**What is Order Management?**

Order management is a system that brokers (or exchanges) data between the front office customer request and the back office fulfillment systems to deliver accurate services to the customer.

Its job is to ensure that all fulfillment systems receive the data that they need, at the precise time they need it, in order to successfully deliver telecommunications services.

A best-in-class order management application is responsible for providing a number of key functions:

1. **Validation of Orders:** It should provide two types of validation:  
   First, the application should ensure that the order is properly structured (we refer to this as syntactical validation).  
   Second, it should ensure that the products and services on the order are allowed to be fulfilled (we refer to this as commercial validation). To create this validation, we’ll use the business rules we define in the shared product catalog.
2. **Decomposition of Orders:** It should convert commercial order information into technical order information. Here again, we’ll use the rules provided in the shared product catalog.
3. **Communication with Fulfillment Systems:**The application should communicate with fulfillment systems to send and receive information in the correct sequence.
4. **Manage Fulfillment Lifecycle and Notifications:** It should manage the entire order lifecycle, providing notifications for events and handling errors that occur.
5. **Zero-touch vs. Manually Driven:** A best-in-class order management application should find the right balance between manual and automated tasks, to maximize accuracy and efficiency.

**Addressing COM and SOM Order Management Requirements**

The TMForum identifies 2 main areas of order management. These are:

* Commercial Order Management, also referred to as COM and
* Service Order Management referred to as SOM

Order Management systems that play in both COM and SOM spaces have very similar requirements, including the key functions we’ve just discussed. Industries Order Management covers both COM and SOM in compliance with the TM Forum.

Because it’s built on a catalog-driven paradigm, Industries Order Management blurs the lines between COM and SOM. As such, if your catalog includes products, Industries Order Management will automatically act in a Commercial Order Management role. If your catalog includes services, Industries Order Management will automatically act in a Service Order Management role. And, when your catalog includes both products and services, then Industries Order Management will act as both a COM and a SOM.

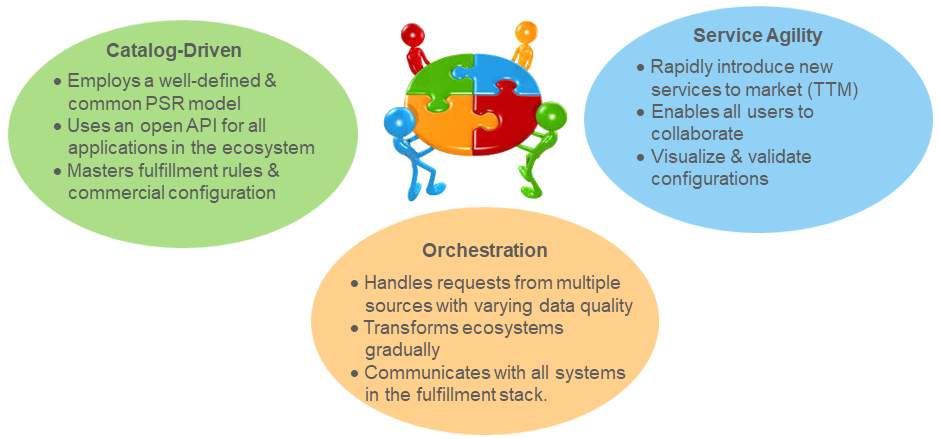
Industries Order Management application has the flexibility to broker data to your fulfillment systems regardless of the context of that data.

