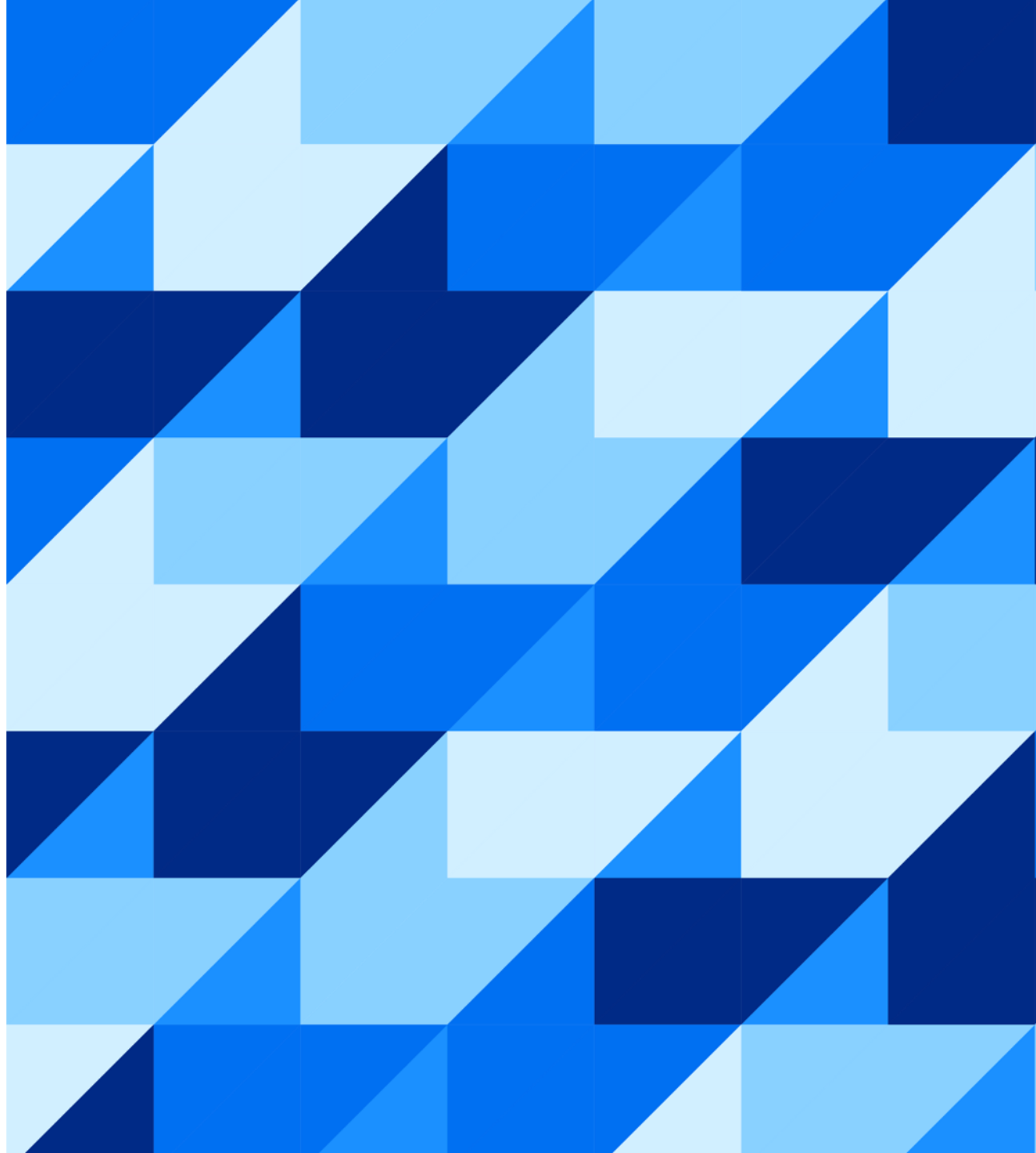




Introduction to PunchOut[®] Catalogs & Account Setup

PUBLIC



Contents

The Catalog Project

- [What Is A Catalog Project?](#)
- [Catalog Project Flow?](#)

Introduction to PunchOut Catalogs

- [What Is A PunchOut Catalog?](#)
- [How Does It work?](#)
- [Sample cXML Messages](#)
- [Understanding The cXML Message Flow](#)
- [Closing the PunchOut Session](#)
- [Catalog Tester](#)

Supplier Account Setup

- Test Account Creation/Management
 - [What is a Test Account?](#)
 - [Creating a Test Account](#)
 - [Switching to the Test Account](#)
- [Creating a PunchOut Connection](#)
- [Native cXML Transaction Setup](#)

Additional Resources

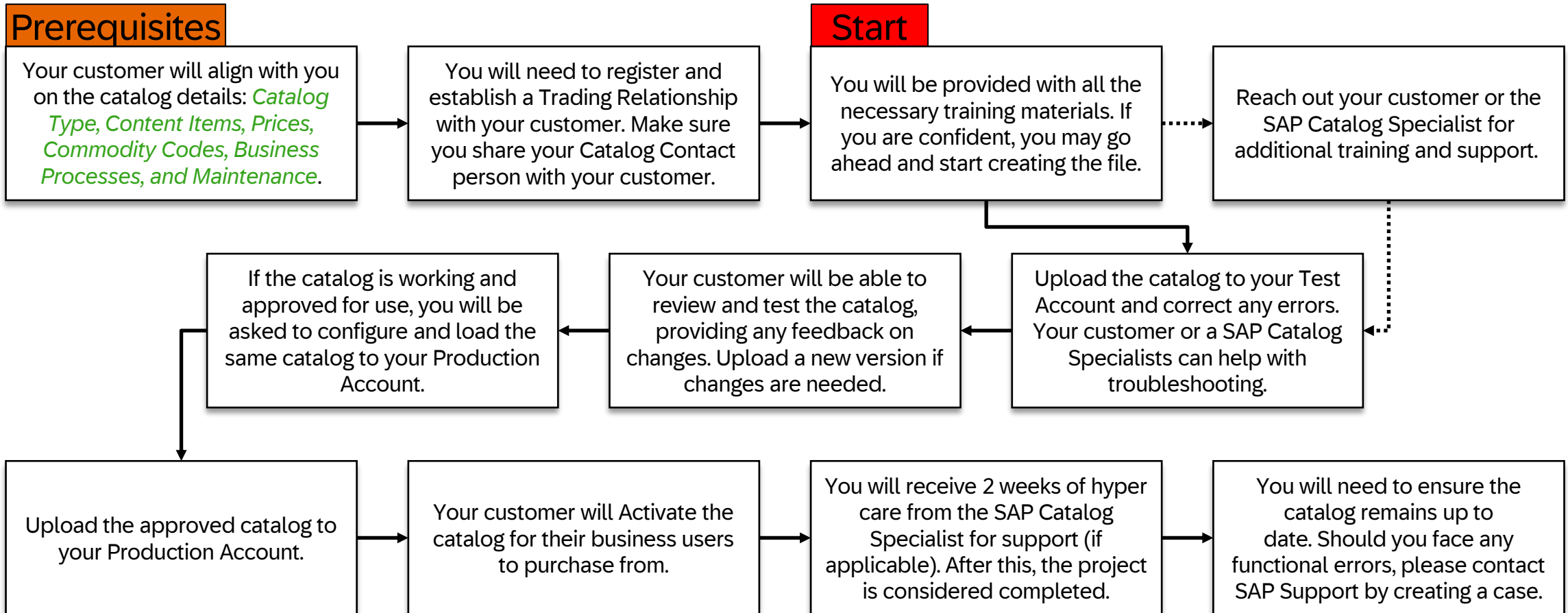
- [Links to Additional Resources](#)

The Catalog Project

What Is A Catalog Project?

- Your customer is using the SAP Ariba Procurement solution to help streamline their purchasing processes.
- As part of this initiative, your organization has been identified as a candidate to provide, manage, and maintain a catalog for them.
- The end goal is to allow your customer to efficiently purchase items from your catalog within their SAP Ariba Procurement application, aiming to see an increase in order accuracy and processing times, and spend compliance.
- You will be able to work with your customer and SAP teams to get support in creating, loading, and troubleshooting your catalog files in the SAP Business Network. Once the catalog is loaded and activated, you will need to ensure it remains updated for your customer.

Catalog Project Flow



Introduction to PunchOut[®] Catalogs

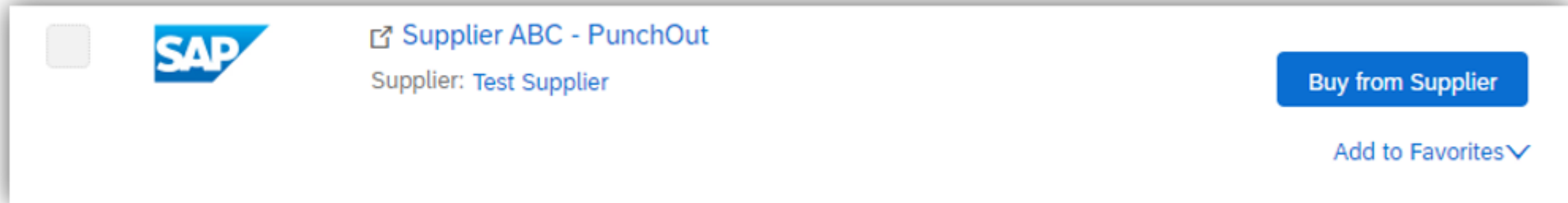
What Is A PunchOut Catalog?

- PunchOut catalogs utilize a supplier's existing ecommerce website/store. A connection is created between the SAP Business Network and the supplier's website. This allows the customer's user to 'punch out' to the site, browse and shop, then return their selected items to their SAP Ariba Procurement application where they will complete the requisition process. After processing and approving the requisition, it becomes a purchase order which is then sent to the supplier.
- For Level 1 (L1) PunchOut catalogs, the User does all their shopping on the Supplier's website. In a Level 2 (L2) scenario, the User shops using the Ariba Procurement catalog database and the Supplier's site.



What Is A PunchOut Catalog?

- Once you have worked through the PunchOut Catalog project, your customer will be able to access your PunchOut catalog via the generated Catalog file.
- Your customer will be able to see your company name and logo within their system.



- Clicking on the 'Buy From Supplier' button will take them to your interactive website.

How Does It Work?

- PunchOut uses the **cXML** protocol, based on XML and created by SAP Ariba.
- cXML is freely available and is well documented for Suppliers.
- Details for PunchOut catalog syntax, DTD listings, guides and other information is available for download at:

www.cxml.org

- Catalog-specific information is available in the chapters titled “*PunchOut site planning*” and “*PunchOut transactions*” in the **cXML Solutions Guide** and the chapter titled “*PunchOut transactions*” in the **cXML User’s Guide**.



How Does It Work?

- A special catalog file called a PunchOut Index file is loaded or generated in the SAP Business Network Supplier account. This file will instruct the system to allow the catalog to connect with the Supplier's site for shopping.
- PunchOut uses cXML to connect and transfer information from the SAP Ariba Procurement application to the Supplier and back. There are 3 cXML documents used by PunchOut to accomplish this:
 - **PunchOutSetupRequest** (POSR) – The POSR identifies the buyer and sends authentication (basically name and password) to the supplier's website to establish a connection.
 - **PunchOutSetupResponse** – The Response document sent from the Supplier tells the SAP Ariba Procurement application what website or items on the site to display for the user.
 - **PunchOutOrderMessage** (POOM) – The POOM converts the Shopping Cart on the Supplier's site and sends the cart or item information back to the SAP Ariba Procurement solution to be submitted for purchase.

