# xpedx.com Next generation

# *B2B Punchout Design Document*

**Authors: Sterling Commerce**

**Date Created:** 5/4/2010

**Last Updated:** 1/10/2011 3:59 PM

**File Name:** C:\Documents and Settings\bfurman\My Documents\Temp\Methodology v1.1\Project Management\TEMPLATE - DOCUMENT - Use Case Definition.docxpedx B2B Punchout Detail Design Doc V1.4.docx

Approval Signatures (Mandatory)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Title** | **Name** | **Signature** | **Date** | **Comments / Issues / Concerns** |
| **xpedx Owner(s)** | Steve Bugher |  |  |  |
| Cheryl Tullis |  |  |  |
| **Sterling Commerce Owner(s)** | Guy Read |  |  |  |
|  |  |  |  |

**Note**: The sign off indicates approval of all sections of the document.

Document Revision History

This chart tracks the changes introduced by the revisions to the document as the project progresses through the stages of the System Development Life Cycle (SDLC).

| Version | **Date** | **Description (Changes Made)** | **Author(s)** |
| --- | --- | --- | --- |
| 0.1 | 05/04/2010 | Initial Draft | Sterling |
| 1.0 | 05/10/2010 | Ready to deliver | Sterling |
| 1.1 | 5/25/2010 | Updated based on feedback dated 5/20/2010 | Sterling |
| 1.2 | 6/1/2010 | Clean and Final Version | Sterling |
| 1.3 | 6/4/2010 | Added assumption regarding order business rules based on Steve’s feedback 20100602 | Sterling |
| 1.4 | 01/06/2011 | Updated with Change Request for multiple logins for OCI and cXML customers. It also includes new functionality around ‘require UNSPSC’, ‘replace characters’ and punchout maintenance screen layout. | Sterling |

Related or Reference Documents

| Document Name | Description | Owner | Location |
| --- | --- | --- | --- |
| SCI\_Xpedx Solution Definition Document v1.5 | Solution Definition document | Sterling Commerce |  |
| b2b-punchout\_cXML\_mapping\_v1.0.xlsx | cXML Setup Request/Response and Cart Punchout mapping document | Sterling Commerce/xpedx |  |
| b2b-punchout\_OCI\_mapping\_v2.1.xlsx | OCI Setup Request/Response and Cart Punchout mapping document | Sterling Commerce/xpedx |  |
| CallCtrCustIntegrationsScreen\_2010-12-01.xlsx | Call center maintenance screen for punchout | Sterling Commerce/xpedx | [JIRA - http://xpedxit.onjira.com/browse/XCNG-390] |
| Punchout Changes Next Gen v3.1docx.docx | Agreed upon requirements for Punchout CR | Sterling Commerce/xpedx | [JIRA - http://xpedxit.onjira.com/browse/XCNG-390] |
| Punchout Customers and XSLT Maps.xls | Customer specific xslts for certain OCI customers | Sterling Commerce/xpedx | [JIRA - http://xpedxit.onjira.com/browse/XCNG-390] |

TABLE OF CONTENTS

1. Introduction 6

1.1 Document Purpose 6

1.2 Document Audience 6

2 B2B Punchout 7

2.1 Functions & Solution 7

2.1.1 Setup at xpedx site 7

2.1.1.1 Customer Setup 7

2.1.1.2 Customer User Setup 8

2.1.1.3 Business Rules 8

*2.1.2* *Setup at customer site* 8

*2.1.3* *Customer user punchout flow* 8

*2.1.4* *Standards supported* 10

2.1.4.1 cXML/Ariba 10

2.1.4.2 OCI/SAP 10

2.1.5 Customer User Pages 10

2.1.6 Other functionality changes 10

2.1.6.1 B2B specific translations 10

2.1.6.2 Punchout Cart Management 11

2.1.6.3 Punchout Change Request Functionality 11

2.2 Master System 11

2.3 Implementation Details 11

2.3.1 Entity objects. 11

2.3.2 Actions involved and Functions 11

2.4 Process Flow 11

2.5 Field Mapping 12

2.5.1 cXML Setup Request/Response/Cart Punchout Mapping 12

2.5.2 OCI Cart Punchout Mapping 12

2.6 Schema 12

2.6.1 cXML Punchout Setup Schema (Customer to Sterling) 12

2.6.2 cXML Punchout Setup Response Schema (Sterling to customer) 13

2.6.3 cXML Cart Punchout Schema (Sterling to Customer) 14

2.6.4 Punchout Event Log Examples: 15

2.6.5 Punchout Authentication 16

2.6.6 OCI Cart Punchout Schema (Sterling to Customer) 16

2.7 Screen Shot 17

2.8 Open Questions 21

2.9 Assumptions 21

3 Connectivity Diagram 22

3.1 Master Catalog Connectivity Diagram 22

3.2 Connectivity Process 22

4 Glossary of Terms 23

5 Appendix 24

5.1 cXML Punchout Setup Request Sample 24

5.2 cXML Punchout Setup Response Sample 24

5.3 cXML Cart Punchout Sample 24

5.4 OCI Punchout Setup Sample 24

5.5 OCI Cart Punchout Sample 24

# Introduction

## Document Purpose

This document is the governing functional design document for the B2B Punchout functionality. It presents significant decisions and constructs used in developing the functionality. Testing, builds, configuration management are not covered in this document.