**Industries Order Management - Our Goals**

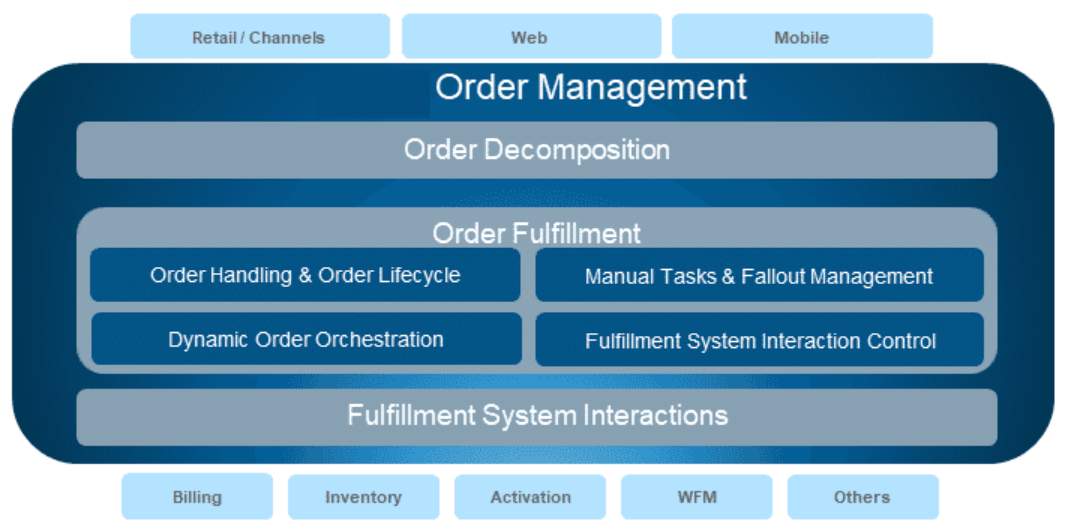
The goal of Industries Order Management is to revolutionize the order management space by providing the first enterprise catalog-driven order management application that:

* Integrates into your end-to-end ecosystem by combining a complete product, service, and resource model to drive sales, CPQ, ordering, and fulfillment functions.
* Provides one user interface with common tools, enabling your teams to collaboratively and rapidly create, manage, and deliver offerings to market.
* Allows for an evolutionary approach to transform both front office and back office systems to become catalog-driven over time. This supports the elimination of data silos and the mastering of both Commercial and Technical entities in a single model.

Critical Service Provider Operational Needs



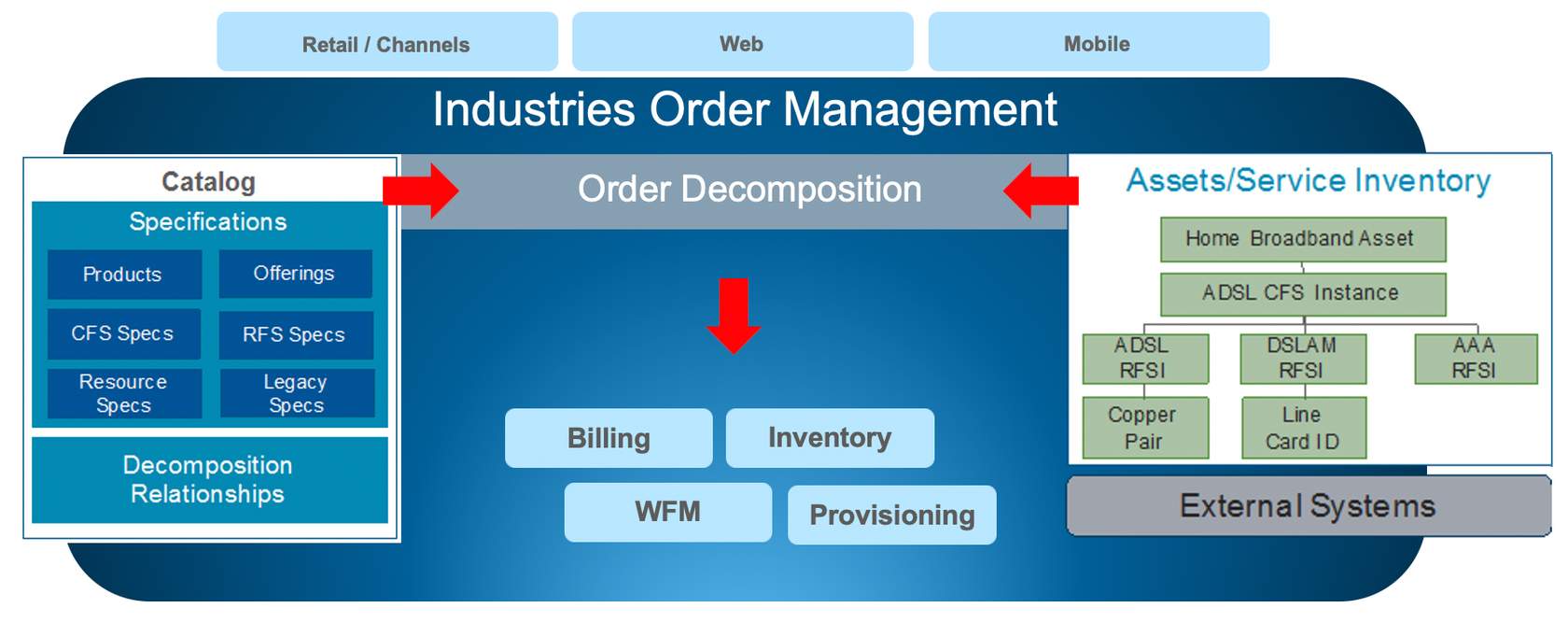
**Industries Order Management Solution Map**



