A Tour of Amazon E-Commerce Service (ECS)

To get you started, we have assembled a set of REST requests that you can access in your browser to take a tour of the major ECS features. To use these links, you should do the following:

1. Do a global change on [your subscription ID here] to insert your own Amazon Web Services (AWS) subscription ID string.

2. Copy the links, one by one, into your browser and activate the link, to see what the ECS response looks like. Of course, you can also modify the URL and re-submit it, to experiment with ECS and get immediate feedback on what is available.

Most of the URLs provided work on all Amazon locales: US, UK, DE, CA, FR and JP. Your subscription ID will work on all sites, so feel free to try out the REST requests for each site.

For each request, we have included a sample of an XML response you could expect to receive from ECS. The Amazon catalog changes on an hourly basis, so the products we have used in our examples may not return the same results to you. For convenience, or if you are not connected to the Internet, you may use the sample responses we have provided to get to know ECS and its features.

Searching the Amazon Catalog

The first operation we'll try is ItemSearch, which allows you to search the Amazon product catalog in various kinds of ways.

Scenario: You're interested in finding golf clubs on the US site manufactured by Callaway.

REST request / URL:

http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ItemSearch &SearchIndex=SportingGoods
&Manufacturer=Callaway

We encourage you to try the link for yourself, but if you are not able to see the results, you can view the sample response instead.

The SearchIndex parameter in this request specified that you were searching for sporting goods. You could also have set SearchIndex to Books, Toys, DVD, Music, VideoGames, Software or any other of Amazon’s stores. Not all stores are available on all sites. For example, the sporting goods store referred to in the REST sample above is only available for the US site.

Note also several keywords, separated by spaces (URL-encoded as %20). Capitalization and order are not significant in the search keywords.

This ItemSearch request returned the XML data for the first ten items in the Amazon catalog that matched your criteria. For each item, an <Item> element contains a description of the product. Note that the first element inside <Item> is the ASIN (Amazon Standard Identification Number)--we’ll be working with ASINs a lot both in this tour and throughout the ECS documentation.

Scenario: You want to find a variety of Harry Potter related merchandise, including books, clothing, DVDs, etc. You also noted from the last search that no product images were included, and you want to see product image URLs for this search.

REST Examples [Show]
The output shows how the ResponseGroup parameter specifies how much data to return. When the parameter is omitted, a default set of data is returned. By setting this parameter to Medium, we requested additional elements, including product image URLs.

You can search in only one store at a time. However, when you set the SearchIndex parameter to Blended, it searches the DVD, Electronics, Toys, Magazines, PCHardware, Tools, SportingGoods, Books, Software, Music, GourmetFood, Kitchen, and Apparel stores. Note how the output consists now of two sections of data:

1. <SearchResultsMap> contains a <SearchIndex> element for each store where the keywords were found. For each store, you are given the ASINs of the three most popular products in that store that matched your keyword(s).

2. For each ASIN in the <SearchResultsMap>, you will find an <Item> element with item attributes and other information.

The first ItemSearch request returned up to ten items that matched your criteria. However a blended ItemSearch request can return up to three items for each store searched.

Let’s try a few more ItemSearch requests.

**Scenario:** You’re looking for a CD which contains a specific track.

REST Examples [Show]

This search uses a special response group, Tracks, which returns a list of CDs with tracks whose titles match the search criteria.

**Scenario:** You want to find used digital cameras made by Nikon.

Note that this sample only works for locales that support the Electronics SearchIndex.

REST Examples [Show]

Note that not all sites may carry a camera that matches the search parameters. This search specifies only used items and returns price and availability information (via the OfferFull response group).

**Scenario:** You’re interested in books about European soccer leagues.

This is a very simple query as shown below which returns standard XML.

REST Examples [Show]

**Looking up Data for Specific Products**

The previous URLs (operation requests in ECS terminology) used the ItemSearch operation. ItemSearch does a search using criteria such as keywords or titles or manufacturer. The ItemLookup operator allows you to look up a specific product, typically by ASIN (Amazon Standard Item Number), but also by SKU, UPC (US only) or EAN. ItemLookup also allows you to filter your results by merchant (US only) or by delivery method (shipped or in-store pickup, US only).

