

Pageflex Services Storefront Extension: XML and JDF Export

This extension enables exporting order and document information from Storefront to either XML or JDF files using a template tagged with variables. It enables the export of one file per order, per shipment, or per document. Files can be exported to local and remote folders. Files can also be posted via HTTP. Additionally JDF files can be transported via HTTP using the JMF transport protocol to a destination server. Creation of the XML and JDF export files are accomplished using templates which use key word substitution. Users can create any number of templates tailored to meet the target system data requirements. When the extension is set to per document XML output a different template can be specified for each document.

Kit Product Support

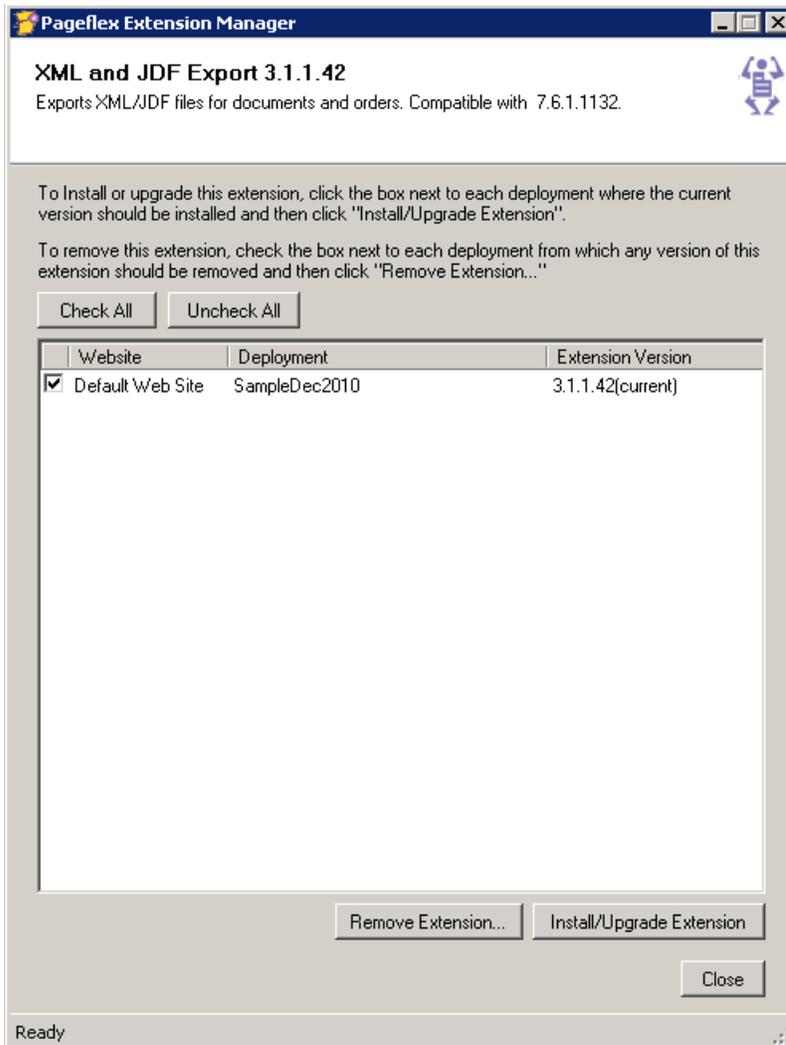
XML and JDF Export 3.1.0 adds support for kit products. When the extension encounters a kit product, the kit is “flattened” to represent the items within the kit. For example, a kit containing a business card and letter results in two entries in the XML output, one for the letter and one for the business card. If the output is set to per document one XML file will be generated for each kit listing its items.

Using the Extension Manager

Use the Extension Manager to install or remove extensions from your deployments.

Installing the XML and JDF Export Extension

Run the **ExtensionManager for XML and JDF Export.exe** application and choose the deployment(s) you would like the extension installed on. Note that you should run this application with local administrator privileges, to do so, right-click and choose **Run as administrator** or right-click the application, select **Properties**, choose the **Compatibility** tab and select the **"Run this program as an administrator"** check box in the Privilege Level section to always run this application as an administrator by default.



Follow the steps below to install or upgrade an extension on one or several deployments.

To install an extension

1. Unzip the XML_and_JDF_Export_x.x.x.x.zip file into a directory of your choice (the x's represent the version number).
2. Run the **ExtensionManager for XML and JDF Export.exe** application in the extension directory.
3. Select the deployment(s) and click **Install/Upgrade Extension** to install the extension on the selected deployment(s).
4. Click **OK** in the dialog to stop the deployment so that the extension can be installed. Note that the extension resets and recompiles the deployment, so it may take awhile for the installation to finish, do not interrupt the installation once it has begun.
5. After the extension is installed, the version number in the Extension Version column next to the deployment name changes to the latest version and the Extension Manager becomes responsive again. Click **Close** to close the Extension Manager.

To remove an extension

1. In the Extension Manager, select the deployment(s) from which you want to remove the extension and click **Remove Extension**.
2. Click **OK** in the dialog to confirm. The extension will be removed and no extension version number is displayed in the column next to the deployment name.
3. Click **Close** to close the Extension Manager.
6. A dialog box is displayed to remind you to log in to each deployment and delete the removed extension on the extension configuration screen on the administrator site in order to complete the removal of the extension. Click **OK** to confirm the dialog.
7. To delete the removed extension on the extension configuration screen, log into the administrator site for the deployment, click **Extensions** on the sidebar, select the extension and click **Delete**. Click **OK** in the dialog to confirm.

Configuring the extension

1. Login to the administrator site for the Storefront deployment on which you installed the extension. For example, <http://myserver/mydeployment/adminLogin.aspx>.
2. Click **Extensions** in the sidebar to go to the **Extensions** page.

Extensions

Manage extension modules.