At a high level, Industries Order Management provides the following layers of functionality:

1. Orders are received from various systems, like retail, web, or mobile applications. We call this omni-channel order submission.
2. Order Decomposition uses the relationships and rules defined in the catalog to enrich the commercial order with the technical information required to fulfill it.
3. Order Fulfillment is then responsible for tracking orders through a lifecycle of states, dynamically generating workflows and controlling and sequencing the interactions with fulfillment systems.
4. The Fulfillment System Interaction Layer provides integration logic for communicating with fulfillment systems, using technologies like DataRaptor, Apache Camel, and Java.

**Order Decomposition**



The process of decomposition includes a number of important functions:

1. As each order is received, Order Decomposition uses the catalog to enrich the commercial order with the technical attributes needed for fulfillment.
2. Next, Order Decomposition checks subscriber inventory to ensure the appropriate fulfillment actions are taken based on the services the customer already has.
3. And finally, Order Decomposition generates a series of sub-orders or fulfillment requests, that include related fulfillment tasks.

**Order Orchestration**

The Industries Order Management Orchestration process:

* Uses the catalog to dynamically generate fulfillment actions based on the order context
* Enforces any dependencies in the fulfillment actions
* Establishes and maintains connections to fulfillment systems and applies the correct logic to handle connections
* Interacts with external fulfillment systems such as Inventory, Activation, Billing, and Workforce Management to exchange the data needed for fulfillment

# Fallout Handling and Manual Tasks

Industries Order Management Fallout Handling and Manual Task Capabilities:

* Automatically detect errors from external fulfillment systems using error codes
* Highlight failed tasks in the process flow, enabling fallout managers to quickly locate and troubleshoot issues
* Automatically assign tasks to queues for manual resolution, allowing users to retry, complete or skip tasks

**Fulfillment Operator**

Responsible for manually enriching an order with information during the normal course of fulfillment. Examples may include approving the credit rating of a customer or manually assigning an available piece of equipment to the order.

Responsible for dealing with errors that occur in the interactions with fulfillment systems. Examples may include addressing connection management issues or resolving specific errors that are returned from a fulfillment system.

# Connection Handling and Integration

Industries Order Management Connection Handling and Integration Logic capabilities:

