Copyright Information

Copyright © 1994-2010 Objectif Lune Inc. All Rights Reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any other language or computer language in whole or in part, in any form or by any means, whether it be electronic, mechanical, magnetic, optical, manual or otherwise, without prior written consent of Objectif Lune Inc.

Objectif Lune Inc. disclaims all warranties as to this software, whether expressed or implied, including without limitation any implied warranties of merchantability, fitness for a particular purpose, functionality, data integrity or protection.

PlanetPress and PrintShop Mail are registered trademarks of Objectif Lune Inc.

PostScript and Acrobat are registered trademarks of Adobe Systems Inc.

Pentium is a registered trademark of Intel Corporation.

Windows is a registered trademark of Microsoft Corporation.

Adobe, Adobe PDF Library, Adobe Acrobat, Adobe Distiller, Adobe Reader, Adobe Illustrator, Adobe Photoshop, Optimized Postscript Stream, the Adobe logo, the Adobe PDF logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Trademarks of other companies mentioned in this documentation appear for identification purposes only and are the property of their respective companies.

<table>
<thead>
<tr>
<th>Title</th>
<th>PrintShop Mail Web - MIS Integration Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td>Revision</td>
<td>2010/01/04</td>
</tr>
</tbody>
</table>
Table of contents

1 Introduction .................................................................................................................................................................................................... 1
2 MIS Connectors ................................................................................................................................................................................................ 2
  2.1 Enabling a MIS connector ......................................................................................................................................................... 3
  2.2 Installed modules ........................................................................................................................................................................ 4
  2.3 Properties .................................................................................................................................................................................... 5
3 XML to File connector ........................................................................................................................................................................ 6
  3.1 XML (eXtensible Markup Language) .......................................................................................................................................... 7
  3.2 Options ........................................................................................................................................................................................ 8
    3.2.1 General information ............................................................................................................................................................... 8
    3.2.2 Push method ............................................................................................................................................................................ 8
    3.2.3 XML options .......................................................................................................................................................................... 9
    3.2.4 Post options ........................................................................................................................................................................... 10
    3.2.5 File options .......................................................................................................................................................................... 11
4 Sample POST receive script ........................................................................................................................................................... 12
1 Introduction

PrintShop Mail Web comes with a framework for managing and describing modules. Modules are used to add or customize functionality in specific areas of the application. One of these areas is the integration of Management Information Systems (MIS).

The MIS Connector Framework is built on top of this module framework and is called at order creation and order status changes. MIS Connectors are used to automatically transfer order information from PrintShop Mail Web to external MIS systems.

This document describes the MIS Connector Framework and the options of the XML to File connector.
2 MIS Connectors

MIS connectors add functionality to the PSM Web order creation and order status change processes. The following diagram outlines these processes.

The system performs the following tasks:

1. Order information is stored in the PSM Web database.
2. The system defines if email notification messages should be sent.
3. The system checks for the active MIS connector, if found the functions of this connector are called.
4. The user interface (UI) of the system is updated.
2.1 Enabling a MIS connector

In order to activate a MIS connector:

1. Log on to PSM Web using an Administrator account.
2. Click on Settings in the PSM Web menu bar, the Settings sections appears.
3. Click Settings in the Pricing and Ordering section of the Settings sub menu. The Pricing and Ordering page appears. This page shows information about the current pricing and ordering settings as well as the active MIS connector (if one is selected).
4. Click Edit.
5. Select a connector from the Active Module pull down menu in the MIS Connector section. The pull down menu shows the available MIS connectors.
6. Click Save to activate the module.

**Pricing and Ordering**

![Settings Form]

Select a module from the list and click Save to activate the connector

**Note:** Orders that reside in your PSM Web environment upon enabling a connector are not handled by the connector and thus their information is not pushed to an external application.
2.2 Installed modules

To view the installed modules:

1. Log on to PSM Web using an Administrator account.
2. Click on Settings in the PSM Web menu bar, the Settings sections appears.
3. Click Overview in the Modules section of the Settings sub menu. The Modules Overview page appears.
   This page shows an overview of the installed modules. The modules are grouped by their module type.
4. Click the plus-icon in front of a module type to view the available modules for that type.
5. To access the properties page of a module simply click its name.

The Modules overview page
2.3 Properties

The properties of a connector differ from connector to connector. They depend on the options and functions of that specific module.

To view and modify the properties of a connector (module) do the following:

1. Log on to PSM Web using an Administrator account.
2. Click on Settings in the PSM Web menu bar, the Settings sections appears.
3. Click Overview in the Modules section of the Settings sub menu. The Modules Overview page appears. This page shows an overview of the installed modules. The modules are grouped by their module type.
4. Click the plus-icon in front of a module type to view the available modules for that type.
5. To access the properties page of a module simply click its name.

The following figure shows the properties page of the XML to File connector.