The examples below use ASINs that were valid products at the time of writing. If you are trying to use an ASIN without success, you can simply go to one of the Amazon Web sites and look up a product. The ASIN will either be displayed on the product detail page or you can extract it from the URL. For example, for the Amazon page URL below:
the ASIN is 1931561648.

**Scenario: You want to find out the list price and current Amazon price for a specific book.**

REST Examples [Show]

The returned data is standard XML that includes all pricing information that is current.

**Scenario: You want to build a detail page for a specific book listed on your Web site.**

REST Examples [Show]

The Large response group returns a great deal of information that includes reviews, ASINs of similar products, customer-created lists that reference this product, and more. We'll come back to some of that data later in this tour.

**Getting Seller Feedback for non-Amazon Vendors**

If you (or your customers) are considering buying products from non-Amazon vendors, you can get feedback information about sellers with the SellerLookup operation.

**Scenario: You want to find out if a seller gets high customer-approval ratings or not.**

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=SellerLookup &SellerId=A35O48Y1QVX6
```

Note that in the output XML, the SellerFeedback/Feedback/Rating element value ranges from 1-5, indicating the number of stars that a given customer gave to the seller (1 star indicates poor service/delivery while 5 stars indicates excellent service). Only the first five feedback records are returned, but you can return additional "pages" of output using the FeedbackPage parameter, as shown below.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=SellerLookup &SellerId=A35O48Y1QVX6
&FeedbackPage=2
```

**Looking at Products with Variations (US only)**

On the US site, many apparel and sporting goods products come in a range of sizes and colors. These products are stored as a parent product that contains the standard attributes, linked to child products that are the variations (the specific size/color product). The scenario below discusses how to retrieve variation product data.

**Scenario: You want to allow customers to purchase shirts in a variety of sizes/colors.**

Step 1: Find the parent ASIN. You can use either ItemSearch or ItemLookup to get the data for a given product.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ItemLookup &IdType=ASIN &ItemId=B0002FHIWG
&MerchantId=All &ResponseGroup=Large,Variations
```
Note that this request returns a new kind of element, `<Variations>` which contains a series of `<Item>` elements for each actual combination that is available as well as its size/color/price/availability. Only the first ten variations are returned, but you can retrieve additional ones by using the `VariationPage` parameter, as shown below:

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ItemLookup &IdType=ASIN &ItemId=B0002FHIWG
&MerchantId=All &ResponseGroup=Large,Variations
&VariationPage=2
```

Step 2: Use the child ASIN to purchase. The parent ASIN cannot be purchased because it does not actually exist (i.e., you cannot buy the generic product without specifying a size, color or other varying characteristic).

### Buying Products with the Shopping Cart

The ECS shopping cart is a mechanism for managing the items a customer wants to buy before sending the customer to Amazon to purchase their selections.

**Scenario:** You want your customers to be able to browse books on your site and you want to create a shopping cart and send them to Amazon to complete their purchase.

**Step 1:** Look up the book, determine its list price and current price and availability.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ItemLookup &IdType=ASIN &ItemId=0486411214
&ResponseGroup=Medium,OfferFull
```

In the returned **XML output**, the `<ItemAttributes>` element contains a `<ListPrice>` element that specifies the standard publisher's list price for the book. However, the presence of the ListPrice element does not indicate that the book is available. Additionally, the ListPrice value does not correlate to the actual price at which the book is being sold.

To determine current price and availability, you must look at the `<Offers>` element. Every vendor who actually sells this book has an `<Offer>` element that specifies the condition of the vendor's offer (used, new, collectible, etc.). Since we did not specify a vendor by including the MerchantId parameter, only the Amazon offer is provided. Within the Offer element, the OfferListing element contains the vendor's price and availability. If the Offer/OfferListing/Availability element starts with the words "usually ships" (US), "usually dispatched" (UK), or equivalent in the other locales, then the product is available and may be purchased.

**Step 2:** Create a shopping cart and add the item to it.

