

# PrintShop Mail Connect

OL™ Software

## User Guide

Version: 1.5



# PrintShop Mail Connect

OL™ Software

User Guide

Version 1.5

Last Revision: 2017-07-06

Objectif Lune, Inc.

2030 Pie-IX, Suite 500

Montréal, QC, Canada, H1V 2C8

+1 (514) 875-5863

[www.objectiflune.com](http://www.objectiflune.com)

All trademarks displayed are the property of their respective owners.

© Objectif Lune, Inc. 1994-2017. All rights reserved. No part of this documentation may be reproduced, transmitted or distributed outside of Objectif Lune Inc. by any means whatsoever without the express written permission of Objectif Lune Inc. Objectif Lune Inc. disclaims responsibility for any errors and omissions in this documentation and accepts no responsibility for damages arising from such inconsistencies or their further consequences of any kind. Objectif Lune Inc. reserves the right to alter the information contained in this documentation without notice.

# Table of Contents

<b>Table of Contents</b> .....	<b>5</b>
<b>Welcome to PrintShop Mail Connect 1.5</b> .....	<b>10</b>
Icons used in this guide .....	10
<b>Setup And Configuration</b> .....	<b>12</b>
System and Hardware Considerations .....	12
System Requirements .....	12
Environment considerations .....	13
Database Considerations .....	15
Network considerations .....	18
Language and Encoding considerations .....	18
Performance Considerations .....	19
Installation and Activation .....	21
Installation Pre-Requisites .....	21
User accounts and security .....	22
The Importance of User Credentials on Installing and Running PrintShop Mail Connect ..	22
Installing PrintShop Mail Connect on Machines without Internet Access .....	24
Installation Wizard .....	26
How to Run Connect Installer in Silent Mode .....	28
Activating a License .....	31
Migrating to a new workstation .....	34
Uninstalling .....	39
Important Note: Stop any Anti-Virus Software before uninstalling Connect. ....	39
Impacts upon other Applications and Services .....	40
Uninstallation Wizard .....	40
<b>The Designer</b> .....	<b>41</b>
Basic Steps .....	41
Templates .....	42
Contexts .....	47
Sections .....	48
Features .....	52
Print .....	53
Pages .....	53
Headers, footers, tear-offs and repeated elements (Master page) .....	54

Stationery (Media) .....	54
Creating a Print template with a Wizard .....	54
Print context .....	58
Print sections .....	62
Pages .....	69
Master Pages .....	76
Media .....	80
Email .....	85
Designing an Email template .....	86
Creating an Email template with a Wizard .....	89
Email context .....	93
Email templates .....	95
Email header settings .....	98
Email attachments .....	102
Content elements .....	104
Element types .....	104
Editing HTML .....	105
Attributes .....	106
Inserting an element .....	107
Selecting an element .....	107
Styling and formatting an element .....	108
Barcode .....	109
Boxes .....	150
Business graphics .....	152
Date .....	154
Hyperlink and mailto link .....	156
Images .....	158
Table .....	163
Text and special characters .....	167
Snippets .....	168
Adding a snippet .....	169
Creating a snippet .....	169
JSON Snippets .....	170
Styling and formatting .....	170
Local formatting versus style sheets .....	170
Layout properties .....	171
Styling templates with CSS files .....	171

How to position elements .....	179
Styling text and paragraphs .....	182
Background color and/or image .....	186
Border .....	187
Colors .....	190
Fonts .....	194
Spacing .....	196
Styling a table .....	197
Rotating elements .....	200
Locale .....	201
Personalizing content .....	202
Variable data .....	203
Conditional content .....	203
Dynamic images .....	203
Snippets .....	203
Scripts .....	204
Loading data .....	204
Variable Data .....	211
Formatting variable data .....	217
Showing content conditionally .....	219
Dynamic Images .....	222
Writing your own scripts .....	223
How scripts work .....	224
Creating a new script .....	224
Writing a script .....	226
Managing scripts .....	228
Testing scripts .....	231
Optimizing scripts .....	234
Loading a snippet via a script .....	238
Control Scripts .....	240
Designer User Interface .....	254
Dialogs .....	256
Menus .....	320
Panels .....	328
Toolbars .....	343
Welcome Screen .....	347
Print Options .....	348

Job Creation Presets .....	401
Output Creation Settings .....	410
Designer JavaScript API .....	428
Designer API .....	429
Control Script API .....	429
Designer Scripts API .....	429
Control Script API .....	477
<b>Generating output .....</b>	<b>493</b>
Print output .....	493
Email output .....	493
Optimizing a template .....	494
Scripts .....	494
Images .....	494
Generating Print output .....	495
Saving Printing options in Printing Presets. ....	496
Connect Printing options that cannot be changed from within the Printer Wizard. ....	497
Print Using Standard Print Output Settings .....	497
Print Using Advanced Printer Wizard .....	499
Adding print output models to the Print Wizard .....	499
Splitting printing into more than one file .....	500
Variables available in the Output .....	501
Generating Tags for Image Output .....	507
Generating Email output .....	509
Email output settings in the Email context and sections .....	510
Email attachments .....	511
Using an ESP with PrintShop Mail Connect .....	513
<b>Print Manager .....</b>	<b>519</b>
Print Manager Introduction Video .....	519
Print Manager usage .....	519
Print Manager Interface .....	519
Menu options .....	520
Printers pane .....	521
Printer Queue pane .....	524
Printer Capabilities pane .....	525
Job Queue pane .....	526
History pane .....	527

Printer Status pane .....	528
Messages pane .....	529
Print Manager Preferences .....	529
<b>Release Notes .....</b>	<b>532</b>
Overview .....	532
Connect 1.5 Designer Enhancements and Fixes .....	532
Connect 1.5 Output Enhancements and Fixes .....	535
Connect 1.5 General Enhancements and Fixes .....	537
Known Issues .....	538
<b>Copyright Information .....</b>	<b>540</b>
<b>Legal Notices and Acknowledgments .....</b>	<b>541</b>

# Welcome to PrintShop Mail Connect 1.5

## Note

Since we are always looking for new ways to make your life easier, we welcome your questions and comments about our products and documentation. Shoot us an email at [doc@ca.objectiflune.com](mailto:doc@ca.objectiflune.com), or visit the online help: [help.objectiflune.com](http://help.objectiflune.com) and use the feedback tool at the bottom of the page.

PrintShop Mail Connect is a tool designed to optimize customer communications. It is designed to improve the creation, distribution, interaction and maintenance of your communications.

The PrintShop Mail Connect **Designer** is designed to create output for print and email within a single template and from many data types. Output presets applied outside the design phase make printing device independent.

The **Designer** has an easy-to-use interface that makes it possible for almost anyone to create multi-channel output. More advanced users may use native HTML, CSS and JavaScript.

This online documentation covers **PrintShop Mail Connect** version 1.5.

## Icons used in this guide

Icons are used throughout this guide to point your attention to certain information.

## Note

Complementary information that is not critical, but may help you better use PrintShop Mail Connect.

**Tip**

Information that is useful or suggests an easier method.

**Technical**

Information that may require specific knowledge to understand.

**Warning**

Information that is potentially critical to using PrintShop Mail Connect. Pay close attention.

# Setup And Configuration

This chapter describes the PrintShop Mail Connect installation and the different considerations that are important in regards to the installation and use of PrintShop Mail Connect.

- "System and Hardware Considerations" below
- "Installation and Activation" on page 21

## System and Hardware Considerations

There are a variety of considerations to be aware of. These are documented in the following pages:

- "System Requirements" below
- "Environment considerations" on the next page
- "Database Considerations" on page 15
- "Network considerations" on page 18
- "Language and Encoding considerations" on page 18
- "Performance Considerations" on page 19

## System Requirements

These are the system requirements for PrintShop Mail Connect 1.5

### **Operating System (64-bit only)**

- Microsoft Windows 2008/2008 R2 Server
- Microsoft Windows 2012/2012 R2 Server
- Microsoft Windows Vista
- Microsoft Windows 7
- Microsoft Windows 8.1
- Microsoft Windows 10

**Note**

Windows 8.0, Windows XP, Windows 2003 and older versions of Windows are not supported by PrintShop Mail Connect.

**Minimum Hardware Requirements**

- NTFS Filesystem (FAT32 is not supported)
- CPU Intel Core i7-4770 Haswell (4 Core)
- 8GB RAM (16GB Recommended)
- Disk Space: At least 10GB (20GB recommended)

**Note**

For tips and tricks on performance, see "Performance Considerations" on page 19.

## Environment considerations

**Virtual Machine Support**

PrintShop Mail Connect supports VMWare Workstation, VMWare Server, VMWare Player, VMWare ESX (including VMotion), Microsoft Hyper-V and Microsoft Hyper-V/Azure infrastructure environments as software installed on the Guest operating system.

**Warning**

Copying (duplicating) a Virtual Machine with Connect installed and using both images simultaneously constitutes an infringement of our End-User License Agreement.

**Note**

While some virtual machine environments (from VMWare and Microsoft) are supported, other virtual environments (such as Parallels, Xen and others) are not supported at this

time.

### **Terminal Server/Service**

PrintShop Mail Connect does not support Terminal Server (or Terminal Service) environment as possible under Windows 2000, 2003 and 2008. This is to say, if Terminal Service is installed on the server where PrintShop Mail Connect is located, unexpected behaviours may occur and will not be supported by Objectif Lune Inc.. Furthermore, using PrintShop Mail Connect in a Terminal Service environment is an infringement of our End-User License Agreement.

### **Remote Desktop**

Tests have demonstrated that PrintShop Mail Connect can be used through Remote Desktop. It is however possible that certain combination of OS could cause issues. If problems are encountered, please contact OL Support and we will investigate.

PrintShop Mail Connect 1.3 and later have been certified under Remote Desktop.

### **64-bit Operating Systems**

PrintShop Mail Connect is a 64-bit software and can only be installed on 64-bit operating systems.

### **Antivirus Considerations**

- Antivirus software may slow down processing or cause issues if they are scanning in temporary folders or those used by PrintShop Mail Connect. Please see KB-002: Antivirus Exclusions for more information.
- Antivirus software might interfere with installation scripts, notably a vbs script to install fonts. McAfee, in particular, should be disabled temporarily during installation in order for MICR fonts to install and the installation to complete successfully.

### **Windows Search Indexing Service**

Tests have concluded that the Windows Search service, used to provide indexing for Windows Search, can interfere with Connect when installing on a virtual machine. If the installation hangs during the last steps, it is necessary to completely disable this service during installation.

- Click on Start, Run.
- Type in **services.msc** and click OK.
- Locate the **Windows Search** service and double-click on it.
- Change the **Startup Type** to **Disable**, and click **Stop** to stop the service.
- Try the installation again.
- Once completely, you may re-enable the service and start it.

### Commandline switches and .ini entries

PrintShop Mail Connect is intended to work stably and reliably, based on Java and the Eclipse framework. To ensure this reliability and robustness, many Java and Eclipse parameters have been tested and tuned, which is reflected in the respective .ini entries and the used command line switches. A collection of valuable settings has been elaborated and found its entry in PrintShop Mail Connect “good switches list” (called the “whitelist”).

The protection of the end user’s system is one of our main goals and therefore we have implemented a very strict verification mechanism, which ensures, that only these whitelisted ini entries and commandline switches are accepted, when one of Connect components is started and run. Please be therefore advised, that any non-whitelisted ini entry or commandline switch will be accepted and will - if tried to be used - lead to the respective application’s “sudden death”. If you should encounter such a behaviour then please double-check your Connect log file/s for respective entries.

## Database Considerations

This page describes the different considerations and pre-requisites for the database back-end used by PrintShop Mail Connect, whether using the MySQL instance provided by the installer, or pre-existing instance.

### Using the MySQL Instance from the Installer

The MySQL Instance provided in the [Installation Wizard](#) is already pre-configured with options to provide the most stable back-end setup.

These are the specific options that have been changed in our version of “my.ini”:

- **max\_connections = 200** : PrintShop Mail Connect uses *a lot* of database connections. This number ensures that even in high volume environments, enough connections will be

available.

- **max\_allowed\_packet = 500M** : In some implementations, especially when using Capture OnTheGo, large packet sizes are required to allow transferring binary files. This substantial packet size maximum setting ensures that the data received by PrintShop Mail Connect will be able to be stored within the database.
- **character-set-server = utf8 , collation-server = utf8\_unicode\_ci , default-character-set=utf8** : These indicate database support for UTF-8/Unicode.

### Using a pre-existing MySQL Instance

If MySQL server is already installed and you wish to use it, the following should be taken into consideration:

- The MySQL account must have access to all permissions using the GRANT Command, including creating databases.
- The database configuration must include the options detailed in the "Using the MySQL Instance from the Installer" on the previous page topic above.
- The SQL instance must be open to access from other computers. This means the bind-address option should not be set to 127.0.0.1 or localhost.

### Using MS SQL Server from the Installer

- When MS SQL is selected, the default values for root user are **sa** and **1433** for the port.
- If db settings from a previous installation are found, the pre-existing settings will be displayed for the matching db type (for MS SQL settings, this will only work if they were created with Server Config Tool 1.5.0 or later, or the Connect installer 1.6.0 or later). If the db type is changed in the configuration page, the default values for this db type will be displayed. If the pre-existing db settings are set to Hsqldb, the default db type selection will be MySQL.
- Selected db settings are stored in the preferences as usual (C:\ProgramData\Objectif Lune\OI Connect\settings\ConnectHostScope\com.objectiflune.repository.eclipselink.generic.preferences)

### Updating With No Local MySQL Product

- When updating a Connect installation from 1.5.0 which contains a Server Product but no local MySQL Product, the DB Configuration Page will detect which db type was set before (especially if the db configuration was switched from MySQL to MS SQL using the

Server Configuration Tool), and default to those settings.

- On Update from 1.4.2 or earlier, the DB Configuration Page will always default to MySQL connection settings, and if the installation was manually tweaked to connect to MS SQL Server, the user has to switch to "Microsoft SQL Server" type and enter connection details again.

### Installing / Updating Connect Using a Local MySQL

- The Configuration page for the local MySQL is displayed.
- MySQL settings are pre-filled with default values if no existing MySQL db configuration is found.
- MySQL settings are pre-filled with existing db configuration settings, if they point to a MySQL db type.

### When modifying Connect

- If local MySQL is removed from an installation, the DB Configuration page will offer additionally the **Microsoft SQL Server** db type with respective default values.
- If local MySQL is added to an installation, the usual MySQL Configuration page with default values will be displayed.

### Important

If a Server Product and a MySQL Product were selected to be installed on Connect 1.5.0, and then the Server Configuration Tool is used to switch the database used by the Server to an external Microsoft SQL, then the Update to 1.6 requires an extra step. The procedure is as follows:

1. Run the **Update to Connect 1.6**. This will assume the local MySQL database needs to be updated and configured, so the user has to enter a root password on the MySQL Configuration Page (can be any password matching Connect security rules).
2. After the update, the **Connect 1.6 Setup** needs to be run once more to modify Connect.
3. On the **Product Selection** page, now the MySQL product can be unselected.
4. When stepping forward in the Wizard, the DB Configuration page will be displayed which allows to configure the Microsoft SQL Server with appropriate settings.

After this modification, the local MySQL is removed, and also the service dependency from Server to MySQL is removed.

### Note

If Connect was initially installed not containing the local MySQL product (i.e. on 1.5 installation an external MySQL was configured as database), then the Update to 1.6 will allow to select either external MySQL or external Microsoft SQL on the DB Configuration Page.

## Network considerations

The following should be taken into consideration in regards to network settings and communications

- If a local proxy is configured (in the **Internet Explorer Options** dialog, the option **Bypass proxy server for local addresses** must be checked, or some features depending on local communication will not work.

## Language and Encoding considerations

Please note the following considerations:

- **Language:**
  - PrintShop Mail Connect is currently offered in several languages. These languages can be switch between via the Preferences dialog. The current languages include:
    - English
    - French
    - German
    - Spanish
    - Italian
    - Portuguese
    - Chinese (Simplified)
    - Chinese (Traditional)
    - Japanese.

The default language is English.

The PrintShop Mail Connect help system (this document) is currently only available in English.

- **Encoding:**
  - Issues can sometimes be encountered in menus and templates when running PrintShop Mail Connect on a non-English operating system. These are due to encoding issues and will be addressed in a later release.

## Performance Considerations

This page is a comprehensive guide to getting the most performance out of PrintShop Mail Connect as well as a rough guideline to indicate when it's best to upgrade.

### Performance Analysis Details

In order to get the most out of PrintShop Mail Connect, it is important to determine how best to maximize performance. The following guidelines will be helpful in extracting the best performance from PrintShop Mail Connect before looking into hardware upgrades or extra PrintShop Mail Connect performance packs.