Page 1/1 First | Prev | Next | Last 18 rows # Rows: 50

<input type="checkbox"/>	Name	Active	Status	Path
<input type="checkbox"/>	Integrated Payment: Purchase Order	<input checked="" type="checkbox"/>	OK	Extensions/PaymentPurchaseOrder.dll
<input type="checkbox"/>	Integrated Shipping: Custom	<input checked="" type="checkbox"/>	OK	Extensions/ShippingCustom.dll
<input type="checkbox"/>	Integrated Shipping: United Parcel Service	<input type="checkbox"/>		Extensions/ShippingUPS.dll
<input type="checkbox"/>	Integrated Shipping: Federal Express	<input type="checkbox"/>		Extensions/ShippingFedEx.dll
<input type="checkbox"/>	Integrated Shipping: ZeroWeight	<input type="checkbox"/>		Extensions/ShippingZeroWeight.dll
<input type="checkbox"/>	CreditCardSample (testing only)	<input type="checkbox"/>		CreditCardSampleExtension\CreditCardSample.dll
<input type="checkbox"/>	Integrated Shipping: DHL	<input type="checkbox"/>		Extensions/ShippingDHL.dll
<input type="checkbox"/>	Services: Authorize.Net Payment	<input type="checkbox"/>		Services\AuthorizeNetPay.dll
<input type="checkbox"/>	Services: Externally Driven HTML Content	<input type="checkbox"/>		Services\ExternallyDrivenHTMLContent.dll
<input type="checkbox"/>	Services: Image Upload Preflight	<input type="checkbox"/>		Services\ImageUploadPreflight.dll
<input type="checkbox"/>	Services: Password Check	<input type="checkbox"/>		Services>PasswordCheck.dll
<input type="checkbox"/>	Services: Payflow Pro Payment Gateway	<input type="checkbox"/>		Services\PayflowPro.dll
<input type="checkbox"/>	Services: Sku Map	<input type="checkbox"/>		Services\SkuMap.dll
<input type="checkbox"/>	Services: iSINI	<input type="checkbox"/>		Services\SuperInsert.dll
<input type="checkbox"/>	Services: Canadian Tax	<input type="checkbox"/>		Services\CanadianTax.dll
<input type="checkbox"/>	Services: Punch Out Integration	<input checked="" type="checkbox"/>	OK	Services\PunchOutIntegration.dll
<input type="checkbox"/>	Services: Inventory Tracking by SKU Number	<input type="checkbox"/>		Services\Inventory.dll
<input type="checkbox"/>	Services: XML and JDF Export	<input checked="" type="checkbox"/>	OK	Services\XmlOutput.dll

Pages: 1

3. Click **Services: XML and JDF Export** to configure the extension.
4. Select **Yes** in the **Active** field to make the extension active.
5. Configure the fields in the extension (described below) and click **Update** to save changes, or **Cancel** to exit without saving.

General Properties and Configuration

The general properties and configuration options are described below.

The screenshot shows a dialog box with two main sections: **General Properties** and **General Configuration**.

General Properties:

- Path: Services\XmlOutput.dll
- Active: Yes No
- Services: XML and JDF Export - 3.1.1.42
- [Open Doc](#) - Up to date.

General Configuration:

- XML template :
- One file per :

General Properties

The extension includes the following general properties:

- **Path:** Lists the location of the DLL for the extension.
- **Active:** Select **Yes** to make the extension active or **No** to make it inactive.
- **Services:** List the name and version number of the extension.
- **Open Doc:** Click this link to open the latest documentation for this extension, served from the Pageflex Services website. Next to this link is a note that lists “Up to date.” If the extension is the latest offering from Pageflex, or changes to a link to download the latest version if it is not.

General Configuration

The following general configuration options are available:

- **XML template:** This is the template that will be used by the extension to substitute the Storefront values for those specified in the template. The XML and JDF output files will be generated from the selected template. The XML template(s) are located in the following directory:
<deployment name>\WebPages\Custom\XmlOutputTemplates.
 This directory is created when the extension is installed. The sample.template XML file is included as the default template. NOTE: If you have a product specific version of this extension for Promail, Printstream, Avanti, etc. you will also have a product specific template that has been tested with that system.
- **One file per:** Choose the action that triggers an XML file to be created. There will be an XML file created for every action.
 - **Order Placed:** Creates an XML file for every order that is placed using the external order id as the file name when the order is placed. Note that this happens prior to any Storefront approval processes.

- **Shipment when order placed:** Creates an XML file for every shipment when the order is placed using the external order id as the file name. This is useful if, for example, you are using split shipping and you expect items in one order to be shipped to multiple locations. Note that this happens prior to any Storefront approval processes.
- **Order Rendered:** Creates an XML file for every order that is rendered using the external order id as the file name. This is useful if, for example, you don't expect every placed order to result in a rendered order. An order is considered rendered when every item in the order has a status of either rendered or rejected.
- **Shipment when order rendered:** Creates an XML file for every shipment when the order is placed using the external order id as the file name. This is also useful for deployments using split shipping. An order is considered rendered when every item in the order has a status of either rendered or rejected.
- **Document:** Creates an XML or JDF file for every document rendered using the external document id as the file name. The most common usage of this setting is to output JDF files. **Note:** If exporting documents, you must also configure the **UseJdfOutput** metadata field for each product that you want to export one file per document on. See the **Configure Export File Per Document** section for details on setting up this field and applying it to products.

XML / JDF Output Configuration

Choose how you want files to be exported and sent.

XML/JDF Output Configuration

Exported files are : XML ▾

Send XML/JDF via : Hotfolder ▾

TIP: If you are exporting non-JDF XML, you must choose hotfolder above.

Path for JDF/XML output : * C:\Pageflex\XMLOut

TIP: This should be a file path if you are sending JDF or XML to a hotfolder, and a URL if you are sending JDF via JMF.

Exclude Rejected documents

TIP: Enable this option to omit from the exported XML information about documents subject to approval that are rejected. This option is exports.

The following options are available:

- **Exported files are:** Choose whether to export XML or JDF files.
 - **XML:** When the selection is XML, the generated file will be an XML file with the .xml extension.
 - **JDF:** The JDF (Job Definition Format) selection generates a file with the .jdf extension. The JDF file can be used to describe the intent of the printed piece, as well as each process step to achieve that intent.
- **Send XML/JDF via:** Choose to send the XML or JDF file via Hotfolder or JMF (Job Message Format).
 - **Hotfolder:** Choose Hotfolder if you choose the XML export type or JDF.
 - **JMF:** You can choose JMF when exporting JDF output, but not XML. This format allows the transport of JDF based messages via HTTP.
- **Path for JDF/XML output:** Enter the directory to place the output XML/JDF file. The output folder should be a local or remote directory if you are sending XML or JDF to a Hotfolder, or a URL if you are exporting JDF via JMF.
- **Exclude Rejected documents:** checking this box will force any rejected documents to be omitted from the XML.

