

Using XML to Overcome Barriers to Collaborative Commerce

Nick **Arnold** <narnold@immediation.com>

Abstract

After a period of uncertainty over the impact of the Internet on traditional resellers and channel partners, it is now clear that dis-intermediation is not the norm. Traditional channel partners, as well as new players, need increasingly integrated information and business processes. For one or more partners to collaborate effectively online, they must synchronize their content, pricing policies, inventory availability and logistics systems. This paper will discuss how XML can be used to facilitate online collaborative selling by overcoming obstacles to business process integration.

1. Using XML to Overcome Barriers to Collaborative Commerce

In this paper, we will examine the business issues related to collaborative selling on the internet. We will then look at how XML can be used to integrate systems and processes in fulfillment of these business needs.

1.1. What is Collaborative Selling?

Collaborative selling involves two or more partners combining their digital assets to present the strongest commercial offering. Contrary to most predictions in the early day of e-commerce, dis-intermediation was not the new wave. In fact, almost all online dis-intermediation efforts have failed. The reality is that partners and sales channels are invaluable and that the real challenge is to migrate these relationships to the online world.

For the purposes of this paper, we will consider only a two party partnership: that of an original manufacturer who sells direct online and who wants to support its sales

channels by controlling the brand, content and messaging at its retail partner sites. For effective online collaboration, these two parties must be integrated in numerous ways. These can be grouped into six principal categories: content management, product management, price, transaction or shopping cart integration, inventory management and fulfillment.

1.2. Content Management

For effective collaborative selling, the following functionality is required:

The manufacturer already has an existing commerce-enabled website. From this single Commerce Platform, the manufacturer needs the ability to produce different content per retailer based on business rules. At a high level, this includes logos and navigation bars for the Manufacturer brand. Some examples are: a Manufacturer home page at Macy's that contains a unique Manufacturer@Macy's logo, another home page at Target that contains a unique Manufacturer@Target logo.

Similarly, depending on the business rules governing the relationship, certain elements of the Manufacturer brand should not be accessible through the navigation bar of the Manufacturer brand. Some examples are: a store locator function for Manufacturer@Target should show only Target stores, the MyManufacturer direct marketing pages should not be accessible at Macy's.

Technically, this can be achieved by creating a retailer profile within the Commerce Platform and passing a retailer identification parameter on the URL linking the retailer partner and Manufacturer. The Commerce Platform will then display the appropriate content.

1.3. Product Management

We assume that for some Manufacturer brands, not all products will be sold at all retailers. In the case of our Manufacturer Brand A, all online retailer partners currently carry 100% of the online catalog. This might not be the case with Brand B or other brands, so the system design needs to accommodate different product availability per retail partner.

The following functionality is required:

For a single Commerce Platform instance, the ability to show specific product lines and/or products for a given retail partner. At a high level, this includes the display of any part of a product line or product within a product line. An example is BrandB@Target where Target.Direct chooses to carry only a part of the entire Brand B product line.

Technically, this can be achieved by creating a Retailer/Product database table. This table would contain information concerning each product and whether it should be shown when the store within a store is generated for any given retail partner. This table is maintained programmatically through a combination of JMS bus technology using a "publish and subscribe" paradigm and XML as the data interchange format. In this case, the server on which the Retailer/Product database is maintained communicates to both the Manufacturer and Retailer systems to obtain the latest information concerning product availability.

1.4. Price

For the Manufacturer's Brand A, retailers have agreed to sell Manufacturer products at the suggested MSRP. This might not be the case for other brands. A further consideration is whether the Manufacturer site is commerce enabled and therefore the MSRP is included in the content, or whether the Manufacturer site is "information only" and therefore there is no price in the content managed by Commerce Platform.

For a commerce-enabled brand where the price is included in the Manufacturer content, there is a requirement to remove this price dynamically and replace it with the retailer's price. For an "information only" site, there is a requirement to append the retailer price and buy button to the "information only" content.

In the case of a commerce-enabled brand, price management could be achieved in the following manner: the software enabling the collaborative selling relationship would perform a real-time query to the retailer commerce engine or other system to obtain the retailer price. In order to provide scalability across many partners and maintenance over time, this would also be done using XML. The software would then replace the price found in the Manufacturer content with the retailer price.