- **RAM Configuration:** By default, each instance of the Merge Engine and Weaver Engine is set to use 640MB of RAM. This means that regardless of speed units, if not enough memory is available, output speed might not be as expected. Assuming that the machine itself is not running any other software, the rule of thumb is the following: The total number of used memory in the machine should be pretty much the maximum available (around 95%).  
For each engine, it's necessary to modify the .ini file that controls its JAVA arguments. Edit as follows:
  - For the Merge Engine: see C:\Program Files\Objectif Lune\OL Connect\MergeEngine\Mergeengine.ini
  - For the Weaver Engine: see C:\Program Files\Objectif Lune\OL Connect\weaverengine\Weaverengine.ini
  - The parameters are -Xms640m for the minimum RAM size, -Xmx640m for the maximum RAM size. Explaining Java arguments is beyond the scope of this document. Please read references [here](#), [here](#) and [here](#) for more details (fair warning: these can get pretty technical!).
- **Template and data mapping optimization:** Some functionality offered by the Designer module are very useful, and sometimes downright awesome, but can cause the

generation of records and of contents items to slow down due to their nature. Here are some of them:

- **Loading external and network resources:** In Designer, using images, javascript or css resources located on a slow network or on a slow internet connection will obviously lead to a loss of speed. While we do our best for caching, a document with 100,000 records which queries a page that takes 1 second to return a different image each time will, naturally, slow output generation down by up to 27 hours.
- **External JavaScript Libraries:** While loading a single JavaScript library from the web is generally very fast (and only done once for the record set), actually running a script on each generated page can take some time. Because yes, JavaScript will run for each record, and often take the same time for each record.
- **Inefficient Selectors:** Using very precise ID selectors in script wizards can be much faster than using a text selector, especially on very large documents. (more details on this in another upcoming page).
- **Complex Scripts:** Custom scripts with large, complex or non-optimized loops can lead to slowing down content creation. While it is sometimes difficult to troubleshoot, there are many resources online to help learn about JavaScript performance and coding mistakes. [Here](#), [here](#), and [here](#) are a few. Note that most resources on the web are about JavaScript in the *browser*, but the greatest majority of the tips do, indeed, apply to scripts in general, wherever they are used.

## High-Performance Hardware

The following is suggested when processing speed is important. Before looking into a Performance Packs to enhance performance, ensure that the below requirements are met.

- **A physical, non-virtualized server.** VMWare servers are great for reducing the numbers of physical machines in your IT space, but they must share the hardware between each other. While you can create a virtual machine that seems as powerful as a physical, it will still be sharing hardware with any other virtual machines, and this will adversely affect performance.
- **MySQL Database on a separate machine.** MySQL's main possible bottleneck is file I/O, and as such a high-performance setup will require this server to be on a separate machine, ideally with a high-performance, low-latency hard drive. A Solid State Drive (SSD) would be recommended.
- **High-Quality 16+ GB Ram.** This is especially true when working with many server instances ("speed units") running in parallel. The more parallel processing, the more RAM is recommended.

- **4 or 8 physical cores.** We're not talking Hyper-Threading here, but physical cores. Hyper-Threading is great with small applications, but the overhead of "switching" between the virtual cores, and the fact that, well, they're virtual, means the performance is much lesser on high-power applications such as OL Connect. In short, a dual-core processor with Hyper-Threading enabled is not equivalent to a quad-core processor.

## Installation and Activation

This topic provides detailed information about the installation and activation of PrintShop Mail Connect 1.5.

### Note

A PDF version of this guide is available for use in offline installations. [Click here to download it.](#)

### Where to Obtain the Installers:

The installers for PrintShop Mail Connect 1.5 can be obtained on DVD or downloaded as follows:

- If you are a **Customer**, the installers can be downloaded from the Objectif Lune Web Activations page: <http://www.objectiflune.com/activations>
- If you are a **Reseller**, the installers can be downloaded from the Objectif Lune Partner Portal: <http://extranet.objectiflune.com/>

For information on licensing, please see [Activating your license](#).

## Installation Pre-Requisites

- Make sure your system meets the [System requirements](#).
- PrintShop Mail Version 1.5 can be installed [under a regular user account with Administrator privileges](#).
- Connect **must** be installed on an NTFS file system.

- PrintShop Mail requires Microsoft .NET Framework 3.5 already be installed on the target system.
- As with any JAVA application, the more RAM available, the faster the product will execute.

#### **Note**

From PrintShop Mail Connect Version 1.2 onwards, the new version (1.1.8) of the Update Client is included by default with all setups.

## User accounts and security

### **Permissions for PrintShop Mail Connect Designer**

PrintShop Mail Connect Designer does not require any special permissions to run besides a regular program. It does not require administrative rights and only needs permission to read/write in any folder where Templates are located.

If generating Print output, PrintShop Mail Connect Designer requires permission on the printer or printer queue to send files.

## The Importance of User Credentials on Installing and Running PrintShop Mail Connect

OL Connect and required credentials depends heavily on the Connect component and respective tasks and what sort of user credentials are needed.

First of all, it is important to distinguish between installation and run-time

### **Installation**

The Connect installer puts all required files, folders, registry entries and much more to their correct places and locations. As many of these locations are protected against malicious accesses, that very user under whose context the Connect installation is started and running, needs very extensive rights on the respective computer. This user must belong to the Local Administrators group on that machine. Here are some required capabilities, this user:

- Must be able to write into the "Programs" folder.
- Must be allowed to check for existing certificates and must also be allowed to install new ones into the global certificate store on that machine.
- Must be able to write into HKLM and any subtree of it in the registry.
- Must be able to INSTALL, START and RUN services and also to MODIFY service settings.
- Must be known in the network the machine belongs to and must also need to be able to use shared network resources like shared drives and/or printers etc.

This list may not be complete, but it gives the extent of the requirements. Generally, the local administrator of the machine will have all these credentials, but there may exist network restrictions and policies, which will block one or more of these capabilities. In such cases, the respective network administrator should provide a valid user account for the installation.

### **User Account**

The user account shall be used to later RUN one of the Connect Server flavors (Server or Server Extension). This dedicated user account has to be entered on the respective installer dialog page and must be allowed to START, STOP and RUN services on this machine. This is different from the credentials of the installation user account, which additionally requires the right to INSTALL services. Please be aware of this fact!

Additionally, the Server user must be able to access any network resources that are required for OL Connect to function properly. This includes e.g. additional drives, printers, scanners, other computers and, where appropriate, internet resources, URLs, mail servers, FTP servers, database servers and everything else planned to be used for the intended operation of Connect. The Server user is the run-time user.

### **Connect Components**

Usually, a standard end user will only be facing Connect Designer and maybe the License Activation Tool. Designer this does not require administrator rights. Either everything required to create documents or also to run some tasks will be already available (installed by the installer) or be accessible in a way, where no specific credentials are required. However some tasks like starting an email campaign will possibly require a respective account at a mail server. But this has generally nothing to do with the credentials of the Designer user.

## **Activation Tool**

To run the Software Activation Tool, administrator rights are required because this tool needs to write the license file in one of the protected folders of Windows. The tool will however allow to restart it with respective credentials if required.

## **MySQL**

MySQL database service is installed by the install user (thus again the requirement of installing, starting, running and modifying services). Once running it will just work.

## **Merge and Weaver Engines**

These components do run under the Designer (if only Designer is installed) or the Server / Extension service(s) and inherit the rights of their parent application.

## **Server (Extension) Configuration Tool**

This component needs to access the settings of the Server. As these are stored and read by the Server, it should be clear that the user used to run the Configuration tool should be the same as the Server Service user as explained above.

# **Installing PrintShop Mail Connect on Machines without Internet Access**

Installing PrintShop Mail Connect1.5 in offline mode requires some extra steps. These are listed below.

## **GoDaddy Root Certificate Authority needs to be installed.**

In order to install PrintShop Mail Connect it is necessary for the GoDaddy Root Certificate Authority to be installed (G2 Certificate) on the host machine and for this to be verified online. When a machine hosting the installation does not have access to the Internet, the installation will fail because the verification cannot be performed. To solve this problem one must first ensure that all Windows updates have been installed on the host machine. Once the Windows updates are confirmed as being up to date, then complete the following steps:

1. Go to <https://certs.godaddy.com/repository> and download the following two certificates to copy to the offline machine:

- GoDaddy Class 2 Certification Authority Root Certificate - G2 - the file is gdroot-g2.crt
  - GoDaddy Secure Server Certificate (Intermediate Certificate) - G2 - the file is gdig2.crt
2. Install the certificates: Right mouse click -> Install Certificate, and follow the steps through the subsequent wizard.
  3. Now copy the PrintShop Mail Connect installer to the offline machine and start the installation as normal

### **Windows certificate validation - Certificate Revocation List retrieval should be switched off**

For your security Objectif Lune digitally signs all relevant files with our own name and certificate. The integrity of these files is checked at various times by different, context related, methods. One of these checks, done during the installation process, uses the Windows certificate validation check. .

The Windows certificate validation process not only checks the integrity of a file against its signature, but also usually checks if the certificate itself is still valid. That check is done against the current Certificate Revocation List (CRL), which needs to be retrieved from the internet. However, if the machine in question does not have internet access, the retrieval of the CRL must fail, which will lead to subsequent validation issues.

To circumvent such issues it is **highly recommended** to switch off the CRL retrieval prior to installing Connect on machines without internet access. There is no security risk associated with this, as the CRLs would never be retrievable without internet access, anyway. Advantage of the switch will not only be found during the installation and operation of Connect, but also in some speed improvements for any application which use signed binaries.

To switch off CRL retrieval on the computer, complete the following steps:

1. Open the "Internet Options" via the Control Panel
2. Select the "Advanced" tab and scroll down to "Security" node.
3. Uncheck the entry "Check for publisher's certificate revocation" under that node.
4. Click the OK button to close the dialog.
5. Re-start the computer.

# Installation Wizard

## Starting the PrintShop Mail Connect installer

The PrintShop Mail Connect installer may be supplied as an ISO image or on a DVD:

- If an ISO image, either burn the ISO onto a DVD or unzip the contents to a folder (keeping the folder structure)
- If on a DVD, either insert the DVD and initiate the installation from there or copy the contents to a folder (keeping the folder structure)

Navigate to the PrintShop Mail\_Connect\_Setup\_x64.exe or and double-click on it. After a short while the Setup Wizard will appear as a guide through the installation steps.

### Note

PrintShop Mail Connect requires prior installation of Microsoft .NET Framework 3.5 (please refer to <https://www.microsoft.com/en-us/download/details.aspx?id=21> for more details).

## Selecting the required components

After clicking the Next button, the component selection page appears, where the different components of PrintShop Mail Connect can be selected for installation. Currently, the following are available:

- **PrintShop Mail Connect Designer:** The [Designer](#) module. It may be used as a standalone with no other installed modules, but it will not have certain capabilities such as automation and commingling.
- **Installation Path:** This is the location where modules are to be installed.

The installer can also calculate how much disk space is required for installing the selected components as well as how much space is available:

- **Disk space required:** Displays the amount of space required on the disk by the selected components.
- **Disk space available on drive:** Displays the amount of space available for installation on the drive currently in the Installation Path.

- **Recalculate disk space:** Click to re-check available disk space. This is useful if space has been made available for the installation while the installer was open.
- **Source repository location:** Displays the path where the installation files are located. This can be a local drive, installation media, or a network path.

## Selection Confirmation

The next page confirms the installation selections made. Click **Next** to start the installation itself.

## End User License Agreement

The next page displays the [End User License Agreement](#), which needs to be read and accepted before clicking **Next**.

Click **Next** to start the actual installation process. This process can take several minutes.

## Completing the installation

This screen describes a summary of the components that have been installed.

- **Configure Update Check** checkbox: This option is enabled by default. It causes the **Product Update Manager** to run after the installation is complete. This allows configuring PrintShop Mail Connect to regularly check for entitled updates.  
**Note:** this checkbox may not be available in the event that an issue was encountered during the installation.
- **Show Log...** : If an issue was encountered during the installation, click this button to obtain details. This information can then be provided to Objectif Lune for troubleshooting.
- When ready, click the **Finish** button to close the installation wizard, and initialize the Product Update Manager, if it was selected.

## The Product Update Manager

If the **Configure Update Check** option has been selected, the following message will be displayed after clicking “Finish” in the setup:

Click “Yes” to install or open the Product Update Manager where the frequency with which the updates can be checked and a proxy server (if required) can be specified.

**Note:** if the Product Update Manager was already installed by another Objectif Lune

application, it will be updated to the latest version and will retain the settings previously specified.

Select the desired options and then click **OK** to query the server and obtain a list of any updates that are available for your software.

- Note that the Product Update Manager can also be called from the “Objectif Lune Update Client” option in the Start menu.
- It can be uninstalled via Control Panel | Programs | Programs and Features.

## Product Activation

After installation, it is necessary to activate the software. See [Activating your license](#) for more information.

### Technical

Before activating the software, please wait 5 minutes for the database to initialize. If the software is activated and the services rebooted too quickly, the database can become corrupted and require a re-installation.

## How to Run Connect Installer in Silent Mode

PrintShop Mail Connect can be installed in a so called "silent mode" to allow an automated setup during a company wide roll-out or comparable situations. The trigger for the Connect Installer to run in silent mode is a text file with the fixed name **install.properties**, which is located either in the same folder as the PrintShop Mail\_Connect\_Setup\_x86\_64.exe or in the unpacked folder of the **installer.exe**.

### Note

Only the installation can be run silently. **Silent mode** does not apply to the uninstallation.

The file needs to be a properties file with the following line types:

- Comment Lines, starting with # (e.g. # The options to configure an external database)
- Key=Value pairs (e.g. install.product.0 = Connect Designer)

For supported keys, please refer to the next paragraph.

#### Note

install.properties file notation must follow Commons Configuration rules. Please refer to [Properties files](#) for more details.

### Required and Optional Properties

Here is an example of an **install.properties** file.

```
# Verbose logging
logging.verbose = true

# Product selection
install.product.0 = Connect Designer
install.product.1 = Connect Server

# Server settings
server.runas.username = Localadmin
server.runas.password = admin

# Database configuration
database.type = mysql
database.host = 192.168.116.10
database.port = 3308
database.username = root
database.password = admin
database.schema = my_ol
```

## Verbose Logging (Optional)

By default, the **Silent Installer** will log in the same way as the GUI installer. That means logging of error and warnings, and certain information during database configuration. A more verbose logging can be switched on by using **logging.verbose = true**.

## Product Selection (Optional)

By default, the **Silent Installer** will install all products which are visible to the user in the respective brand (except for the Server Extension, because only Server or Server Extension can be installed at the same time).

However, it is possible to define the products to be installed using their visible product names, and using a counter for the **install.product** property, e.g.

- install.product.0 = Connect Designer
- install.product.1 = Connect Server

Other possible product names are Connect Print Manager, Connect Server Extension and MySQL Product.

## Server / Extension configuration (required if Server / Extension is selected for install)

For both, Server and Server Extension, the user credentials who will be running the Server service need to be provided:

- server.runas.username
- server.runas.password

Additionally for the Server Extension, some properties to define the Master Server are required:

- server.master.host
- server.master.port
- server.master.authenticate = true\_or\_false
- server.master.username
- server.master.password

## Database configuration

If the MySQL Product is part of the installation, the following properties should be defined:

- database.type = mysql (required)
- database.password (required, needs to match the security rules)
- database.port (optional, the default is 3306. The defined port needs to be available.)

Additional properties are required when the configuration of an external MySQL database is required (this is the case if a Server product but not MySQL product is installed):

- database.host
- database.username

Optionally, the "schema" name can be defined (the default is objectiflune):

- database.schema

## Activating a License

PrintShop Mail Connect includes 30 day trial periods during which it is not necessary to have a license for reviewing basic functionality. If a modification to the license is required, such as to allow an extension to the trial period, or for extra functionality, then a new activation code will need to be requested.

### Obtaining the PrintShop Mail Connect Magic Number

To obtain an activation file the OL™ Magic Number must first be retrieved. The Magic Number is a machine-specific code that is generated based on the computer's hardware and software using a top-secret Objectif Lune family recipe. Each physical computer or virtual computer should have a different Magic Number, thus require a separate license file to be functional.

**To get the PrintShop Mail Connect Magic Number**, open the PrintShop Mail Connect Designer application:

- Open the **Start Menu**
- Click on **All Programs**, then **Objectif Lune**, then **PrintShop Mail Connect**
- Open the **PrintShop Mail Connect Designer [version]** shortcut.

- When the application opens, if it has never been activated or the activation has expired, the **Software Activation** dialog appears:
  - **License Information** subsection:
    - **Magic Number**: Displays the PrintShop Mail Connect Magic Number.
    - **Copy to Clipboard**: Click to copy the Magic Number to the clipboard. It can then be pasted in the activation request email using the CTRL+V keyboard shortcut.
  - **Licensed Products** subsection:
    - **Name**: Displays the name of the application or module relevant to this activation.
    - **Serial Number**: Displays the activation serial number if the product has been activated in the past.
    - **Expiration Date**: Displays the date when the activation will expire (or the current date if the product is not activated)
    - **Web Activations**: Click to be taken to the online activation page (not yet functional).
  - **End-User License Agreement** (Appears only when loading a license file):
    - **License**: This box displays the EULA. Please note that this agreement is legally binding.
    - **I agree**: Select to accept the EULA. This option **must** be selected to install the license.
    - **I don't agree**: Select if you do not accept the EULA. You cannot install the license if this option is selected.
  - **Load License File**: Click to browse to the .olconnectlicense file, once it has been received.
  - **Install License**: Click to install the license and activate the software (only available when a license file is loaded).
  - **Close**: Click to cancel this dialog. If a license file has been loaded, it will not automatically be installed.