Sample cXML Messages: PunchOutSetupRequest - Header

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/version/cXML.dtd">
<cXML payloadID="0012345@www.buyer.com" timestamp="2025-06-21T12:57:09-07:00">
  <Header>
    <From>
      <Credential domain="NetworkID">
        <Identity>AN01999999999</Identity>
      </Credential>
    </From>
    <To>
      <Credential domain="NetworkID">
        <Identity>AN01234567891</Identity>
      </Credential>
    </To>
    <Sender>
      <Credential domain="AribaNetworkUserId">
        <Identity>sysadmin@ariba.com</Identity>
        <SharedSecret>*****</SharedSecret>
      </Credential>
      <UserAgent>Ariba Buyer 8.2</UserAgent>
    </Sender>
  </Header>
```

The ANID in the **<From>** section will change per customer to reflect their SAP Business Network Account ID. It will differ in the Production or Test accounts. Example: AN01999999999 (Prod) or AN01999999999-T (Test)

The ANID in the **<To>** section will reflect your SAP Business Network Account. It will differ in your Production or Test accounts.

The **<Sender>** section will pass the **<Identity>** **sysadmin@ariba.com** which can be used to identify the request coming from SAP Ariba.

The **<SharedSecret>** is a private “password” between your SAP Business Network Account and your back-end system.
Do not share this with your customers!

Sample cXML Messages: PunchOutSetupRequest - Body

```

<Request>
  <PunchOutSetupRequest operation="create">
    <BuyerCookie>CK007</BuyerCookie>
    <Extrinsic name="CostCenter">928</Extrinsic>
    <Extrinsic name="UserEmail">catalog_tester@buyer.com</Extrinsic>
    <Extrinsic name="UniqueName">catalog_tester</Extrinsic>
    <BrowserFormPost>
      <URL>https://service.ariba.com/CatalogTester.aw/10000007</URL>
    </BrowserFormPost>
    <SupplierSetup>
      <URL>https://www.supplier.com/b2b/sappunchout</URL>
    </SupplierSetup>
    <ShipTo>
      <Address addressID="001">
        <Name xml:lang="en-US">Catalog Tester</Name>
        <PostalAddress>
          <DeliverTo>Catalog Tester</DeliverTo>
          <Street>1234 Catalog Tester Way</Street>
          <City>Any Town</City>
          <State>CA</State>
          <PostalCode>12345</PostalCode>
          <Country isoCountryCode="US">United States</Country>
        </PostalAddress>
      </Address>
    </ShipTo>
    <SelectedItem>
      <ItemID>
        <SupplierPartID>PunchOut007</SupplierPartID>
        <SupplierPartAuxiliaryID/>
      </ItemID>
    </SelectedItem>
  </PunchOutSetupRequest>
</Request>
</cXML>

```

The information passed on the PunchOutSetupRequest is combination of standard fields and customer specific values.

The fields highlighted in **red** will always be present.

The fields in **black** are subject to change with each customer and PunchOut instance.

Not all customers will be able to pass information in the fields in **black**.

Sample cXML Messages: PunchOutSetupResponse

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.2.014/cXML.dtd">
<cXML xml:lang="en-US" payloadID="6543210" timestamp="2025-06-21T12:58:09-07:00">
  <Response>
    <Status code="200" text="success"></Status>
    <PunchOutSetupResponse>
      <StartPage>
        <URL>https://www.supplier.com/b2b/customer/sessionid=7006</URL>
      </StartPage>
    </PunchOutSetupResponse>
  </Response>
</cXML>
```

The PunchOutSetupResponse should contain a clear status code and text description. This will help when troubleshooting connectivity issues.

The StartPage URL is the landing page that will load for the customer.

Depending on the PunchOut Level 1 or Level 2, this may take the user to the Home, Aisle, Shelf, or Product page.

Sample cXML Messages: PunchOutOrderMessage - Header

```
<?xml version="1.0"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/version/cXML.dtd">
<cXML payloadID="0012345@www.supplier.com" timestamp="2025-06-21T12:59:49-07:00">
  <Header>
    <From>
      <Credential domain="NetworkID">
        <Identity>AN01234567891</Identity>
      </Credential>
    </From>
    <To>
      <Credential domain=" NetworkID">
        <Identity>AN01999999999</Identity>
      </Credential>
    </To>
    <Sender>
      <Credential domain="www.supplier.com">
        <Identity>PunchOutOrderMessage</Identity>
      </Credential>
      <UserAgent>Supplier PunchOut Site V18.1</UserAgent>
    </Sender>
  </Header>
```

The PunchOutOrderMessage is sent from the Supplier to the Buyer.

One of the notable changes in the header is the **<From>** and **<To>** credentials being reversed.

Sample cXML Messages: PunchOutOrderMessage - Body

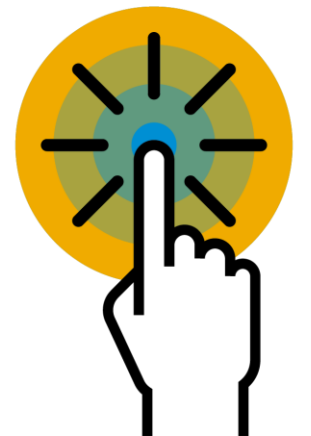
```
<Message>
  <PunchOutOrderMessage>
    <BuyerCookie>1EPRED9RSTRPQ</BuyerCookie>
    <PunchOutOrderMessageHeader operationAllowed="edit">
      <Total>
        <Money currency="USD">16.99</Money>
      </Total>
    </PunchOutOrderMessageHeader>
    <ItemIn quantity="1">
      <ItemID>
        <SupplierPartID>AGT007</SupplierPartID>
        <SupplierPartAuxiliaryID>L2K</SupplierPartAuxiliaryID>
      </ItemID>
      <ItemDetail>
        <UnitPrice>
          <Money currency="USD">16.99</Money>
        </UnitPrice>
        <Description xml:lang="en-US">Blue Ballpoint Pen - Gold</Description>
        <UnitOfMeasure>EA</UnitOfMeasure>
        <Classification domain="UNSPSC">44121704</Classification>
      </ItemDetail>
      <SupplierID domain="NetworkID"> AN01234567891</SupplierID>
    </ItemIn>
  </PunchOutOrderMessage>
</Message>
</cXML>
```

The body of the PunchOutOrderMessage contains all the cart information and the product details.

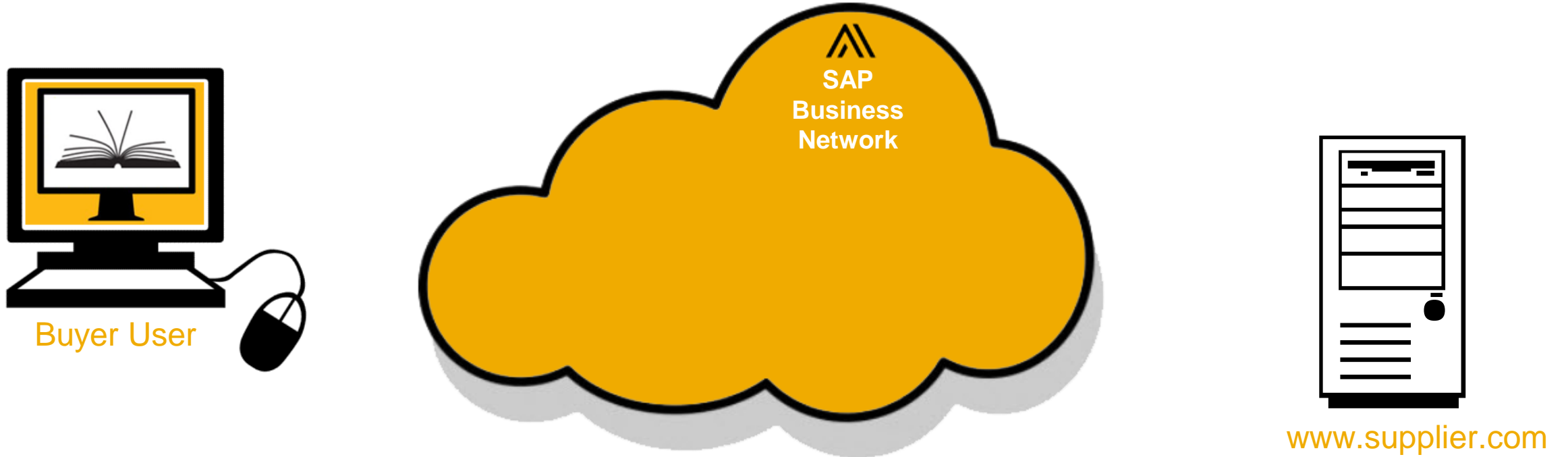
Your customer may request additional information to be passed back to them, please check with your customer or SAP Catalog Specialist for additional guidance.

Understanding The cXML Message Flow

- The steps in the next section shows how a simple **Level 1** PunchOut catalog works and the role the cXML messages play.
- The User **chooses** the Supplier's **catalog link** in the **SAP Ariba catalog** and is taken to the **Suppliers site** where they **shop**. The User then **returns** the selected items back to their **SAP Ariba Procurement application**.
- Once the **items** are back in the SAP Ariba Procurement application, the **User** can **submit** the **Cart** to become a **Requisition**, and ultimately a **Purchase Order**, which is sent to the Supplier.

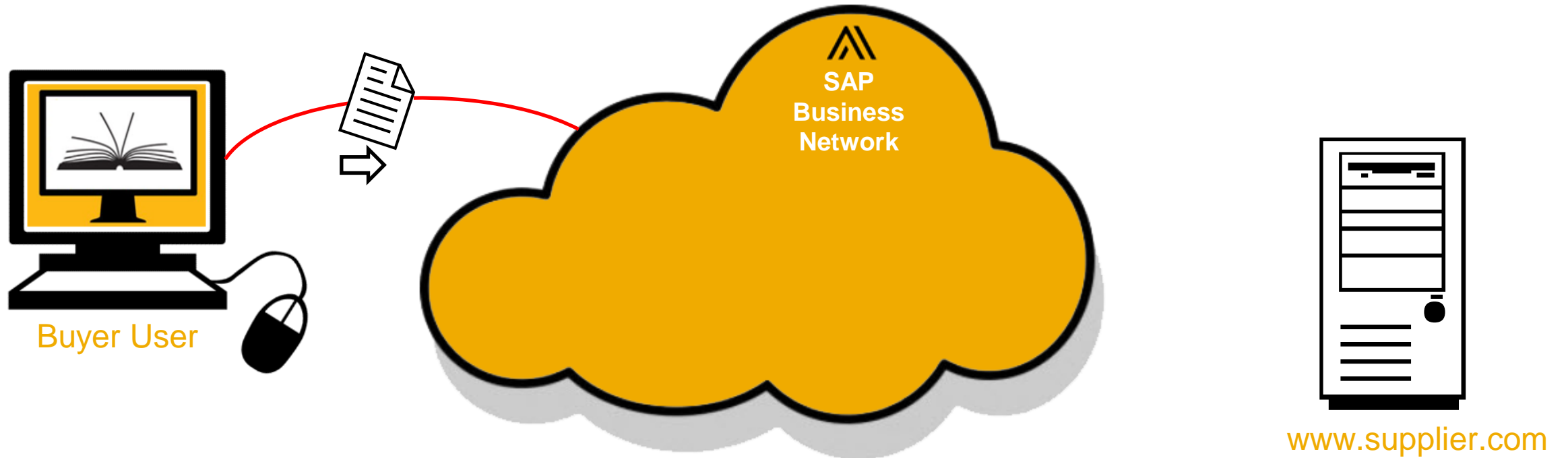


Understanding The cXML Message Flow



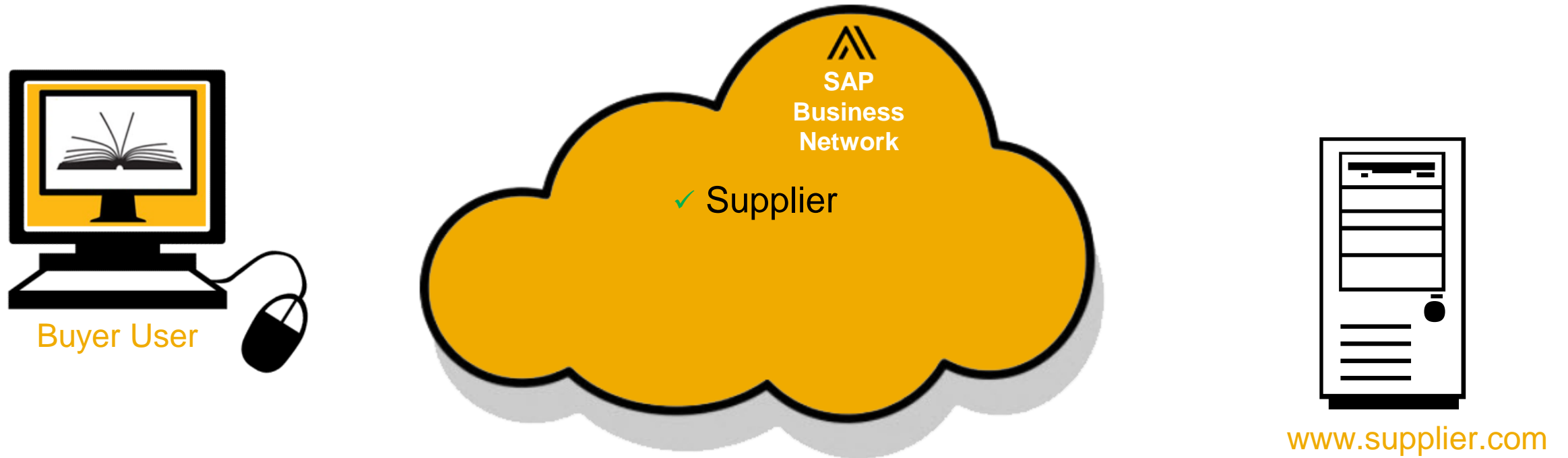
The Buyer User finds the Supplier's link in their SAP Ariba Procurement solution's catalog and clicks "Buy from Supplier."