The document will also serve the purpose of keeping a list of assumptions that were made during design discussions.

## Document Audience

This document is intended for management and technical staff working on this project, xpedx IT and Business, webMethods, Legacy(MAX and ACCESS), HP, IW, xpedx/IP Network Team. Sterling will use the document during design and configuration for design consideration.

# B2B Punchout

## Functions & Solution

B2B Punchout is a standardized process and protocol for integrating a product catalog from xpedx into a customers’s procurement system. Punchout is part of the cXML standard supported by Ariba, but is similar in purpose and method to OCI (Open Catalog Interface), which is supported by SAP.

To enable a customer for punchout the solution at xpedx involves the following steps -

1. Setup at xpedx site
2. Setup at the customer site
3. Customer User Punchout Flow

## Setup at xpedx site

## Customer Setup

On xpedx.com, the Customer profile screen contains the following setup elements related to punchout. The screen has been finalized based on the Change Request and contains the following list of updated fields.

|  |  |
| --- | --- |
| **Setup Parameter** | **Description** |
| Network Identity | Used to identify the organization on the ariba network. This is used by customers that use the cXML format also. |
| Shared secret | Is used to authenticate the network id. Shared between xpedx and the customer. This is the password that is to be used for all the punchout users associated to that customer. Any change in this field means that Sterling will reset the passwords of the punchout users. |
|  |  |
|  |  |
| XSLT Filename | The name of the xslt file that is used to process the punchout cart before it is posted to the customer. |
| Path of the User ID | For cXML only- It is the xpath of the element that contains the user id that is to be used when processing the request as different customers send their user ids is different fields. The default path on the cXML for the email is “//Request/PunchOutSetupRequest/Contact/Email”. But in case this field has a value, then we look only in the xpath defined here. In majority of the cases, the user Id comes in as the value of the Extrinsic element that has the “name” attribute = “UserEmail”). E.g. <Extrinsic name="**UserEmail**">[abc@punchoutcustomer.com](mailto:abc@punchoutcustomer.com) </Extrinsic> |
| Show MyItems | A flag to indicate whether the MyItems link is enabled for the customer. |
| UNSPSC Code Required | For Both – cXML and OCI. A flag that indicates whether a punchout customer is able to add items to his punchout cart that do not have a UNSPSC code defined at the Master Catalog level. When this flag is set to 'True', we need to check for the presence of a UNSPSC code for this Item in the Master Catalog. The item can only be added to the cart if the UNSPSC is found in the catalog. Otherwise an appropriate message is displayed. |
| Replace Character | For both – cXML and OCI. It is a list of characters that the customer is unable to accept in the response. When any of these character is encountered, Sterling will remove it from the string. |
|  |  |
| OCI/SAP parameter flag | This flag indicates that this customer sends the actual user id/password that should be used for authentication for OCI is on different parameters (the standard parameters are “id” and “pwd”). |
| OCI/SAP Username Parameter | Name of the parameter that contains the actual userid to be used. (e.g. jdoe) |
| OCI/SAP User Email Template | This contains a value such as (@target.com). It is to be prefixed with the value of the parameter above (OCI/SAP Username Parameter) before attempting to authenticate (e.g. [jdoe@target.com](mailto:jdoe@target.com)) |
| OCI/SAP Pwd Parameter | The name of the parameter that contains the actual users password to be used. |
| Comments | This is a generic hold all text area that is for internal use by the users that have access to the Punchout Maintenance screen. No functionality is driven by this field. |

## Customer User Setup

`

Once the Customer/Ship To have been setup, we need to mark one or more users of the customer as a punchout user. This is managed on the profile of the users of that customer. The key fields to be used when tying up the incoming setup request to a user is done using the following two fields.

|  |  |
| --- | --- |
| **Setup Parameter** | **Description** |
| Punchout user flag | Identifies whether this customer user is enabled for punchout. |
| emailID | Uniquely identifies a punchout user in the system associated to the customer |

Assumption – multiple users per customer.

Assumption – this is setup as a regular customer user with appropriate permissions to ship tos. Additionally, a flag is enabled on the user profile to mark this user as a punchout user.

## Business Rules

Has this been covered in the order business rules document? Yes

## Setup at customer site

Customer sets up xpedx as a supplier in their procurement system based on parameters sent by xpedx such as network ID/shared secret/ URL to point to/etc.