HTTP POST Configuration (optional)

Choose the address to post the exported XML to, for example, if you will be posting it to a web service. This is optional.



The screenshot shows a dialog box titled "HTTP Post Configuration". It contains a text input field labeled "Address to post to :" with the value "example.com/webservice.aspx" entered. Below the input field is a checkbox labeled "Post only (do not write file to disk)". At the bottom of the dialog, there is a tip: "TIP: If you are posting the exported XML to a web service, enter the URL here."

If you need to send the XML to machine that you do not have file system access to, you can use HTTP POST. Enter the fully qualified URL to the page or web service on the remote machine that will be accepting the POST. By default the XML will also be written to disk unless you select the **Post only (do not write file to disk)** check box. If you select this check box, the file will not be written to disk.

NOTE: if you have a product specific version of this extension for say Promail, Printstream, Avanti, EPMS then the post action may behave differently if there were special setting required for that communication, like particular SOAP headers. The “generic” version of the extension simply does a generic HTTP POST.

Baseline testing of HTTP posting can be difficult. To that end we include a very simple utility we threw together called PostCatcher. See Appendix A for instructions.

PDF Output Configuration

Choose the path for the output file hotfolder, and, if using JDF, the path to use in the JDF file.

PDF Output Configuration

Path for output file hotfolder :

TIP: If your output device requires that output files be placed in a hotfolder (rather than accessed via HTTP) please enter the path to said hotfolder relative to the output device above. See the documentation for more information.

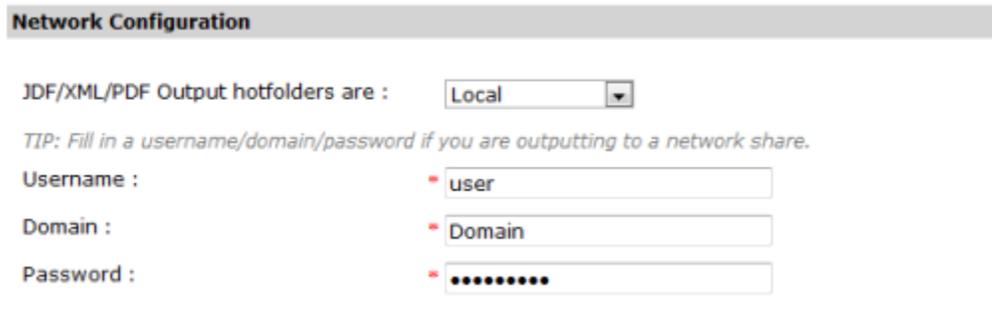
Path to use in JDF file :

TIP: Assuming that your output device needs a local path to find output files, enter the path to use in the JDF or XML file for output. This should end with a trailing \ or / (depending on your path).

- **Path for the output file hotfolder:** Enter the directory to copy the final output files for an order so they are accessible to your remote system. This is a directory if exporting XML or JDF via the **Hotfolder** selection in the **XML / JDF Output Configuration** section described above. If you are transporting the final output files via HTTP, this can be a directory inside WebPages/Custom folder in your deployment.
- **Path to use in JDF file:** Enter the directory to place the JDF file. This path should end with a forward slash “/” or back slash “\” depending upon your requirements. This can also be a URL if your external system supports obtaining output file(s) via HTTP.

Network Configuration

Enter the network configuration information for the hotfolders, such as whether they are on the local system or on a network share. If they are on a network share you will have to enter a username, domain, and password for the extension to use to access the share.



Network Configuration

JDF/XML/PDF Output hotfolders are :

TIP: Fill in a username/domain/password if you are outputting to a network share.

Username :

Domain :

Password :

The following options are available:

- **JDF/XML/PDF Output hotfolders are:** Select whether the hotfolders are on the local machine or on a network share.
 - **Local:** File folders located on the same machine should be set to **Local**. In this case, the username, domain, and password can be left blank.
 - **Network Share:** Select this option if files are copied to remote folders. If this option is selected, enter the username, domain, and password of a user with the appropriate network permissions to both the remote destination machine and the local machine. If using a workgroup, you should use two machines in the same workgroup and use workgroup-level authentication on both, or the same username and password on both machines.
- **Username:** The username to use to access a network share hotfolder.
- **Domain:** The domain to use for this user.
- **Password:** The password for this user.

Email Error Notifications

The extension can be configured to send email notifications on failures.

Email Error Notifications

Enable Notifications:	=	<input type="text" value="YES"/>
From:	=	<input type="text" value="vmyles@pageflex.com"/>
To:	=	<input type="text" value="vmyles@pageflex.com"/>
CC:	=	<input type="text"/>
BCC:	=	<input type="text"/>
Subject:	=	<input type="text" value="failed"/>

TIP: To, CC, and BCC fields support comma separated values for multiple email addresses

The following options are available:

- **Enable Notifications:** Select whether email is to be sent on error.
- **From:** The email address the email should come from.
- **To:** Who the messages should be delivered to.
- **CC:** Optionally, someone to be CC'd on the email.
- **BCC:** Optionally, someone to be BCC on the email.
- **Subject:** The text for the subject line of the email.

The body of the email will be the exception as captured by the extension.

Regenerating\Resending XML

There may be instances where due to an error in one of the systems you need to regenerate and/or resend the XML.

The screenshot shows the 'Order G00001044 Details' page with tabs for 'Ordered Items', 'Payment', 'Shipping', and 'Events'. Below the tabs, there is a link to 'View order details.' and a link to 'Back to Orders'. The 'Order Information' section displays the following details:

Order ID:	G00001044
Order Status:	In Process
User Logon:	test
User Name:	Pat Sampleton
Email Address:	pat@sample.com

Below the order information, there is a checkbox labeled 'Make visible to user' which is checked, and a 'Yes' label. A red box highlights the 'Regenerate XML' button.

At the bottom of the page, there is a section for 'Ordered Items' with several buttons: 'Pull For Data', 'Set Status To "Ready for Production"', 'Submit Job', and 'Set Status To "Print'.

On the Ordered Items tab for an order there will be a “Regenerate XML” button. Pressing this button will trigger the exact same actions that the extension is configured to take.

Configure Export File Per Document

In order to use the export file per document setting, you must configure a metadata field on your deployment to turn on this functionality for each product. The metadata field is called “**UseJdfOutput**”. The name of this field refers to JDF for legacy compatibility, but applies to XML as well.