Understanding The cXML Message Flow



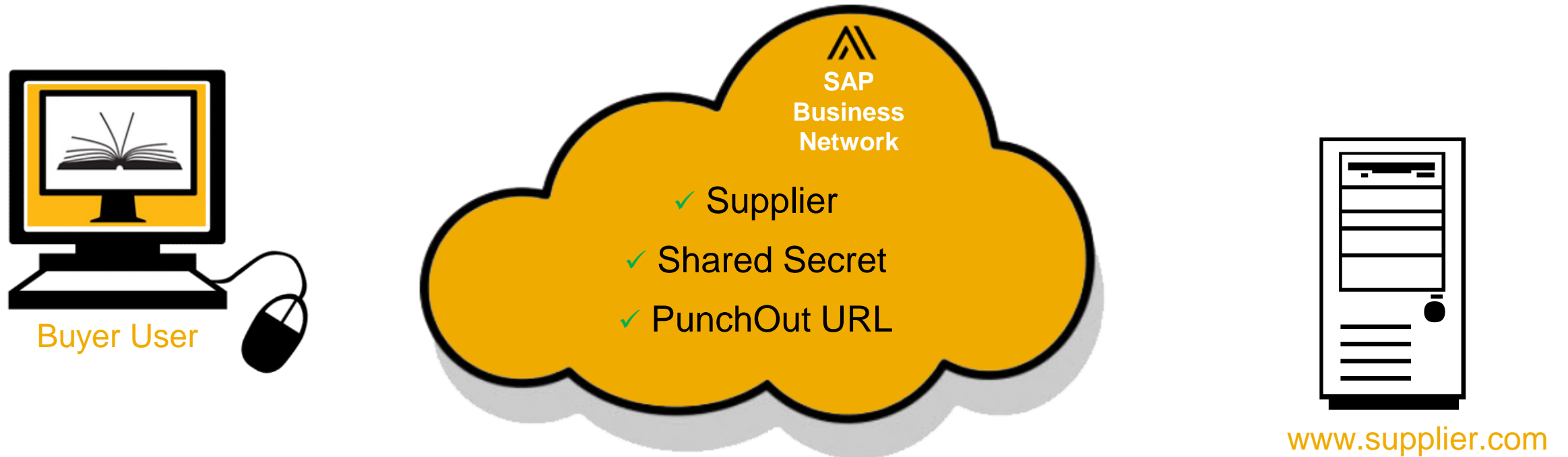
The **PunchOut Setup Request (POSR)** is routed from the SAP Ariba Procurement solution to the SAP Business Network.

Understanding The cXML Message Flow



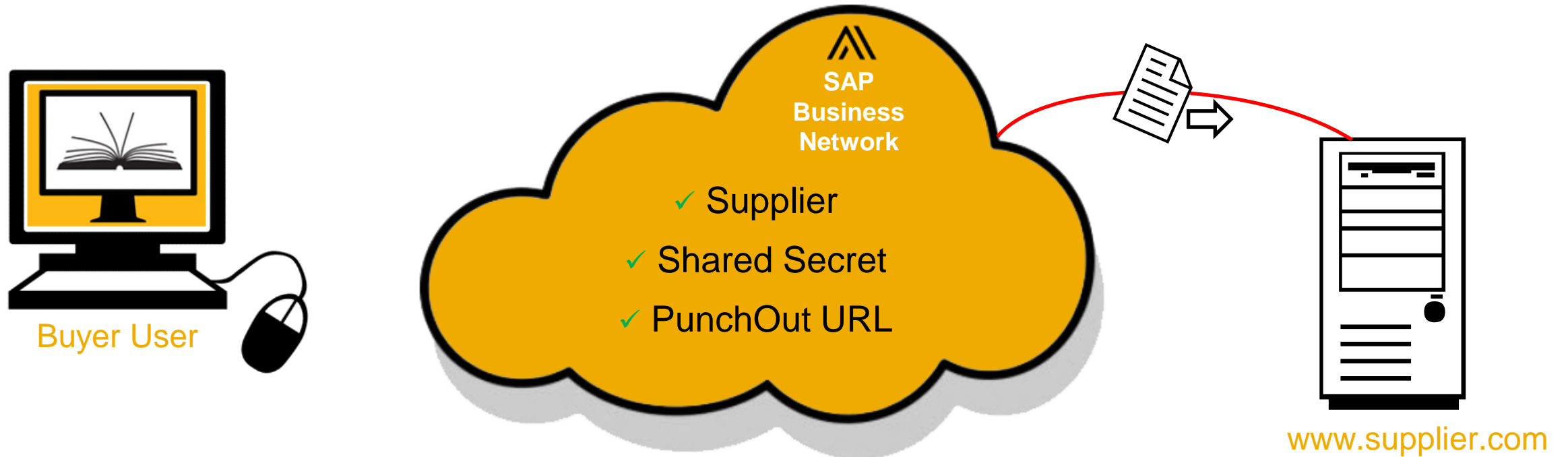
The SAP Business Network looks up the ANID in the **<To>** element of the **POSR** and finds the Supplier record.

Understanding The cXML Message Flow



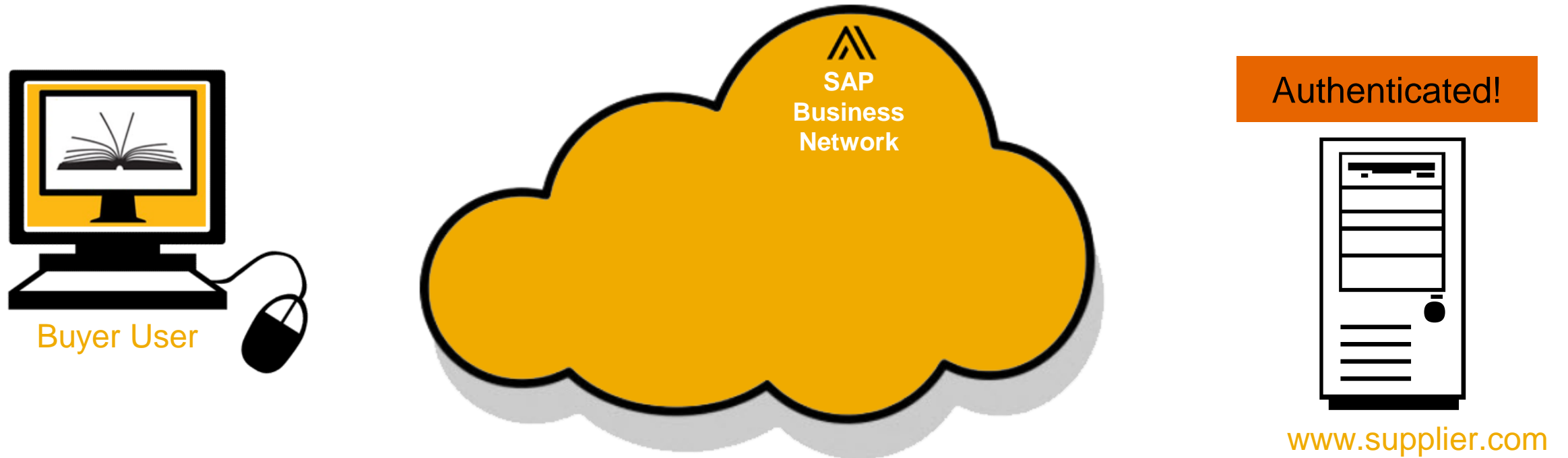
In the Business Network Supplier record, the Supplier's Shared Secret is mapped to the **POSR** and then it looks up the Supplier's PunchOut URL.

Understanding The cXML Message Flow



The **POSR** is then routed and presented to the Supplier's specified PunchOut URL.

Understanding The cXML Message Flow



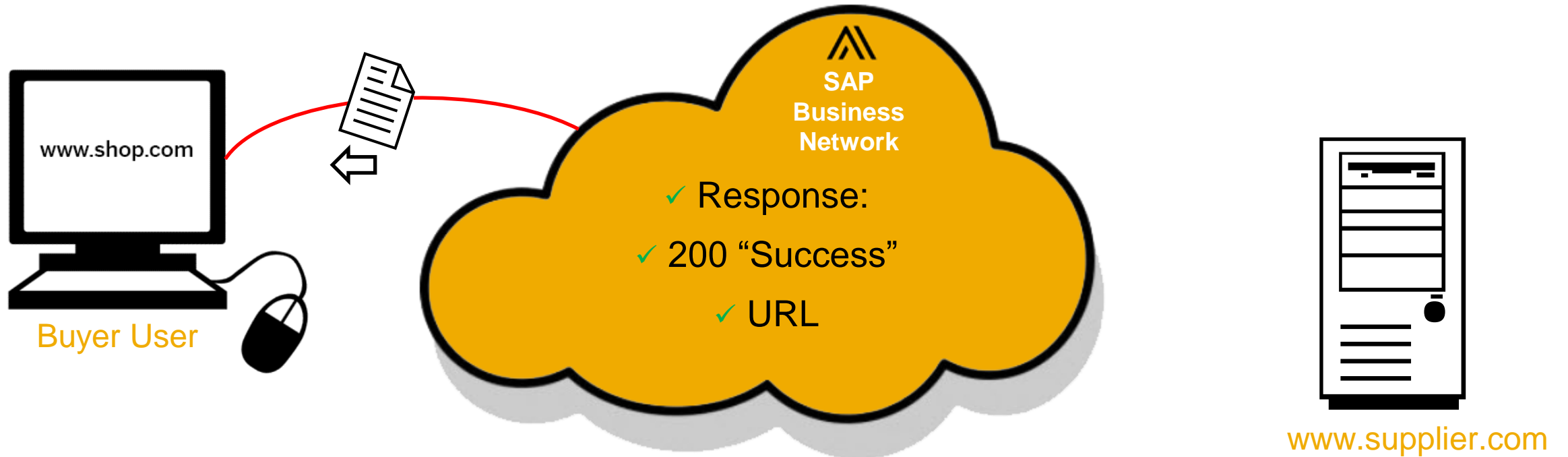
The Supplier authenticates the **POSR** using the **<From>** ANID (as a User Name) and their own Shared Secret (as the Password).