#### Note

The **Software Activation** dialog can also be reached through a shortcut located in **All**

**Programs**, then **Objectif Lune**, then **PrintShop Mail Connect** and is named **Software Activation**. Since it does not load the software, it is faster to access for the initial activation.

## Requesting a license

After getting the Magic Number, a license request must be done for PrintShop Mail Connect:

- **Customers** must submit their Magic Number and serial number to Objectif Lune via the Web Activations page: <http://www.objectiflune.com/activations>. The OL Customer Care team will then send the PrintShop Mail Connect license file via email.
- **Resellers** can create an evaluation license via the Objectif Lune Partner Portal by following the instructions there: <http://extranet.objectiflune.com/>

Note that if you do not have a serial number, one will be issued to you by the OL Activations team.

Accepting the license will activate it, after which the PrintShop Mail Connect services will need to be restarted. Note that in some case the service may not restart on its own. To resolve this issue, restart the computer, or start the service manually from the computer's Control Panel.

## Activating PrintShop Mail Connect

To activate PrintShop Mail Connect, simply save the license file somewhere on your computer where you can easily find it, such as on your desktop. You can then load the license by double-clicking on it, or through the start menu:

- Open the **Start Menu**
- Click on **All Programs**, then **Objectif Lune**, then **PrintShop Mail Connect**
- Open the **PrintShop Mail Connect Designer [version]** shortcut. The “PrintShop Mail Connect Software Activation” tool displays information about the license and the [End-User License Agreement](#) (EULA).
- Click the **Load License File** button.
- Read the EULA and click I agree option to accept it.
- Click **Install License** to activate the license. The license will then be registered on the computer and you will be able to start using the software.

### **Warning**

After installation message will appear warning that the Server services will need to be restarted. Just click OK to proceed.

## **Migrating to a new workstation**

The purpose of this document is to provide a strategy for transferring a Connect installation to a new workstation. The following guide applies to OLConnect v1.x.

### **Before installing the software**

Before upgrading to a new version, even on a new workstation, consult the product's release note to find out about new features, bug fixes, system requirements, known issues and much more. Simply go to the [product page](#) and look for "Release notes" in the Downloads section.

You should also consult the following pages for some technical considerations before installing:

- [Network Considerations](#)
- [Database Considerations](#)
- [Environment Considerations](#)
- [Installation Pre-Requisites](#)
- Antivirus Exclusions

### **Downloading an Installing the Software**

In order to migrate to a new workstation, the software must already be installed on the new workstation. Follow the [Installation and Activation Guide](#) to download and install the newest version of PrintShop Mail Connect on the new workstation.

### **Backing Up files from the current workstation**

The first step in migrating to a new workstation would be to make sure all necessary production files and resources are backed up and copied over to the new system.

## Technical

Although it is not necessary to convert all of your documents when upgrading to the latest version, it is strongly recommended to converting them to the current version you have installed and re-sending them to the Workflow Tools. We recommend that your documents be maintained in a supported format preferably with the latest version.

## Backing up Connect Resources

The following resources are used by Connect and can be backed up from their respective folders:

- **Job Presets (.OL-jobpreset):**  
C:\Users\<<UserName>\Connect\workspace\configurations\JobCreationConfig
- **Output Presets (.OL-outputpreset):**  
C:\Users\<<UserName>\Connect\workspace\configurations\PrinterDefinitionConfig
- **OL Connect Print Manager Configuration files (.OL-ipdsprinter)**  
C:\Users\<<UserName>\Connect\workspace\configurations\PrinterConfig
- **OL Printer Definition Files (.OL-printerdef)**  
C:\Users\<<UserName>\Connect\workspace\configurations\PrinterDefinitionConfig
- **OMR Marks Configuration Files (.hcf)**  
C:\Users\<<UserName>\Connect\workspace\configurations\HCFFiles

## Other Resources

- **OL Connect Designer Templates**, copied from the folder where they reside.
- All Postscript, TrueType, Open Type and other **host based fonts** used in templates must be reinstalled on the new workstation.
- Import all **dynamic images** and make sure their paths match those in the old server.
- Make sure the new workstation can also access network or remote images, JavaScript, CSS, JSON, and HTML resources referenced in the Connect templates.

## Secondary Software and Licenses

The following only apply for specific secondary products and licenses that interacts or is integrated into the main product.

### Image, Fax and Search Modules

- Reconfigure the Image and Fax outputs with the new host information.
- Import the Search Profile and rebuild the database in order to generate the database structure required by the Workflow.

### Capture

- Download the latest version of the [Anoto PenDirector](#).
- Before installing the PenDirector, make sure the pen's docking station isn't plugged into the server. Then install the PenDirector.
- Stop the Messenger 8 service on old and new server from the Workflow menu bar > Tools > Service Console > Messenger > right-click and select Stop.
- Import the following files and folders from the old server into their equivalent location on the new server:  
C:\ProgramData\Objectif Lune\PlanetPress Workflow 8\PlanetPress Watch\capture\PPCaptureDefault.mdb  
C:\ProgramData\Objectif Lune\PlanetPress Workflow 8\PlanetPress Watch\DocumentManager  
C:\ProgramData\Objectif Lune\PlanetPress Workflow 8\PlanetPress Watch\PGC
- If Capture was previously using an external MySQL or Microsoft SQL Server, reconfigure the ODBC connection details as previously from the Workflow Preferences by clicking on the Workflow button on top left corner and clicking on Preferences, then reconfigure the PlanetPress Capture options under Behavior > PlanetPress Capture > Use ODBC Database
- Start the Messenger 8 service on new server from the Workflow menu bar > Tools > Service Console > Messenger > right-click and select Start.

### OL Connect Send

- Re-install OL Connect Send on the new Workstation. This should reinstall the OL Connect Send plugins in the Workflow Tool
- Reconfigure the Server URL and port during the OL Connect Send Printer Driver setup

- Re-run the OL Connect Send printer driver setup on client system and select the Repair option to point the clients to the new Server URL.

### Configuring the Connect Engines

Any changes made to the Server preferences required the OLConnect\_Server service to be restarted to take effect.

- Stop the OLConnect\_Server service from Control Panel > Administrative Tools > Services > OLConnect\_Server > Stop
- Configure the Merge and Weaver Engines scheduling preferences as in the previous installation
  - Open the Server Configuration from :  
C:\Program Files\Objectif Lune\OL Connect\Connect Server\Server.exe
  - Configure the Merge and Weaver engines preferences under Scheduling
  - Configure any other options for the Clean-up Service
- Configure the minimum (Xms) and maximum (Xmx) memory utilization for the Server, Merge and Weaver engines as previously or better:
  - Edit the following Xms and Xmx fields in the following configuration files:
    - C:\Program Files\Objectif Lune\OL Connect\Connect Server\Server.ini
    - C:\Program Files\Objectif Lune\OL Connect\MergeEngine\Mergeengine.ini
    - C:\Program Files\Objectif Lune\OL Connect\weaverengine\weaverengine.ini
- Now start the **OLConnect\_Server** service

### Configuring the Server Extensions

In the case where the OLConnect MySQL is installed on the new Master Server, it is important to reconnect all Server Extension systems to the new Master Server.

Perform the following action on each Server Extension:

- Stop the OLConnect\_ServerExtension service from **Control Panel > Administrative Tools > Services > OLConnect\_ServerExtension > Stop**

- Open the Server Extension Configuration from:  
C:\Program Files\Objectif Lune\OL Connect\Connect Server Extension\ServerExtension.exe
- Click on Database Connection and configure the JDBC Database connection settings so that the hostname points to the new Master server
- Click on Scheduling and type in the location of the new Master server
- Start the **OLConnect\_ServerExtension** service

## Transferring Software Licenses

Once all the above resources have been transferred over to the new server, it is recommended to thoroughly test the new system with sample files under normal production load to identify points of improvement and make sure the output match the user's expectation. Output generated at this point will normally bear a watermark which can be removed by transferring licenses from the old server to the new one.

- To transfer Connect and Workflow licenses, the user is usually required to complete a License Transfer Agreement which can be obtained from their [local Customer Care department](#)
- Upgrades cannot be activated using the automated Activation Manager. Contact your local Customer Care department.

To apply the license file received from the Activation Team:

- Start the PReS Connect, PlanetPress Connect or PrintShopMail Connect Software Activation module:  
C:\Program Files\Objectif Lune\OL Connect\Connect Software Activation\SoftwareActivation.exe
- Click on Load License File to import the license.OLConnectLicense
- Start the Software Activation module on the Extension servers, where applicable
- Click on Load License File to import the above same license.OLConnectLicense
- Restart the OLConnect\_Server service and restart the OLConnectServer\_Extension service on the Extension servers, where applicable
- The number of Expected Remote Merge and Weaver engines should now be configurable in the Connect Server Configuration module (C:\Program Files\Objectif Lune\OL Connect\Connect Server Configuration\ServerConfig.exe)

## To apply the PlanetPress Capture License

- Open the Workflow Configuration
- Click on Help on the Menu Bar and click on PlanetPress Capture License manager to import your license.

## **Uninstalling PrintShop Mail Connect from the previous workstation**

It is recommended to keep the previous install for a few days until everything is completed. However, once your transition is successful and complete, the OL Connect software must be uninstalled from the original server.

# Uninstalling

This topic provides some important information about uninstalling (removing) PrintShop Mail Connect1.5.

To uninstall PrintShop Mail Connect select the application from within the Add/Remove programs option under the Control Panel. This will start the **PrintShop Mail Connect Setup Wizard** in uninstall mode.

### **Note**

The **PrintShop Mail Connect Setup Wizard** might take some seconds to appear.

## **Important Note: Stop any Anti-Virus Software before uninstalling Connect.**

Some anti-virus systems are known to block the uninstallation of MySQL datafiles, as well as blocking the uninstallation of the MySQL database application itself. Therefore it is **highly recommended** that any anti-virus application be stopped prior to uninstalling PrintShop Mail Connect, as otherwise the Connect uninstallation might not work correctly.

# Impacts upon other Applications and Services

- The Uninstall will terminate the installed Server / MySQL service(s)
- The following applications / services should be stopped in a controlled fashion, before running the PrintShop Mail Connect Uninstall:
  1. PrintShop Mail Connect
  2. Any Connect Workflow using PrintShop Mail Connect plugins which connect to this server.
  3. PrintShop Mail Connect Server Extensions on remote systems which connect to this machine as the Master Server.
  4. Connect products on remote systems which refer to this MySQL database.

## Uninstallation Wizard

The uninstallation is done by running the PrintShop Mail Connect Setup Wizard in uninstall mode. The Wizard consists of the following pages:

1. **PrintShop Mail Connect Setup** An information page, listing what will be uninstalled, and also warning about impacts upon running Applications and Services.
2. **Data Management:** A page that provides options for backing up or deleting Connect data. Selections are as follows:
  - **Delete Connect Workspace Data:** Check this box to delete the Workspace data for the current user, or for selected users (as determined by the "Select Users" button)
    - **Backup Connect Workspace Data for all specified Users:** Check this box to backup the Workspace data for the specified users (as previously determined) into a compressed ZIP file (whose location can be customized), before deletion of the full Workspace data.
  - **Delete MySQL objectlune Data:** Check this box to delete the MySQL database installed with PrintShop Mail Connect.
    - **Backup MySQL Date:** If the deletion check box is selected, this option appears to allow backing up the MySQL database to a customizable location, prior to uninstallation.

# The Designer

The Designer is a WYSIWYG (what you see is what you get) editor that lets you create templates for various output channels: Print, Email and Web. A template may contain designs for multiple output channels: a letter intended for print and an e-mail variant of the same message, for example. Content, like the body of the message or letter, can be shared across these contexts. Templates are personalized using scripts and variable data. More advanced users may use native HTML, CSS and JavaScript.

The following topics will help to quickly familiarize yourself with the Designer.

- "Basic Steps" below. These are the basic steps for creating and developing a template.
- "Features" on page 52. These are some of the key features in the Designer.
- "Designer User Interface" on page 254. This part gives an overview of all elements in the Designer User Interface, like menus, dialogs and panes.

More help can be found here:

- [Tutorials](#) On Video: watch an introductory video, overview tutorials or practical how-to videos.
- [Forum](#): Browse the forum and feel free to ask questions about the use of Connect software
- [Demo site](#). Download demonstrations of OL products.

## Basic Steps

With the Designer you can create templates for personalized letters and emails, and generate output from them.

These are the basic steps for creating and developing a template:

### 1. Create a template

Create a template, using one of the Template Wizards. See "Creating a template" on the facing page.

## 2. Fill the template

Add text, images and other elements to the template and style them. See "Content elements" on page 104 and "Styling and formatting" on page 170.

## 3. Personalize the content

Personalize the content using variable data. See "Personalizing content" on page 202.

## 4. Generate output

Adjust the settings, test the template and generate output: letters and emails. See "Generating output" on page 493.

### Note

Steps 2 and 3 are not necessarily to be followed in this order. For example, as you add elements to a template, you may start personalizing them right away, before adding other elements to the template.

# Templates

The Designer is a WYSIWYG (what you see is what you get) tool to create templates. This topic gets you started. It explains how to create a template, what is found in a template file, and how output can be generated.

## Creating a template

In the **Welcome** screen that appears after startup, get off to a flying start choosing **Browse Template Wizards**. Scroll down to see all the Template Wizards. After deciding which output channel – printer email – will be prevalent in your template, select a template.

The Template Wizards can also be accessed from the menu: click **File**, click **New**, expand the **Template** folder, and then expand one of the **templates** folders.

There are Wizards for the three types of output channels, or **contexts** as they are called in the Designer: Print, Email and Web.

See:

- "Creating an Email template with a Wizard" on page 89
- "Creating a Print template with a Wizard" on page 54

- Creating a Web template with a Wizard

After creating a template you can add the other contexts (see "Contexts" on page 47), as well as extra sections (see "Sections" on page 48), to the template.

It is, however, not possible to use a Template Wizard when adding a context or section to an existing template.

### Tip

If an Email context is going to be part of the template, it is recommended to start with an Email Template Wizard; see "Creating an Email template with a Wizard" on page 89. After creating a template, contexts can be added to it, but that can not be done with a wizard.

## Saving a template

A Designer template file has the extension **.OL-template**. It is a zip file that includes up to 3 contexts, all the related resources and scripts, and (optionally) a link to a Data Mapping Configuration.

To save a template for the first time, select **File > Save as**. After that you can save the template by selecting **File > Save** or pressing **Ctrl+S**.

When more than one resource (template or data mapping configuration) is open and the Designer software is closed, the Save Resources dialog appears. This dialog displays a list of all open resources with their names and file location. Selected resources will be saved, deselected resources will have all their changes since they were last saved dismissed.

## Auto Save

**After** a template has been saved for the first time, Connect Designer can auto save the template with a regular interval. To configure Auto Save:

1. Select the menu option **Window > Preferences > Save**.
2. Under **Auto save**, check the option **Enable** to activate the Auto Save function.
3. Change how often it saves the template by typing a number of minutes.

## Auto Backup

Connect Designer can automatically create a backup file when you **manually** save a template. To configure Auto Backup:

1. Select the menu option **Window > Preferences > Save**.
2. Under **Auto backup**, check the option **Enable** to activate the Auto Backup function.
3. Type the number of revisions to keep.
4. Select the directory in which the backups should be stored.

Backup files have the same name as the original template with two underscores and a progressive number (without leading zeros) at the end: **originalname\_\_1.OL-template**, **originalname\_\_2.OL-template**, etc.

### Note

The Auto Save function does **not** cause backup files to be created.

## Sharing a template

To share a template, you can send the template file itself, or save the template to a package file, optionally together with a Job Creation Preset and an Output Creation Preset. (See "[Job Creation Presets](#)" on page 401 and "[Output Creation Settings](#)" on page 410 for more details.)

To create a package file, select **File > Package file**, see "Package files" on the next page. The package file has the extension .OL-package.

## Generating output from the Designer

Output can be generated directly from the Designer; see "Generating Print output" on page 495, "Generating Email output" on page 509 and Generating Web output.

To test a template first, select **Context > Preflight**. Preflights executes the template without actually producing output and it displays any issues once it's done (see also: "Testing scripts" on page 231).

## Package files

The Package file dialog saves templates and print presets as a package file. Print presets make it possible to do such things as filtering and sorting records, grouping documents and splitting the print jobs into smaller print jobs, as well as the more standard selection of printing options, such as binding, OMR markings and the like. See "[Job Creation Presets](#)" on page 401 and "[Output Creation Settings](#)" on page 410 for more details.