## Customer user punchout flow

1. When a user wants more information or needs to configure a product, the user clicks on the xpedx link and is transferred - in a secure, logged in state - to the xpedx.com landing page.
2. From here on the browsing experience is identical to the regular customer user with some of the links such as managing profile information and checkout turned off. Based on the customer profile settings, a B2B customer user may be able to access MyItems lists as well.
3. Once the user has prepared the cart of items that they’d like to order, they can punch back into the procurement system and in the process transfer the cart back into the procurement system.

## Standards supported

## cXML/Ariba

The login setup for cXML is done via a two way handshake. The customer eProcurement system sends a setup request with the necessary information that identifies the user populated. In the Setup response, xpedx provides information such as the landing page for the punchout session. The Procurement system then redirects the customer user to this landing page.



## OCI/SAP

The login setup for OCI on the current dotcom is done through a HTTP GET with the URL parameters that contain all the information needed to setup the user’s session on xpedx.com. For a list of the parameters see the mapping section.

The customer system has to browse to a URL with the authentication information and the HOOK\_URL. Once the catalog browsing is done and the punch-out cart is submitted, a HTTP Form POST is done to the HOOK\_URL supplied by the customer.



## Customer User Pages

The customer user, once punched in has access to the following pages

Catalog Pages

Cart Pages – Ability to view old carts. add and edit cart.

My Items Page – based on punchout setup for customer

## Other functionality changes

## B2B specific translations

1. **UoM Replacement** – In addition to the regular UoM conversions on orders/carts, b2b has an additional translation of UoMs. In today’s dotcom a table stores customer replacement UoMs. This is a one-to-one replacement that is applied to outgoing carts after all standard UoM conversions. It is a simple replacement and involves no conversion factors.
2. **UNSPSC Replacement** – Certain customers define their own set of UNSPSC codes that they’d prefer to receive/send for items that they are purchasing. Similar to the UoM replacement table, there is a UNSPSC Replacement table stored in the current dotcom. The replacement of customer preferred UNSPSC code to Master data UNSPSC code is done on outbound punchout cart.
   1. This is done before we apply the OCI or cXML map to the cart output.
   2. The replacement logic picks up the legacy part # and xpedx UNSPSC and tries to find an entry in UNSPSC Replacement table.
   3. If found, the UNSPSC is replaced with the Customer UNSPSC from the above row.
   4. If not found, then the logic looks for a more generic row using just the xpedx UNSPSC and replaces it with Customer UNSPSC if found.
   5. All entries in this table are managed via USD tickets and there is no user interface to update / view the values.
   6. The content of this table is not fed to any other system at this point.
   7. Below is the list of Master Customers using UNSPSC replace currently:
      1. American Express (Ariba)(Punchout)
      2. Kaiser Foundation Hospitals - FS (PunchOut)
      3. Toys R Us (PunchOut)

## Punchout Cart Management

Punchout users will have access to their previously used carts via the cart list pages. Since b2b orders at xpedx are placed using a mechanism other than converting carts to orders, this list of punchout carts will become unmanageable. To resolve this, there will be a batch job that purges carts of punchout users that are more than a week old.

## Punchout Change Request Functionality

## Master System

N/A

## Implementation Details

## Entity objects.

## Actions involved and Functions

## Process Flow

## Field Mapping

The following mapping contains the fields that are sent as part of the requests from the customer to Sterling and the punchout response sent from Sterling to the customer. In the case of cXML, there is a setup handshake that OCI implemented at xpedx doesn’t support.

## cXML Setup Request/Response/Cart Punchout Mapping



## OCI Cart Punchout Mapping



## Schema

For samples of the following transactions, see the Appendix section.

## cXML Punchout Setup Schema (Customer to Sterling)

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE cXML SYSTEM "http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd">

<cXML

version=""

payloadID=""

timestamp=""

>

<Header>

<From>

<Credential domain="">

<Identity/>

</Credential>

</From>

<To>

<Credential domain="">

<Identity/>

</Credential>

</To>

<Sender>

<Credential domain="">

<Identity/>

</Credential>

<UserAgent />

</Sender>

</Header>

<Message>

<PunchOutOrderMessage>

<BuyerCookie/>

<PunchOutOrderMessageHeader operationAllowed="">

<Total>

<Money currency=""/>

</Total>

</PunchOutOrderMessageHeader>

<ItemIn quantity=""

lineNumber="">

<ItemID>

<SupplierPartID/>

<SupplierPartAuxiliaryID/>

</ItemID>

<ItemDetail>

<UnitPrice>

<Money currency=""/>

</UnitPrice>

<Description xml:lang="en-US"/>

<UnitOfMeasure/>

<Classification domain="UNSPSC"/>

</ItemDetail>

</ItemIn>

</PunchOutOrderMessage>

</Message>

</cXML>

## cXML Punchout Setup Response Schema (Sterling to customer)

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE cXML SYSTEM "http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd">

<cXML

payloadID=""

timestamp=""