Create UseJdfOutput Metadata Field

Follow the steps below to create this metadata field. Note that while the field is named **UseJdfOutput** it does not refer specifically to just JDF output. The field name has been retained for legacy compatibility.

To create the UseJdfOutput metadata field:

1. Log into the administrator site.
2. Click **Metadata Fields** on the sidebar.
3. On the **Product** tab, click **Add New**.
4. Create a new field called **UseJdfOutput** with the following properties. Use the default values unless specified otherwise.
 - a. Enter a **Field Name** of “UseJdfOutput”.
 - b. Under **Control and Options**, choose the **Control Type** “Drop down list box”.
 - c. For **Display Label**, enter “Enable Document Export”.
 - d. In options, add two new table values, Display Text “YES” with a value of “YES” and Display Text “NO” with a value of “NO”.
 - e. Under **Default Value**, choose “Same for all users” the value “NO (NO)”.

Note that if you want all products to export files per document by default, you can set this value to “YES (YES)”, but be aware that this may increase the load on your server, which may slow down your deployment significantly, as all products will then export files for every document that they produce.

- f. Click **Update** to save changes.

The field definition should match the following screen:

Name and Use

Field Name:

On User Forms: as which is

On Admin Forms: as which is

Control and Options

Control Type:

Display Label:

Width: pixels

Style:

Options:

2 items

	Display Text	Value	Help
<input type="checkbox"/>	YES	YES	
<input type="checkbox"/>	NO	NO	

Default Content

Default Value:

Value:

Validation

Validation:

Validation Error text:

After saving the above field properties, the list of metadata fields should contain the **UseJdfOutput** field as follows:

Metadata Fields
Product Asset
Help

Define metadata fields for all product definitions.

Show "Excluded" fields

Field Name	Display Label	Control Type	User Forms	Admin Forms	Reserved
ProductCode	ProductCode	Single Line Text	Control	Control	
UseJdfOutput	Enable Document Export	Dropdown List	Control	Control	

Tip: Click and drag to change the order in which fields will be displayed on the form.

Enable File Export Per Document for a Product

To enable XML or JDF file export per document for a product, you have to set the **UseJdfOutput** metadata field for the product to “Yes”. By default, this field is set to “No” (you can change this default in the metadata field settings as described above, if desired). If this field is set to “No”, no XML file will be exported per document for this product. Note that for legacy reasons, the name of this field remains **UseJdfOutput**. However, selection of JDF versus XML in the extension configuration is not relevant to the usage of this metadata field.

To enable file export per document for a product:

1. Click **Products** on the sidebar and click the link to the product on which you want to enable document export.
2. Click **Metadata** on the product settings page.



3. Choose **YES** in the **Enable Document Export** field that you created to produce an XML file when the extension is set to export one file per document.

Metadata Fields for Product-Specific Settings

Additional metadata fields can optionally be set up to allow individual products to override the extension settings for various configuration options. When exporting on a per-document basis, if a value is entered for a particular product, the value entered for the product will take precedence over the corresponding global extension value. If the value is left blank, the setting from the extension configuration will be used. The following fields can be overridden:

- **XMLTemplateName:** Must be either a full path to the template file, or a filename corresponding to a file in the template directory for the deployment.
- **PathForOutput:** The directory to which the XML will be saved.

- **OutputHotfolder:** The directory to which output files will be copied.
- **OutputPathForJdfFile:** The URL for sending via JMF.
- **OutputLocalOrNetwork:** Should be set to “Local” or “Network Share”.
- **XMLExportAuthDomain:** Domain name to be used for authentication.
- **XMLExportAuthUser:** User name to be used for authentication.
- **XMLExportPass:** Password for the user specified above.

Create these fields in the same way that you created the **UseJdfOutput** metadata field above, using the configuration options described below. Then, modify them on a Product-by-Product basis in the same way as the **UseJdfOutput** metadata field.

XMLTemplateName

Configure the **XMLTemplateName** metadata field as follows to match the screen below:

- Enter a **Field Name** of “XMLTemplateName”.
- Enter a **Display Label** of “XMLTemplateName”.
- Click **Update** to save changes.

Name and Use	
Field Name:	<input type="text" value="XMLTemplateName"/>
On User Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always hidden"/>
On Admin Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always visible"/>
Control and Options	
Control Type:	<input type="text" value="Single line text"/>
Display Label:	<input type="text" value="XMLTemplateName"/>
Width:	<input type="text" value="200"/> pixels
Style:	<input type="text" value="(default)"/>
Default Content	
Default Value:	<input type="text" value="None"/>
Validation	
Validation:	<input type="text" value="None"/>
Validation Error text:	<input type="text"/>
<input type="button" value="Update"/> <input type="button" value="Cancel"/>	

PathForOutput

Configure the **PathForOutput** metadata field as follows to match the screen below:

- Enter a **Field Name** of “PathForOutput”.
- Enter a **Display Label** of “PathForOutput”.
- Click **Update** to save changes.

Name and Use

Field Name:

On User Forms: as which is

On Admin Forms: as which is

Control and Options

Control Type:

Display Label:

Width: pixels

Style:

Default Content

Default Value:

Validation

Validation:

Validation Error text:

OutputHotfolder

Configure the **OutputHotfolder** metadata field as follows to match the screen below:

- Enter a **Field Name** of “OutputHotfolder”.
- Enter a **Display Label** of “OutputHotfolder”.
- Click **Update** to save changes.

Name and Use	
Field Name:	<input type="text" value="OutputHotfolder"/>
On User Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always hidden"/>
On Admin Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always visible"/>
Control and Options	
Control Type:	<input type="text" value="Single line text"/>
Display Label:	<input type="text" value="OutputHotfolder"/>
Width:	<input type="text" value="200"/> pixels
Style:	<input type="text" value="(default)"/>
Default Content	
Default Value:	<input type="text" value="None"/>
Validation	
Validation:	<input type="text" value="None"/>
Validation Error text:	<input type="text"/>
<input type="button" value="Update"/> <input type="button" value="Cancel"/>	

OutputPathForJdfFile

Configure the **OutputPathForJdfFile** metadata field as follows to match the screen below:

- Enter a **Field Name** of “OutputPathForJdfFile”.
- Enter a **Display Label** of “OutputPathForJdfFile”.
- Click **Update** to save changes.