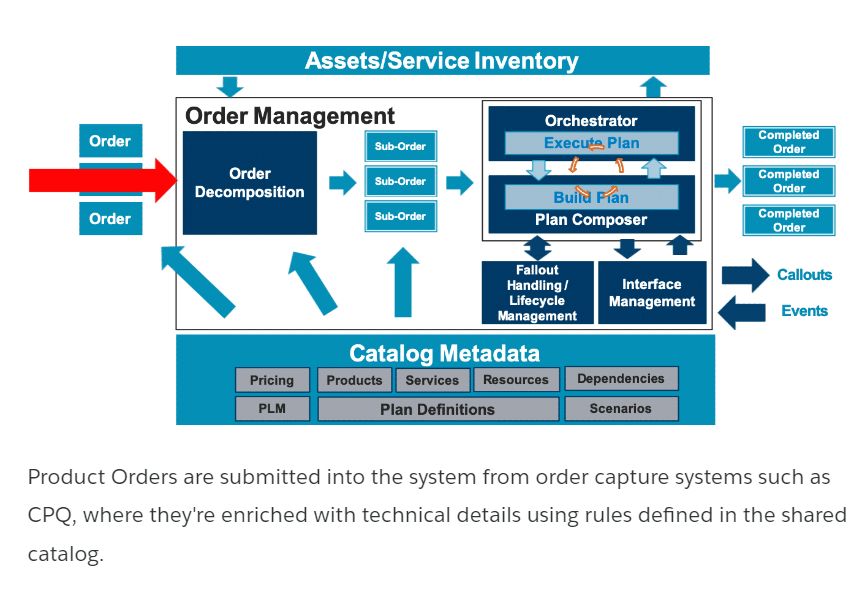
* Automatically detect connection failures
* Queue up fulfillment requests for retry rather than generating fallout
* Attempt to re-establish failed connections according to pre-defined retry policies
* Restart fulfillment requests when connections are successfully re-established
* Provide comprehensive protocol and integration technology such as DataRaptors