To save a template to a package file:

1. Select **File > Package files**.
2. Select the template to send. By default the currently active template is listed. Click **Browse** to select another template. You may select more than one template in the Browse dialog, and each of them is added to a package file. A template file has the extension .OL-template.
3. Use the drop-down to select a Job Creation Preset to send. Click **Browse** to select a preset that is not in the default location for presets. A Job Creation Preset file has the extension .OL-jobpreset.
4. Use the drop-down to select an Output Creation Preset. Click **Browse** to select a preset that is not in the default location for presets. An Output Creation Preset file has the extension .OL-outputpreset.
5. Click **OK** to save the package file.

## Resources

This page clarifies the difference between Internal, External and Web resources that may be used in a template, and explains how to refer to them in HTML and in scripts.

### Internal resources

Internal resources are files that are added to and saved with the template. To add images, fonts, style sheets, and snippets to your template, you can drag or copy/paste them into the Resources Pane. See also: "[Images](#)" on page 158, "[Snippets](#)" on page 168, "[Styling templates with CSS files](#)" on page 171 and "[Fonts](#)" on page 194.

Resource files can also be dragged or copy/pasted **out** of the the application to save them on a local hard drive.

Once imported, internal resources are accessed using a relative path, depending where they're called from. Resources can be located in the following folders:

- images/ contains the files in the Images folder.
- fonts/ contains the files in the Fonts folder.
- css/ contains the files in the StyleSheets folder.
- js/ contains the files in the JavaScripts folder.
- snippets/ contains the files in the Snippets folder.

When referring to them, normally you would simply use the path directly with the file name. The structure within those folders is maintained, so if you create a "signatures" folder within the "Images" folder, you need to use that structure, for example in HTML: ``. In scripts, you can refer to them in the same way, for example:

```
results.loadhtml("snippets/en/navbar.html");
```

See also: "Loading a snippet via a script" on page 238 and "Writing your own scripts" on page 223.

### Note

When referring to images or fonts from a CSS file, you need to remember that the current path is `css/`, meaning you can't just call `images/image.jpg`. Use a relative path, for example: `#header { background-image: url('../images/image.jpg'); }`

## External resources

External resources are not stored in the template, but on the local hard drive or on a network drive. They are accessed using a path. The path must have forward slashes, for example `` or `var json_variables = loadjson("file:///d:/jsondata/variables.json");`. The complete syntax is: `file://<host>/<path>`. If the host is "localhost", it can be omitted, as it is in the example, resulting in `file:///<path>`. The empty string is interpreted as 'the machine from which the URL is being interpreted'.

Network paths are similar: `results.loadhtml`

```
("file://servername/sharename/folder/snippet.html");
```

 (note that in this case **file** is followed by 2 slashes only).

## Some limitations

- Style sheets cannot refer to external resources.
- The Connect Server user needs access to whichever network path is used. If the network path is on a domain, the Connect Server must be identified with domain credentials that have access to the domain resources.

For more information on network paths, please see this Wikipedia entry: [file URI scheme](#).

## **Web resources**

Web resources are simply accessed using a full URL. This URL needs to be publicly accessible: if you type in that URL in a browser on the server, it needs to be visible.

Authentication is possible only through URL Parameters

(<http://www.example.com/data.json?user=username&password=password>) or through HTTP Basic Auth (<http://username:password@www.example.com/data.json>).

## Contexts

Contexts are parts of a template that are each used to generate a specific type of output: Email or Print.

- The Print context outputs documents to either a physical printer a PDF file; see "Print context" on page 58.
- The Email context outputs HTML email, composed of HTML code with embedded CSS. See "Email context" on page 93.

When a new template is made, the Context appropriate to that new template is automatically created, including one section. After a template has been created, the other two contexts can be added to it; see "Adding a context" on the facing page.

### **Tip**

If an Email context is going to be part of the template, it is recommended to start with an Email Template Wizard; see "Creating an Email template with a Wizard" on page 89. After creating a template, contexts can be added to it, but that can not be done with a wizard.

## Outputting and combining contexts

All three contexts can be present in any template and they can all be used to output documents; see "Generating Email output" on page 509, "Generating Print output" on page 495 and "Generating Web output".

They can even be combined in output.

If present in the same template, a Print context and a Web context can be attached to an Email context.

Outputting other combinations of contexts, and selecting sections based on a value in the data, can be done via a Control Script; see "Control Scripts" on page 240.

## Adding a context

To add a context, right-click the **Contexts** folder on the **Resources** pane and click **New print context**, **New email context** or **New web context**. Only one context of each type can be present in a template. Each context, however, can hold more than one section; see "Sections" below.

## Deleting a context

To delete a context, right-click the context on the **Resources** pane and click **Delete**.

### Warning

No backup files are maintained in the template. The only way to recover a deleted section, is to click **Undo** on the **Edit** menu, until the deleted section is restored. After closing and reopening the template it is no longer possible to restore the deleted context this way.

To prevent losing any work, it is recommended to configure the auto-save and auto-backup functions in the preferences (see "Saving Preferences" on page 293).

## Sections

Sections are parts of one of the contexts in a template: Print or Email. They contain the main text flow for the contents. In each of the contexts there can be multiple sections. A Print context, for example, may consist of two sections: a covering letter and a policy.

## Adding a section

To add a section to a context, right-click the context (Email or Print) on the **Resources** pane, and then click **New section**.

It is not possible to use a Template Wizard when adding a section to an existing template.

### Tip

If an Email context is going to be part of the template, it is recommended to start with an Email Template Wizard; see "Creating an Email template with a Wizard" on page 89. After creating a template, contexts can be added to it, but that can not be done with a wizard.

## Editing a section

To open a section, expand the **Contexts** folder on the **Resources** pane, expand the respective context (**Print** or **Email**) and double-click a section to open it.

Each section can contain text, images and many other elements (see "Content elements" on page 104), including variable data and other dynamic elements (see "Personalizing content" on page 202).

## Copying a section

Copying a section, either within the same template or from another template, can only be done manually. You have to copy the source of the HTML file:

1. Open the section that you want to copy and go to the **Source** tab in the workspace.
2. Copy the contents of the **Source** tab (press **Ctrl+A** to select everything and then **Ctrl+C** to copy the selection).
3. Add a new section (see [Adding a section](#)).
4. Go to the **Source** tab and paste the contents of the other section here (press **Ctrl+V**).
5. When copying a section to another template, add the related source files, such as images, to the other template as well.

## Deleting a section

To delete a section:

- On the **Resources** pane, expand the **Contexts** folder, expand the folder of the respective context, right-click the name of the section, and then click **Delete**.

### Warning

No backup files are maintained in the template. The only way to recover a deleted section, is to click **Undo** on the **Edit** menu, until the deleted section is restored. After closing and reopening the template it is no longer possible to restore the deleted context this way.

### Renaming a section

To rename a section:

- On the **Resources** pane, expand the **Contexts** folder, expand the folder of the respective context, right-click the name of the section, and then click **Rename**.

### Note

Sections cannot have an integer as name. The name should always include alphanumeric characters.

### Section properties

Which properties apply to a section, depends on the context it is part of. See also: "Print sections" on page 62, "Email templates" on page 95, and Web pages.

To change the properties for a section:

- On the **Resources** pane, expand the **Contexts** folder, expand the folder of the respective context, right-click the name of the section, and then click one of the options.

### Applying a style sheet to a section

In order for a style sheet to be applied to a specific section, it needs to be included in that section. There are two ways to do this.

### Drag & drop a style sheet

1. Click and hold the mouse button on the style sheet on the **Resources** pane.
2. Move the mouse cursor within the **Resources** pane to the section to which the style sheet should be applied.
3. Release the mouse button.

### Using the Includes dialog

1. On the Resources pane, right-click the section, then click **Includes**.
2. Choose which CSS files should be applied to this section. You can also change the order in which the CSS files are read. This can have an effect on which CSS rule is applied in the end.

### Arranging sections

Changing the order of the sections in a context can have an effect on how they are outputted; see: "Print sections" on page 62, "Email templates" on page 95 and Web pages.

To rearrange sections in a context:

- On the **Resources** pane, expand the Contexts folder, expand the folder of the respective context, and then drag and drop sections to change the order they are in. Alternatively, right-click a section and click **Arrange**. In the Arrange Sections dialog you can change the order of the sections in the same context by clicking the name of a section and moving it using the **Up** and **Down** buttons.

### Outputting sections

Which sections are added to the output, depends on the type of context they are in.

When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record. The sections are added to the output in the order in which they appear on the **Resources** pane. See "Generating Print output" on page 495.

In email output, only one section can be executed at a time. The section that will be output is the section that has been set as the 'default'. See "Generating Email output" on page 509 and "Email templates" on page 95.

It is, however, possible to include or exclude sections when the output is generated, or to set another section as the 'default', depending on a value in the data. A Control Script can do this; see "Control Scripts" on page 240.

See "Generating output" on page 493 to learn how to generate Print documents or Email.

## Features

The Designer is Connect's module to create templates for personalized customer communications. These are some of the key features in the Designer:

"Templates" on page 42. Start creating, using and sharing templates.

"Contexts" on page 47. A context contains one or more designs for one output channel:

- "Print" on the next page. This topic helps you design and fill sections in the Print context.
- "Email" on page 85. This topics helps you design an email template.
- Web. This topic helps you design a web page.

"Sections" on page 48. Sections in one context are designed for the same output channel.

"Content elements" on page 104. Elements make up the biggest part of the content of each design.

"Snippets" on page 168. Snippets help share content between contexts, or insert content conditionally.

"Styling and formatting" on page 170. Make your Designer templates look pretty and give them the same look and feel with style sheets.

"Personalizing content" on page 202. Personalize your customer communications using variable data.

"Writing your own scripts" on page 223. Scripting can take personalization much further. Learn how to script via this topic.

"Generating output" on page 493. Learn the ins and outs of generating output from each of the contexts.

# Print

With the Designer you can create one or more Print templates and merge the template with a data set to generate personal letters, invoices, policies etc.

The Print **context** is the folder in the Designer that can contain one or more Print sections.

Print templates, also called Print *sections*, are part of the Print context. They are meant to be printed to a printer or printer stream, or to a PDF file (see "Generating Print output" on page 495).

The Print context can also be added to Email output as a PDF attachment; see "Generating Email output" on page 509. When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record.

When a Print template is created (see "Creating a Print template with a Wizard" on the facing page), or when a Print context is added to an existing template (see "Adding a context" on page 48) the Print context folder is created along with other folders and files that are specific to a Print context (see "Print context" on page 58).

Only one Print section is created at the start, but you can add as many Print sections as you need; see "Print sections" on page 62.

## Pages

Unlike emails, Print sections can contain multiple *pages*. Pages are naturally limited by their size and margins. If the content of a section doesn't fit on one page, the overflow goes to the next page. This happens automatically, based on the section's page size and margins; see "Page settings: size, margins and bleed" on page 70.

Although generally the same content elements can be used in all three contexts (see "Content elements" on page 104), the specific characteristics of pages make it possible to use special elements, such as page numbers; see "Page numbers" on page 72.

See "Pages" on page 69 for an overview of settings and elements that are specific for pages.

## Headers, footers, tear-offs and repeated elements (Master page)

In Print sections, there are often elements that need to be repeated across pages, like headers, footers and logos. In addition, some elements should appear on each first page, or only on pages in between the first and the last page, or only on the last page. Examples are a different header on the first page, and a tear-off slip that should show up on the last page.

This is what Master Pages are used for. Master Pages can only be used in the Print context.

See "Master Pages" on page 76 for an explanation of how to fill them and how to apply them to different pages.

## Stationery (Media)

When the output of a Print context is meant to be printed on paper that already has graphical and text elements on it (called stationery, or preprinted sheets), you can add a copy of this media, in the form of a PDF file, to the Media folder.

Media can be applied to pages in a Print section, to make them appear as a background to those pages. This ensures that elements added to the Print context will correspond to their correct location on the preprinted media.

When both Media and a Master Page are used on a certain page, they will both be displayed on the Preview tab of the workspace, the Master Page being 'in front' of the Media and the Print section on top. To open the Preview tab, click it at the bottom of the Workspace or select **View > Preview View** on the menu.

The Media will not be printed, unless this is specifically requested through the printer settings in the Print Wizard; see "Generating Print output" on page 495.

See "Media" on page 80 for further explanation about how to add Media and how to apply them to different pages.

## Creating a Print template with a Wizard

A Print template may consist of various parts, such as a covering letter and a policy. Start with one of the Template Wizards for the first part; other parts can be added later.

To create a Print template with a Template Wizard:

1.
  - In the **Welcome** screen that appears after startup:
    - Choose **Browse Template Wizards** and scroll down until you see the Print Template wizards and select the Postcard or Formal Letter wizard.
    - Or choose **Create a New Template** and select the PDF-based Print wizard.
  - Alternatively, on the **File** menu, click **New**, expand the **Template** folder, and then:
    - Select the PDF-based Print wizard.
    - Or expand the **Basic Print templates** folder, select Postcard or Formal Letter and click **Next**.

See "Print Template Wizards" below for information about the various types of Template wizards.

2. Make adjustments to the initial settings (the options for each type of template are listed below). Click **Next** to go to the next settings page if there is one.
3. Click **Finish** to create the template.

See "Print context" on page 58 and "Print sections" on page 62 for more information about Print templates.

### Tip

Use the **Outline** pane at the left to see which elements are present in the template and to select an element.

Use the **Attributes** pane at the right to see the current element's ID, class and some other properties.

Use the **Styles** pane next to the Attributes pane to see which styles are applied to the currently selected element.

## Print Template Wizards

There are three Print Template wizards: one for a formal letter, one for a postcard and one for a Print template based on a PDF that you provide.

## Postcard

The Postcard Wizard lets you choose a page size and two background images, one for the front and one for the back of the postcard.

When you click **Finish**, the Wizard creates:

- A Print context with one section in it, that has duplex printing (printing on both sides) enabled. See "Printing on both sides" on page 61.
- Two Master Pages that each contain a background image. The first Master Page is applied to the front of every page in the Print section. The second Master Page is applied to the back of every page in the Print section. See "Master Pages" on page 76.
- **Scripts** and **selectors** for variable data. The **Scripts** pane shows, for example, a script called "first\_name". This script replaces the text "@first\_name@" on the front of the postcard by the value of a field called "first\_name" when you open a data set that has a field with that name. See "Variable Data" on page 211.
- A script called Dynamic Front Image Sample. This script shows how to toggle the image on the front page dynamically. See also "Writing your own scripts" on page 223.
- One empty Media. Media, also called Virtual Stationery, can be applied to all pages in the Print section. See "Media" on page 80.

The Wizard opens the Print section, so that you can fill it with text and other elements; see "Content elements" on page 104. It already has two Positioned Boxes on it: one on the front, for text, and one on the back, for the address.

See "Print context" on page 58 and "Print sections" on page 62 for more information about Print templates.

## Formal letter

The Formal Letter Wizard first lets you select the page settings, see "Page settings: size, margins and bleed" on page 70.

These settings are fairly self-explanatory, except perhaps these:

- Duplex means double-sided printing.
- The margins define where your text flow will go. The actual printable space on a page depends on your printer.

- The bleed is the printable space **around** a page. It can be used on some printers to ensure that no unprinted edges occur in the final trimmed document. Printers that can't print a bleed, will misinterpret this setting. Set the bleed to zero to avoid this.
- The number of sections is the number of parts in the Print context. Although this Template wizard can add multiple Print sections to the Print context, it will only add content to the first section.

On the next settings page (click **Next** to go there), you can type a subject, the sender's name and the sender's title. These will appear in the letter. You can also:

- Click the **Browse** button to select a signature image. This image will appear above the sender's name and title.
- Select Virtual Stationery: a PDF file with the letterhead stationery. Also see Media.

When you click **Finish**, the Wizard creates:

- A Print context with one section in it; see "Print context" on the facing page and "Print sections" on page 62.
- One empty Master Page. Master Pages are used for headers and footers, for images and other elements that have to appear on more than one page, and for special elements like tear-offs. See "Master Pages" on page 76.
- One Media. You can see this on the **Resources** pane: expand the **Media** folder. **Media 1** is the Virtual Stationery that you have selected in the Wizard. It is applied to all pages in the Print section, as can be seen in the Sheet Configuration dialog. (To open this dialog, expand the **Contexts** folder on the **Resources** pane; expand the **Print** folder and right-click "Section 1"; then select **Sheet Configuration**.) See "Media" on page 80.
- **Selectors** for variable data, for example: @Recipient@. You will want to replace these by the names of fields in your data. See "Variable Data" on page 211.

The Wizard opens the Print section. You can add text and other elements; see "Content elements" on page 104.

The formal letter template already has an address on it. The address lines are paragraphs, located in one cell in a table with the ID **address-block-table**. As the table has no borders, it is initially invisible. The address lines will stick to the bottom of that cell, even when the address has fewer lines. See "Styling and formatting" on page 170 to learn how to style elements.

### Tip

Click the **Edges** button on the toolbar to make borders of elements visible on the Design tab. The borders will not be visible on the Preview tab.

## PDF-based Print template

The PDF-based Print template wizard creates a document from an existing PDF file: a brochure, voucher, letter, etc. The PDF is used as the background image of the Print section (see "Using a PDF file as background image" on page 65). Variable and personalized elements, like a reseller address, voucher codes and so on, can be added in front of it (see "Personalizing content" on page 202 and "Variable Data" on page 211).