Understanding The cXML Message Flow



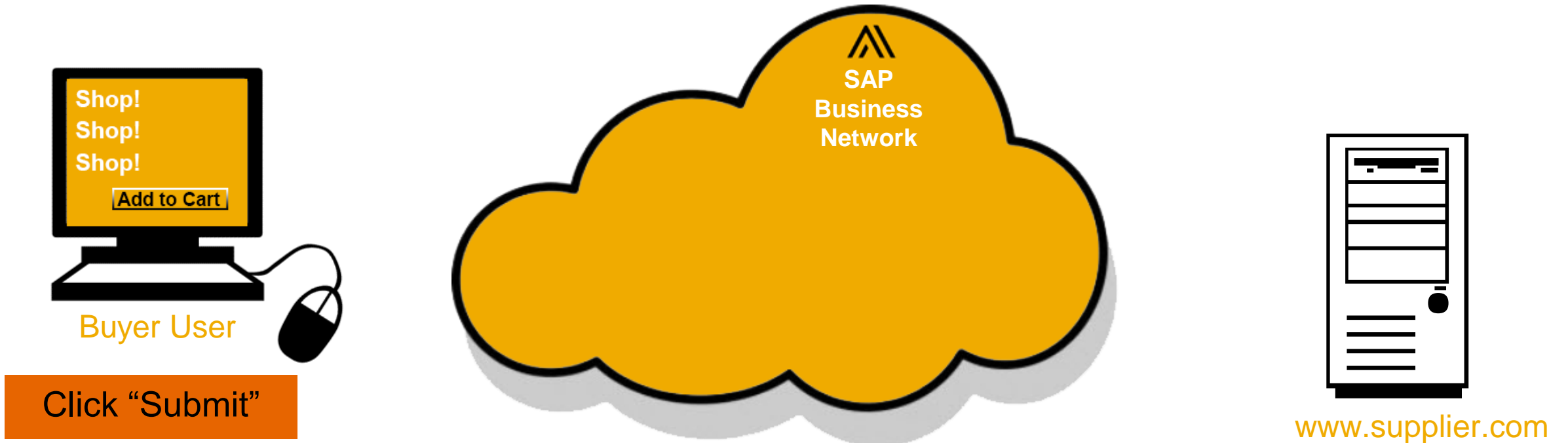
The Supplier then evaluates any other information and sends back a **PunchOutSetupResponse** with the URL of the catalog to display to the User.

Understanding The cXML Message Flow



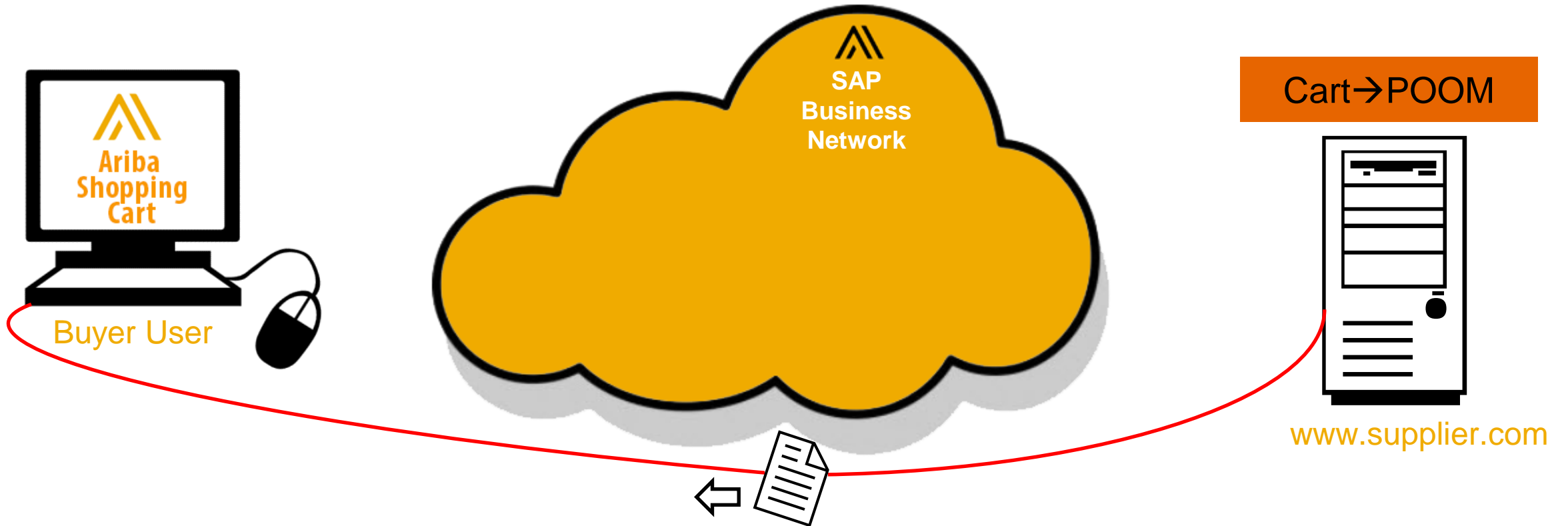
The SAP Business Network validates the **Response**, then sends the URL of the Supplier's Catalog to the Buyer's Ariba Procurement application to display to the User.

Understanding The cXML Message Flow



The User shops the Supplier's Catalog (now in the SAP Ariba Procurement window) and adds items to the Shopping Cart. When they are done, they submit the Cart back to the SAP Ariba Procurement solution.

Understanding The cXML Message Flow



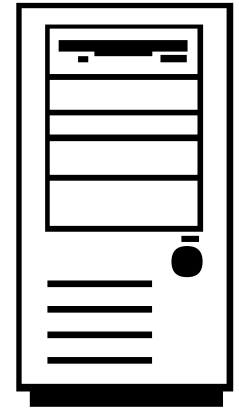
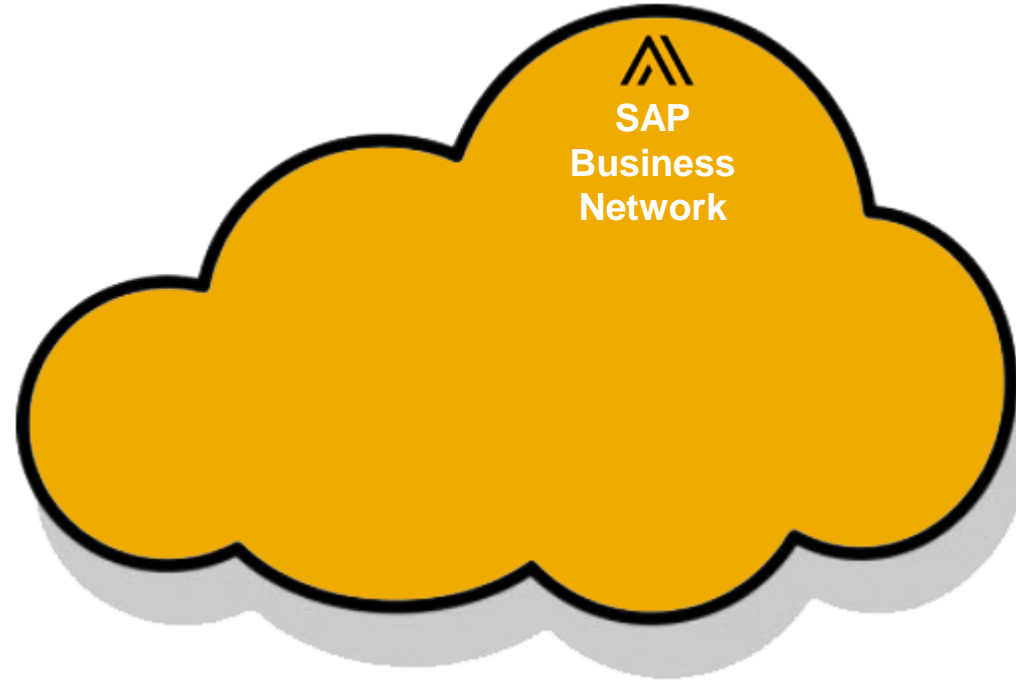
The Supplier converts their Shopping Cart to cXML, closes the connection and sends a **PunchOutOrderMessage (POOM)** directly to the SAP Ariba Procurement solution, which then adds the items from the Supplier into the User's SAP Ariba Procurement Shopping Cart.

Understanding The cXML Message Flow



Buyer User

Ready to submit
SAP Ariba Cart to
become a
Requisition
and buy!



www.supplier.com

Closing the PunchOut Session

PunchOut Catalog Functionality: Closing a Session

Important Change to PunchOut Catalogs!

- As a result of a recent change SAP Ariba **recommends** all suppliers implement a “**Close Session**” button on their PunchOut sites.
 - A recent enhancement to the SAP Ariba procurement application will see a change in how customer’s users will access and close supplier PunchOut sites.
 - SAP Ariba procurement applications will now allow PunchOut sites to open via a browser redirect rather than within an iframe, based on customer settings.
 - To close the session customers can either use the browser’s “Back” button, or implement a “Close Session” button.
 - Previously suppliers did not need to have a “Close Session” button on their site, as this was managed within SAP Ariba outside of the iframe (removed).
- By following SAP Ariba’s recommendation and implementing a “Close Session” button, you will enhance the user experience, avoid unnecessary questions from users, and ensure continued smooth interactions between your customers and your PunchOut site.

PunchOut Catalog Functionality: Closing a Session

- This button should behave differently based on the value of the cXML PunchOutSetupRequest operation attribute:
 - **operation="create"** (PunchOut to create a new shopping cart): Halt the session by sending a cXML PunchOutOrderMessage document that contains no ItemIn elements (an empty shopping cart).
 - **operation="edit"** (PunchOut to modify an existing cart): Halt the session by sending a cXML PunchOutOrderMessage document containing the existing ItemOut elements as ItemIn elements and include the cXML "supplier cookie" if it exists. That way, the shopping cart appears unchanged in the procurement application. If the customer has made changes then a revised PunchOutOrderMessage reflecting the changes needs to be triggered and returned.
 - **operation="inspect"** (PunchOut to inspect an existing cart): Halt the session by sending a cXML PunchOutOrderMessage document that contains no ItemIn elements (an empty shopping cart). The shopping cart is read-only in the procurement application, so it is unchanged.

Catalog Tester

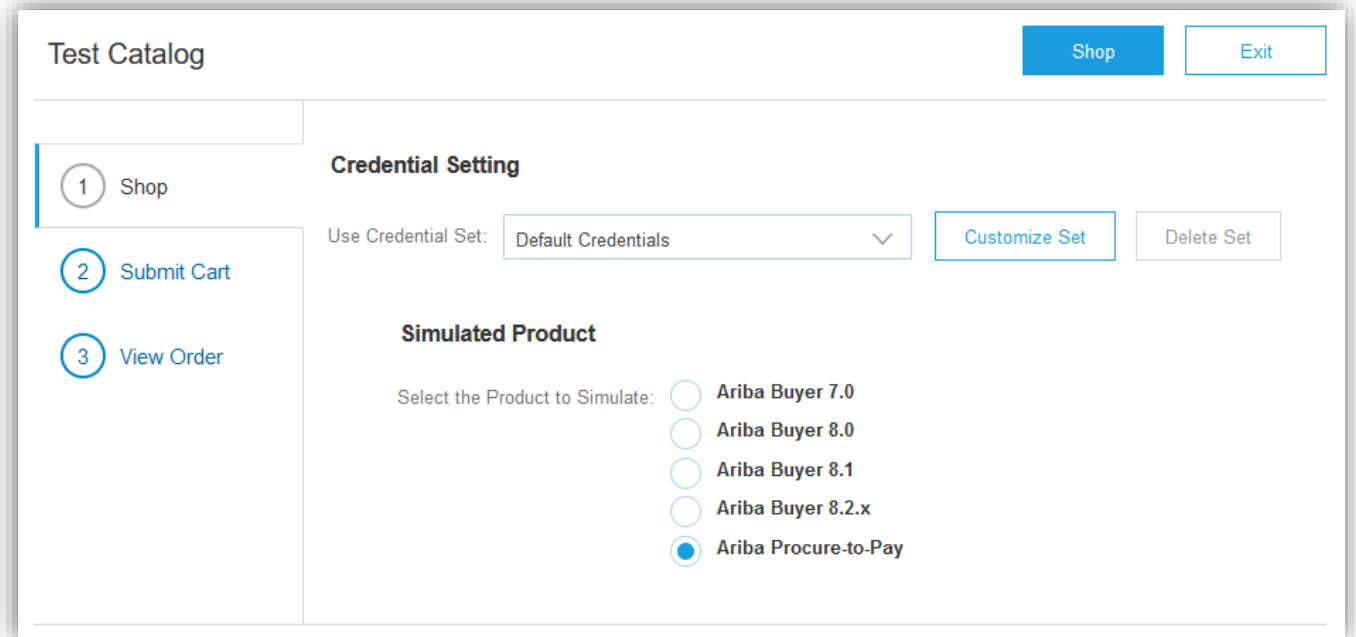
The supplier test account is equipped with a built in Catalog Tester. This Catalog Tester is able to generate a cXML PunchOutSetupRequest and send it to your default PunchOut URL. You will also be able to test other PunchOut catalog functions such as the ability to send and edit a PunchOutOrderMessage.