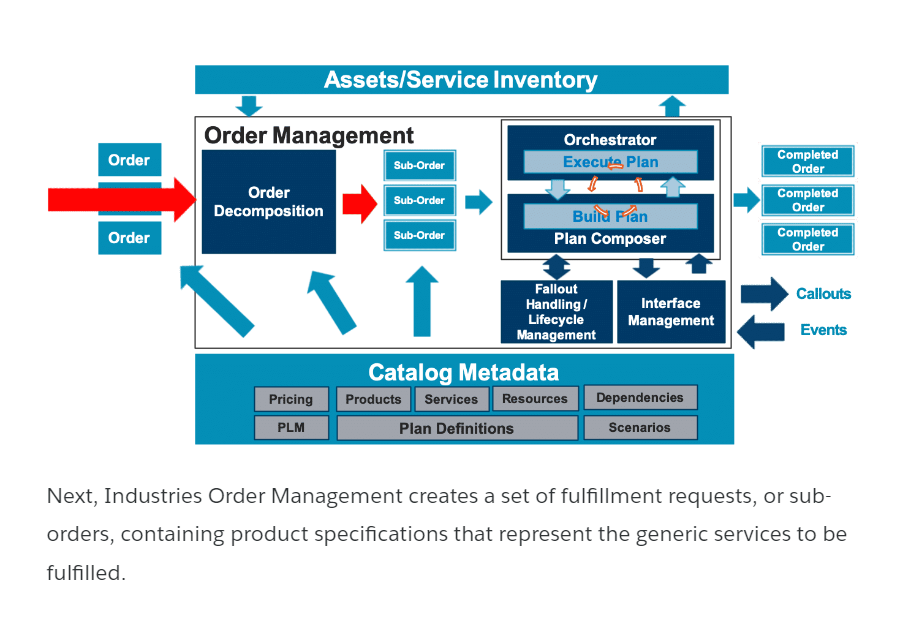
# User Experience

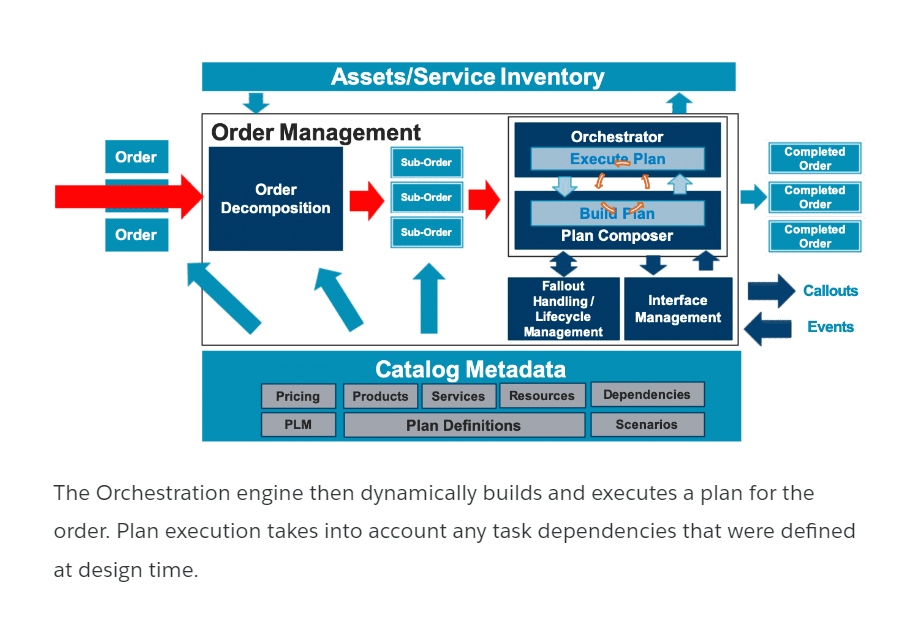
Industries Order Management provides a seamless user experience with:

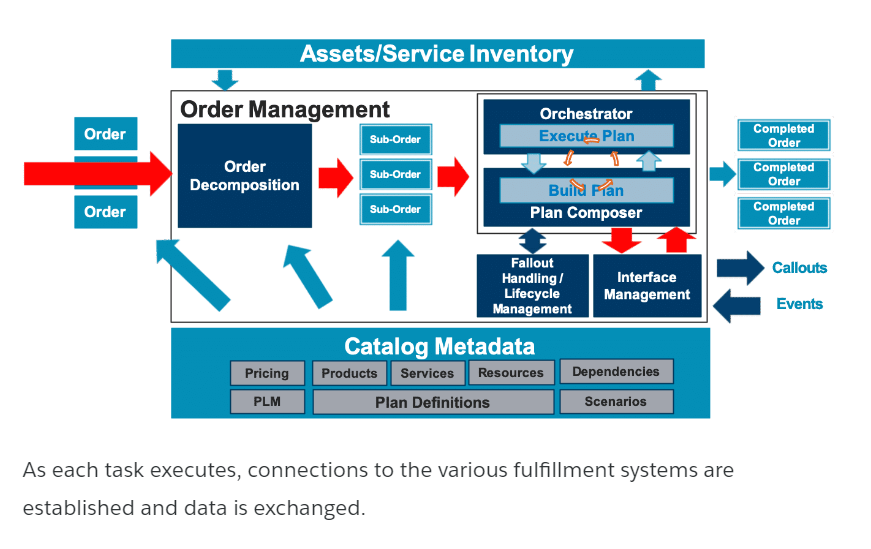
* Single-sign on for all users of the system including those performing fulfillment related roles such as fallout managers and fulfillment designers
* Transparent navigation and consistent application look and feel for both design-time and run-time functions
* Centralization of all product, service, and resource configurations in the catalog allow users to easily collaborate