>

<Response>

<Status

code=""

text=""/>

<PunchOutSetupResponse>

<StartPage>

<URL/>

</StartPage>

</PunchOutSetupResponse>

</Response>

</cXML>

## cXML Cart Punchout Schema (Sterling to Customer)

<?xml version="1.0" encoding="utf-8"?>

<!DOCTYPE cXML SYSTEM "http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd">

<cXML

version=""

payloadID=""

timestamp=""

>

<Header>

<From>

<Credential domain="">

<Identity/>

</Credential>

</From>

<To>

<Credential domain="">

<Identity/>

</Credential>

</To>

<Sender>

<Credential domain="">

<Identity/>

</Credential>

<UserAgent />

</Sender>

</Header>

<Message>

<PunchOutOrderMessage>

<BuyerCookie/>

<PunchOutOrderMessageHeader operationAllowed="">

<Total>

<Money currency=""/>

</Total>

</PunchOutOrderMessageHeader>

<ItemIn

quantity=""

lineNumber=""

>

<ItemID>

<SupplierPartID/>

<SupplierPartAuxiliaryID/>

</ItemID>

<ItemDetail>

<UnitPrice>

<Money currency=""/>

</UnitPrice>

<Description xml:lang=""/>

<UnitOfMeasure/>

<Classification domain=""/>

</ItemDetail>

</ItemIn>

</PunchOutOrderMessage>

</Message>

</cXML>

## Punchout Event Log Examples:

**1-Punchout Setup Request Example:**

<?xml version = '1.0' encoding = 'UTF-8'?> <!DOCTYPE cXML SYSTEM "<http://xml.cxml.org/schemas/cXML/1.1.007/cXML.dtd>"> <cXML version="1.1.007" xml:lang="en-US" payloadID="Tue May 25 10:51:06 PDT 2010" timestamp="Tue May 25 10:51:06 PDT 2010">    <Header>       <From>          <Credential domain="DUNS">             <Identity>006985808</Identity>          </Credential>       </From>       <To>          <Credential domain="DUNS">             <Identity>006985808</Identity>          </Credential>       </To>       <Sender>          <Credential domain="Oracle Exchange ">             <Identity>exchange.oracle.com</Identity>             <SharedSecret>xxxxxxx</SharedSecret>          </Credential>          <UserAgent>Oracle Exchange </UserAgent>       </Sender>    </Header>    <Request>       <**PunchOutSetupRequest** operation="create">          <BuyerCookie>12345678</BuyerCookie>          <Extrinsic name="User">BMGR8874</Extrinsic>          <BrowserFormPost>             <URL><http://tru-ip-orafinap.tru.com:8005/OA_HTML/OA.jsp?OAFunc=ICX_CAT_PUNCHOUT_CALLBACK&#38;OAHP=ICX_POR_HOMEPAGE_MENU&#38;OASF=ICX_CAT_PUNCHOUT_CALLBACK&#38;transactionid=1867600726</URL>>          </BrowserFormPost>          <Contact>             <Name xml:lang="en-US">BRU 8874, CATONSVILLE MD</Name>             <Email>[BMGR8874@TOYSRUS.COM</Email](mailto:BMGR8874@TOYSRUS.COM%3c/Email)>          </Contact>          <SupplierSetup>             <URL>[https://punchout.xpedx.com/punchout.aspx</URL](https://punchout.xpedx.com/punchout.aspx%3c/URL)>          </SupplierSetup>       </PunchOutSetupRequest>    </Request> </cXML>

**2- Punchout Setup Response Example:**

<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE cXML SYSTEM "[http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd"><cXML](http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd) payloadID="none" timestamp="5/25/2010 12:51:11 PM"><Response><Status code="200" text="OK" /><**PunchOutSetupResponse**><StartPage><URL>[https://punchout.xpedx.com/?id=BMGR8874@toysrus.com&amp;pwd=xxxxxxx&amp;punchoutidentity=006985808&amp;FromCredentialIdentity=006985808&amp;SenderCredentialIdentity=exchange.oracle.com&amp;buyercookie=12345678&amp;hook\_url=http%3a%2f%2ftru-ip-orafinap.tru.com%3a8005%2fOA\_HTML%2fOA.jsp%3fOAFunc%3dICX\_CAT\_PUNCHOUT\_CALLBACK%26amp%3bOAHP%3dICX\_POR\_HOMEPAGE\_MENU%26amp%3bOASF%3dICX\_CAT\_PUNCHOUT\_CALLBACK%26amp%3btransactionid%3d1867600726</URL></StartPage></PunchOutSetupResponse></Response></cXML](https://punchout.xpedx.com/?id=BMGR8874@toysrus.com&amp;pwd=xxxxxxx&amp;punchoutidentity=006985808&amp;FromCredentialIdentity=006985808&amp;SenderCredentialIdentity=exchange.oracle.com&amp;buyercookie=12345678&amp;hook_url=http%3a%2f%2ftru-ip-orafinap.tru.com%3a8005%2fOA_HTML%2fOA.jsp%3fOAFunc%3dICX_CAT_PUNCHOUT_CALLBACK%26amp%3bOAHP%3dICX_POR_HOMEPAGE_MENU%26amp%3bOASF%3dICX_CAT_PUNCHOUT_CALLBACK%26amp%3btransactionid%3d1867600726%3c/URL%3e%3c/StartPage%3e%3c/PunchOutSetupResponse%3e%3c/Response%3e%3c/cXML)>