Name and Use

Field Name:

On User Forms: as which is

On Admin Forms: as which is

Control and Options

Control Type:

Display Label:

Width: pixels

Style:

Default Content

Default Value:

Validation

Validation:

Validation Error text:

OutputLocalOrNetwork

Configure the **OutputLocalOrNetwork** metadata field as follows to match the screen below:

- Enter a **Field Name** of “OutputLocalOrNetwork”.
- Under **Control and Options**, choose the **Control Type** “Drop down list box”.
- Enter a **Display Label** of “OutputLocalOrNetwork”.
- In options, add two new table values: Display Text “Local” with a value of “Local” and Display Text “Network Share” with a value of “Network Share”.
- Click **Update** to save changes.

Name and Use

Field Name:

On User Forms: as which is

On Admin Forms: as which is

Control and Options

Control Type:

Display Label:

Width: pixels

Style:

Options:

2 items

Display Text	Value
<input type="text" value="Local"/>	<input type="text" value="Local"/>
<input type="text" value="Network Share"/>	<input type="text" value="Network Share"/>

[Help](#)

Default Content

Default Value:

Validation

Validation:

Validation Error text:

XMLExportAuthDomain

Configure the **XMLExportAuthDomain** metadata field as follows to match the screen below:

- Enter a **Field Name** of “XMLExportAuthDomain”.
- Enter a **Display Label** of “XMLExportAuthDomain”.
- Click **Update** to save changes.

Name and Use	
Field Name:	<input type="text" value="XMLExportAuthDomain"/>
On User Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always hidden"/>
On Admin Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always visible"/>
Control and Options	
Control Type:	<input type="text" value="Single line text"/>
Display Label:	<input type="text" value="XMLExportAuthDomain"/>
Width:	<input type="text" value="200"/> pixels
Style:	<input type="text" value="(default)"/>
Default Content	
Default Value:	<input type="text" value="None"/>
Validation	
Validation:	<input type="text" value="None"/>
Validation Error text:	<input type="text"/>
<input type="button" value="Update"/> <input type="button" value="Cancel"/>	

XMLExportAuthUser

Configure the **XMLExportAuthUser** metadata field as follows to match the screen below:

- Enter a **Field Name** of “XMLExportAuthUser”.
- Enter a **Display Label** of “XMLExportAuthUser”.
- Click **Update** to save changes.

Name and Use

Field Name:

On User Forms: as which is

On Admin Forms: as which is

Control and Options

Control Type:

Display Label:

Width: pixels

Style:

Default Content

Default Value:

Validation

Validation:

Validation Error text:

XMLExportPass

Configure the **XMLExportPass** metadata field as follows to match the screen below:

- Enter a **Field Name** of “XMLExportPass”.
- Under **Control and Options**, choose the **Control Type** “Password text”.
- Enter a **Display Label** of “XMLExportPass”.
- Click **Update** to save changes.

Name and Use	
Field Name:	<input type="text" value="XMLExportPass"/>
On User Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always hidden"/>
On Admin Forms:	<input type="text" value="Include"/> as <input type="text" value="HTML Control"/> which is <input type="text" value="Always visible"/>
Control and Options	
Control Type:	<input type="text" value="Password text"/>
Display Label:	<input type="text" value="XMLExportPass"/>
Width:	<input type="text" value="200"/> pixels
Style:	<input type="text" value="(default)"/>
Default Content	
Default Value:	<input type="text" value="None"/>
Validation	
Validation:	<input type="text" value="None"/>
Validation Error text:	<input type="text"/>
<input type="button" value="Update"/> <input type="button" value="Cancel"/>	

Modify the XML Template

The XML and JDF file is created from the XML template you chose when configuring the extension. This sample template is located in the WebPages\Custom\XmlOutputTemplates directory for the deployment. Open this file in a text editor to modify it. If you have a product specific version of this extension for Promail, Printstream, Avanti, etc. you will also have a product specific template which would be the template we used to qualify that particular integration. You can still modify those templates to suit your needs.

The sample XML template shows some of the information that can be obtained from the Storefront order or document. The extension uses a substitution method to fill in the required values according to tagged variables. This is similar to how Notifications work in Storefront. It might be helpful to review the “Customizing Notifications” and “Example of Notifications” sections of the *Pageflex Storefront Deployer Guide* to familiarize yourself with how variable fields work in Notifications as they are used in a similar way in this extension. For example the following block from the XML sample file for an order is as follows:

```

<Order>
  <ExternalID><Order:OrderID></ExternalID>
  <UserID><OrderProperty:OwnerID></UserID>
  <UserEmail><User:UserProfileEmailAddress></UserEmail>
  <OrderQuantity><Order:TotalQty></OrderQuantity>
  <CustomerPO><Payment:PaymentPurchaseOrderNumber></CustomerPO>
  <LineItemCount><Order:DocumentCount></LineItemCount>
  <ShippingMethod><Shipping:ShippingDeliveryMethod></ShippingMethod>
  <OrderDate><Order:OrderDate></OrderDate>
  <OrderTime><Order:OrderTime></OrderTime>
  <OrderTotal><Order:PriceTotal></OrderTotal>
  <Shipping><OrderProperty:ShippingCharge></Shipping>
  <Handling><OrderProperty:HandlingCharge></Handling>
  <Tax><OrderProperty:Tax></Tax>

```

The external tags are the XML tags that will be output in the file. For example: ExternalID, UserID, and UserEmail. The values within these tags, such as Order:OrderID, OrderProperty:OwnerID, and User:UserProfileEmailAddress, are the Storefront variables that will be populated with a combination of values, including data entered by a user, product information, and Storefront assigned values. For example, a completed block in an XML file for the above might look like this:

```

<Order>
  <ExternalID>G-CDB8B78C</ExternalID>
  <UserID>1</UserID>
  <UserEmail>pat@sample.com</UserEmail>
  <OrderQuantity>100</OrderQuantity>
  <CustomerPO></CustomerPO>
  <LineItemCount>1</LineItemCount>
  <ShippingMethod>Custom_01</ShippingMethod>
  <OrderDate>20080813</OrderDate>
  <OrderTime>1555</OrderTime>
  <OrderTotal>32.60</OrderTotal>
  <Shipping>2.60</Shipping>
  <Handling>4.00</Handling>
  <Tax>0.00</Tax>

```

Since the XML file may include information for a multitude of items, tags can be nested within loops to provide data for all the items in an order. See the Loops section below for details.