```
http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=CartCreate &Item.1.ASIN=0486411214
&Item.1.Quantity=1
```

This operation creates a cart and returns to you a `<Cart>` element containing a **CartId** and **HMAC** -- identifiers that you must use for any future operations with the cart. For now, let's presume that the customer wants to purchase the contents of their cart immediately.

If the product was actually not available when you attempted to add it to the cart, it would be added to the cart as a `<SavedForLaterItem>` rather than a `<CartItem>` in the `<Cart>` element.
Step 3: Transfer the customer to Amazon to complete the purchase.

Purchasing the cart couldn't be simpler. Just transfer the customer to the Cart/PurchaseURL that was returned by the CartCreate operation. To test this for yourself, copy the PurchaseURL to your browser and access the link. If you are not signed in to Amazon, you may be asked to do so. If you are an existing Amazon customer, the contents of the new cart will be merged with your existing cart. If you are not an existing Amazon customer, you will be asked to create a new account.

Of course, since this is just a test, you probably don't want to complete the purchase process. Creating a cart does not obligate you to buy anything, it just sets the purchase process in motion. But clicking through to the PurchaseURL link will allow you to verify that the cart was created correctly and that a customer could, in fact, purchase the item or items they added to the cart.

If you are a member of the Amazon Associates program, you would want to specify your Associate tag in the cart operations, to ensure that you receive commissions on sales referred from your Web site.

ECS provides full management capabilities for the shopping cart--adding items, modifying quantities, deleting items, etc.

The previous example was to purchase an Amazon product. For non-Amazon vendors, it is not sufficient to just specify the ASIN when adding an item to the cart.

**Scenario: You wish to purchase a grill from a non-Amazon vendor.**

Step 1: Look up the grill and availability.

http://webservices.amazon.com/onca/xml?Service=AWSECommerceService &SubscriptionId=[your subscription ID here] &Operation=ItemLookup &IdType=ASIN &ItemId=B00006LS6X &MerchantId=All &ResponseGroup=Medium,OfferFull

Here is a locale-specific difference in ECS. Setting the MerchantId parameter to All returns all vendors including Amazon for the US, but returns all vendors EXCEPT Amazon for UK, DE and JP.

At time of writing, this ASIN produced several offers. As for the Amazon product above, each <Offer> contains an <OfferListing>. However an additional element, the OfferListingId is required to purchase non-Amazon products.

Step 2: Create a cart and add the grill to the cart.

This request uses an OfferListingId rather than ASIN when specifying the product to add to the cart. While ASINs are permanent identifiers, offer listings are not. You will therefore need to note the offer listing ID from the XML output returned by the previous REST request and copy it into this request:

http://webservices.amazon.com/onca/xml?Service=AWSECommerceService &SubscriptionId=[your subscription ID here] &Operation=CartCreate &Item.1.OfferListingId=5C1cZKzqzVg2Yp6OhAvSD2%2BFg1KTz4vNkr9L%2 &Item.1.Quantity=1

As before you can test that the cart is correct by linking to Amazon through the PurchaseURL to verify that the product was correctly added to the cart.

**Looking up Customers' Wish Lists and Registries**

**Scenario: You know a customer's name and you want to look up the customer's wish list.**
Step 1: Get the ID of the wish list.

http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ListSearch &ListType=WishList &Name=Jeff%20Bezos

This search returns customer information, including WishList, but you could also search for WeddingRegistry or BabyRegistry lists in the same way. You can also specify an Email parameter (in place of the name parameter) to search by email address.

Step 2: Once you have identified the ListId, you can then retrieve the contents of the list with a request as shown below.

http://webservices.amazon.com/onca/xml?Service=AWSECommerceService
&SubscriptionId=[your subscription ID here]
&Operation=ListLookup &ListType=WishList &ListId=BUWBWH9K2H77 &ResponseGroup=Request,ListFull

The End of the Tour

This is the end of our tour of ECS. For more information on how to create your own Web site or application that takes advantage of ECS, consult our Getting Started Guide!