![Properties page of the XML to File connector](image-url)
3 XML to File connector

The XML To File connector is a generic MIS connector that ships with PrintShop Mail Web. It uses common technologies and methods to communicate between applications. The XML To File connector has the following features:

- Write order information to a XML file.
- Use a HTTP POST call to submit XML data to a predefined URL.
- A combination of the above (write and POST)
- Apply a XSLT style sheet to the XML data before writing it to disk and/or submitting the data (optional)
### 3.1 XML (eXtensible Markup Language)

XML stands for eXtensible Markup Language. Since XML data is stored in plain text format, XML provides a software- and hardware-independent way of sharing data. This makes it much easier to create data that different applications can work with. It also makes it easier to expand or upgrade a system to new operating systems, servers, applications, and new browsers.

The XML to File connector collects the order information and writes this data to XML files. The sample below shows a part of a XML file generated by the XML to File connector. The names of the elements match the fields names in the PrintShop Mail Web database tables.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<PrintShopCommunication>
  <PSW_to_MIS>
    <Order>
      <Properties>
        <fnCompanyID>1</fnCompanyID>
        <fcCompanyName>OL BV</fcCompanyName>
        <fcShippingContactName>Peter Parker</fcShippingContactName>
        <fcShippingEmailAddress>pparker@finco.com</fcShippingEmailAddress>
        <fnShippingCompanyID>1</fnShippingCompanyID>
        <fcShippingCompanyName>OL BV</fcShippingCompanyName>
        <fnShippingDepartmentID>0</fnShippingDepartmentID>
        <fcShippingAddress1>Daltonstraat 42-44</fcShippingAddress1>
        <fcShippingAddress2></fcShippingAddress2>
        <fcShippingZipCode>3846 BX</fcShippingZipCode>
        <fcShippingCity>Harderwijk</fcShippingCity>
        ...
      </Properties>
      <Jobs>
        <Job>
          <fnJobID>1</fnJobID>
          <fnTemplateID>1</fnTemplateID>
          <fnPersonID>0</fnPersonID>
          <fbDatabaseInput>0</fbDatabaseInput>
          <fbUserInput>1</fbUserInput>
          <fcTemplateName>Business Card</fcTemplateName>
          <fnEngineID>4</fnEngineID>
          <fnPublicationTypeID>1</fnPublicationTypeID>
          <fcTemplateProductCode></fcTemplateProductCode>
          <fcEngineShort>PSM</fcEngineShort>
          <fcPublicationType>Business Cards</fcPublicationType>
          <fcPublicationTypeProductCode></fcPublicationTypeProductCode>
          <UserInputFields>
            <Field name="First">Peter</Field>
            <Field name="Last">Parker</Field>
            <Field name="Email">pparker@finco.com</Field>
            <Field name="Title">Web Developer</Field>
          </UserInputFields>
        </Job>
      </Jobs>
    </Order>
  </PSW_to_MIS>
</PrintShopCommunication>
```
3.2 Options

This section describes the following options of the XML to File MIS connector:

- General information
- Push method
- XML options
- Post options
- File options

3.2.1 General information

The first section of the properties and edit properties page shows general information about the connector. The information is static and cannot be changed through the PSM Web interface.

**XML to File**

<table>
<thead>
<tr>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General</strong></td>
</tr>
<tr>
<td>Module Type:</td>
</tr>
<tr>
<td>Status:</td>
</tr>
<tr>
<td>Version:</td>
</tr>
<tr>
<td>Date:</td>
</tr>
<tr>
<td>Name:</td>
</tr>
<tr>
<td>Copyright:</td>
</tr>
<tr>
<td>Creator:</td>
</tr>
<tr>
<td>Comments:</td>
</tr>
</tbody>
</table>

*The General Information section of the XML to File properties*

3.2.2 Push method

The *Push method* defines the method used for providing the order information to the outer world. You can select one of the following options:

- Write to folder
- Post
- Write to folder and Post

**Push method**

<table>
<thead>
<tr>
<th>Push method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method:</td>
</tr>
</tbody>
</table>

*The Push method section of the XML to File properties*
**Write to folder**

When the *Write to folder* option is selected an XML with the order information is created in a central location. The element names in the XML data match the fields names in the PrintShop Web database tables (tblorder). The files are stored in the *Orders* folder located in the *Files* folder of the PrintShop Web web site folder. A file is created for each order using the orders internal database ID (primary key). External systems could monitor this folder and act when files are added to this folder. For this you could use applications like PlanetPress Watch.

![Diagram of XML files created by XML to File connector]

*The location of the XML files created by the XML to File connector*

**Post**

The XML to File connector has the ability to POST the XML data to a HTTP server. This enables the connector to be truly server/application independent. All the HTTP server needs to do is have the capability to receive HTTP POST calls and be able to pass the call on to some form of program (ASP, CGI, Java Servlets, etc).

The combination of XML via HTTP calls acts as a kind of middle ware or glue to tie systems together.

The connector lets you specify the URL of the HTTP server. A sample POST receive script (written in PHP) is stored in the folder of the XML to File connector.

**3.2.3 XML options**

XSLT is a language for transforming XML documents into XML documents or even plain text. With XSLT you can add/remove elements and attributes to or from the output file. You can also rearrange and sort elements, perform tests and make decisions about which elements to remove and keep. More information can be found at: [http://www.w3schools.com/](http://www.w3schools.com/)

By activating the *Apply XSLT style sheet* option the connector will apply the XSLT style sheet stated in the *Style sheet* text area to the XML output. It is applied before writing the information to disk or before sending it using a HTTP POST call. This way the XML structure can be modified to match the requirements of the receiving application.
The XML options section

Sample XML files may be obtained by setting up a demo version of PSM Web on your local machine and using the *Write to file* option. The resulting file than can be used in an application like XML Spy or Cooktop to develop the XSLT style sheet.