**3- PunchoutOrderMessage Example** (response sent back in hidden field named "cxml-urlencoded" and posted on a form)

<FORM name='Form1' method='post' action='http://tru-ip-orafinap.tru.com:8005/OA\_HTML/OA.jsp?OAFunc=ICX\_CAT\_PUNCHOUT\_CALLBACK&OAHP=ICX\_POR\_HOMEPAGE\_MENU&OASF=ICX\_CAT\_PUNCHOUT\_CALLBACK&transactionid=1867600726'><input name='cxml-urlencoded' type='hidden' value='<?xml version="1.0" encoding="utf-8"?><!DOCTYPE cXML SYSTEM "[http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd"><cXML](http://xml.cXML.org/schemas/cXML/1.2.014/cXML.dtd) version="1.2.014" payloadID="none" timestamp="2010-52-25T12:52:07-CST"><Header><From><Credential domain="NetworkId"><Identity>006985808</Identity></Credential></From><To><Credential domain="NetworkId"><Identity>006985808</Identity></Credential></To><Sender><Credential domain="DUNS"><Identity>exchange%2Eoracle%2Ecom</Identity></Credential><UserAgent /></Sender></Header><Message><**PunchOutOrderMessage**><BuyerCookie>12345678</BuyerCookie><PunchOutOrderMessageHeader operationAllowed="edit"><Total><Money currency="USD">226.56</Money></Total></PunchOutOrderMessageHeader><ItemIn quantity="12" lineNumber="1"><ItemID><SupplierPartID>711855</SupplierPartID><SupplierPartAuxiliaryID>1014035</SupplierPartAuxiliaryID></ItemID><ItemDetail><UnitPrice><Money currency="USD">18.88</Money></UnitPrice><Description xml:lang="en-US">Buff Shopping Bag L 24X6X30 Co-Branded (250 Each/Case)</Description><UnitOfMeasure>CT</UnitOfMeasure><Classification domain="UNSPSC">53121608</Classification></ItemDetail></ItemIn></PunchOutOrderMessage></Message></cXML>'/></FORM>

After this step, the session is ended.

## Punchout Authentication

We support two types of punchout integrations (OCI using html form post, and XML over http). Each type of integration one has its own authentication mechanism.

1- OCI method: in this setup, the userid, pwd, punchoutidentity, hook\_url, etc are all passed via url querystring variables using form post. The data is posted directly to the b2b home page and special handling is provided when the request is a punchout request. If the userid and pwd are valid, the user is authenticated and redirected to the requested page. if not, an error is displayed on the logon page.

2- XML over https: in this setup the authentication information is submitted inside an XML document which is parsed during the PunchoutSetupRequest phase. The request is submitted to a special page (Punchout.aspx) that expects an xml document. Several nodes in the xml request are validated including userid, pwd, punchoutidentity, secret, etc.

First, the application checks if the punchoutidentity and shared secret are valid. If so, it checks if useremail supplied is valid. If so, the PunchoutSetupRequest is authorized and a start page URL (retrieved from the mastercustomer database record) is returned to the calling system. The useremail can either be specified in the " //Request/PunchOutSetupRequest/Contact/Email" node (see cXML.dtd) or inside an extrinsic value specified on the mastercustomer record.

## OCI Cart Punchout Schema (Sterling to Customer)

NEW\_ITEM-DESCRIPTION[1]

NEW\_ITEM-MATNR[1]

NEW\_ITEM-MATGROUP[1]

NEW\_ITEM-QUANTITY[1]

NEW\_ITEM-UNIT[1]

NEW\_ITEM-PRICE[1]

NEW\_ITEM-PRICEUNIT[1]

NEW\_ITEM-CURRENCY[1]

NEW\_ITEM-LEADTIME[1]

NEW\_ITEM-VENDOR[1]

NEW\_ITEM-VENDORMAT[1]

NEW\_ITEM-CUSTOMER\_SKU[1]

NEW\_ITEM-MANUFACTURER\_SKU[1]

NEW\_ITEM-CONTRACT[1]

NEW\_ITEM-CONTRACT\_ITEM[1]

NEW\_ITEM-SERVICE[1]

NEW\_ITEM-EXT\_QUOTE\_ID[1]

NEW\_ITEM-EXT\_QUOTE\_ITEM[1]

NEW\_ITEM-EXT\_PRODUCT\_ID[1]

NEW\_ITEM-LONGTEXT\_1:132[]

NEW\_ITEM-EXT\_SCHEMA\_TYPE[1]

NEW\_ITEM-EXT\_CATEGORY\_ID[1]



## Screen Shot

[TBD] Insert final screenshots from xpedx into the catalog portions.

