Find References**

Product administrators use the **Find References** button to find references to a given product in active promotions, product bundles, assets, quote line items, order line items and opportunity line items. The button assists administrators in determining the effect of changing product selling period and end of life dates.

The default number of rows returned for each reference object is 20, but the limit can be increased by setting the FindReferencesRowCountLimit custom setting in CPQ Configuration Settings Overview. The maximum number of rows returned is 1000.

**Cautions**

In order to find order, quote and opportunity line items references, each line item's fulfilment status (vlocity\_cmt\_\_FulfilmentStatus\_\_c) must be set to Draft, In Progress, PONR (point of no return), or Pending.

In order to find asset references, the asset's provisioning status (vlocity\_cmt\_\_ProvisioningStatus\_\_c) must be set to Active, Changed or Deleted. These values are normally set by order management processes. If the statuses have null or other values, the references will not display.

**Product Selling Periods in the Cart**

To support MACD and asset-based ordering, the Cart can display current, future, past and retired products based on the Order (or Quote or Opportunity) Start Date.

* Current and future products display in the cart normally, but **future** products are indicated with a **green** **clock** icon.
* **Past** products are products that have passed their Selling End Date but have not passed their End of Life Date. They are indicated by a **yellow** **clock**icon. They cannot be added to the cart or configured for new orders, but they can be configured or deleted during MACD or asset-based ordering.
* **Retired** products are products that have passed their End of Life Date and they are indicated by a **red clock** icon. In a new order, they cannot be added to the cart or configured but during MACD or asset-based ordering they can be deleted.

Similar to products added to the cart, the product list also displays current and future products based on the Order Start Date.

**Caution:**Product selling period functionality does not override other standard controls for displaying products in the Products list. In order to display, products must be marked Orderable, Active and have a price list entry. The price list entry Effective From date must be current, as well.

* Products that have reached the end of their selling period or are at their end of life date will not display in the product list and can’t be added to the cart.
* You can add a future-dated product to the cart and submit the order normally; however, Future Dated Ordering functionality must be implemented in order to manage the fulfilment of future products.

**EPC Products**

We recommend (but do not enforce) that all products be created with an object type.

New attributes and fields can be assigned directly to a product, if desired.

Attributes and fields inherited from an Object Type cannot be unassigned.

Commercial products must have a price, be Orderable and Active.

Products are bundled using a “Has-A” Containment relationship as either a child or a realization.

We recommend using Specification Type to designate Products, Offers, Resources, and Services.

Pricing Challenges and the Salesforce Industries Approach

On a daily basis, your company faces many pricing challenges that include discounting prices:

In bundles without changing the base price of each product when purchased separately

Over time without ongoing data entry

Only for a limited time

Based on whether the account is customer or employee, in a certain region, or has a certain service level agreement

**Challenges and the Salesforce Industries Approach**

The Salesforce Industries solution offers:

Reusablility: A component-oriented system with reusable items which live independently of products

Pricing types:  Categories of pricing such as penalties, charges, and adjustments to existing charges

Frequency: Settings that determine the frequency of the charge

Easier updating: Ability to transition from older pricing to newer pricing for less expense and disruption

With Salesforce Industries' pricing model, you can:

Use different methods to price bundles of products

Assign more than one base price to a product

Separate pricing for business needs, such as employee discounts or customer service level agreements

Analyze Product Prices and Their Settings

A product price is a set of pricing components assigned to a product. Salesforce Industries' pricing approach:

Provides components to create all the aspects of pricing

Separates the price and its nuances from the product

Products List in the Cart

To make sure a product appears in the PRODUCTS list of the cart, assign a price to it.

Base Price and Display Text

To mark a price as the base price, select the Base Price option when you create the price list entry. The display text of the base price appears along with the product in the PRODUCTS list of the cart.

Prices Not Marked as Base Price

If a product has no price list entries marked as a base price, the prices in the PRODUCTS list will display as zero. However, the charge will show in the cart.

Price Lists

By using price lists:

You can assign more than one base price to a product by creating price list entries stored in different price lists.

You can create price lists based on the needs of your business. For example, you may wish to separate customer pricing from wholesale pricing, or from employee pricing.

**Price List, Price Book**

How does a price list relate to a Salesforce price book? Every price list is associated with a price book, because the price book is required by Salesforce. Salesforce Industries uses the Salesforce price book as a pass-through.

**Parent/Child Price Lists**

A price list can be a parent that has child price lists. The parent price list will contain pricing elements that you want to use in all its child price lists.

**Context Rules and Child Price Lists**

You can use a context rule to apply a price list entry located in a child price list based on whether the account meets a specific condition.

For example, suppose you want to set different prices, depending on the geographic location of the account. You can write a context rule to retrieve price list entries from the California child price for California accounts, and from the Nevada child price list for Nevada accounts. Context rules are in the Rules module.