- This Catalog Tester is only available for the Default PunchOut URL configured in your account, and will not function with customer specific URLs.
- The built in Catalog Tester has it's own unique ANID available to all suppliers: AN01000002779-T.
- You will need to ensure that the Tester's ANID is mapped to a customer or dummy account in your system for testing purposes.

Using the Catalog Tester

After the catalog index file has been loaded you will be able to use the built in Catalog Tester, by following these steps:

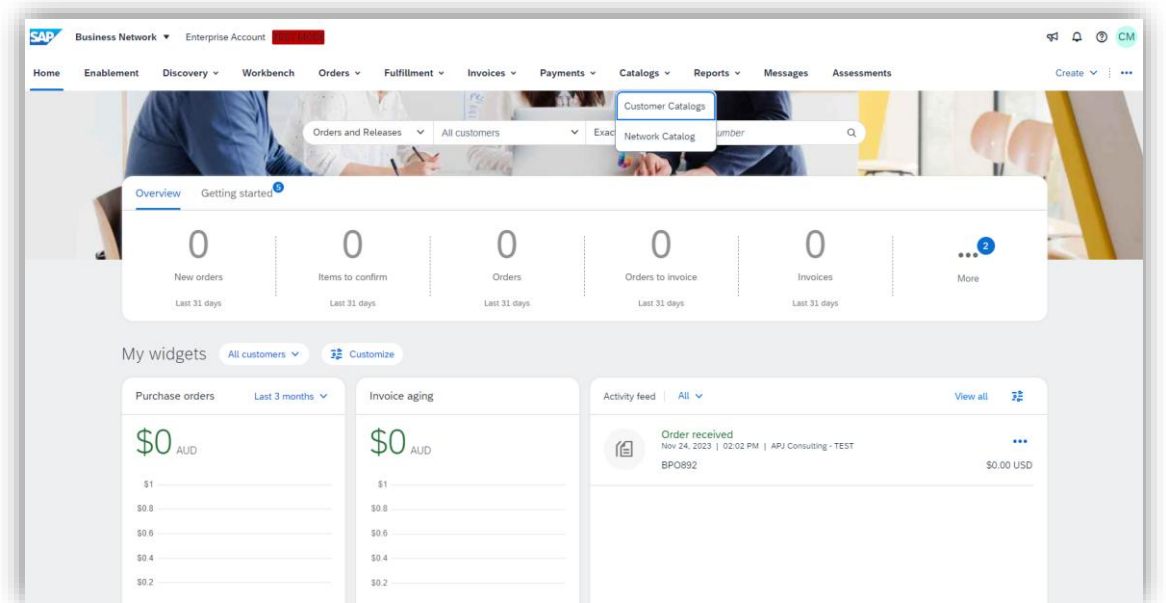
- 1) Select the radio button next to a catalog subscription.
- 2) Click the Test button at the bottom of the screen.
- 3) You can leave the Default Credentials and Simulated Product as is.
- 4) Click the Shop button to trigger the cXML PunchOutSetupRequest being sent to your PunchOut URL for validation.
- 5) If everything is correctly configured your site should load, and you will be able to add an item to the shopping cart and bring it back into your account via the cXML PunchOutOrderMessage.



Supplier Account Setup

What is A Test Account

- It is recommended to load the catalog to the Test account first. This will help ensure your catalog is error free and working when it is later loaded to the Production account.
- Every SAP Business Network account has a Production account by default. However, only the **account administrator** can create and manage additional test accounts.
- Ensure you have access to a Test account, or contact your account administrator for support.



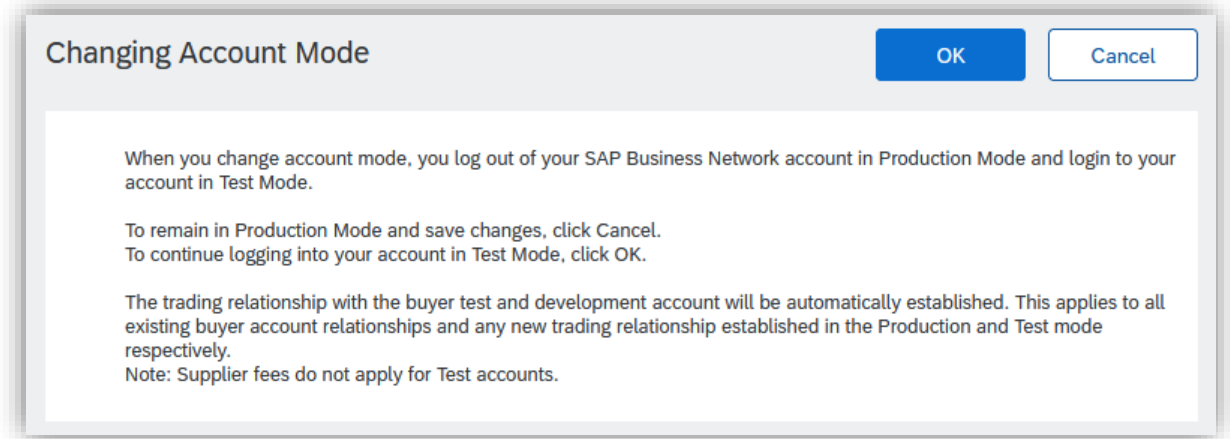
Test accounts are recognizable by the TEST MODE icon in the main banner.

TEST MODE

Creating A Test Account

Creating Test Account Instructions

- 1) Click on your name/initials in the top right.
 - 2) Click “Switch to Test ID”
 - 3) Click “OK” to acknowledge the action.
 - 4) It will then ask you to setup a username and password which is only for the test account.
- Note:** the default username will simply add “test-” to the existing username.
- 5) Once this is done you will be logged into the test account and can upload the catalogue file.

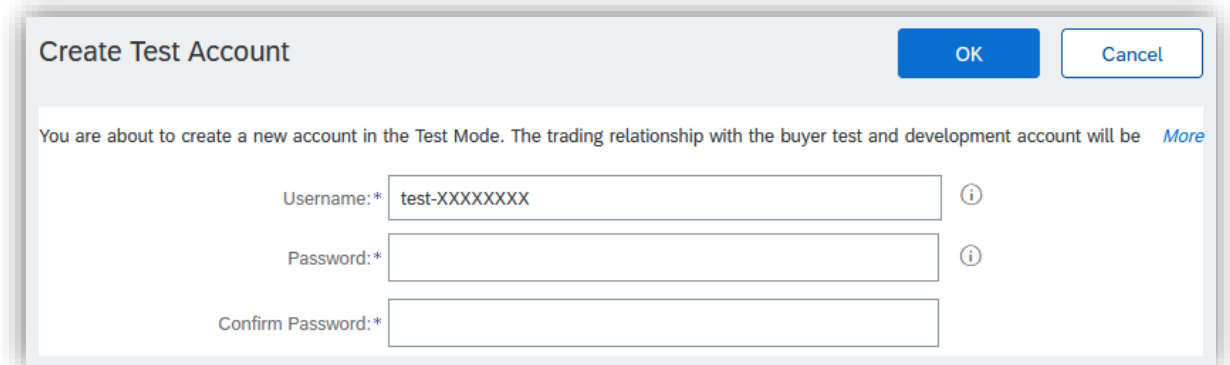


Changing Account Mode [OK] [Cancel]

When you change account mode, you log out of your SAP Business Network account in Production Mode and login to your account in Test Mode.

To remain in Production Mode and save changes, click Cancel.
To continue logging into your account in Test Mode, click OK.

The trading relationship with the buyer test and development account will be automatically established. This applies to all existing buyer account relationships and any new trading relationship established in the Production and Test mode respectively.
Note: Supplier fees do not apply for Test accounts.



Create Test Account [OK] [Cancel]

You are about to create a new account in the Test Mode. The trading relationship with the buyer test and development account will be [More](#)

Username:* test-XXXXXXXX [i]

Password:* [i]

Confirm Password:*

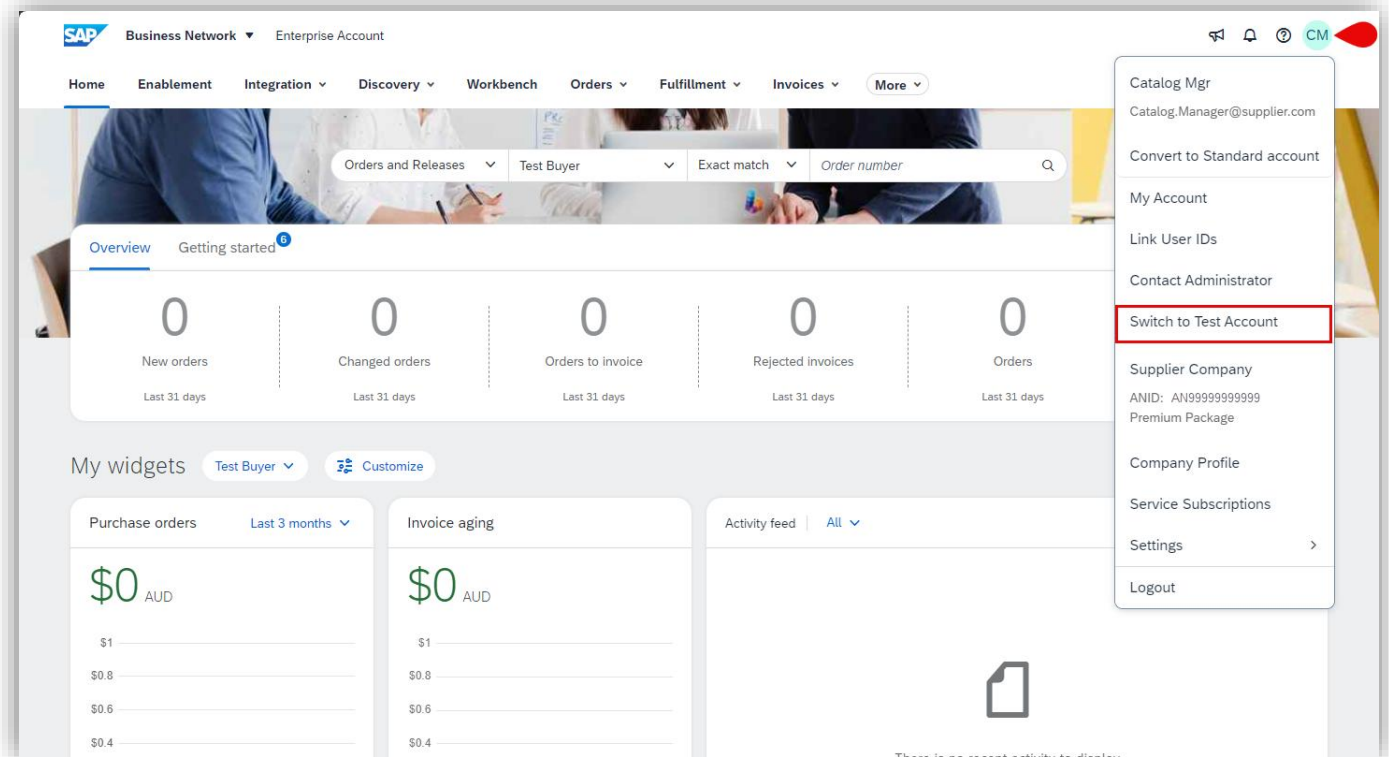
Switching To The Test Account

Once you have setup you will be able to switch between your Production and Test account.