```xml
<!--This is an sample stylesheet-->  
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">  
<xsl:output method="xml" indent="yes"/>
<xsl:template match="/">  
  <PrintShopCommunication>
    <Order>
      <xsl:attribute name="id"><xsl:value-of select="//fnOrderID"></xsl:attribute>
      <xsl:attribute name="statusid"><xsl:value-of select="//fnOrderStatusID"></xsl:attribute>
      <xsl:attribute name="status"><xsl:value-of select="//fcOrderStatus"></xsl:attribute>
    </Order>
    <Information>
      <Created><xsl:value-of select="//fdCreated"></xsl:Created>
    </Information>
  </PrintShopCommunication>
</xsl:template>
</xsl:stylesheet>
```

The sample XSLT code creates the following XML output:

```xml
<PrintShopCommunication>   <Order id="19" statusid="1" status="cOrder">     <Information>       <Created>2007-10-08 16:01:41</Created>     </Information>   </Order> </PrintShopCommunication>
```

### 3.2.4 Post options

The URL field lets you state the HTTP server address that is used when the POST method is selected.
3.2.5 File options

The File extension field lets you specify the extension that is used for the data file. By default the file extension will be XML as the connector exports XML files. A XSLT style sheet could be created to output text files (f.e. Comma separated or tab delimited), in this case you can change the file extension to txt.

<table>
<thead>
<tr>
<th>File options</th>
</tr>
</thead>
<tbody>
<tr>
<td>File extension:</td>
</tr>
<tr>
<td>xml</td>
</tr>
</tbody>
</table>

The File options
## 4 Sample POST receive script

The following script is a sample on how to receive POST data in PHP. The code receives the XML data sent by the XML to File connector and writes this information to disk using the ID of the order. The sample file can be found at the following location:

C:\Program Files\PrintShop Mail Suite 7\PrintShop Mail Web\Website\modules\mis\xml_to_file_mis\sample

```php
<?php
/**
 * Sample POST receive page
 * This demo script creates a xml file based on the OrderID of the received XML.
 * The file is saved directly under the "C" directory
 * To use this script, the following must be applied
 * The module "XML to File" must be activated on "Settings"
 * Either "Post" or "Move to folder and Post" must be selected as "PushMethod"
 * "http://localhost/modules/mis/xml_to_file_mis/sample/receive.php" must be entered in "URL"
 * Finally, as you change the status of the order, then "OrderID.xml"
 * will be created in "C" directory
 */

$aJobXML = simplexml_load_string($_POST['PSW_XML']);
if ($aJobXML === false) {
    echo "Cannot parse XML";
    exit();
}

$aOrderID = $aJobXML->xpath('//fnOrderID');
if ($aOrderID === false) {
    echo "Cannot extract fnOrderID from XML";
    exit();
}

cPath = "C:" . intval($aOrderID[0]) . ".xml";
$handle = @fopen($cPath, 'w');
if ($handle === false) {
    echo "Cannot open or create xml file ($cPath)"
    exit();
}

$nWritten = @fwrite($handle, $_POST['PSW_XML']);
if ($nWritten !== FALSE) {
    echo "Cannot write to xml file ($cPath)"
}
fclose($handle);
chmod($cPath, 0777);
?>
```

The `$_POST` variable is an array of variable names and values sent by the HTTP POST method. PrintShop Mail Web sends the XML stream in the `PSW_XML` POST variable. The script reads the value of this variable and stores the data in a local variable.

The `simplexml_load_string` function is used to read the XML data and to define the ID of the order. The SimpleXML extension of PHP provides a set of functions to convert XML to an object that can be processed with normal property selectors and array iterators. More information can be found at the following url: [http://www.php.net/simplexml](http://www.php.net/simplexml)
A xpath expression is used to retrieve a specific node of the XML order data (the fnOrderID element). You can use the same method to retrieve any of the elements from the XML data and use it values according to your needs.

At the end the XML data is written to a XML file using the order id as the file name.