Pricing Variables and Pricing Elements

Pricing Variables

Pricing variables enable you to set a type of price that you can associate with a charge. Type determines:

Whether it is a regular charge or a penalty fee

How frequent – whether the charge is for one time or ongoing

Whether it is a price charged to the customer or a cost the company must bear

Whether payment of the charge is accomplished with currency or loyalty points

**Pricing Elements**

After you have chosen the type of price, you choose from a list the amounts in the price list that are associated with that pricing type. A pricing element combines the pricing variable with the amount and currency.

You can create multiple price list entries for the same product using the same price list. To prevent these price list entries from interfering with each other, each price list entry can have an effectivity date range. When you use this technique, the product's price can change over time as each price list entry becomes effective.

**Using Effectivity Date Ranges**

You can use the **Effective From** and **Effective Until** fields to assign a set of base prices that decrease over time

**Gaps In Effectivity Date Ranges**

If there are gaps between the effectivity date ranges for the base price, the product will not appear in the product list of the Cart. For example, a gap between May 16 and May 23.

# The Basics of Bundles

A bundle is a logical grouping of products into one "package". The top level of the bundle is considered the parent product. All products under this level are considered child products.

**How are the prices of bundles totaled?**

**Rolled Up Pricing**

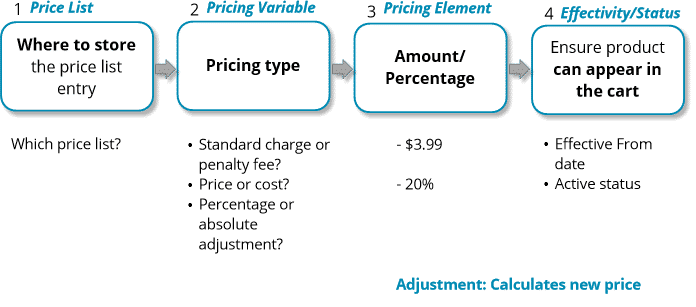
The prices of child products and the parent product always roll up into the total for the bundle.

# Changing Prices of Child Products

You can change the price of any product or child product in a bundle without altering its base price. Your options are to:

* **Adjust** the price: uses the base price to calculate a percentage discount or an amount discount; or
* **Override** the price: overrides the base price.

**Adjusting Prices of Child Products**



# Manually Change Prices in the Cart

You can manually change a price in the Cart by:

* Adjusting it with a percentage or amount
* Overriding the price

When you discount a price that is a recurring charge, you can also assign:

* A time plan to limit the time frame of the discount
* A time policy to determine how the discount begins and ends

You can also delete any manual changes you have made to prices in the Cart.

Context rules can be used to control the conditions under which you can make manual pricing adjustments. You can also use context rules to essentially "turn off" this capability and ensure that no manual adjustments are made.

**Pricing Elements**

You use pricing components to assign prices to products and to adjust base prices by percentage or amount. Pricing components are reusable.

**Pricing Elements**

The basic types of pricing elements are:

* **Charges** to assign a base price
* **Adjustments** to adjust a base price
* **Overrides** to override a base price

If you don’t see the price you need in the list of prices with their currency and types, then you must create a new pricing element.

**Best Practices**

The name of the pricing element should indicate as much information about the price as possible, such as the currency, amount, and whether it is a one-time or recurring charge. Additionally, if the charge is recurring, make sure to specify when it recurs.

For example, a recurring monthly charge of $9.99 could be named “$9.99 RM” or “$9.99 Monthly”. The code is a unique code which identifies the individual pricing element.

**Time Plans and Time Policies**

**Time Plan Settings**

The time plan is the length of time for pricing to apply to a product. For example, a 2-year subscription to cellular service has a 24-month time plan.

Settings for a time plan include:

* Total duration of the time
* The units of measure for the duration: Day, Week, Month, Year

**Time Policy Settings**

A time policy indicates when the price starts and stops being applied. When you create a time policy, you have these setting options:

**Start Policy**:

* Purchase Date: Typically, the date on which the customer submits payment and signs the agreement
* Cycle Start Date: Date on which the customer’s next billing cycle begins
* First Day of Month: First day of month
* Activation Start: Date on which the customer activates the service or device

**End Policy**:

* End of Plan Duration: The last day of the plan’s duration
* Cycle End Date: The last day of the billing cycle
* Set by Order Management
* Last Day of Month: The last day of the month

**Type**:

* Start Proratable (on/off)
* End Proratable (on/off)

**Start Time Delayed:**

* Start Time Delayed (on/off)
* Delay Offset: Amount, supporting positive and negative numbers
* Delay Offset Unit of Measure: Day, Week, Month, or Year

# Promotion Challenges

**Promotions** is a separately purchased module that gives you flexibility and automated functionality for designing and deploying promotions and discounts in your Industries CPQ implementation.