Your Catalog should be loaded and tested in your Test Account first, and later to Production.

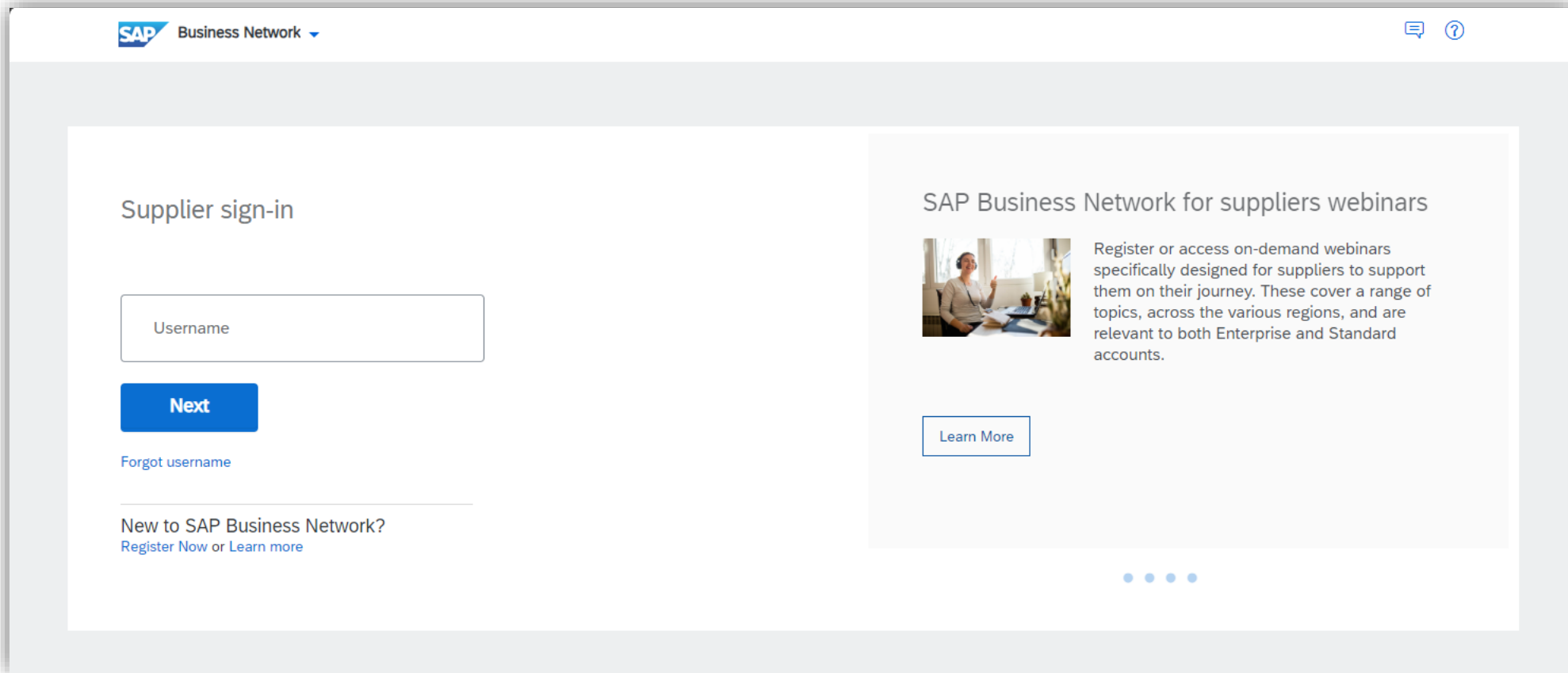
Note: If you are instructed to load a Catalog to a Production account, you will not need to switch to Test.

- 1) Find your initials in the upper right corner and click for the pull down menu
- 2) Click **“Switch To Test Account”**
If you don’t see a “Switch to Test Account” Contact your SAP Business Network Administrator
- 3) You will get the warning;
“You are about to switch to Test Mode.”
- 4) Click **“OK”**



Creating A PunchOut Connection

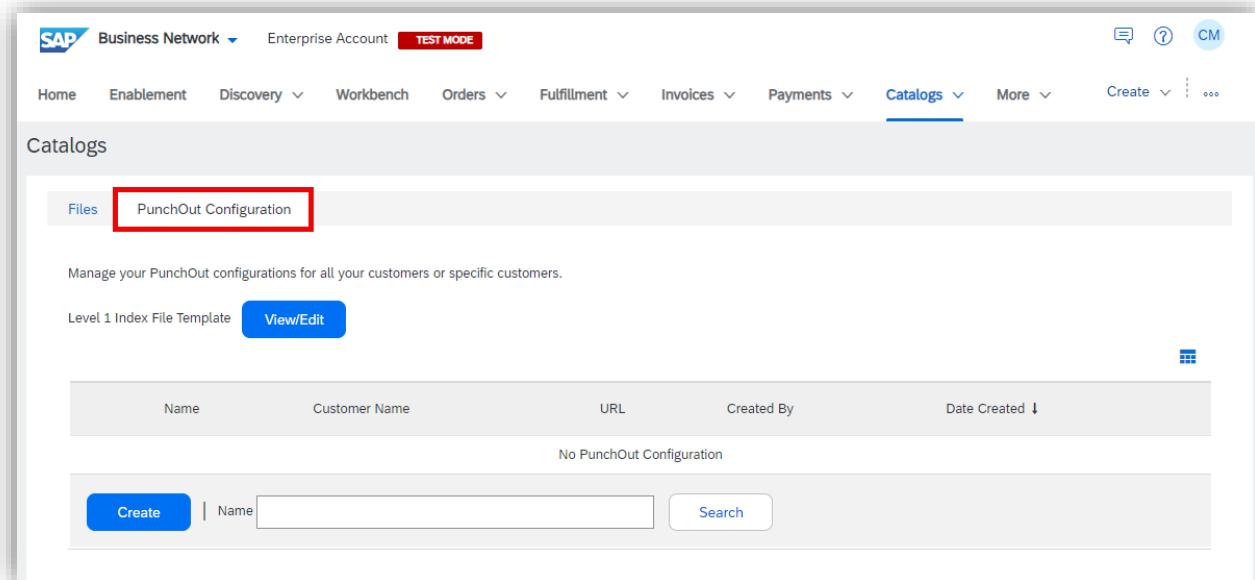
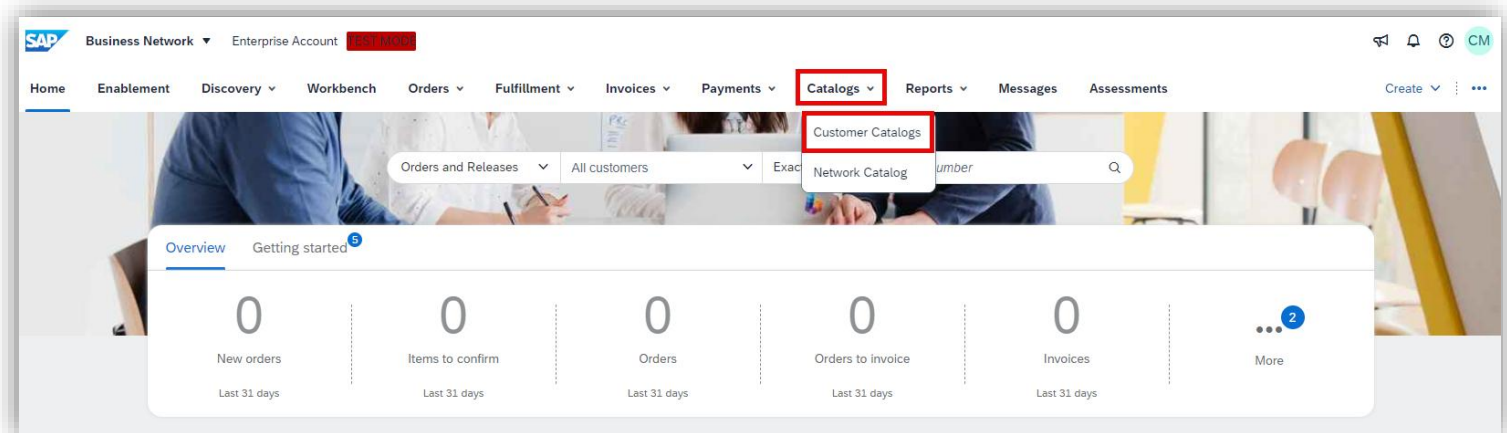
- Login to your **SAP Business Supplier Account** (<https://supplier.ariba.com>)



- Assistance from your IT Manager or 3rd party may be needed for the following slides.

Creating A PunchOut Connection

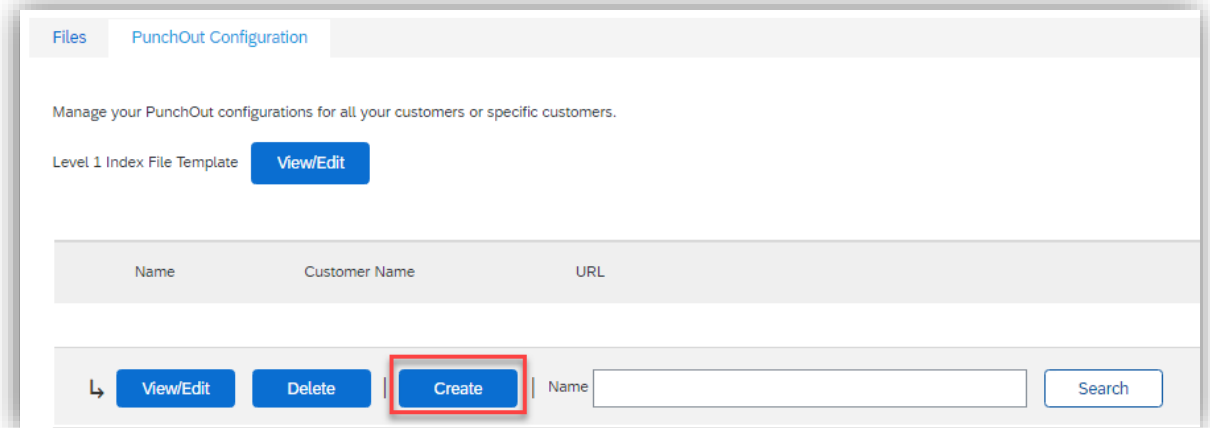
- Go to **Catalogs** tab, choose **Customer Catalogs**, and click **PunchOut Configuration**.
- This area allows you to create your **default** PunchOut connection or a **unique** connection with your customer (if applicable).
- Please note the next steps need to be completed in your **Test** and **Production** accounts.



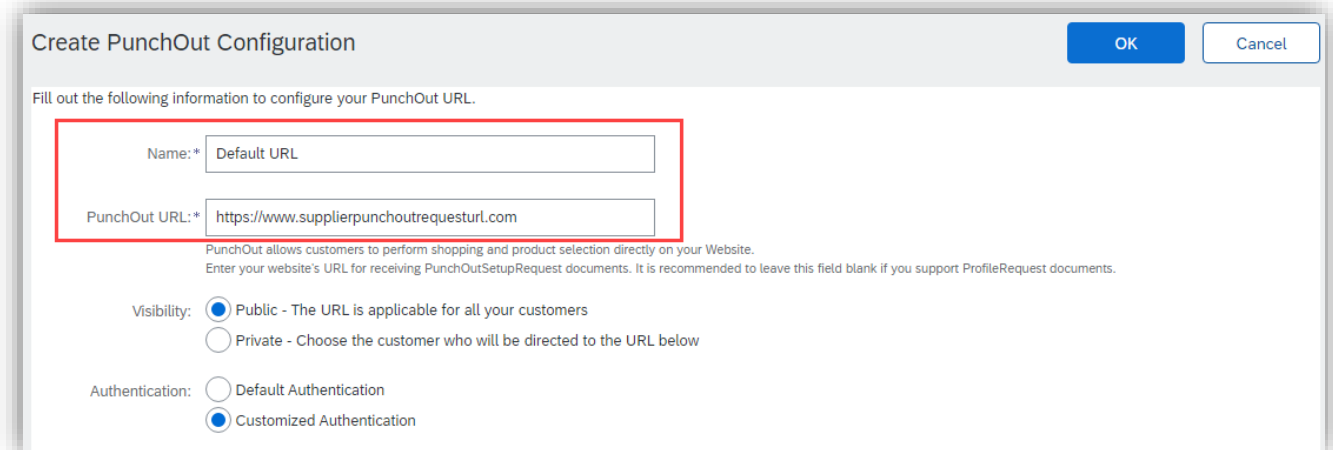
Creating A PunchOut Connection

To setup your PunchOut Catalog URL:

- Click on **Create**.
- In the **Name** field, write a customer specific name or “Default URL” as seen in the image.
- Add your company’s complete **PunchOut Request URL** within the PunchOut URL field.
 - This can be your **default** URL that will be used for **all** of your customers, each time they initiate a PunchOut session **OR** a **unique** URL for a specific customer.