Variable Substitution Syntax

The substitution syntax for filling elements within an XML form is:

```
<Category:FieldName>
```

Category is one of the 11 categories of fields described below, and **FieldName** is a specific field relevant to that category. For example, if the subject of the e-mail is set to be:

```
Ticket for document <DocumentProperty:ExternalID>
```

The placeholder `<DocumentProperty:ExternalID>` will be replaced with the external ID property of the relevant document:

```
Ticket for document D-B7425725
```

Note that both the category and field names are case-sensitive.

Loops

Document, Shipment, Output File, and Metadata loops can all be used to include variable information for multiple items in an order. See below for details.

Document Loops

Document Information tags must be used within a document loop tag in XML and JDF files.

A document loop sets a condition to repeat information for each document in an order with the `<DocumentLoop>` tag.

For example, you could include the ID and price of every document in an order with the following tag in the body of your e-mail:

```

<DocumentLoop>
  <DocumentProperty:ExternalID> - $<DocumentProperty:Price>
</DocumentLoop>

```

If your order had three documents: D-1, which cost \$100, D-2, \$200, and D-3, \$300, the document loop code block would expand to the following when the notification is sent:

D-1 - \$100

D-2 - \$200

D-3 - \$300

The tags inside the loop are evaluated once for each document in the order.

Metadata Loops

Metadata loops are a special XML tag used to include repeated metadata information about a product. For example, if you have an optional finishing process on some products but not on others, you could use a metadata loop to include this information. If a `<DocumentLoop>` tag is used, the `<MetadataLoop>` must be inside that `<DocumentLoop>`.

There are two areas to address: setup of the XML/JDF template, and setup of the product metadata. Here's an example XML snippet:

```
<FinishingProcesses>
<MetadataLoop name="MyFinishingProcess">
  <FinishingProcess>
    <Name>$1</Name>
    <Value>$2</Value>
  </FinishingProcess>
</MetadataLoop>
</FinishingProcesses>
```

In your product, create a metadata field (for example "MyFinishingProcess") and assign it a value. The MetadataLoop will be repeated for every line in the value. Each line can contain any number of elements, separated by a | character. To continue, set the value of the "MyFinishingProcess" field to:

```
SORT|<PrintingField:PrintingQuantity>
FOLD|TRIPLE
```

Now, each time a document based on this product is ordered, the example XML snippet is replaced with:

```
<FinishingProcesses>
  <FinishingProcess>
    <Name>SORT</Name>
    <Value>500</Value>
  </FinishingProcess>
  <FinishingProcess>
    <Name>FOLD</Name>
    <Value>TRIPLE</Value>
  </FinishingProcess>
</FinishingProcesses>
```

As you can see, the text inside the metadata field can contain variables expressed in the same syntax as other XML variables described in this document.

Shipment Loops

Shipment fields do not need to be enclosed in a shipment loop tag unless you are using split shipping.

Example

```
<ShipmentLoop>
Shipping Delivery Method: <Shipping:ShippingDeliveryMethod>
Name: <Shipping:ShippingFirstName> <Shipping:ShippingLastName>
Address:
<Shipping:ShippingAddress1>
<Shipping:ShippingAddress2>
<Shipping:ShippingCity>, <Shipping:ShippingState>
<Shipping:ShippingPostalCode>
</ShipmentLoop>
```

Additionally, you can nest a document loop inside of a shipment loop to provide information about documents sent to differing addresses, for example:

```
<ShipmentLoop>
The Following documents:
<DocumentLoop>
<Document:ExternalID>
<Document:ProductName>
</DocumentLoop>
Will be sent to the following address:
<Shipping:ShippingFirstName>
<Shipping:ShippingLastName>
<Shipping:ShippingAddress1>
<Shipping:ShippingCity>
<Shipping:ShippingPostalCode>
<Shipping:ShippingDeliveryMethod>
</ShipmentLoop>
```

The markup text above creates the following result:

```
The Following documents:
D00001072
BusinessCard
D00001073
Party Invitation
Will be sent to the following address:
John
Smith
Pageflex
Marlborough
```

01752

Overnight (next day air): \$12.60

The Following documents:

D00001074

Travel Flyer

D00001075

Property Details Flyer

Will be sent to the following address:

Pat

Sampleton

Sample Corp.

Schenectady

12345

Standard (5-7 business days): \$6.00

Shipment field categories

Shipping information should now use the *Shipment* or *ShipmentField* (which are synonymous) and *ShipmentProperty* notation in your template, instead of the older *Shipping* notation. So to retrieve the shipping method, the placeholder would be **<Shipment:ShippingDeliveryMethod>**.

When exporting one file for an entire order, you must use the `<ShipmentLoop>` and `</ShipmentLoop>` tags. You can (although it is not required) enclose the `<DocumentLoop>` and `</DocumentLoop>` within the Shipment Loop. When this is done, the Shipment Loop will create separate blocks of XML for each shipment, and the fields within the Document Loop inside will be specific to the documents in the respective shipments. Note that if the Document Loop is instead outside the Shipment Loop, it will instead process all of the documents in the order. See the Loops section for a description of ShipmentLoops and DocumentLoops.

Output File Loops

Output file loops are particular to the XML\JDF extension. They do not work in other extensions or in Storefront. You must use the special extension fields described below within an `<OutputFileLoop>`. Fields within this loop will appear once for each output file associated with the document.

If you are using a `<DocumentLoop>`, the `<OutputFileLoop>` must be enclosed. For example:

```
<DocumentLoop>
  <OutputFileLoop>
    <OutputProperty:OutputPathForJdfFile>
  </OutputFileLoop>
</DocumentLoop>
```

Special Output Fields

Two special fields are included in this extension:

- `<OutputProperty:OutputPathForJdfFile>`: You would use this if need a path other than the actual path the file is being written to. For example you may be writing the files to D:\Output but the JDF requires the path on the print server, say C:\Files\Output to be in the RunList. This is the tag you should use.
- `<OutputProperty:ShortFileName>`: Some systems need just the file name (not the path) as part of the RunList attributes. You can use this tag to get that file name in your JDF file.

To use these two tags, you must place them inside an `<OutputFileLoop>`, which must itself reside inside your `<DocumentLoop>` if present. Fields within this loop will appear once for each output file associated with the document. See the Loops section for a description of OutputFileLoops and DocumentLoops.