By default, the PDF itself is added to the **Image** folder located in the **Resources** pane. Uncheck the option **Save with template** if the PDF should not be imported in the template. If not saved with the template, the image will remain external. Note that external images need to be available when the template is merged with a record set to generate output, and that their location should be accessible from the machine on which the template's output is produced. External images are updated (retrieved) at the time the output is generated.

After clicking **Next**, you can change the settings for the page. The initial page size and bleed area are taken from the selected PDF.

When you click **Finish**, the Wizard creates:

- A Print context with one section in it; see "Print context" below and "Print sections" on page 62. The selected PDF is used as the background of the Print section; see "Using a PDF file as background image" on page 65. For each page in the PDF one page is created in the Print section.
- One empty Master Page. Master Pages are used for headers, footers, images and other elements that have to appear on more than one page, and for special elements like tear-offs. See "Master Pages" on page 76.
- One empty Media. Media, also called Virtual Stationery, can be applied to all pages in the Print section. See "Media" on page 80.

## Print context

The Print context is the folder in the Designer that can contain one or more Print templates.

Print templates, also called *Print sections*, are part of the Print context. They are meant to be printed to a printer or printer stream, or to a PDF file (see "Generating Print output" on page 495).

The Print context can also be added to Email output as a PDF attachment; see "Generating Email output" on page 509. When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record.

## Creating the Print context

You can start creating a Print template with a Wizard (see "Creating a Print template with a Wizard" on page 54), or add the Print context to an existing template (see "Adding a context" on page 48).

### Tip

Editing PDF files in the Designer is not possible, but when they're used as a section's background, you can add text and other elements, such as a barcode, to them. To create a new Print template from a PDF file, use the PDF-based Print template (see "Creating a Print template with a Wizard" on page 54). To use a PDF file as background image for an existing section, see "Using a PDF file as background image" on page 65.

When a Print template is created, the following happens:

- The Print context is created and one **Print section** is added to it. You can see this on the **Resources** pane: expand the **Contexts** folder, and then expand the **Print** folder. The Print context can contain multiple sections: a covering letter and a policy, for example, or one section that is meant to be attached to an email as a PDF file and another one that is going to be printed out on paper. Only one Print section is added to it at the beginning, but you can add as many print sections as you need; see "Print context" on the previous page. See "Print sections" on page 62 to learn how to fill a Print section.
- One **Master Page** is added to the template, as can be seen on the **Resources** pane, in the **Master Page** folder.

In Print sections, there are often elements that need to be repeated across pages, like headers, footers and logos. In addition, some elements should appear on each first page, or only on pages in between the first and the last page, or only on the last page.

Examples are a different header on the first page, and a tear-off slip that should show up on the last page.

This is what Master Pages are used for. Master Pages can only be used in the Print context.

See "Master Pages" on page 76.

Initially, the (empty) master page that has been created with the Print context will be applied to all pages in the Print section, but more Master Pages can be added and applied to different pages.

- One **Media** is added to the template, as is visible on the **Resources** pane, in the **Media** folder. This folder can hold the company's stationery in the form of PDF files. When applied to a page in a Print section, Media can help prevent the contents of a Print section from colliding with the contents of the stationery. See "Media" on page 80 to learn how to add Media and, optionally, print them.  
Initially, the (empty) media that has been created with the Print context, is applied to all pages in the Print section. You can add more Media and apply them each to different pages.
- One **Stylesheet**, named `context_print_styles.css`, is added to the template, as you can see on the Resources pane, in the **Stylesheets** folder. This stylesheet is meant to be used for styles that are only applied to elements in the Print context. See also "Styling templates with CSS files" on page 171.

## Print settings in the Print context and sections

The following settings in the Print context and Print sections have an impact on how the Print context is printed.

## Arranging and selecting sections

The Print context can contain one or more Print sections. When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record. The sections are added to the output in the order in which they appear on the **Resources** pane. This order can be changed; see "Print sections" on page 62.

It is also possible to exclude sections from the output, or to include a section only on a certain condition that depends on a value in the data. This can be done using a Control Script; see "Control Scripts" on page 240.

## Printing on both sides

To print a Print section on both sides of the paper, that Print section needs to have the Duplex printing option to be enabled; see "Enabling double-sided printing" on page 68. This setting can not be changed in a Job Creation Preset or an Output Creation Preset.

### Note

Your printer must support duplex for this option to work.

## Setting the binding style for the Print context

The Print context , as well as each of the Print sections, can have its own Finishing settings. In printing, Finishing is the way pages are bound together after they have been printed. Which binding styles can be applied depends on the type of printer that you are using.

To set the binding style of the Print **context**:

1. On the **Resources** pane, expand the **Contexts** folder, and right-click the **Print** context.
2. Click **Properties**.
3. Choose a Binding style and, if applicable, the number of holes.

To set the binding style of a Print **section**, see "Setting the binding style for a Print section" on page 67.

## Overriding binding styles in a job creation preset

A *Job Creation Preset* can override the binding styles set for the Print sections and for the Print context as a whole. To bind output in another way than defined in the template's settings:

1. Create a Job Creation Preset that overrides the settings of one or more sections: select **File > Presets** and see "[Job Creation Presets](#)" on page 401 for more details.
2. Select that Job Creation Preset in the Print wizard; see "Generating Print output" on page 495.

## Setting the bleed

The **bleed** is the printable space around a page. It can be used on some printers to ensure that no unprinted edges occur in the final trimmed document. The bleed is one of the settings for a section. See "Page settings: size, margins and bleed" on page 70.

## Print sections

Print templates, also called *Print sections*, are part of the Print context. They are meant to be printed to a printer or printer stream, or to a PDF file (see "Generating Print output" on page 495).

The Print context can also be added to Email output as a PDF attachment; see "Generating Email output" on page 509. When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record.

## Pages

Unlike emails, Print sections can contain multiple *pages*. Pages are naturally limited by their size and margins. If the content of a section doesn't fit on one page, the overflow goes to the next page. This happens automatically, based on the section's page size and margins; see "Page settings: size, margins and bleed" on page 70.

Although generally the same content elements can be used in all three contexts (see "Content elements" on page 104), the specific characteristics of pages make it possible to use special elements, such as page numbers; see "Page numbers" on page 72.

See "Pages" on page 69 for an overview of settings and elements that are specific for pages.

## Using headers, footers, tear-offs and repeated elements

In Print sections, there are often elements that need to be repeated across pages, like headers, footers and logos. In addition, some elements should appear on each first page, or only on pages in between the first and the last page, or only on the last page. Examples are a different header on the first page, and a tear-off slip that should show up on the last page.

This is what Master Pages are used for. Master Pages can only be used in the Print context.

See "Master Pages" on page 76 for an explanation of how to fill them and how to apply them to different pages.

## Using stationery (Media)

When the output of a Print context is meant to be printed on paper that already has graphical and text elements on it (called stationery, or preprinted sheets), you can add a copy of this media, in the form of a PDF file, to the Media folder.

Media can be applied to pages in a Print section, to make them appear as a background to those pages. This ensures that elements added to the Print context will correspond to their correct location on the preprinted media.

### Note

When both Media and a Master Page are used on a certain page, they will both be displayed on the Preview tab of the workspace, the Master Page being 'in front' of the Media and the Print section on top. To open the Preview tab, click it at the bottom of the Workspace or select **View > Preview View** on the menu.

See "Media" on page 80 for a further explanation about how to add Media and how to apply them to different pages.

Note: The Media will not be printed, unless this is specifically requested through the printer settings; see "Generating Print output" on page 495.

## Adding a Print section

The Print context can contain multiple sections: a covering letter and a policy, for example, or one section that is meant to be attached to an email as a PDF file and another one that is meant to be printed out on paper. When a Print template is created (see "Creating a Print template with a Wizard" on page 54 and "Print context" on page 58), only one Print section is added to it, but you can add as many print sections as you need.

To add a section to a context:

- On the **Resources** pane, expand the **Contexts** folder, right-click the **Print** context , and then click **New section**.

The first Master Page (see "Master Pages" on page 76) and Media (see "Media" on page 80) will automatically be applied to all pages in the new section, but this can be changed, see

"Applying a Master Page to a page in a Print section" on page 79 and "Applying Media to a page in a Print section" on page 83.

### Tip

Editing PDF files in the Designer is not possible, but when they're used as a section's background, you can add text and other elements, such as a barcode, to them. To create a new Print template from a PDF file, use the PDF-based Print template (see "Creating a Print template with a Wizard" on page 54). To use a PDF file as background image for an existing section, see "Using a PDF file as background image" on the next page.

### Note

Via a Control Script, sections can be added to a Print context dynamically; see "Control Scripts" on page 240.

## Deleting a Print section

To delete a Print section:

- On the **Resources** pane, expand the **Contexts** folder, expand the **Print** context, right-click the name of the section, and then click **Delete**.

### Warning

No backup files are maintained in the template. The only way to recover a deleted section, is to click **Undo** on the **Edit** menu, until the deleted section is restored. After closing and reopening the template it is no longer possible to restore the deleted context this way.

To prevent losing any work, it is recommended to configure the auto-save and auto-backup functions in the preferences (see "Saving Preferences" on page 293).

## Arranging Print sections

When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record. The sections are added to the

output in the order in which they appear on the **Resources** pane, so changing the order of the sections in the Print context changes the order in which they are outputted to the final document.

To rearrange sections in a context:

- On the **Resources** pane, expand the **Print** context and drag and drop sections to change the order they are in.
- Alternatively, on the **Resources** pane, right-click a section in the **Print** context and click **Arrange**. In the Arrange Sections dialog you can change the order of the sections by clicking the name of a section and moving it using the **Up** and **Down** buttons.

### Styling and formatting a Print section

The contents of a Print section can be formatted directly, or styled with Cascading Style Sheets (CSS). See "Styling and formatting" on page 170.

In order for a style sheet to be applied to a specific section, it needs to be included in that section. There are two ways to do this.

### Drag & drop a style sheet

1. Click and hold the mouse button on the style sheet on the **Resources** pane.
2. Move the mouse cursor within the **Resources** pane to the section to which the style sheet should be applied.
3. Release the mouse button.

### Using the Includes dialog

1. On the Resources pane, right-click the section, then click **Includes**.
2. Choose which CSS files should be applied to this section. You can also change the order in which the CSS files are read. This can have an effect on which CSS rule is applied in the end.

### Using a PDF file as background image

In the Print context, a PDF file can be used as a section's background. It is different from the Media in that the section considers the PDF to be content, so the number of pages in the section will be the same as the number of pages taken from the PDF file.

With this feature it is possible to create a Print template from an arbitrary PDF file. Of course, the PDF file itself can't be edited in a Designer template, but when it is used as a section's background, text and other elements, such as a barcode, can be added to it.

To use a PDF file as background image:

1. On the **Resources** pane, expand the **Print** context, right-click the print section and click **Background**.
2. Click the downward pointing arrow after **Image** and select **From PDF resource**.

#### Note

**From DataMapper input** cannot be used in PrintShopMail Connect. The DataMapper is included only in PlanetPress Connect and PreS Connect so far.

3. For a PDF resource, you have to specify where it is located. Click the **Select Image** button.

Click **Resources**, **Disk** or **Url**, depending on where the image is located.

- **Resources** lists the images that are present in the **Images** folder on the **Resources** pane.
- **Disk** lets you choose an image file that resides in a folder on a hard drive that is accessible from your computer. Click the **Browse** button to select an image. As an alternative it is possible to enter the path manually. The complete syntax is: file://<host>/<path>. Note: if the host is "localhost", it can be omitted, resulting in file:///<path>, for example: file:///c:/resources/images/image.jpg. Check the option **Save with template** to insert the image into the **Images** folder on the **Resources** pane.
- **Url** allows you to choose an image from a specific web address. Select the protocol (**http** or **https**), and then enter the web address (for example, <http://www.mysite.com/images/image.jpg>).

#### Note

It is not possible to use a remotely stored PDF file as a section's background,

because the number of pages in a PDF file can not be determined via the http and http protocols. Therefore, with an external image, the option **Save with template** is always checked.

4. Select the PDF's **position**:

- **Fit to page** stretches the PDF to fit the page size.
- **Centered** centers the PDF on the page, vertically and horizontally.
- **Absolute** places the PDF at a specific location on the page. Use the **Top** field to specify the distance between the top side of the page and the top side of the PDF, and the **Left** field to specify the distance between the left side of the page and the left side of the PDF.

5. Optionally, if the PDF has more than one page, you can set the range of **pages** that should be used.

**Note**

The number of pages in the Print section is automatically adjusted to the number of pages in the PDF file that are being used as the section's background image.

6. Finally, click **OK**.

**Note**

To set the background of a section in script, you need a Control Script; see "Control Script API" on page 477.

### Setting the binding style for a Print section

In printing, Finishing is the binding style, or the way pages are bound together. Each Print section can have its own Finishing settings, as well as the Print context as a whole; see "Setting the binding style for the Print context" on page 61.

To set the binding style of a Print **section**:

1. On the **Resources** pane, expand the **Contexts** folder, expand the **Print** context and right-click the Print section.
2. Click **Finishing**.
3. Choose a Binding style and, if applicable, the number of holes.

To set the binding style of the Print **context**, see "Setting the binding style for the Print context" on page 61.

### Overriding binding styles in a job creation preset

A *Job Creation Preset* can override the binding styles set for the Print sections and for the Print context as a whole. To bind output in another way than defined in the template's settings:

1. Create a Job Creation Preset that overrides the settings of one or more sections: select **File > Presets** and see "[Job Creation Presets](#)" on page 401 for more details.
2. Select that Job Creation Preset in the Print wizard; see "Generating Print output" on page 495.

### Enabling double-sided printing

To print a Print section on both sides of the paper, that Print section needs to have the Duplex printing option to be enabled. This is an option in the Sheet Configuration dialog. (See "Sheet Configuration dialog" on page 309.)

#### Note

Your printer must support duplex for this option to work.

To enable duplex printing:

1. On the **Resources** pane, expand the **Print** context, right-click the print section and click **Sheet configuration**.
2. Check **Duplex** to enable content to be printed on the back of each sheet.

3. When duplex printing is enabled, further options become available.
  - Check **Tumble** to duplex pages as in a calendar.
  - Check **Facing pages** to have the side margins switched alternately, so that after printing and binding the pages, they look like in a magazine or book. See "Pages" below to find out how to set a left and right margin on a page.
  - If an odd page count is generated, the last page (which is a duplex backside) has only the master page. To suppress the master page on this last page and exclude this page from page counting, check the option **Omit Master Page Back in case of an empty back page**.

## Pages

Unlike emails, Print sections can contain multiple *pages*. Pages are naturally limited by their size and margins. If the content of a section doesn't fit on one page, the overflow goes to the next page. This happens automatically, based on the section's page size and margins; see "Page settings: size, margins and bleed" on the facing page.

Although generally the same content elements can be used in all three contexts (see "Content elements" on page 104), the specific characteristics of pages make it possible to use special elements, such as page numbers; see "Page numbers" on page 72.

The widow/orphan setting lets you control how many lines of a paragraph stick together, when content has to move to another page; see "Preventing widows and orphans" on page 73. You can also avoid or force a page break before or after an entire element, see "Page breaks" on page 75.

Each page in a print section has a natural position: it is the first page, the last page, a 'middle' page (a page between the first and the last page) or a single page. For each of those positions, a different Master Page and Media can be set. A Master Page functions as a page's background, with for example a header and footer. A Media represents preprinted paper that a page can be printed on. See "Master Pages" on page 76 and "Media" on page 80.

### Page specific content elements

The specific characteristics of pages make it possible to use these special elements:

- **Page numbers** can only be used in a Print context. See "Page numbers" on page 72 to learn how to add and change them.

- Conditional content and dynamic tables, when used in a Print section, may or may not leave an empty space at the bottom of the last page. To fill that space, if there is any, an image or advert can be used as a **whitespace element**; see "Whitespace elements: using optional space at the end of the last page" on the next page.
- Dynamic tables can be used in all contexts, but **transport lines** are only useful in a Print context; see Dynamic table.

## Positioning and aligning elements

Sometimes, in a Print template, you don't want content to move up or down with the text flow. To prevent that, put that content in a Positioned Box. See "Content elements" on page 104.

When it comes to positioning elements on a page, Guides can be useful, as well as Tables. See "How to position elements" on page 179.

## Page settings: size, margins and bleed

On paper, whether it is real or virtual, content is naturally limited by the page size and margins.

These, as well as the bleed, are set per Print section, as follows:

- On the **Resources** pane, right-click a section in the **Print** context and click **Properties**.

For the **page size**, click the drop-down to select a page size from a list of common paper sizes. Changing the width or height automatically sets the page size to Custom.

**Margins** define where your text flow will go. Static elements can go everywhere on a page, that is to say, within the printable space on a page that depends on the printer.

The **bleed** is the printable space around a page. It can be used on some printers to ensure that no unprinted edges occur in the final trimmed document. Note: Printers that can't print a bleed, will misinterpret this setting. Set the bleed to zero to avoid this.