The screenshot shows the 'PunchOut Configuration' page. At the top, there's a 'Files' tab and a 'PunchOut Configuration' tab. Below the tabs, a message says 'Manage your PunchOut configurations for all your customers or specific customers.' There's a 'Level 1 Index File Template' with a 'View/Edit' button. Below this is a table with columns 'Name', 'Customer Name', and 'URL'. At the bottom, there are buttons for 'View/Edit', 'Delete', and 'Create' (highlighted with a red box). To the right of the 'Create' button is a 'Name' input field and a 'Search' button.

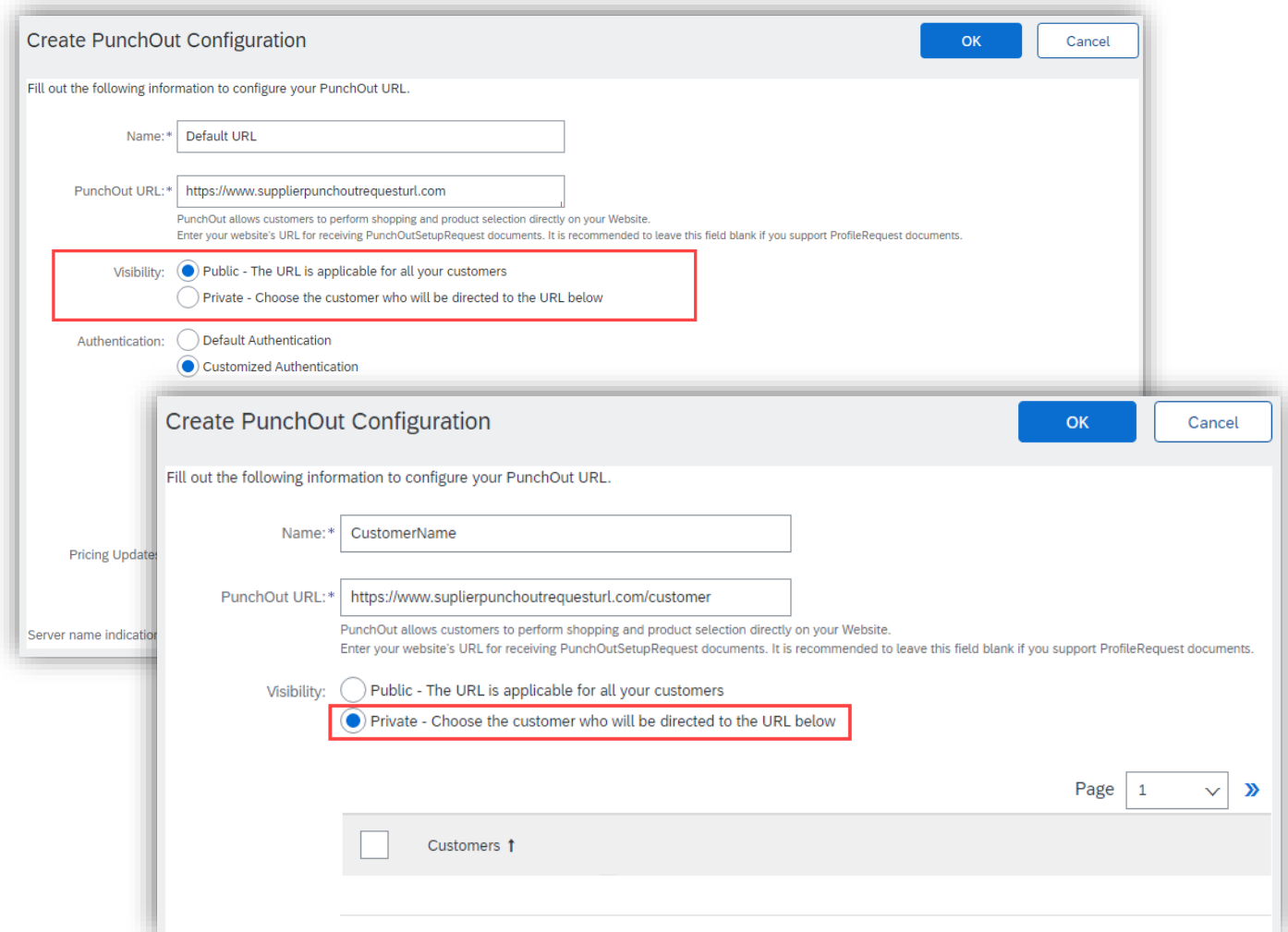


The screenshot shows the 'Create PunchOut Configuration' dialog. It has 'OK' and 'Cancel' buttons at the top right. The main text says 'Fill out the following information to configure your PunchOut URL.' Below this, there are two input fields: 'Name:*' with the value 'Default URL' and 'PunchOut URL:*' with the value 'https://www.supplierpunchoutrequesturl.com'. These two fields are highlighted with a red box. Below the input fields, there's a note: 'PunchOut allows customers to perform shopping and product selection directly on your Website. Enter your website's URL for receiving PunchOutSetupRequest documents. It is recommended to leave this field blank if you support ProfileRequest documents.' There are two sections: 'Visibility' with radio buttons for 'Public - The URL is applicable for all your customers' (selected) and 'Private - Choose the customer who will be directed to the URL below'; and 'Authentication' with radio buttons for 'Default Authentication' and 'Customized Authentication' (selected).

Creating A PunchOut Connection

- Set the **Visibility** of the PunchOut URL.
 - **Public** – can only be set once and will become the default URL for all customers unless otherwise specified.
 - **Private** – can be used several times. When selected, you will be able to specify a chosen customer from your list of accepted Trading Relationships.

Note: You can only use 1 URL (Public or Private) per customer.



The image displays two screenshots of the 'Create PunchOut Configuration' dialog box, illustrating the configuration process for a PunchOut connection.

Top Screenshot (Public Configuration):

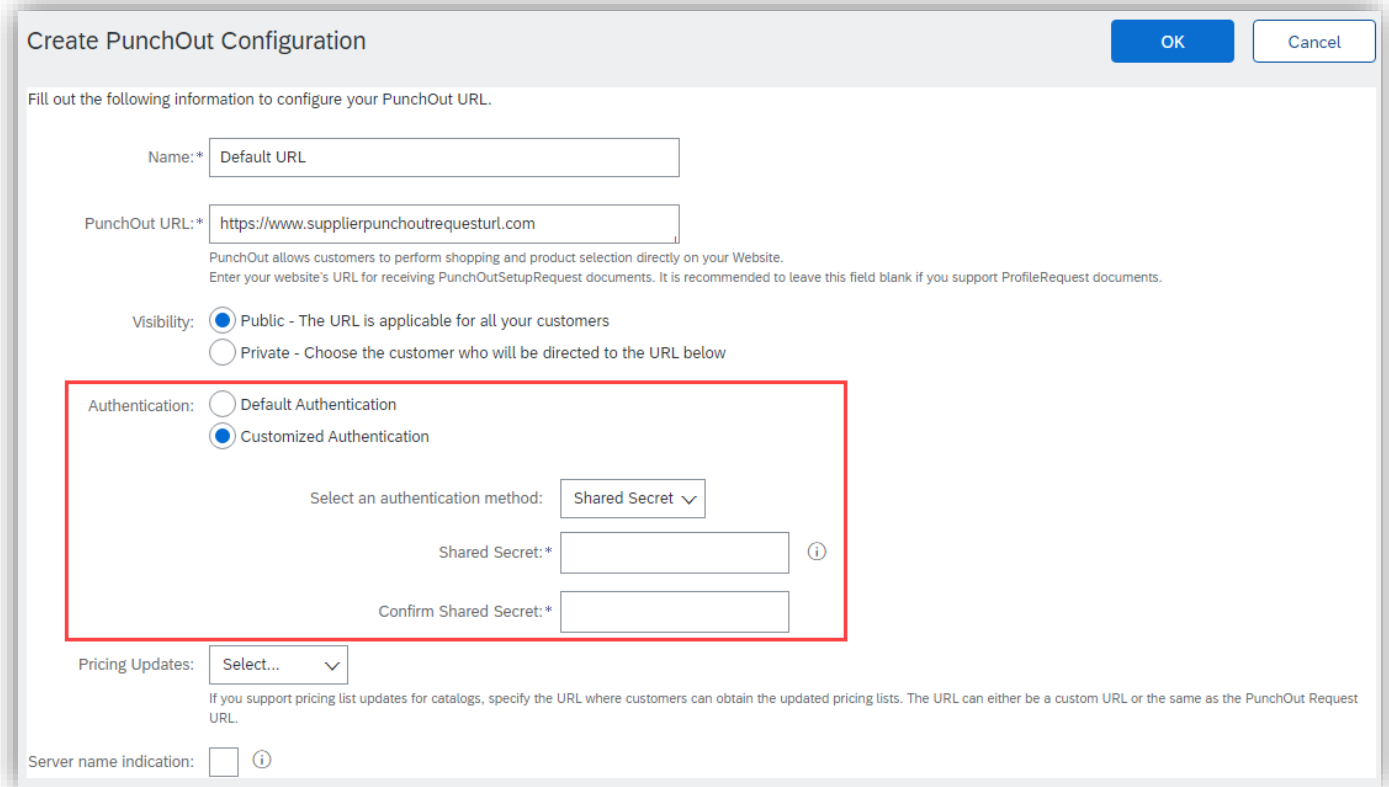
- Title:** Create PunchOut Configuration
- Buttons:** OK, Cancel
- Instructions:** Fill out the following information to configure your PunchOut URL.
- Name:** Default URL
- PunchOut URL:** <https://www.supplierpunchoutrequesturl.com>
- Visibility:** ☒ Public - The URL is applicable for all your customers. ☐ Private - Choose the customer who will be directed to the URL below.
- Authentication:** ☐ Default Authentication. ☒ Customized Authentication.

Bottom Screenshot (Private Configuration):

- Title:** Create PunchOut Configuration
- Buttons:** OK, Cancel
- Instructions:** Fill out the following information to configure your PunchOut URL.
- Name:** CustomerName
- PunchOut URL:** <https://www.supplierpunchoutrequesturl.com/customer>
- Visibility:** ☐ Public - The URL is applicable for all your customers. ☒ Private - Choose the customer who will be directed to the URL below.
- Page:** 1
- Customers:** Customers ↑

Creating A PunchOut Connection

- Next, select an **Authentication** option and method. The Preferred method is **Shared Secret**.
 - **Customized Authentication:** This shared secret is used purely for PunchOut catalog connectivity. These credentials will be sent to your PunchOut site to authenticate requests from **ALL** Buyers.
 - **Default Authentication:** This Account Level authentication is used for all cXML transactional documents and is maintained in another section of your Supplier Account. More detail regarding this topic and setup is covered in section [Native cXML Transaction Setup](#).



Create PunchOut Configuration

Fill out the following information to configure your PunchOut URL.

Name:* Default URL

PunchOut URL:* https://www.supplierpunchoutrequesturl.com

PunchOut allows customers to perform shopping and product selection directly on your Website. Enter your website's URL for receiving PunchOutSetupRequest documents. It is recommended to leave this field blank if you support ProfileRequest documents.