- `<DocumentProperty:ListOfOutputFiles>`: This tag *does not* go inside an Output File Loop. This tag invokes a function which goes to the final output location, enumerates the files and returns a list of files separated by newlines.

Document field categories

The following categories pertain to details about the ordered document. If you are exporting your XML or JDF file once per order or once per shipment, then document properties must be inside a `<DocumentLoop>` tag closed with `</DocumentLoop>`. All XML inside this tag will be repeated for each document in the order. If you are exporting one file per document, the `<DocumentLoop>` tag is unnecessary.

- **DocumentProperty**: If you use the category DocumentProperty, then the field name can be any of the DocumentProperty fields you can access with the API, for example, Description, the shopping cart Description column for this document.
- **VariableValue**: If you use the category VariableValue, then the field name can be any field from the Form Filling (for Pageflex print products) or Description (for hosted and user-uploaded PDF products). Note that if you include a field that is not present on the form filling or description step of the product on which a particular document is based, its value will be returned as blank in the e-mail.
- **PrintingField**: If you use the category PrintingField, then the field name can be any field from the Printing step of the document. Note that if you include a field that is not present in the printing step of the product on which a particular document is based, its value will be returned as blank in the e-mail.
- **ProductField**: If you use the category ProductField, then the field name can be any field from the metadata fields of the product. Note that if you include a field that is not present in the metadata of the product on which a

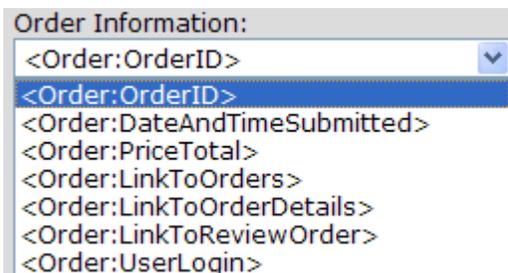
particular document is based, its value will be returned as blank in the e-mail.

- **ProductProperty:** If you use the category ProductProperty, then the field name can be any of the ProductProperty fields you can access with the API, for example, DisplayName, the product's name in the product catalog.

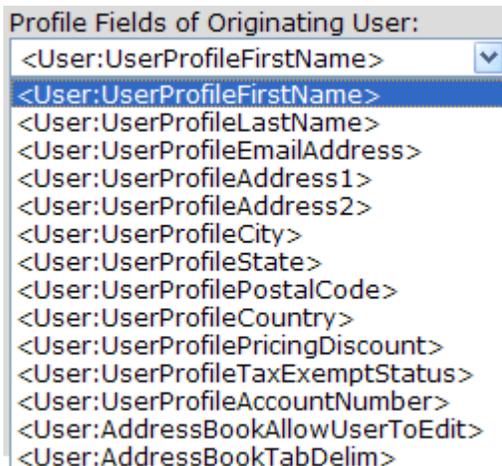
Order field categories

The following categories pertain to details about the order that the document is part of, and the user who placed the order.

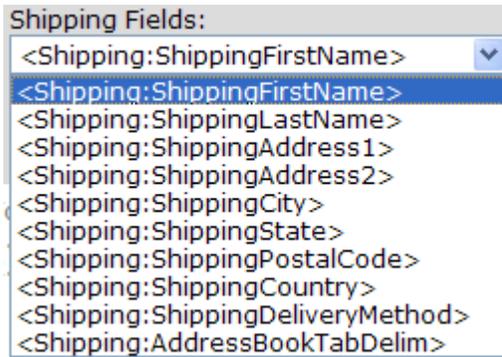
- **Order:** You can use any FieldName that is available in the Storefront Notifications Set Message screen, as shown below.



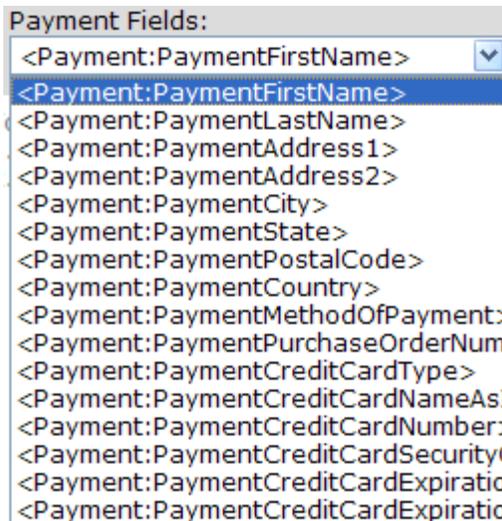
- **User:** These are details about the user who placed the order. As in the Storefront Notifications Set Message screen, the FieldName can be any user profile field that you have set up, in addition to the built-in fields.



- **Shipment:** As in the Storefront Notifications Set Message screen, the FieldName can be any shipping field that you have set up, in addition to the built-in fields, except that the prefix used in the template must be *Shipment* or *ShipmentField* and not *Shipping* as shown below.



- **Payment:** As in the Storefront Notifications Set Message screen, the FieldName can be any payment field that you have set up, in addition to the built-in fields.



- **UserProperty:** The Storefront API chapter in the *Pageflex Programming Reference Guide* includes numerous properties that you can access, described in the Accessing Fields and Properties section. If you use the category UserProperty, then the field name can be any of the UserProperty fields you can access with the API. For example, UserProperty:LogonName will access the logon name of the user who placed the order.
- **OrderProperty:** If you use the category OrderProperty, then the field name can be any of the OrderProperty fields you can access with the API, for example, OrderProperty:HandlingCharge.

Extension Operation

After the extension has been configured and the user places an order, XML or JDF files are output to the hotfolder directory you entered on the output configuration page. The types of files output depend upon the entries specified in the extension configuration.

For example, the screen shot below shows a directory with XML and JDF output files created when the order was rendered. Note that the files use the external order id as the file name. In the pdf subdirectory are the document PDF and JDF files.

pdf	File Folder	8/14/2008 9:00 AM
G-CDB4B78C.xml	2 KB XML Document	8/13/2008 4:15 PM
G-CDB6B78C.xml	2 KB XML Document	8/13/2008 4:10 PM
G-CDB8B78C.xml	2 KB XML Document	8/13/2008 3:59 PM
G-CDBAB78B.jdf	2 KB JDF File	8/13/2008 4:23 PM
G-CDC2B78C.jdf	2 KB JDF File	8/13/2008 4:21 PM

Document PDF files are copied from Storefront to the local directory specified. If the metadata property *UseJdfOutput* is set to *YES* then a corresponding JDF or XML file will be created as well.