### Tip

By default, measurements settings are in inches (in). You could also type measures in centimeters (add 'cm' to the measurement, for example: 20cm) or in millimeters (for

example: 150mm).

To change the default unit for measurement settings to centimeters or millimeters: on the **Window** menu, click **Preferences**, click **Print**, and then click **Measurements**.

### Whitespace elements: using optional space at the end of the last page

Print sections with conditional content and dynamic tables (see "Personalizing content" on page 202) can have a variable amount of space at the bottom of the last page. It is useful to fill the empty space at the bottom with transpromotional material, but of course you don't want extra pages created just for promotional data. 'Whitespace elements' are elements that will only appear on the page if there is enough space for them.

To convert an element into a whitespace element:

1. Import the promotional image or snippet; see "Images" on page 158 and "Snippets" on page 168.
2. Insert the promotional image or snippet in the content.

#### Note

- Only a top-level element (for example, a paragraph that is not inside a table or div) can function as a whitespace element.
- Do not place the promotional image or snippet inside an absolute positioned box. Whitespace only works for elements that are part of the text flow, not for absolute-positioned boxes.

3. Select the image or the element that holds the promotional content: click it, or use the breadcrumbs, or select it on the **Outline** tab; see "Selecting an element" on page 107.
4. On the **Attributes** pane, check the option **Whitespace element**.
5. (Optional.) Add extra space at the top of the element: on the menu **Format**, click the option relevant to the selected element (Image for an image, Paragraph for a paragraph, etc.) and adjust the spacing (padding and/or margins).  
Do not add an empty paragraph to provide space between the whitespace element and the variable content. The extra paragraph would be considered content and could end up on a separate page, together with the whitespace element.

## Page numbers

### Inserting page numbers

Page numbers can be added to a Print section, but they are usually added to a Master Page, because headers and footers are designed on Master Pages; see also: "Master Pages" on page 76.

To insert a page number, select **Insert > Special character > Markers** on the menu, and then click one of the options to decide with what kind of page number the marker will be replaced:

- **Page number:** The current page number in the document. If a page is empty or does not display a page number, it is still added to the page count.
- **Page count:** The total number of pages in the document, including pages with no contents or without a page number.
- **Content page number:** The current page number in the document, counting only pages with contents that are supplied by the Print section. A page that has a Master Page (as set in the Sheet Configuration dialog, see "Applying a Master Page to a page in a Print section" on page 79) but no contents, is not included in the Content page count.
- **Content page count:** This is the total number of pages in the current document that have contents, supplied by the Print section. A page that has a Master Page but no contents, is not included in the Content page count.
- **Sheet number:** The current sheet number in the document. A sheet is a physical piece of paper, with two sides (or pages). This is equivalent to half the page number, for example if there are 10 pages, there will be 5 sheets.
- **Sheet count:** This marker is replaced by the total number of sheets in the document, whether or not they have contents.

#### Note

When a marker is inserted, a class is added to the element in which the marker is inserted. Do not delete that class. It enables the software to quickly find and replace the marker when generating output. The respective classes are: `pagenumber`, `pagecount`, `contentpagenumber`, `contentpagecount`, `sheetnumber`, and `sheetcount`.

## Creating a table of contents

A table of contents can only be created in a template script. The script should make use of the `pageRef()` function. For an example, see "pageRef()" on page 462. If you don't know how to write a script, see "Writing your own scripts" on page 223.

## Configuring page numbers

By default the page numbers are Arabic numerals (1, 2, 3, etc.) without leading zeros nor prefix, and page numbering starts with page 1 for each section. But this can be changed. To do that:

1. On the **Resources** pane, right-click a section in the **Print** context and click **Numbering**.
2. Uncheck **Restart Numbering** if you want the page numbers to get consecutive page numbers, instead of restarting the page numbering with this section.

### Note

Even if a section is disabled, so it doesn't produce any output, this setting is still taken into account for the other sections. This means that if Restart Numbering is checked on a disabled section, the page numbering will be restarted on the next section.

Disabling a section can only be done in a Control Script (see "Control Scripts" on page 240). Control Scripts can also change where page numbers restart.

3. Use the **Format** drop-down to select uppercase or lowercase letters or Roman numerals instead of Arabic numerals.
4. In **Leading Zeros**, type zeros to indicate how many digits the page numbers should have. Any page number that has fewer digits will be preceded by leading zeros.
5. Type the **Number prefix**. Optionally, check Add Prefix to Page Counts, to add the prefix to the total number of pages, too.
6. Close the dialog.

## Preventing widows and orphans

Widows and orphans are lines at the beginning or at the end of a paragraph respectively, dangling at the bottom or at the top of a page, separated from the rest of the paragraph.

By default, to prevent orphans and widows, lines are moved to the next page as soon as two lines get separated from the rest of the paragraph.

This setting can be changed for the entire Print context, per paragraph and in tables.

#### Note

Widows and orphans are ignored if the **page-break-inside** property of the paragraph is set to **avoid**.

### In the entire Print context

To prevent widows and orphans in the entire Print context:

1. On the menu, select **Edit > Stylesheets**.
2. Select the **Print** context.
3. Click **New** (or, when there are already CSS rules for paragraphs, click the selector **p** and click **Edit**).
4. Click **Format**.
5. After **Widows and Orphans**, type the number of lines that should be considered a widow or orphan (this amounts to the minimum number of lines that may be separated from a paragraph, minus one).

Alternatively, manually set the **widows** and **orphans** properties in a style sheet:

1. Open the style sheet for the Print context: on the **Resources** pane, expand the **Styles** folder and double-click `context_print_styles.css`.
2. Add a CSS rule, like the following:  

```
p { widows: 4; orphans: 3 }
```

### Per paragraph

To change the widow or orphan setting for one paragraph only:

1. Select the paragraph, using the breadcrumbs or the **Outline** pane (next to the **Resources** pane).

2. Select **Format > Paragraph**, on the menu.
3. After **Widows** and **Orphans**, type the number of lines to be considered a widow or orphan (this amounts to the minimum number of lines that may be separated from a paragraph, minus one).

## In tables

The CSS properties **widows** and **orphans** can be used in tables. They are not available in the Table Formatting dialog, however, so they must be added manually, either directly in the `style` attribute of the `<table>` element (on the **Source** tab in the Workspace) or in a style sheet rule, as follows:

1. On the menu, select **Edit > Stylesheets**.
2. Select the **Print** context.
3. Click **New** (or, when there are already CSS rules for tables, click the selector **table** and click **Edit**).
4. Click the **Advanced** button.
5. Add a rule for widows and/or orphans, typing the name of the CSS property in the left column and the value in the right column.
6. Close the dialogs.

## Page breaks

A page break occurs automatically when the contents of a section don't fit on one page.

### Inserting a page break

To insert a page break before or after a certain element, set the `page-break-before` property or the `page-break-after` property of that element (a paragraph for example; see also "Styling text and paragraphs" on page 182):

1. Select the element (see "Selecting an element" on page 107).
2. On the **Format** menu select the respective element to open the Formatting dialog.
3. In the Breaks group, set the **before** or **after** property.
  - **Before**: Sets whether a page break should occur **before** the element. This is equivalent to the `page-break-before` property in CSS; see [CSS page-break-before property](#) for an explanation of the available options.

- **After:** Sets whether a page break should occur **after** the element. Equivalent to the page-break-after property in CSS; see [CSS page-break-after property](#) for an explanation of the available options.

Click the button **Advanced** to add CSS properties and values to the inline style tag directly.

#### Note

You cannot use these properties on an empty <div> or on absolutely positioned elements.

### Preventing a page break

To prevent a page break inside a certain element, set the page-break-inside property of that element to **avoid**:

- Select the element (see "Selecting an element" on page 107).
- On the **Format** menu, select the respective element to open the Formatting dialog.
- In the **Breaks** group, set the **inside** property to **avoid**, to prevent a page break inside the element. This is equivalent to the **page-break-inside** property in CSS; see [CSS page-break-inside property](#) for an explanation of all available options.

### Adding blank pages to a section

How to add a blank page to a section is described in a how-to: [Create blank page on field value](#).

## Master Pages

In Print sections, there are often elements that need to be repeated across pages, like headers, footers and logos. In addition, some elements should appear only on specific pages, such as only the first page, or the last page, or only on pages in-between. Examples are a different header on the first page, and a tear-off slip that shows up on the last page.

This is what Master Pages are used for. Master Pages can only be used in the Print context (see "Print context" on page 58).

Master Pages resemble Print sections, and they are edited in much the same way (see "Editing a Master Page" on the next page) but they contain a single page and do not have any text flow.

Only one Master Page can be applied per page in printed output. Then a Print template is created, one master page is added to it automatically. You can add more Master Pages; see "Adding a Master Page" below. Initially, the original Master Page will be applied to all pages, but different Master Pages can be applied to different pages; see "Applying a Master Page to a page in a Print section" on page 79.

## Examples

This how-to demonstrates the use of Master Pages to show terms and conditions on the back of the first page of a Print section only:

- [Showing a Terms and Conditions on the back of the first page only.](#)

How to use Master Pages to add a tear-off slip to the first page of an invoice is explained in the following how-to :

- [A tear-off section on the first page of an invoice.](#)

## Adding a Master Page

When a Print template is created, one master page is added to it automatically. Adding more Master Pages can be done as follows:

- On the **Resources** pane, right-click the **Master pages** folder and click **New Master Page**.
- Type a name for the master page.
- Optionally, set the margin for the header and footer. See "Master Pages" on the previous page.
- Click **OK**.

Initially, the master page that has been created together with the Print context will be applied to all pages in the Print section. After adding more Master Pages, different Master Pages can be applied to different pages; see "Applying a Master Page to a page in a Print section" on page 79.

## Editing a Master Page

Master Pages are edited just like sections, in the workspace. To open a Master Page, expand the **Master pages** folder on the **Resources** pane, and double-click the Master Page to open it.

A Master Page can contain text, images and other elements (see "Content elements" on page 104), including variable data and dynamic images (see "Personalizing content" on page 202). All elements on a Master Page should have an absolute position or be inside an element that has an absolute position. It is good practice to position elements on a Master Page by placing them in a Positioned Box (see "Content elements" on page 104).

Keep in mind that a Master Page always remains a single page. Its content cannot overflow to a next page. Content that doesn't fit, will not be displayed.

#### Note

Editing the Master Page is optional. One Master Page must always exist in a Print template, but if you don't need it, you can leave it empty.

### Adding a header and footer

Headers and footers are not designed as part of the contents of a Print section, but as part of a Master Page, which is then applied to a page in a print section.

To create a header and footer:

1. First insert elements that form the header or footer, such as the company logo and address, on the Master Page; see "Editing a Master Page" on the previous page.
2. Next, define the margins for the header and footer. The margins for a header and footer are set in the Master Page properties. This does not change the content placement within the Master Page itself; in Master Pages, elements can go everywhere on the page. Instead, the header and footer of the Master Page limit the text flow on pages in the Print sections to which this Master Page is applied. Pages in a Print section that use this Master Page cannot display content in the space that is reserved by the Master Page for the header and footer, so that content in the Print section does not collide with the content of the header and footer. To set a margin for the header and/or footer:
  1. On the **Resources** pane, expand the **Master pages** folder, right-click the master page, and click **Properties**.
  2. Fill out the height of the header and/or the footer. The contents of a print section will not appear in the space reserved for the header and/or footer on the corresponding master page.

3. Finally, apply the master page to a specific page in a print section. See "Applying a Master Page to a page in a Print section" below.

### Applying a Master Page to a page in a Print section

Every page in a print section has a natural position: it can be the first page, the last page, one of the pages in between (a 'middle page'), or a single page. For each of those positions, you can set a different Master Page and Media (see "Media" on the facing page). It can even have two master pages, if printing is done on both sides (called duplex printing).

To apply Master Pages to specific page positions in a Print section:

1. On the **Resources** pane, expand the **Print** context; right-click the Print section, and click **Sheet configuration**.
2. Optionally, check **Duplex** to enable content to be printed on the back of each sheet. Your printer must support duplex for this option to work. If Duplex is enabled, you can also check **Tumble** to duplex pages as in a calendar, and **Facing pages** to have the margins of the section switch alternately, so that pages are printed as if in a magazine or book.
3. If the option **Same for all positions** is checked, the same Master Page will be applied to every page in the print section (and to both the front and the back side of the page if duplex printing is enabled). Uncheck this option.
4. Decide which Master Page should be linked to which sheet (position): click the downward pointing arrow after **Master Page Front** and select a Master Page. If Duplex is enabled, you can also select a Master Page for the back of the sheet and consequently, check **Omit Master Page Back in case of an empty back page** to omit the specified Master Page on the last backside of a section if that page is empty and to skip that page from the page count.
5. Optionally, decide which Media should be linked to each sheet.
6. Click OK to save the settings and close the dialog.

### Deleting a Master Page

To delete a Master Page, expand the **Master pages** folder on the **Resources** pane, right-click the master page, and click **Delete**.

Note that one Master Page as well as one Media must always exist in a Print template. Just leave it empty if you don't need it.

# Media

When the output of a Print context is meant to be printed on paper that already has graphical and text elements on it (called stationery, or preprinted sheets), you can add a copy of this media, in the form of a PDF file, to the Media folder.

Media can be applied to pages in a Print section, to make them appear as a background to those pages. This ensures that elements added to the Print context will correspond to their correct location on the preprinted media.

For further explanation about how to apply Media to different pages, see "Applying Media to a page in a Print section" on page 83.

Media will not be printed, unless you want them to; see below.

Per Media, a front and back can be specified and you can specify on what kind of paper the output is meant to be printed on. This includes paper weight, quality, coating and finishing; see "Adding Media" below.

## Adding Media

To add a Media, right-click the **Media** folder on the **Resources** pane and select **New Media**.

The new Media is of course empty. You can specify two PDF files for the Media: one for the front, and, optionally, another for the back.

## Specifying and positioning Media

### Specifying a PDF for the front: the fast way

To quickly select a PDF file for the front of a Media, import the PDF file by dragging it from the Windows Explorer to the **Images** folder on the **Resources** pane.

Then drag that the PDF file from the **Images** folder and drop it on one of the Media in the **Media** folder. With this method you can not set any options.

To be able to specify a PDF file for both the front and the back of the Media, and to specify a position for the Media's PDF files, you have edit the properties of the Media.

## Setting Media properties

Media have a number of properties that you can set, as described below. What you **cannot** set are a Media's page size and margins. The page size and margins are derived from the section to which the Media is applied.

You can, however, specify a PDF file (or any other type of image file) for both the front and the back of the Media, and specify how the virtual stationery should be positioned on the page. This is done as follows:

1. On the **Resources** pane, expand the **Contexts** folder, expand the **Media** folder, right-click the Media and click **Virtual Stationery**.
2. Click the **Select Image** button to select a PDF image file.
3. Click **Resources**, **Disk** or **Url**, depending on where the image is located.
  - **Resources** lists the PDF files that are present in the **Images** folder on the **Resources** pane.
  - **Disk** lets you choose an image file that resides in a folder on a hard drive that is accessible from your computer. Click the **Browse** button to select an image. As an alternative it is possible to enter the path manually. The complete syntax is: file://<host>/<path>. Note: if the host is "localhost", it can be omitted, resulting in file:///<path>, for example: file:///c:/resources/images/image.jpg. Check the option **Save with template** to insert the image into the **Images** folder on the **Resources** pane.
  - **Url** allows you to choose an image from a specific web address. Select the protocol (**http** or **https**), and then enter the web address (for example, <http://www.mysite.com/images/image.jpg>).

### Note

It is not possible to use a remotely stored PDF file as virtual stationery, because the number of pages in a PDF file can not be determined via the http and https protocols. Therefore, with an external image, the option **Save with template** is always checked.

4. Select a PDF file.
5. If the PDF file consists of more than one page, select the desired page.

6. Click **Finish**.
7. For each of the PDF files, select a position:
  - **Fit to page** stretches the PDF to fit the page size.
  - **Centered** centers the PDF on the page, vertically and horizontally.
  - **Absolute** places the PDF at a specific location on the page. Use the **Top** field to specify the distance between the top side of the page and the top side of the PDF, and the **Left** field to specify the distance between the left side of the page and the left side of the PDF.
8. Finally, click **OK**.

### Setting the paper's characteristics

To set a Media's paper characteristics:

1. On the **Resources** pane, expand the **Contexts** folder, expand the **Media** folder, and right-click the Media. Click **Characteristics**.
2. Specify the paper's characteristics:
  - **Media Type**: The type of paper, such as Plain, Continuous, Envelope, Labels, Stationery, etc.
  - **Weight**: The intended weight of the media in grammage (g/m<sup>2</sup>).
  - **Front Coating**: The pre-process coating applied to the front surface of the media, such as Glossy, High Gloss, Matte, Satin, etc.
  - **Back Coating**: The pre-process coating applied to the back surface of the media.
  - **Texture**: The intended texture of the media, such as Antique, Calenared, Linen, Stipple or Vellum.
  - **Grade**: The intended grade of the media, such as Gloss-coated paper, Uncoated white paper, etc.
  - **Hole Name**: A predefined hole pattern that specifies the pre-punched holes in the media, such as R2-generic, R2m-MIB, R4i-US, etc.
3. Click **OK**.