***Customer Setup Page***

This will include elements from the current Punchout Maintenance and Master Customer Maintenance Pages. The element relevant to punchout from the Master Customer Maintenance Page is the WebMethods Customer Id. This Id comes across on the punchout message to identify the customer.

There will be a new set of flags on this page that indicate whether this customer’s punchout request is in cXML or OCI format.

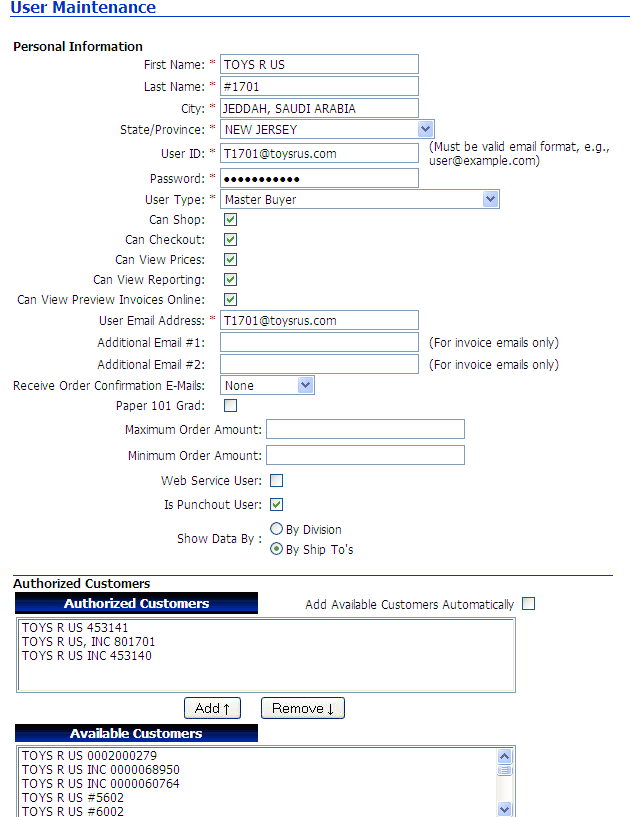
Web Methods Customer ID is what the customer sends as the Buyer ID on B2B transactions, which links it to the xpedx.com master customer.