Note: Files output with the same name will be overwritten with the latest. Versioning of files will not be done.

D-CDB6B78F_00001.pdf	425 KB	Adobe Acrobat Doc...	8/14/2008 9:00 AM
D-CDB7B78F_00001.pdf	425 KB	Adobe Acrobat Doc...	8/13/2008 4:45 PM
D-CDB8B78F_00001.pdf	425 KB	Adobe Acrobat Doc...	8/13/2008 4:39 PM
D-CDB9B78F_00001.pdf	425 KB	Adobe Acrobat Doc...	8/13/2008 4:37 PM
D-CDBAB78F_00001.pdf	425 KB	Adobe Acrobat Doc...	8/13/2008 4:25 PM
D-CDCAB789_00001.pdf	425 KB	Adobe Acrobat Doc...	8/13/2008 4:15 PM
D-CDCAB797_00001.pdf			8/13/2008 4:23 PM
D-CDCAB798_00001.pdf			8/13/2008 4:21 PM
D-CDB6B78F.jdf			8/14/2008 9:00 AM
D-CDB7B78F.jdf			8/13/2008 4:45 PM
D-CDB8B78F.jdf	3 KB	JDF File	8/13/2008 4:39 PM
D-CDB9B78F.jdf	3 KB	JDF File	8/13/2008 4:37 PM
D-CDBAB78F.jdf	3 KB	JDF File	8/13/2008 4:30 PM

Type: Adobe Acrobat Document
 Title: Pageflex Server [document: D-CDBAB78F_00001]
 Date Modified: 8/13/2008 4:25 PM
 Size: 424 KB

Appendix A - PostCatcher

Many times we hear that the HTTP posting is “not working”. It can be difficult to determine where in the multi-machine configuration the problem lies. To that end we put together this very simple utility to help test that our extension is posting the XML correctly.

It makes sense to start troubleshooting on the same machine the Storefront deployment is on. That eliminates network issues. Create a virtual directory on your Storefront server. It does not matter where you put it or what you name it. Copy the supplied PostCatcher.aspx into the virtual directory. By default the page simply listens for the incoming XML and streams it back to the caller. So if posting is successful your XML will show up in the response recorded in the Storefront log. There is also a section of code in there that allows capturing the XML to a file. Find the block where it says “uncomment this” and uncomment that code. You also need to put in a file path that is sane for where the file is installed and permissions need to be such that the web process is allowed to create the file.

```
// Read XML posted via HTTP
StreamReader reader = new StreamReader(Page.Request.InputStream);
String xmlData = reader.ReadToEnd();
reader.Close();

/**
 * uncomment this and change the file path to something sane for your system
 * to write the XML to a file
 StreamWriter outfile = new StreamWriter(@"C:\temp\log.txt", false);
 outfile.Write(xmlData);
 outfile.Close();
 **/

//write back the XML we recieved as a response
Response.Write(xmlData);
}
~"
```

In the extension configuration on the Storefront deployment simply enter the fully qualified URL where the PostCatcher is installed. Example:

<http://<server>/<Virtual Directory>/Postcatcher.aspx>

HTTP Post Configuration

Address to post to :

com/PostTest/PostCatcher.aspx

Post only (do not write file to disk)

TIP: If you are posting the exported XML to a web service, enter the URL here.

PDF Output Configuration

Then place an order and process it through to where XML will be exported. After a short time you should be able to log at your Storefront logs and see an entry something like this:

Event ID	EventCategory	LogonName	EventDateTime	EventMsgExtra
761		test	10/12/2011 3:05:39 PM	XML post acknowledgement: <Order> <ExternalID>G00001042</ExternalID> <UserID>1</UserID> <UserEmail>pat@sample.com</Us...
760		test	10/12/2011	Price After = 36 (from calcprices.services.pageflex.com)

Alternately there should be an error which should at least point in the next troubleshooting direction.

Version History

3.6.4 10/3/2012

- Changed javascript int PageLoad for "Regenerate XML" button to accommodate the new 8.0.0 table structure

3.6.3 9/27/2012

- Rebuilt for new PageflexServices.DLL

3.6.2 7/25/2012

- Added EPMS tag to output documents (ExportDocuments) function..

3.6.1 4/20/2012

- New build to include updated PageflexServices.dll which contains a fix for some instances of XML output not being produced.

3.6.0 2/1/2012

- Added two addition configuration options "Order Approved" and "Shipment when order approved" to produce XML
- Generation of xml when final step count is reached in OrderApproval_After2

3.5.3 2/1/2012

- Added additional configuration option to log additional debug messages
- Updated code to accommodate additional debugging

3.5.2

- Updated for Storefront 7.7.
- Added ability to send email on errors.
- Added ability to regenerate and resend XML.
- Restored "exclude Rejected Docs" which got lost somewhere\
- Added error handling to HTTP POST code.
- Included PostCatcher.aspx for testing.
- Updated documentation.

3.1.2

- Updated extension documentation.

3.1.1

- Updated for Storefront 7.6.1 and added OutputLoop capability to order-level exports

3.1.0

- Updated for Storefront 7.6 and kits

3.0.0

- Updated for Storefront 7.5 and split shipping

2.0.0

- Added Order Rendered option for file generation

1.1.1

- Output is copied to hotfolder for XML as well as JDF output

1.1

- Added MetadataLoop tag

1.0

- Initial Release

Pageflex Services Storefront Extension: XML and JDF Export

Information in this document is subject to change without notice.

NEITHER PAGEFLEX NOR MSDH INC. MAKES ANY WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Neither Pageflex nor MSDH Inc. shall be liable for errors herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

© Copyright 2009 MSDH Inc. and Pageflex, Marlborough, MA. All rights reserved. No part of this document may be photocopied, reproduced, or translated without the prior written consent of Pageflex.

MSDH Inc., 500 Nickerson Road, Marlborough, MA 01752
Pageflex by MSDH, 500 Nickerson Road, Marlborough, MA 01752
Worldwide phone number: 1-617-520-8600
Phone number in the U.S. and Canada: 800-551-0549

Internet: <http://www.pageflex.com>

Printed and bound in the United States of America.

8/31/2012