

Buy with Prime for Salesforce implementation overview

Overview: Buy with Prime built an integration with Salesforce that you can use to incorporate Buy with Prime into your existing checkout experience and business operations. The integration supports Prime search and filtering while shoppers are browsing, and building a mixed-cart of items that are Prime eligible and other items. You can use this document to help develop an estimate for the scope and cost of the project.

What areas of an ecommerce business are involved in Buy with Prime?

Buy with Prime is a combination of the ecommerce services that you already use: shopper authentication, shopper wallet, and a 3PL that offers post-order management experiences (for example, self-service returns). Because of this, Buy with Prime touches your product pages (Prime badge, delivery estimate), checkout (signin, address, wallet, place order), and order management system (OMS) (fulfillment, order updates). Ideally, downstream services, such as accounting, analytics, and customer service can rely on the OMS as the source of truth and continue to work as they do today.

What are the general steps to implement Buy with Prime for Salesforce?

1. Create an account for Buy with Prime and generate Buy with Prime API credentials.
2. Inbound inventory to Amazon fulfillment centers (or use your existing inventory).
3. Download and install the Buy with Prime for Salesforce cartridge library.
4. Configure syncing jobs to dynamically enroll or remove products from Buy with Prime and confirm their current status from the Salesforce Business Manager.
5. Use (or merge) Buy with Prime UI elements to add Prime badging and delivery estimates to product listing and detail pages.
6. Use (or merge) Buy with Prime UI elements, such as Prime badging, delivery estimates, shopper signin, and shipping methods, to cart and checkout pages.
7. Use (or merge) Buy with Prime calls to update place order to make a real-time inventory reservation for Prime items on the order.
8. Use (or merge) Buy with Prime UI elements such as tracking link on order confirmation and order history pages.
9. Update OMS to add Buy with Prime as a fulfillment provider.
10. Update OMS to add Amazon Pay (or Amazon Pay supported payment processor such as Adyen) as Payment Service Provider.
11. Update OMS to read the Buy with Prime flag from orders coming from the storefront, split line items (multiple 3PLs for one order), and submit fulfillment requests.
12. Set up the Buy with Prime Event Listening infrastructure to subscribe to Buy with Prime (and optionally Multi-Channel Fulfillment) generated events, including order statuses.
13. Set up Buy with Prime events to initiate OMS actions (for example, shipped status initiates payment capture, or item_returned initiates refund), and Buy with Prime inventory and buyability events initiate updates in the storefront.

What notable changes does a brand have to make when implementing Buy with Prime?