# Advantages of Promotions

Promotions are separate and distinct from products. This means that product offers and bundles can remain intact. Any promotions you create can apply to products without altering the products.

**Managing Promotions**

You can manage promotions in the Cart by:

* Searching for promotions in the **PROMOTIONS** list of the Cart
* Deleting promotions placed in the Cart
* Applying a promotion to a product already placed in the Cart

Disqualified promotions can be seen in the Cart. If the customer can meet the requirements for the promotion, it becomes qualified and available.

**Automated Functionality**

Automated functionality can be based on:

* **Time plan/commitment duration** of the promotion:
  + You can charge a penalty for early termination of a subscription using penalty context rules.
  + You can create a follow-on promotion that automatically begins when a previous promotion ends.
* **Service continuation:** Subscriptions can be automatically canceled or continue after a promotion ends based on settings within the promotion.
* What the promotion applies to:  promotions can apply to **assets**as well as to items in the Cart.

# Promotion Basics and Building Stages

You can create promotions on existing products and product bundles.

Promotions have settings for:

* The duration of the subscription and the way the subscription begins and ends
* Different ways to discount the products in the promotion and how long each discount lasts
* Whether the promotion:
  + Should or should not appear in the list of promotions to manually add to the Cart
  + Is automatically applied to a customer's account
  + Begins immediately after a previous promotion
  + Can be applied to a product the customer has already purchased

**What is the difference between a product and a promotion?**

A product:

* Can be used on an ongoing basis and typically does not expire soon
* Contains products or quantities of each product that are not necessarily limited
* Can contain multiple bundles

Products can be bundled together with the pricing of child products totaled (or “rolled up”) and added to the price of the parent product.

Typically, promotions are individual products or product bundles you create for a limited:

* Time
* Customer group
* Subset of products

You can use promotions to apply to products you have already created.

**When do you use a product and when do you use a promotion?**

A product is useful when there are no limitations.

Create a promotion when you wish to provide discounts for a limited time, have a limited group of products to sell, or a limited set of customers you wish to reach.

**Product Time Plans and Cardinality**

**Discount Duration**

When a product's charge recurs on a regular basis, such as every month, you can decide how long a promotional discount on the charge will last. It can last the entire length of the promotion, or it can end before the promotion commitment duration ends.

**Changing Min/Max/Default Amounts of a Child Product**

You can override the original cardinality of any product in a promotion. In this example, two products that are normally optional in the bundle are required in the promotion.

# Automated Functionality for Promotions

**Penalty for Cancelled Promotion**

You can use a context rule to issue a penalty against an account that has cancelled promotional discounts. We cover penalty context rules in the Rules module.

**One Promotion Begins After a Previous One Ends**

You can create a follow-on promotion that does not appear in the Cart. Instead, you can set it to begin as soon as a previous promotion ends.

**Manually Opt Out to End Subscription**

If the promotion is set to **Manually Opt Out**,  the customer must contact you to request the subscription to end.  If they do not, the promotional discounts to the promotion end, but the subscription continues.

**Manually Opt In To Continue Subscription**

If the promotion is set to **Manually Opt In**, the customer must contact you to request the subscription to continue. If they do not, the promotional discounts to the promotion and the subscription ends.

# Adjusting the Prices of Products in a Promotion

You can change the price of any product in a promotion by adjusting the price by a percentage or an amount. In addition to adjusting the price, you can set how long the adjusted price will last by assigning a time plan. Say, 20% off for 3 months or $5 off monthly for 6 months, for example.

When adjusting the price of a product in a promotion it is not necessary to run the Product Hierarchy Maintenance, Clear Managed Platform Cache or the Refresh Platform Cache jobs as explained in the EPC module.

# Overriding the Prices of Products in a Promotion

You have the option of overriding the price of a product in a promotion rather than adjusting it by a percentage or amount. So, instead of adjusting the price to 20% off $34.99 for 3 months you override the $34.99 price to $24.99. And, you can assign a time plan to the override as well.

# Refining Promotions

There are two setting options for the **Update Scope** field when building out a promotion for a product bundle. The Update Scope field controls how the promotion is applied to the bundle in the Cart.

* 1

1. **Update Item Only:** the parent product is the only product that is updated for the promotion. No promotional changes are made to the child products in the Cart.
2. **Update Item and Child Items:** the parent product and its child products are updated for the promotion in the Cart.

# Deleting Promotions

There are two settings in which to delete promotions in the Cart.

* 1

1. **Deep Delete:**a deep delete removes the promotion and its changes along with the products pertaining to the promotion. This is the default setting for DeleteServices.
2. **Shallow Delete:** when you delete the promotion it only deletes the promotion and its changes (discounts and cardinality). The products that the promotion applies to remain in the Cart. To change the delete to a shallow delete, edit the DeleteServices setting.