### Rename Media

To rename Media:

- On the **Resources** pane, expand the **Contexts** folder, expand the **Media** folder, right-click the Media and click **Rename**. Type the new name and click **OK**.
- Alternatively, on the **Resources** pane, expand the **Contexts** folder, expand the **Media** folder, right-click the Media and click **Properties**. Type the new name in the **Name** field and click **OK**.

### Applying Media to a page in a Print section

Every page in a print section has a natural position: it can be the first page, the last page, one of the pages in between (a 'middle page'), or a single page. For each of those positions, you can set different Media.

To apply Media to specific page positions in a Print section:

1. On the **Resources** pane, expand the **Print** context; right-click the Print section, and click **Sheet configuration**.
2. Optionally, check **Duplex** to enable content to be printed on the back of each sheet. Your printer must support duplex for this option to work. If Duplex is enabled, you can also check **Tumble** to duplex pages as in a calendar, and **Facing pages** to have the margins of the section switch alternately, so that pages are printed as if in a magazine or book.
3. If the option **Same for all positions** is checked, the same Media will be applied to every page in the print section. Uncheck this option.
4. Decide which Media should be linked to each sheet position: click the downward pointing arrow after **Media** and select a Media.
5. Optionally, decide which Master Page should be linked to each sheet; see "Master Pages" on page 76.

#### Note

When both Media and a Master Page are used on a certain page, they will both be displayed on the Preview tab of the workspace, the Master Page being 'in front' of the Media and the Print section on top. To open the Preview tab, click it at the bottom of the Workspace or select **View > Preview View** on the menu.

### Dynamically changing the Media

In addition to applying Media to sheets via the settings, it is possible to change Media dynamically, based on a value in a data field, in a script. The script has already been made;

you only have to change the name of the Media and the section in the script, and write the condition on which the Media has to be replaced.

1. On the **Resources** pane, expand the **Contexts** folder, expand the **Print** context, right-click the print section and click **Sheet configuration**.
2. Decide which pages should have dynamically switching media: every first page in the Print section, every last page, one of the pages in between (a 'middle page'), or a single page. (Uncheck the option **Same for all positions**, to see all page positions.)
3. In the area for the respective sheet position, click the **Edit script** button next to **Media**. The Script Wizard appears with a standard script:

```
results.attr("content", "Media 1");
```

Media 1 will have been replaced with the name of the media selected for the chosen sheet position.

The field **Selector** in the Script Wizard contains the name of the section and the sheet position that you have chosen.

4. Change the script so that on a certain condition, another media will be selected for the content. For instance:

```
if(record.fields.GENDER === 'M') {  
    results.attr("content", "Media 2");  
}
```

This script changes the media to Media 2 for male customers.

See "Writing your own scripts" on page 223 if you are not familiar with how scripts are written.

5. Click **Apply**, open the tab **Preview** and browse through the records to see if the script functions as expected.
6. When you click **OK**, the script will be added to the **Scripts** pane.

## Printing virtual stationery

Media are not printed, unless you want them to. Printing the virtual stationery is one of the settings in a Job Creation Preset. To have the virtual stationery printed as part of the Print output:

1. Create a job creation preset that indicates that Media has to be printed: select **File > Presets** and see "[Job Creation Presets](#)" on page 401 for more details.
2. Select that job creation preset in the Print Wizard; see "Generating Print output" on page 495.

# Email

With the Designer you can create one or more Email templates and merge the template with a data set to generate personalized emails.

The Email **context** is the folder in the Designer that can contain one or more Email templates, also called Email **sections**. The HTML generated by this context is meant to be compatible with as many clients and as many devices as possible.

## Email template

It is strongly recommended to start creating an Email template with a Wizard; see "Creating an Email template with a Wizard" on page 89. Designing HTML email that displays properly on a variety of devices and screen sizes is challenging. Building an email is not like building for the web. While web browsers comply with standards (to a significant extent), email clients do not. Different email clients interpret the same HTML and CSS styles in totally different ways.

When an Email template is created, either with a Wizard or by adding an Email context to an existing template (see "Adding a context" on page 48), the Email context folder is created along with other files that are specific to an Email context; see "Email context" on page 93.

Only one Email section is created at the start, but you can add as many Email sections as you need; see "Email templates" on page 95. However, when the Designer merges a data set to generate output from the Email context, it can merge only one of the templates with each record; see "Generating Email output" on page 509.

Email templates are personalized just like any other template; see "Variable Data" on page 211.

## Sending email

When the template is ready, you can change the email settings (see "Email header settings" on page 98) and send the email. To test a template, you can send a test email first.

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

See "Email attachments" on page 511 and "Generating Email output" on page 509.

## Designing an Email template

With the Designer you can design Email templates. It is strongly recommended to start creating an Email template with an Email Template Wizard, because it is challenging to design HTML email that looks good on all email clients, devices and screen sizes that customers use when they are reading their email.

This topic explains why designing HTML email design is as challenging as it is, which solutions are used in the Email Template Wizards and it lists good practices, for example regarding the use of images in HTML email. It will help you to create the best possible Email templates in the Designer.

### HTML email challenges

Creating HTML email isn't like designing for the Web. That's because email clients aren't like web browsers. Email clients pass HTML email through a preprocessor to remove anything that could be dangerous, introduce privacy concerns or cause the email client to behave unexpectedly. This includes removing javascript, object and embed tags, and unrecognized tags. Most preprocessors are overly restrictive and remove anything with the slightest potential to affect the layout of their email client. Next, the HTML has to be rendered so that it is safe to show within the email client. Unfortunately, desktop, webmail, and mobile clients all use different rendering engines, which support different subsets of HTML and CSS. More often than not, the result of these operations is that they completely break the HTML email's layout.

### Designing HTML email in PrintShop MailDesigner

The problem of HTML email is that preprocessing and rendering engines break the HTML email's layout. HTML tables, however, are mostly left untroubled. As they are supported by every major email client, they are pretty much the only way to design HTML emails that are universally supported. That's why Tables are heavily used to position text and images in HTML email.

Nesting tables (putting tables in table cells) and applying CSS styles to each table cell to make the email look good on all screen sizes is a precision work that can be a tedious and demanding. Connect's Designer offers the following tools to make designing HTML email easier.

## Email templates: Slate and others

The most obvious solution offered in the Designer is to use one of the templates provided with the Designer; see "Creating an Email template with a Wizard" on page 89. The layout of these templates has been tested and proven to look good in any email client, on any device and screen size. The Tables in these templates are nested (put inside another table) and they have no visible borders, so readers won't notice them.

### Tip

Click the **Edges** button on the toolbar to make borders of elements visible on the Design tab. The borders will not be visible on the Preview tab or in the output.

## Emmet

Emmet is a plugin that enables the lightning-fast creation of HTML code through the use of a simple and effective shortcut language. The Emmet functionality is available in the HTML and CSS source editors of Connect Designer. Emmet transforms abbreviations for HTML elements and CSS properties to the respective source code. The expansion of abbreviations is invoked with the **Tab** key.

In the Source tab of the Workspace, you could for example type `div.row`. This is the abbreviation for a `<div>` element with the class `row`. On pressing the Tab key, this abbreviation is transformed to:

```
<div class="row"></div>
```

To quickly enter a table with the ID 'green', one row, and two cells in that row, type:

```
table#green>tr>td*2
```

On pressing the Tab key, this is transformed to:

```
<table id="green">
  <tr>
    <td></td>
    <td></td>
  </tr>
</table>
```

All standard abbreviations can be found in Emmet's documentation: [Abbreviations](#).

To learn more about Emmet, please see their website: [Emmet.io](http://emmet.io) and the Emmet.io documentation: <http://docs.emmet.io/>.

## **Preferences**

To change the way Emmet works in the Designer, select **Window > Preferences**, and in the Preferences dialog, select **Emmet**; see "Emmet Preferences" on page 288.

## **Using CSS files with HTML email**

Email clients do not read CSS files and some even remove a <style> tag when it is present in the email's header. Nevertheless, CSS files can be used with the Email context in the Designer. When generating output from the Email context, the Designer converts all CSS rules that apply to the content of the email to inline style tags, as if local formatting was applied.

## **Using images in email campaigns: tips**

### **Host images on a public server**

In the Designer you can add images as resource to the template document. When used in email messages these images are automatically embedded on sending the email. These embedded images appear instantly when viewing the message in your email client. There is, however, a downside to this method: embedded images can't be used to track email open rates. Email services like mandrillapp.com embed a tiny tracer image at the bottom of your message. Each time a recipient opens the email the tracer image (aka beacon image) is downloaded and yet another 'open' is registered. On mobile devices this happens when the user clicks the Display Images button. So, when tracking open rates in your email campaigns, store your images on a publicly-accessible server (preferably your own server - you could set up a process in Workflow to serve images and track open rates) or a reputable image hosting service, like photobucket.com. Don't forget to set the Alternate Text for your images on the Attributes pane.

### **Do not capture your email in one big image**

Most e-mail clients do not automatically download images, so do not capture your email in one big image. The recipient initially sees a blank message and probably deletes it right away.

### **Do not resize images in your email**

Many mail clients do not support image resizing and will show the image in its original dimensions. Resize the images before you link to or embed them.

## Use background images wisely

Most mail clients do not support background images: a very good reason to stay away from them in your mainstream email campaign. There is one situation in which they do come in handy. Both iPhone and Android default mail have solid CSS support and cover most of the mobile marketplace. You could use background images to substitute images when viewed on these devices. This is done by hiding the actual image and showing a mobile-friendly image as background image instead. This is a technique used in Responsive Email Design.

## Creating an Email template with a Wizard

With the Designer you can design Email templates as well as PDF attachments. PDF attachments are designed in the Print context; see "Print context" on page 58.

It is strongly recommended to start creating an Email template with a Wizard, because designing HTML email that displays properly on a variety of devices and screen sizes is challenging. Email clients can, and will, interpret the same HTML and (inline) CSS in totally different ways (see "Designing an Email template" on page 86).

With an Email Template Wizard you can easily create an Email template that outputs emails that look good on virtually any email client, device and screen size.

After creating an Email template, the other contexts can be added to it, as well as other sections (see "Contexts" on page 47 and "Email templates" on page 95).

To create an Email template with a Template Wizard:

1. In the **Welcome** screen that appears after startup:
  - Choose **Browse Template Wizards**.  
Scroll down until you see the Email Template Wizards. There are three types of Email Template Wizards:
    - Basic Email templates
    - Banded Email templates
    - Slate: Responsive Email templates by Litmus.
  - Or choose **Create a New Template** and select the Email template. This starts the Basic Action Email wizard.

Alternatively, on the **File** menu, click **New**, and:

- Select Email Template. This starts the Basic Action Email wizard.
- Or expand the **Template** folder, and then expand the **Basic Email templates** folder, the **Banded Email templates** folder, or the **Slate: Responsive Email Templates by Litmus** folder.

See "Email Template Wizards" on the next page for information about the various types of Template Wizards.

2. Select a template and click **Next**. If you don't know what template to choose, see below; the characteristics of each kind of template are described further down in this topic.
3. Make adjustments to the initial settings (the options for each type of template are listed below). Click **Next** to go to the next settings page if there is one.
4. Click **Finish** to create the template.

The Wizard creates:

- An Email context with one section in it. The section contains dummy text and one or more **selectors** for variable data, for example: "Hello @first@". You will want to replace those by the names of fields in your data. See "Variable Data" on page 211.  
The Invoice email template also contains a Dynamic Table; see Dynamic table.
- One **script**, named "To". Double-click that script on the **Scripts** pane to open it. This script ensures that the email is sent to an email address that is specified in a data field called "email-to". After loading data, you can change the script so that it uses the actual field in your data that holds the customer's email address. See "Email header settings" on page 98
- A style sheet, named context\_html\_email\_styles.css, and another style sheet depending on which Template Wizard was used. The style sheets can be found in the **Stylesheets** folder on the **Resources** pane.

The Wizard opens the Email section, so that you can fill it with text and other elements; see "Content elements" on page 104, "Email context" on page 93, and "Email templates" on page 95.

### Tip

Use the **Outline** pane at the left to see which elements are present in the template and to select an element.

Use the **Attributes** pane at the right to see the current element's ID, class and some other properties.

Use the **Styles** pane next to the Attributes pane to see which styles are applied to the currently selected element.

Note that the contents of the email are arranged in tables. The many tables in an Email template ensure that the email looks good on virtually any email client, device and screen size. As the tables have no borders, they are initially invisible.

### Tip

Click the **Edges** button on the toolbar to make borders of elements visible on the Design tab. The borders will not be visible on the Preview tab.

## Email Template Wizards

There are Wizards for three kinds of Email templates: for **Basic Email**, for **Banded Email**, and **Slate** templates for responsive email designed by Litmus.

### Slate: Responsive Email Templates by Litmus

Scroll past the Web Template Wizards to see the Slate: Responsive Email templates, created by Litmus (see <https://litmus.com/resources/free-responsive-email-templates>).

More than 50% of emails are opened on mobile. These five responsive HTML email templates are optimized for small screens and they look great in any inbox. They've been tested in Litmus and are completely bulletproof.

### Tip

After creating the email template, click the Responsive Design View icon  at the top of the workspace to see how the email looks on different screen sizes.

The only thing you can set in advance for a Slate template is the color of the call-to-action button. Enter a hexadecimal color code. The color value must be a valid [HTML Color Name](#), or a valid hexadecimal color code. To get a hexadecimal color code, you could use an online color picker tool (such as [w3schools' Color Picker](#)). The color can be changed later; see "Colors" on page 190.

## Basic Email and Banded Email

The difference between Basic and Banded email is that the contents of a Basic email extend to the email's margin, rather than to the edge of the window in which it is read, as the contents of Banded emails do.

The Banded Email **Action** Template is a simple call-to-action email with a message, header and a button linking to a website, such as an informational or landing page.

The Banded Email **Invoice** Template is an invoice with an optional Welcome message and Pay Now button.

## Settings

For a **Blank** email you can not specify any settings in the Wizard.

For an **Action** or **Invoice** email, the Email Template Wizard lets you choose:

- The subject. You can change and personalize the subject later, see "Email header settings" on page 98.
- The text for the header. The header is the colored part at the top. The text can be edited later.
- The color of the header and the color of the button. The color value must be a valid [HTML Color Name](#), or a valid hexadecimal color code. To get a hexadecimal color code, you could use an online color picker tool (such as [W3Schools' Color Picker](#)). The color can be changed later; see "Colors" on page 190.
- The web address where the recipient of the email will be taken after clicking the button in the email. Type the URL in the **Link** field.

In addition, for an **Invoice** email you can change the following content settings:

- **Show Welcome Message.** Check this option to insert a salutation and one paragraph with dummy text in the email.

- **Detail Table Name.** Type the name of a detail table to fill the lines of the invoice with data. In the Designer, a detail table is a field in the Data Model that contains a variable number of items (usually transactional data).

#### Note

Detail tables can not be used in PrintShopMail Connect. They are created with the DataMapper module that is included in PlanetPress Connect and PreS Connect.

## Email context

In the Designer the Email context is the folder that contains Email templates. From the Email context, output can be generated in the form of email (see below).

When an Email template is created (see "Creating an Email template with a Wizard" on page 89) or when an Email context is added to a template (see "Adding a context" on page 48) the following happens:

- The Email context is created and one **Email section** is added to it. You can see this on the **Resources** pane: expand the **Contexts** folder, and then expand the **Email** folder. See "Email templates" on page 95 to learn how to fill an Email section. Although only one email can be sent per record when generating Email output, the Email context can contain multiple sections. One Email section is created at the start, but you can add more; see "Adding an Email template" on page 95.
- A style sheet, named context\_htmlmail\_styles.css, is added to the template. Depending on which Template Wizard was used to create the template, another style sheet can be added as well. Style sheets are located in the folder **Stylesheets** on the **Resources** pane. These style sheets are meant to be used for styles that are only applied to elements in the Email context.

The Wizard opens the Email section, so that you can fill it with text and other elements; see "Content elements" on page 104 and "Email templates" on page 95.

### Sending email

When the template is ready, you can generate Email output; See "Generating Email output" on page 509. To test a template, you can send a test email first.

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

#### Note

To split the Print context into multiple attachments, or to attach multiple Web sections as separate attachments, you need to create a Control Script that specifies **parts**; see "Parts: splitting and renaming email attachments" on page 246.

See "Email attachments" on page 511.

### Email output settings

The following settings in an Email context influence how the Email output is generated.

#### Compressing PDF attachments

For PDF attachments, generated from the Print context, you can set the Print Context Image Compression to determine the quality of the files, and with that, the size of the files.

To set the Print Context Image Compression:

1. On the **Resources** pane, expand the **Contexts** folder, and right-click the **Email** context.
2. Click **Properties**.
3. Change the properties of the PDF file that will be attached when the Print context is attached to the email.

**Lossless** is the maximum quality. Note that this will produce a larger PDF file. Uncheck this option to be able to set a lower quality.