The finalized maintenance screen is shown below

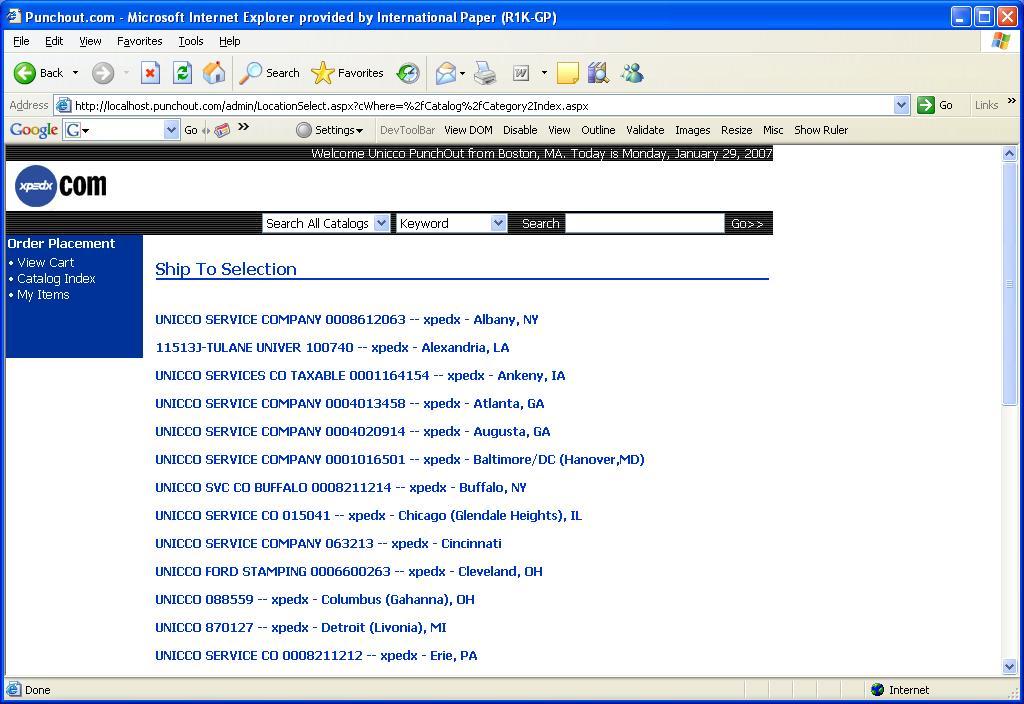


***Customer User Setup Page***

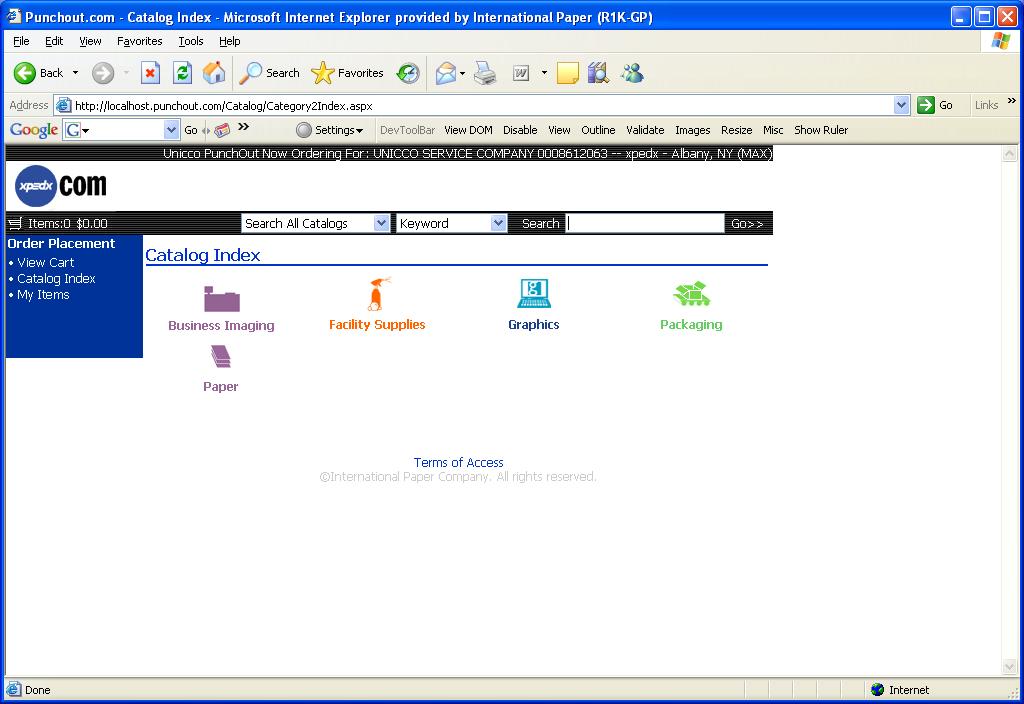
The relevant elements specific for the punchout are the flag indicating that this is a punchout customer as well as the User Email Address field that is used to identify which user is attempting to log in.