In the case of an "information only" brand, price management could be achieved in

the following manner: the software enabling the collaborative selling relationship would perform a real-time query to the retailer commerce engine or other system to obtain the retailer price. The software would then append this price, plus the retailer buy button, to the Manufacturer content.

In order to achieve the above functionality, it is assumed that the Commerce Platform will either have product driven templates or that the HTML of each product will contain unique tags identifying products in a standardized manner, such as the UPC.

1.5. Transaction or Shopping Cart Integration

Again, there are two scenarios. One scenario addresses the requirements for a commerce enabled Manufacturer brand, such as Manufacturer Brand A; a second scenario addresses the requirements related to "information only", non-commerce enabled Manufacturer Brand B.

For a commerce enabled brand (Brand A), the following functionality is required: when an end user is browsing the Manufacturer offering at a retail partner site and clicks on the Manufacturer buy button, the product must be loaded into the retailer shopping cart for further processing. The form this takes is negotiated on a retailer-by-retailer basis and includes, but not be limited to the following behavior (1) the product is loaded into the retailer shopping cart and the end user stays on the Manufacturer product page; (2) a pop up window is launched and the end user can confirm that he wants to load the product in the (retailer) shopping cart; (3) the end user clicks on the buy button, the product is loaded into the retailer shopping cart and the end user is served the appropriate retailer shopping cart page. Again, XML is used to model the product identification synchronization and data interchange between the two sites and commerce engines.

Combined with Java, XML plays an important role in all these cases: (1) perform the mapping between the Manufacturer brand product identifier (UPC) and the corresponding retailer product identifier; (2) access and serve any pop up windows as required; (3) re-direct the end user to the appropriate "next page" as required.

1.6. Inventory Management

In some cases, the retail partner does the fulfillment directly from their own stocks and there is no requirement for integrated inventory management. An example is the Manufacturer@Macy's collaborative offering, where there is no inventory availability check prior to loading the product in the Macy's shopping cart. In other cases, depending on the business relationship between Manufacturer and the retailer partner, there could be the following requirements: (1) When an end user clicks from the retail site to the collaborative offering, only those Manufacturer products for which the retailer has inventory will be shown; (2) When an end user clicks from the retail site to the collaborative offering, if the retailer does not have inventory but the product is available for back order, the text on the detailed product page will be modified accordingly; (3) When an end user clicks from the retail site to the collaborative and the retailer has no inventory for a given product but Manufacturer or a 3rd party does have inventory, the product is shown as normally available, but the remote stock is reserved at check out on the retailer site.

Technically, all these requirements can be addressed by using XML to perform a real-time inventory check on the retailer, Manufacturer or 3rd party inventory and pass an inventory status parameter on the URL to the Commerce Platform which will perform the necessary content management functions.

It is assumed that (1) all products are assigned to a product category and (2) each page should contain within the URL a category or product parameter in order to perform an inventory check..

1.7. Fulfillment

In some cases, the retail partner does the fulfillment directly from their own stocks and there is no requirement for integrated order routing and fulfillment. In other cases, the retailer might want Manufacturer or a 3rd party to do the fulfillment for a certain range of products. This functionality can be achieved by creating a retailer/product/fulfillment table that indicates for a specific product at a specific retailer, the order should be routed to X party for fulfillment according to Y order processing format. This can be achieved by using JMS and XML to monitor the transaction on the retail partner site and automatically route the order according to the business rules contained in the retailer/product/fulfillment table.

Biography

Nick **Arnold**

Director, Strategic Business Consulting

iMediation, Inc.

New York

USA

Email: narnold@immediation.com

Nick Arnold - Nick Arnold is Director of Strategic Business Consulting for iMediation. iMediation is a pioneer in delivering software and services for managing Internet Selling Relationships. Prior to that, Nick managed the production, research and development, and Internet sections for the OECD's publishing activities. Prior to joining the OECD in 1990, Nick worked in New York in a variety of positions within the information management field.