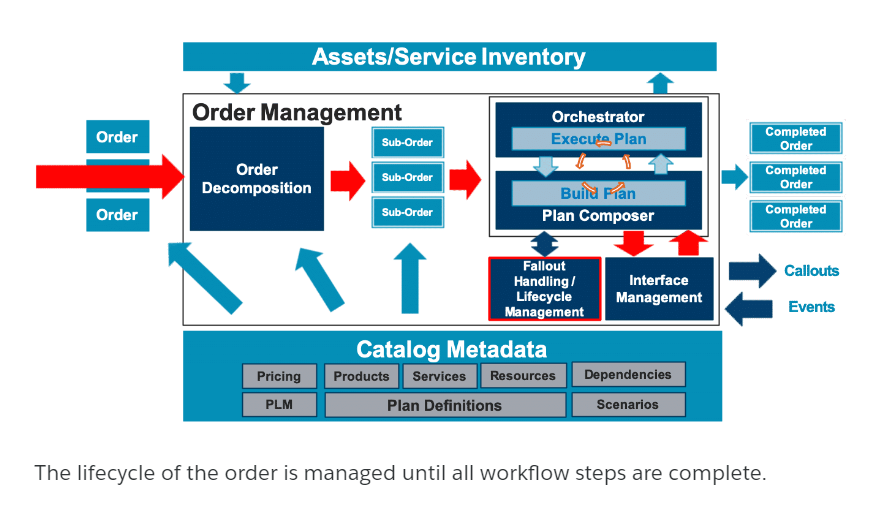
# Order Management Architecture

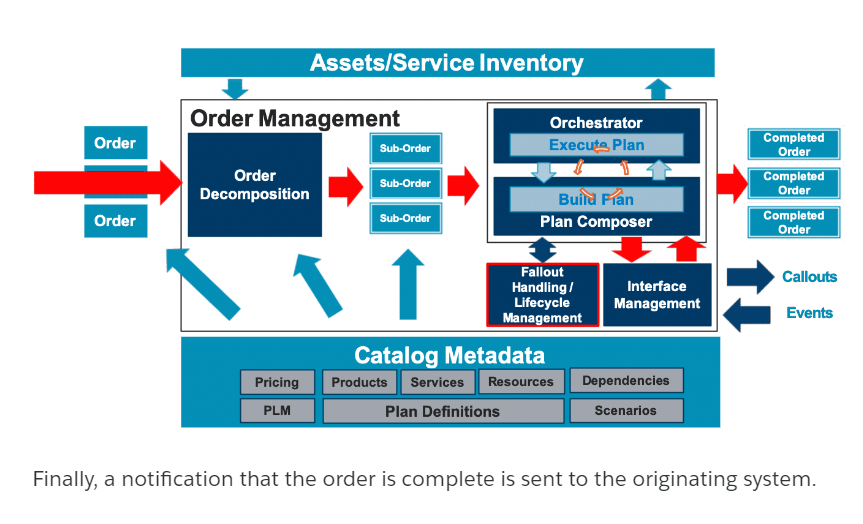












# Industries Order Management and the Shared Catalog

The Shared Catalog is the foundation for Industries Configure Price Quote (CPQ) and Industries Order Management (IOM). The Shared Catalog describes all of the incarnations of a product, as it travels from a twinkle in the customer's eyes to the physical implementation in the customer's hands. The Shared Catalog supports all levels of product definitions, including offers, product specifications, services, and resource specifications. Using Industries Order Management, product specifications are translated into various technical and fulfillment products that are tuned to each of the downstream systems that use them.

**Terminology**

* **Commercial products** - Products in the Shared Catalog that are commercial (what the customer sees).
* **Technical products** - Products in the Shared Catalog that are technical (what the customer does not see, but are critical for downstream order fulfillment systems).
* **Shared Catalog**- The Shared Catalog includes both commercial and technical products. The Product Console provides create, read, update, and delete (CRUD) functionality for the Shared Catalog.

**Industries Order Management**

The first catalog-driven order management system for the enterprise.

From a technical perspective, Industries Order Management has two key facets:

1. Order decomposition
2. Order orchestration

Industries Order Management should not be thought of as completely separate from the Shared Catalog. Ideally, Industries Order Management starts with the catalog and the product model it uses.

**Technical Attributes**  
Some things in life (and technology) are easy to understand. Others are a bit blurrier. Try to grasp the key points below, but don't sweat the small stuff!

**Commercial Products**

As you learned earlier, commercial products are most easily understood as the products in the Shared Catalog that the customer sees and understands. Commercial products are the things you order from a catalog and toss into your shopping cart.

**Technical products** are not as obvious to some. For now, consider the following additional helpful points:

* Technical products are also in the Shared Catalog. (They don't have their own catalog.)
* Similar to the way customers understand commercial products, downstream fulfillment systems understand technical products.
* The process of mapping commercial products to technical products in order to communicate with downstream order fulfillment systems is called order decomposition.

One last concept before moving on: product attributes are required for order decomposition to work. Industries Order Management relies on product attributes to create mappings between commercial product fields and/or attributes and their corresponding technical product attributes.