******

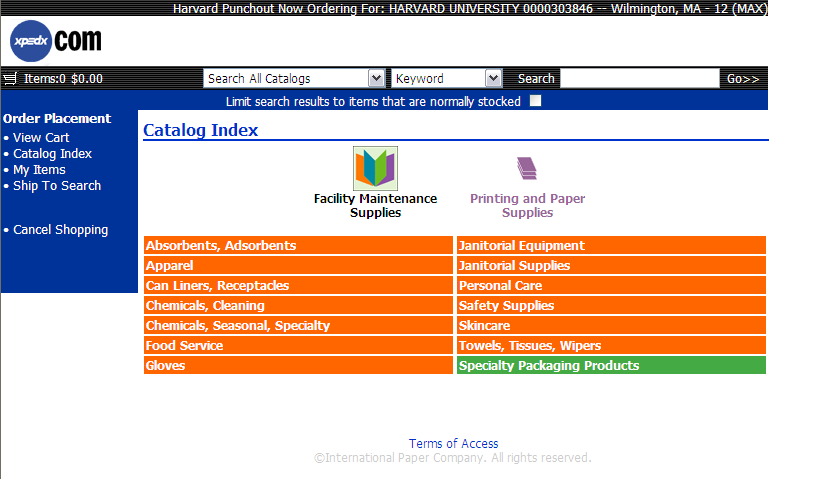
***Customer User Landing Page***



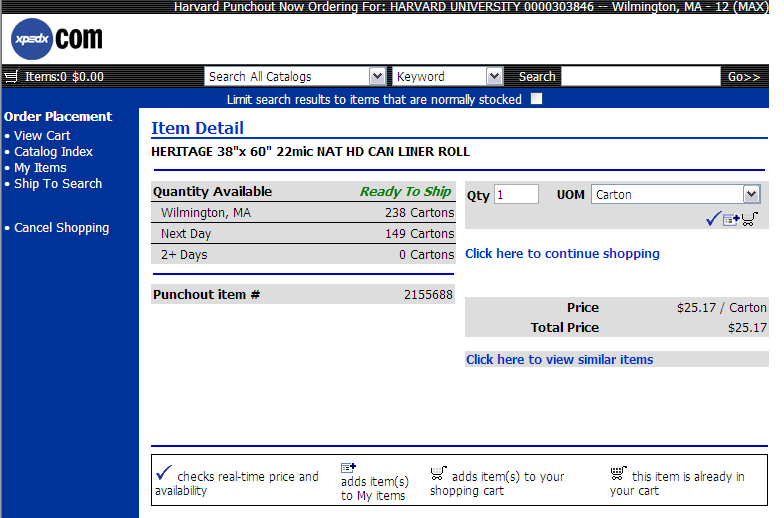
***Customer User Catalog Home Page***



***Customer User Catalog Browsing Page***



***Customer User Item Detail Page***



## Open Questions

1. How many variations of cXML login are we going to support (ootb -1)? [Prashant – Sterling has proposed using only login method. This may require changes on the customer side. xpedx to confirm process]
2. How many variations of OCI/cXML output are we going to support (ootb -1)? Need further discussion as this will require customer specific output formats that are currently not supported in Sterling. [Whitlock: We use different variations of the OCI map based on customer requirements; need to understand how we will modify OCI maps and how we will integrate the various OCI maps already in use?] There were initially two xslt files created to support SAP and standard xml punchout. This list grew based on different customer requirements. There are currently 5 variations of the xslt files.
3. Do any customer use MPC during B2B transactions? Resolved. [Bugher: should not be used for Punchout since the customer is using our Legacy part number (or their customer part number if configured) from the Cart and will send that in the PO]
4. Connectivity from Customer to Sterling. More importantly Sterling to Customer on cart punch-out.
5. Process to handle rogue orders ? Resolved. This is covered in the B2B Transactions document.
6. Screenshots for catalog, cart and myitems pages for the punchout user.
7. How is the National Account Number used on Master Customer Maintenance ? Resolved. Not related to punchout. [Whitlock: National Account number on the Master Customer Maintenance Screen has no role for punchout users, it’s only used to link Preview invoices]
8. How is the BuyerId/wM Customer Id used? Is it relevant to punchout ? Resolved. Not relevant to B2B Punchout. [Whitlock: The Buyer ID and wM Custoemr ID are the same and it is how Web Methods links to the master account. No it’s not relevant only if the customer sends the order electronically.]
9. Have the business rules been covered in the Order Business Rules document ? Resolved. [Whitlock: yes previous B2B JAD Sessions covered all applicable business rules.]
10. Is there a requirement to store and use original price? Resolved. [Bugher: I don’t think so]
11. The b2b specific conversions for UoM replacement and UNSPSC – what is the data source and how is it managed? Resolved. [Colin: The data source is a table in .com. There system of record according to George is xpedx.com. There is no GUI on the site to enter this.]
12. What is the “Start Page URL” shown in the Call Center Customer Integration Screen xls ? How does it get populated into Sterling and how is it used?

## Assumptions

1. No editing existing carts by punching in directly to cart. Only possible by finding it in recent carts.
2. This document has been prepared with the assumption that we will be supporting one standard method of login using cXML and one link based login for OCI.
3. The login method for OCI implemented in the current dotcom doesn’t follow the standards for setup of the punchout session. It is assumed that this process will be carried forward as is for NG.
4. Pricing Rules/Coupons are not applicable for punchout customers. Sterling ootb capability only support targeting by customer not user.
5. UNSPSC and UoM Replacements are done only on incoming/outgoing messages. The browsing experience still continues to use the standard values as a customer logging on directly into xpedx.com would see.
6. B2B Punchout sessions will invoke the same business rules as Web Channel sessions as applies to orders, for example, minimum order charges.
7. The presence or absence of fields from a customers punchout cart is governed by the XSLT provided. Currently we have been given a superset of all the fields that may exist on the outgoing message (contained in Section 2.5.2. The expectation is that if there is anything customer specific, a new XSLT will be created and modified to meet customer requirements.
8. Sterling doesn’t support defaulting punchout sessions to the first punchout user. The correct credentials need to be sent on the setup messages.

# Connectivity Diagram

## Master Catalog Connectivity Diagram

[TBD – Insert Connectivity Diagram from Tim]

## Connectivity Process

1. Will Sterling be talking directly to the customers via firewalls, or will there be some connectivity portion before punchout requests hit the site?

# Glossary of Terms

|  |  |  |
| --- | --- | --- |
| S. No. | Term | Definition |
| 1. | WSDL | Web Services Definition Language |
| 2. | UE (User Exit) | Hooks to write custom code in Sterling |
| 3. | MQ | Message Queue |
| 4. | BR1 | Business Release 1 |
| 5. | IW | Industrial Wisdom – UI firm engaged on the project. |
| 6. | UoM | Unit of Measure |
|  |  |  |

# Appendix

## cXML Punchout Setup Request Sample



## cXML Punchout Setup Response Sample



## cXML Cart Punchout Sample

******

## OCI Punchout Setup Sample

<http://punchoutstg.xpedx.com/?id=xxxx@punchout.com&pwd=Punchout123&hook_url=cartinfo.aspx>

## OCI Cart Punchout Sample