Visibility: ☒ Public - The URL is applicable for all your customers
☐ Private - Choose the customer who will be directed to the URL below

Authentication: ☐ Default Authentication
☒ Customized Authentication

Select an authentication method: Shared Secret

Shared Secret:*

Confirm Shared Secret:*

Pricing Updates: Select...

If you support pricing list updates for catalogs, specify the URL where customers can obtain the updated pricing lists. The URL can either be a custom URL or the same as the PunchOut Request URL.

Server name indication: ☐

- If using the customized option, type in your Shared Secret, and confirm it.

Creating A PunchOut Connection

- Uncheck if your URL is not **Server Name Indication** (SNI) enabled.
- Click OK

Create PunchOut Configuration

OKCancel

Fill out the following information to configure your PunchOut URL.

Name: *Default URL

PunchOut URL: *https://www.supplierpunchoutrequesturl.com

PunchOut allows customers to perform shopping and product selection directly on your Website.
Enter your website's URL for receiving PunchOutSetupRequest documents. It is recommended to leave this field blank if you support ProfileRequest documents.

Visibility: ☒ Public - The URL is applicable for all your customers
☐ Private - Choose the customer who will be directed to the URL below

Authentication: ☐ Default Authentication
☒ Customized Authentication

Select an authentication method: Shared Secret

Shared Secret: *

Confirm Shared Secret: *

Pricing Updates: Select...

If you support pricing list updates for catalogs, specify the URL where customers can obtain the updated pricing lists. The URL can either be a custom URL or the same as the PunchOut Request URL.

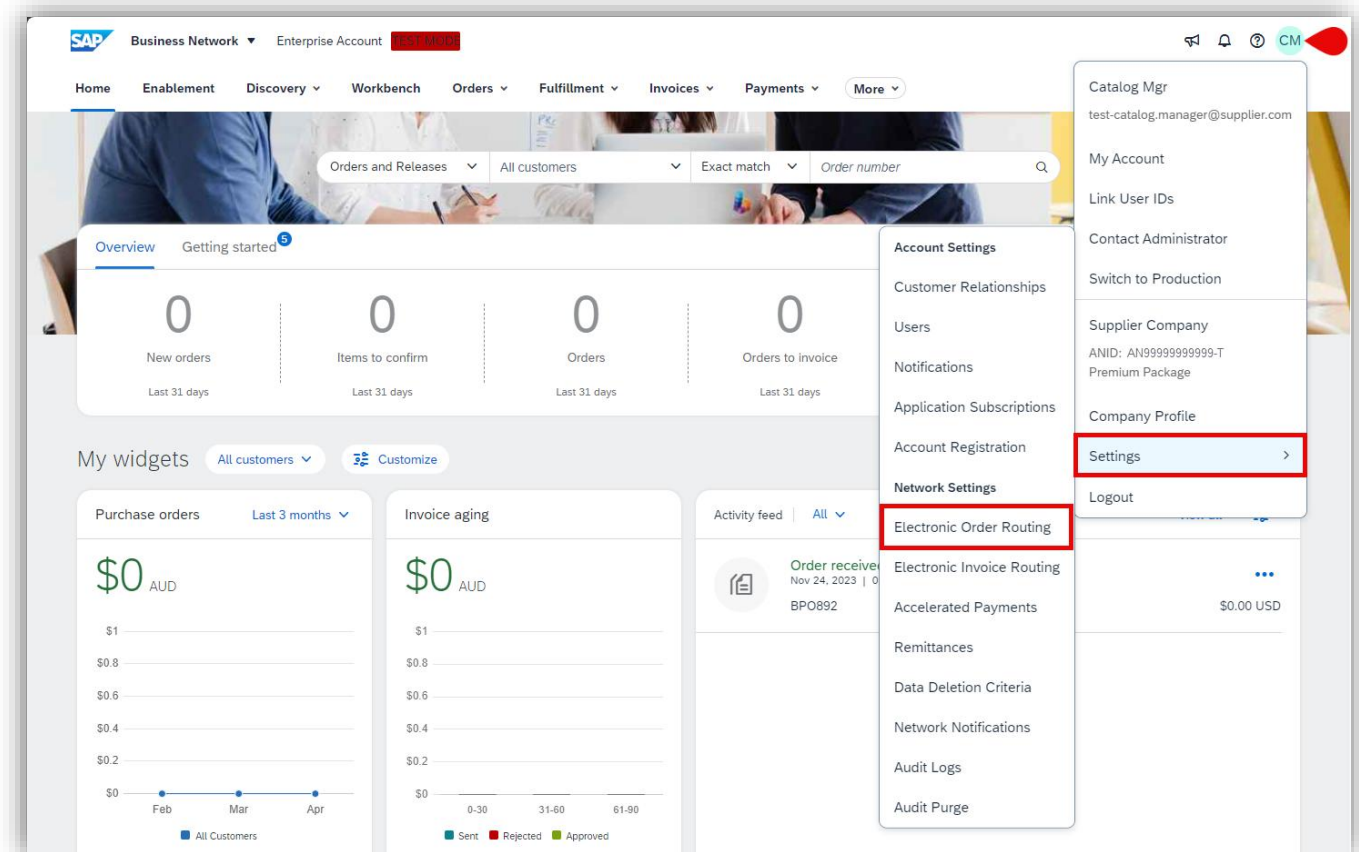
Server name indication: ☐

© 2024 SAP SE or an SAP affiliate company. All rights reserved. PUBLIC

42

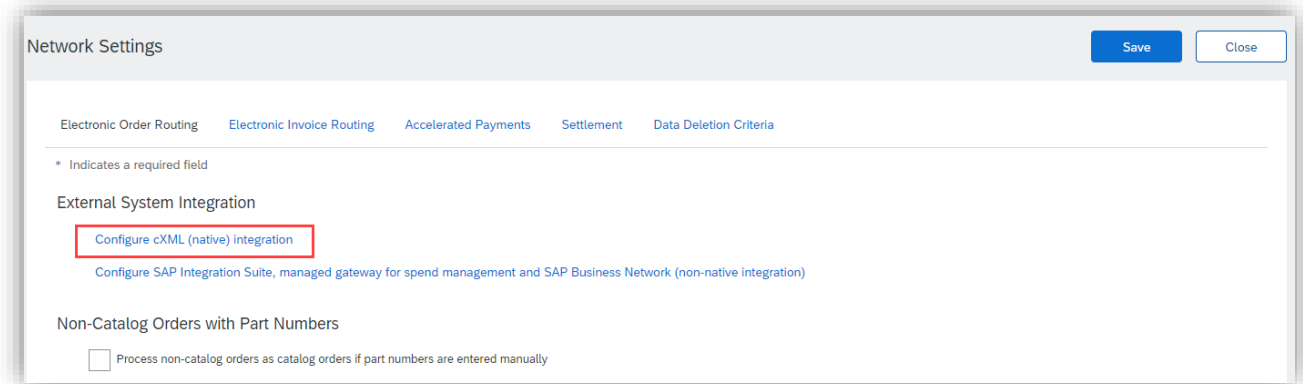
Native cXML Transaction Setup

- This option refers to the account level **Default Authentication**. This option would be used for all cXML transactional documents across all customers.
- To access this authentication setup:
 - Click your initials
 - Select Settings
 - Click Electronic Order Routing



Native cXML Transaction Setup

- Click Configure cXML (native) Integration
- Select your Authentication Method. The Preferred method is Shared Secret.
- Type in your Shared Secret credentials.
 - This Shared Secret will be sent to your systems to authenticate requests.
- Click OK.



Network Settings

Electronic Order Routing Electronic Invoice Routing Accelerated Payments Settlement Data Deletion Criteria

* Indicates a required field

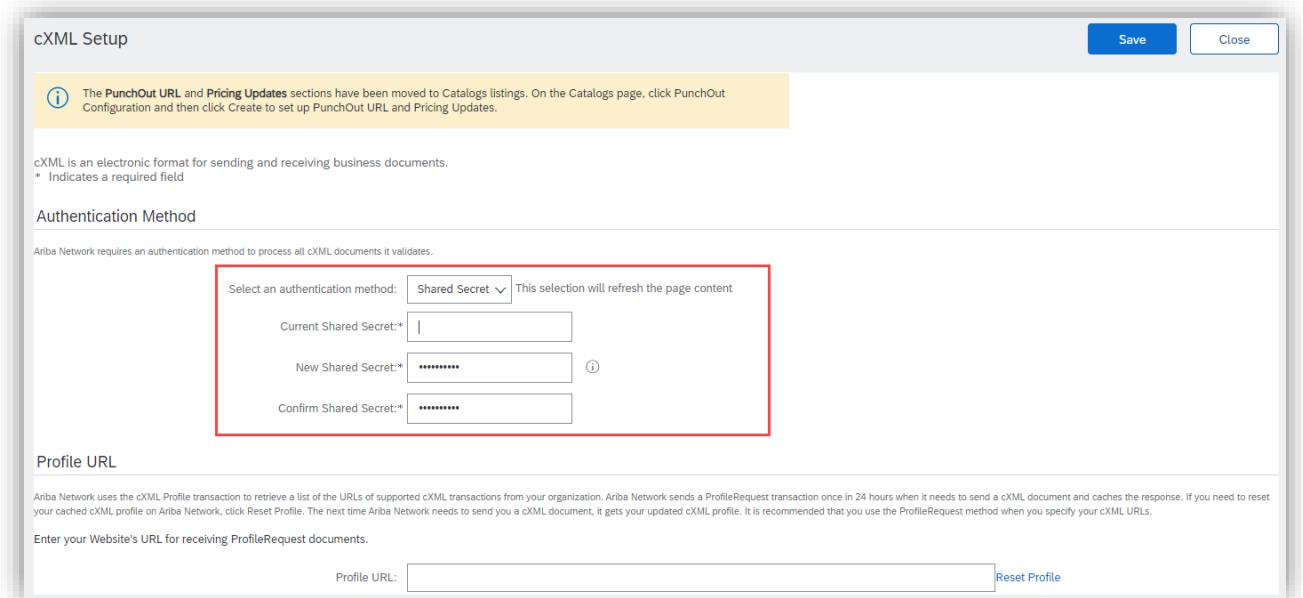
External System Integration

Configure cXML (native) integration

Configure SAP Integration Suite, managed gateway for spend management and SAP Business Network (non-native integration)

Non-Catalog Orders with Part Numbers

☐ Process non-catalog orders as catalog orders if part numbers are entered manually



cXML Setup

The PunchOut URL and Pricing Updates sections have been moved to Catalogs listings. On the Catalogs page, click PunchOut Configuration and then click Create to set up PunchOut URL and Pricing Updates.

cXML is an electronic format for sending and receiving business documents.

* Indicates a required field

Authentication Method

Ariba Network requires an authentication method to process all cXML documents it validates.

Select an authentication method: Shared Secret This selection will refresh the page content

Current Shared Secret:*

New Shared Secret:*

Confirm Shared Secret:*

Profile URL

Ariba Network uses the cXML Profile transaction to retrieve a list of the URLs of supported cXML transactions from your organization. Ariba Network sends a ProfileRequest transaction once in 24 hours when it needs to send a cXML document and caches the response. If you need to reset your cached cXML profile on Ariba Network, click Reset Profile. The next time Ariba Network needs to send you a cXML document, it gets your updated cXML profile. It is recommended that you use the ProfileRequest method when you specify your cXML URLs.

Enter your Website's URL for receiving ProfileRequest documents.

Profile URL: Reset Profile

Additional Resources

Additional Resources

Continue your PunchOut Catalog creation journey, and move to the next phase if needed.

The PunchOut Catalog Documents			
Introduction to PunchOut Catalogs & Account Setup (Currently viewing this document)	The PunchOut Wizard	Customer Specific Templates	Troubleshooting and Updating Catalogs
Details the Punchout fundamentals and account configuration	Details the Catalog validation flow and using an online template and wizard to create your PunchOut Index file.	Details the process of manually creating and uploading the PunchOut Index file. To view this document, please go to the Customer specific Supplier Information Portal or contact your Customer directly.	Details the Catalog upload and update process and the XLS to CIF conversion. This document also illustrates different troubleshooting scenarios and resolutions.

Thank you.