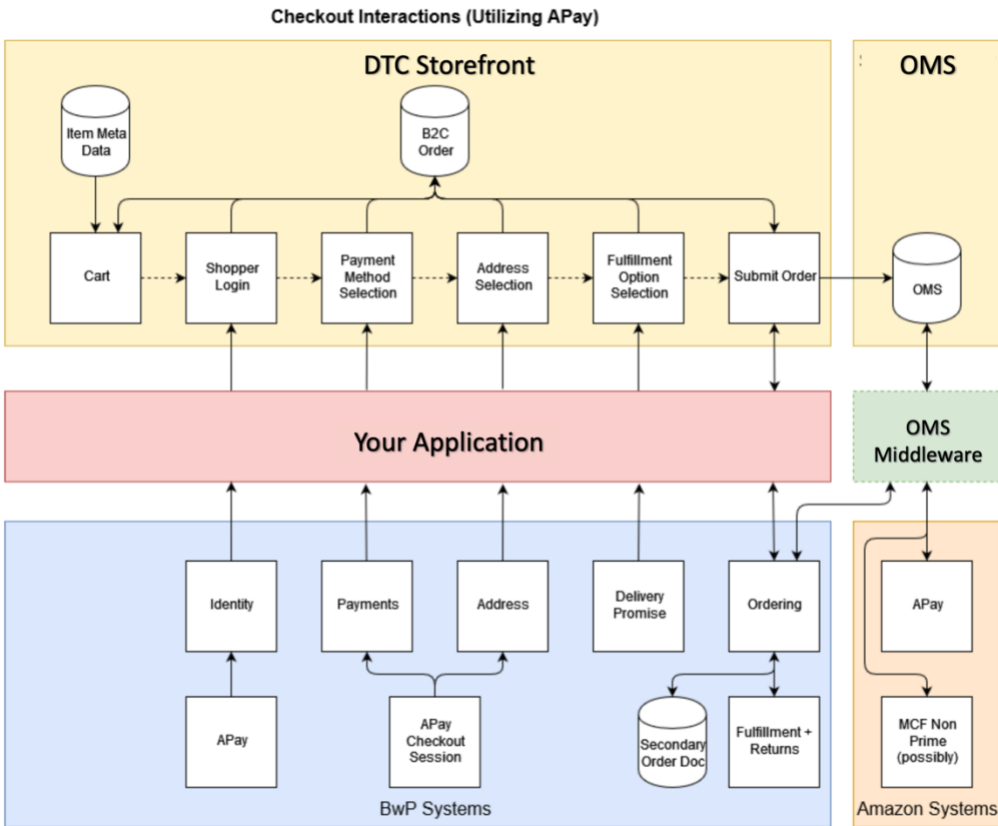
Most of the updates to the Storefront, OMS, and back-office services are standard. Adding a user experience to product detail pages or a new 3PL work as expected. Buy with Prime can improve the shopper experience by sharing relevant information during the checkout process, rather than hiding it behind the scenes. For example, with most ecommerce shopping experiences, the shopper doesn't know who's shipping the product and when it will arrive. Usually that information is buried in the OMS, even though shoppers benefit from having that information before they place their order. Potential changes for improving the shopper experience include the following:

1. Fulfilling split shipments: Many OMSs support splitting an order with multiple line items into multiple shipments that could be fulfilled by different warehouses or providers. For example, OMSs support sending one sub-shipment to a physical store and the other to be fulfilled by a 3PL. If you haven't configured this in your OMS, this could be significant change because it has downstream implications, including how returns are handled.
2. Adding a new fulfillment node: You need to add Buy with Prime as a new fulfillment node in your OMS. Orders coming from the storefront must have line-items flagged with this Buy with Prime fulfillment node.
3. Adding a payment processor: Today, Buy with Prime works with Amazon Pay or Adyen as a payment service provider. If you use a different payment service provider, you need to add a new one.
4. Displaying split shipments: Buy with Prime requires you to clearly signal to shoppers which items in their cart and order history are delivered with Buy with Prime. We don't want shoppers to be disappointed or confused if other items on their order ship using a different fulfillment provider and don't arrive on the promised delivery date.
5. Event handling: We provide infrastructure to subscribe to Buy with Prime events, which relies on AWS Eventbridge. You can translate these events for your systems. For example, you could build a middleware layer that calls the OMSs RESTful APIs to update order status.
6. Operational processes: You should expect and prepare for adjustments to areas of your operations. For example, Merchandisers or Product Information Specialists might need to evaluate whether products should be enrolled in Buy with Prime or not. Supply Chain Managers might need to adjust forecasting or inventory positioning. Customer Service Agents might need to adjust how they handle shopper orders, especially where orders contain both items that are Prime eligible and other items.

How does the post-order shopper experience work?

When a shopper clicks place order, the storefront makes a call to Buy with Prime to validate the order and reserve inventory before sending the order with Buy with Prime items flagged to the OMS. In the Buy with Prime response, we return a link to the self-service Buy with Prime order details page, which allows the storefront and OMS to store a link for shoppers to later review, cancel, or initiate a return for an item. From here, we expect the storefront to pass the order to the OMS, and OMS to then own further communication with Buy with Prime APIs to signal the official placement of the order and any other order update. Each time order status changes, we emit an event that can initiate actions in the OMS. For example, when the order is shipped, the OMS captures payment. If an item is returned, the OMS initiates a refund.

The following diagrams show at a high level the relationship between your storefront, the underlying Buy with Prime API, and the OMS. These likely adjust slightly based on your tech stack, but provide a helpful overview.



Post Order Interactions