The **quality** is set in a percentage of the maximum quality.

**Tile Size** is the size of the files in which the image that is being compressed is divided. (If the image height or width is not an even multiple of the tile size, partial tiles are used on the edges.) Image data for each tile is individually compressed and can be individually decompressed. When low Quality values are used to optimize images smaller than 1024 x 1024 pixels, using the largest tile size will produce better results.

## Setting a default section for output

When generating output from the Email context, only one of the Email templates can be merged with each record. One of the Email sections is the 'default'; see "Setting a default Email template for output" on page 97.

## Email templates

Email templates (also called Email **sections**) are part of the Email context in a template. The Email context outputs HTML email with embedded formatting to an email client through the use of an email server. Since email clients are numerous and do not support same features, the HTML generated by this context is not optimized for any specific client - rather, it's meant to be compatible with as many clients and as many devices as possible.

In Email templates, many content elements can be used; see "Content elements" on page 104. However, special attention must be paid to the way elements are positioned. In Email sections, it is advisable to position elements using Tables and to put text in table cells.

Email templates are personalized just like any other template; see "Variable Data" on page 211.

The subject, recipients (To, CC and BCC), sender and reply-to address are specified with Email Script Wizards; see "Email header settings" on page 98.

An Email context can contain multiple templates. When generating output from the Email context, however, only one of the Email templates can be merged with each record. Set the 'default' Email section (see below) before generating Email output; see also "Generating Email output" on page 509.

## Adding an Email template

When an Email template is created (see "Creating an Email template with a Wizard" on page 89), only one Email section is added to it. An Email context may contain various templates, but per record only one of those can be sent when you generate Email output.

It is not possible to add an Email section to an existing Email context with the help of a Template Wizard.

To provide alternative content for your email, you could use Conditional Content (see "Showing content conditionally" on page 219), or Snippets and a script (see "Snippets" on page 168 and "Loading a snippet via a script" on page 238).

If you would like to start with a template that is identical to the one you already have, consider copying it (see "Copying a section" on page 49).

To add a section to the Email context:

- On the **Resources** pane, expand the **Contexts** folder, right-click the **Email** folder, and then click **New Email**.

### Deleting an Email template

To delete an Email section:

- On the **Resources** pane, expand the **Contexts** folder, expand the **Email** context, right-click the name of the section, and then click **Delete**.

#### Warning

No backup files are maintained in the template. The only way to recover a deleted section, is to click **Undo** on the **Edit** menu, until the deleted section is restored. After closing and reopening the template it is no longer possible to restore the deleted context this way.

To prevent losing any work, it is recommended to configure the auto-save and auto-backup functions in the preferences (see "Saving Preferences" on page 293).

### Styling and formatting an Email template

The contents of an Email section can be formatted directly, or styled with Cascading Style Sheets (CSS). See "Styling and formatting" on page 170.

Email clients do not read CSS files and some even remove a `<style>` tag when it is present in the email's header. Nevertheless, CSS files can be used with the Email context in the Designer. When generating output from the Email context, the Designer converts all CSS rules that apply to the content of the email to inline style tags, as if local formatting was applied.

### Tip

Before you can style an element, you have to select it. In an Email context it can be difficult to select an element by clicking on it. Use the **breadcrumbs** at the top and the **Outline** pane at the left, to select an element. See "Selecting an element" on page 107.

In order for a style sheet to be applied to a specific section, it needs to be included in that section. There are two ways to do this.

### Drag & drop a style sheet

1. Click and hold the mouse button on the style sheet on the **Resources** pane.
2. Move the mouse cursor within the **Resources** pane to the section to which the style sheet should be applied.
3. Release the mouse button.

### Using the Includes dialog

1. On the Resources pane, right-click the section, then click **Includes**.
2. Choose which CSS files should be applied to this section. You can also change the order in which the CSS files are read. This can have an effect on which CSS rule is applied in the end.

### Setting a default Email template for output

An Email context can contain multiple templates. When generating output from the Email context, however, only one of the Email templates can be merged with each record.

To select the Email section that will be output by default:

- On the **Resources** pane, expand the **Email** context, right-click a section and click **Set as Default**.

### Tip

Use a Control Script to dynamically select an Email section for output depending on the

value of a data field.

## Email header settings

Email header settings define the information that goes into the header of each email that is generated from an Email template.

The default Email SMTP settings and the sender's name and address are defined in the Connect Designer preferences and can be adjusted per run in the Send Email and Send Test Email dialogs.

The subject, the recipients (To, CC and BCC), the sender and the reply-to address can be specified with Email Script wizards.

Tip: Load data first, so you can create Email Scripts that use a field in your data. See "Loading data" on page 204.

### Email SMTP settings

Simple Mail Transfer Protocol (SMTP) is the standard protocol for sending emails across the Internet.

Default SMTP settings can be specified in the Preferences dialog: select **Window > Preferences**, expand the **Email** preferences and click **SMTP**.

You can add as many presets as needed, for example for different Email Service Providers (see "Using an ESP with PrintShop Mail Connect" on page 513). To do this, click the Add button at the right. Then fill out the following settings:

- **Name:** The name of the preset. This will show up in the Send Email dialog.
- **Host:** The SMTP server through which the emails are to be sent. This can be a host (mail.domain.com) or an IP address.
- **Port:** You can specify a port number. This will be added to the host name, for example: smtp.mandrillapp.com:465.

- **Use authentication:** Check this option and fill in the user name if a user name and password are needed to send emails through the host. (The password has to be specified in the Send Email or Send Test Email dialog.)
- **Start TLS:** This option is enabled if authentication is checked. It sends emails through Transport Layer Security (TLS), which is sometimes referred to as SSL.

When you click the **Restore** button, the presets for a number of Email Service Providers will appear.

#### Note

When updating the software from a version prior to version 1.5, pre-existing presets will be maintained in the new version.

In the "Send Email" on page 306 and "Send Test Email" on page 307 dialogs you will be able to choose one of the presets and adjust the settings to your needs.

## Subject

To specify a subject for an email template:

1. On the **Resources** pane, expand the **Contexts** folder, and expand the **Email** section.
2. Right-click the section of which the subject should change and click **Properties**. Now you can change the subject.

## Creating a dynamic subject with variable data

To replace an Email section's subject by a dynamic subject:

1. On the Scripts pane, click the black triangle on the **New** button and click **Email Subject Script**. A new script named Subject is added to the Scripts pane.
2. Double-click the new script to open it.
3. Select a data field and type a prefix and/or suffix.

The result of this script will appear in the email as the subject line.

To create a dynamic subject without variable data, or to create a subject that depends on the value of a data field, click **Expand** and modify the script. If you don't know how to write a script, see "Writing your own scripts" on page 223.

## Recipients: To, CC and BCC

The Email To Script Wizard defines the email address to which the email will be sent. The Email CC Script Wizard and the Email BCC Script Wizard define additional email addresses to which the email will be sent.

The To and CC email addresses will be visible to all of the recipients. The BCC ('blind carbon copy') email address(es) will not be visible to any of the other recipients.

To specify recipients for Email output:

1. On the Scripts pane, click the black triangle on the **New** button and click **Email To Script**, **Email CC Script** or **Email BCC script**. A new script is added to the Scripts pane.
2. Double-click the new script to open it.
3. Select a data field that holds an email address.

The result of this script goes in the email's To, CC, or BCC address field, respectively. It should be a valid, fully-formed email address.

## Sender

### From address

A default **From** name and email address can be specified in the Preferences dialog: select **Window > Preferences**, expand the **Email** preferences and click **General**.

This name and email address will appear as the default in the "Send Email" on page 306 and "Send Test Email" on page 307 dialogs.

To dynamically specify a From address you have to use the Email From Script Wizard:

1. On the Scripts pane, click the black triangle on the **New** button and click **Email From Script**. A new script is added to the Scripts pane.
2. Double-click the new script to open it.
3. Select a data field that holds an email address.

The result of this script overwrites the address given in the Send Email dialog or Send Test Email dialog. It should be a valid, fully-formed email address.

### **Reply-To address**

The Reply-To address is often used when sending email campaigns and to do tracking of email replies.

The Reply-To address has to be specified in a script:

1. On the Scripts pane, click the black triangle on the **New** button and click **Email From Script** or **Email Reply-To Script**. A new script is added to the Scripts pane.
2. Double-click the new script to open it.
3. Select a data field that holds an email address.

The result of this script should be a valid, fully-formed email address.

### **Email PDF password**

The Email PDF Password Script Wizard defines a password with which to protect the PDF generated when using the Print context as PDF Attachment option in the Send Email or Send Test Email dialogs (see "Generating Email output" on page 509). The result of the script will be the password necessary to open the PDF when it is received by email.

To define a password to protect the generated PDF attachment:

1. On the Scripts pane, click the black triangle on the **New** button and click **Email PDF password Script**. A new script is added to the Scripts pane.
2. Double-click the new script to open it.
3. Select a data field and optionally, type a prefix and/or suffix.

### **Password types**

PDF allows for two types of passwords to be set on a secured PDF file: a user password and owner password. The user password allows a limited access to the file (e.g. printing or copying text from the PDF is not allowed). The owner password allows normal access to the file. The Email PDF password script sets both the user and owner password to the same value, so that when the recipient provides the password, he can manipulate the file without limitations.

### Note

If a template has a Control Script that creates multiple PDF attachments, all the attachments are secured by the same password.

### Note

Via a Control Script it is possible to set a different user password and owner password, see "Control Script: Securing PDF attachments" on page 252, "Control Scripts" on page 240 and "Control Script API" on page 477.

## Email attachments

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

By default, when adding the Print context to an email, all Print sections are output to a single PDF file, named after the email subject, which is then attached to the email. The PDF can be protected with a password (see "Email PDF password" on the previous page).

### Note

To split the Print context into multiple attachments, or to attach multiple Web sections as separate attachments, you need to create a Control Script that specifies **parts**; see "Parts: splitting and renaming email attachments" on page 246.

This topic explains how to attach files other than those generated by the Print or Web context. This is also described in a how-to; see [Add custom email attachments](#).

### Attaching external files

To attach files other than those generated by the Print or Web context to Email output:

1. Add the files to the template; see Adding images, or put them in a folder that is available to the machine that outputs the emails.
2. Create a script: on the **Scripts** pane at the bottom left, click **New**. A new script appears in the list. Double-click on it to open it. If you are not familiar with scripts, see "Writing your own scripts" on page 223 for an explanation of how scripts work.
3. Change the name of the script, so that it reflects what the script does.
4. Choose the option **Selector** and in the **Selector** field, type **head**.
5. Write a script that appends a <link> element to the `results` (the selector is `head`, so the `results` contain the <head> of the email).
  - Make sure to set the **rel** attribute to **related**.
  - The **href** attribute determines where the file comes from. For resources inside of the template, use `'images/file.extension'`, or `'fonts/myfont.otf'`, etc. For external resources, you need the full path to the file, such as `'file:///c:/resources/attachments/instructions.pdf'`. Of course, you can also use dynamic calls such as `'file:///c:/clientfiles/' + record.fields.client_id + '/invoices/' + record.fields.invoice_number + '.pdf'`.

## Examples

The following script attaches a PDF file named `letter-CU00048376.pdf` to each generated email. The PDF file is located in the Images folder on the Resources panel.

```
results.append("<link rel='related' href='images/letter-CU00048376.pdf'>");
```

If that same file would be located on the C: drive, the script should refer to it as follows:

```
href='file:///C:/letter-CU00048376.pdf'.
```

The link doesn't have to be static; you could use data from the record set to build the link, for example:

```
var customerID = record.fields.ID;
results.append('<link rel="related" href="images/letter-' +
customerID + '.pdf">');
```

# Content elements

Once you have created template, it can be filled with all kinds of elements, from text to barcodes and from tables to fields on a web form. All types of elements are listed on this page; see below.

There are several ways to insert elements, see "Inserting an element" on page 107.

Each element can have an ID and a class, as well as a number of other properties, depending on the element's type. When an element is selected, its properties can be changed; see "Selecting an element" on page 107, "Attributes" on page 106 and "Styling and formatting an element" on page 108.

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file. It is possible to edit the source of the HTML file directly in the Designer; see "Editing HTML" on the next page.

## Element types

The following types of content can be added to the content of a template:

- "Images" on page 158 and "Dynamic Images" on page 222
- "Text and special characters" on page 167
- "Date" on page 154
- "Table" on page 163
- "Boxes" on page 150: Positioned Box, Inline Box, Div and Span

### Tip

Wrapping elements in a box (see "Boxes" on page 150) or in a semantic HTML element makes it easier to target them in a script or a style sheet. Place the cursor in the element or select multiple elements. Then, on the menu, click **Insert > Wrap in Box**. You can now use the wrapper element as a script's or style's `selector`; see "Using the Text Script Wizard" on page 214 and "Styling and formatting" on page 170.

- "Hyperlink and mailto link" on page 156

- "Barcode" on page 109
- "Whitespace elements: using optional space at the end of the last page" on page 71 (Print context only)
- "Page numbers" on page 72 (Print context only)
- Article, Section, Header, Footer, Nav and Aside are HTML5 semantic elements; see [http://www.w3schools.com/html/html5\\_semantic\\_elements.asp](http://www.w3schools.com/html/html5_semantic_elements.asp)
- Other HTML elements: Heading, Address and Pre
- "Snippets" on page 168: a Snippet is a small, ready-to-use piece of content in a file
- Business graphics

Most elements are suitable for use in all contexts. There are a few exceptions, however. Forms and Form elements can be used on web pages only, whereas Whitespace elements and Page numbers can only be used in a Print context. Positioned boxes are well suited for Print sections, but are to be avoided in the Email context.

Whether it is best to use a Table or Box to position text, images and other elements, depends on the context in which they are used; see "How to position elements" on page 179 for more information.

## Editing HTML

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file.

To see this, toggle to the **Design** tab in the workspace. Click anywhere in the content. Take a look at the *breadcrumbs* at the top of the workspace. The breadcrumbs show the HTML tag of the clicked element, as well as the HTML tags of other elements to which the clicked element belongs. The clicked element is at the end of the line.

To edit the HTML text directly:

- In the workspace, toggle to the **Source** tab.

On this tab you can view and edit the content of the template in the form of plain text with HTML tags (note the angle brackets: <>). You may add and edit the text and the HTML tags, classes, ID's and other attributes.

To learn more about HTML, see for example <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Introduction> and <http://www.w3schools.com/html/default.asp>.

Many video courses and hands-on courses about HTML (and CSS) are offered on the Internet as well, some for free. Go, for example, to [www.codeschool.com](http://www.codeschool.com) or [www.codecademy.com](http://www.codecademy.com) and look for HTML (and CSS) courses.

## Attributes

### ID and class

Every element in the content of a template can have an **ID** and a **class**. ID's and classes are particularly useful with regard to variable data (see "Personalizing content" on page 202) and styling (see "Styling templates with CSS files" on page 171).

You can specify an ID and/or class when you add the element to the content.

To add an ID and/or class to an element that has already been added to a template, select the element (see "Selecting an element" on the next page) and type an ID and/or a class in the respective fields on the **Attributes** pane at the top right.

### Other attributes

Apart from the ID and class, elements can have a varying number of properties, or 'attributes' as they're called in HTML (see "Editing HTML" on the previous page). Which properties an element has, depends on the element itself. An image, for example, has at least four attributes: `src` (the image's URL), `alt` (alternate text), `width` and `height`. These attributes are visible on the **Attributes** pane when you click an image in the content.

For each type of element, a small selection of attributes is visible on the **Attributes** pane at the top right.

### Changing attributes via script

Many attributes can be changed via the user interface. Another way to change attributes is by using a script.

Any of the Script Wizards can produce a script that changes an attribute of an HTML element. Set the **Options** in the Script Wizard to **Attribute**, to output the script's results to the value of a specific attribute. See "Using the Text Script Wizard" on page 214.

In code, you can change an element's attribute using the function `attr()`; see "Writing your own scripts" on page 223 and "Designer Scripts API" on page 429.

## Inserting an element

To insert an element in the content of a template:

1. Click the respective toolbar button. Alternatively, click the element on the **Insert** menu.
2. Add an ID and/or a class. ID's and classes are particularly useful with regard to variable data (see "Personalizing content" on page 202) and styling (see "Styling templates with CSS files" on page 171).
3. Use the **Location** drop-down (if available) to select where to insert the element.
  - **At cursor position** inserts it where the cursor is located in the template.
  - **Before element** inserts it before the HTML element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be before the `<p>` tag.\*
  - **After start tag** inserts it within the current HTML element, at the beginning, just after the start tag.\*
  - **Before end tag** inserts it within the current HTML element, at the end, just before the end tag.\*
  - **After element** inserts it after the element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be after the end tag of the paragraph (`</p>`).\*

\* If the current element is located inside another element, use the **Elements** drop-down to select which element is used for the insertion location. The list displays every element in the breadcrumbs, from the current selection point until the root of the body.

## Selecting an element

When an element is selected, the **Attributes** pane shows the attributes of that element, and the **Styles** pane, next to the **Attributes** pane, shows which styles are applied to it.

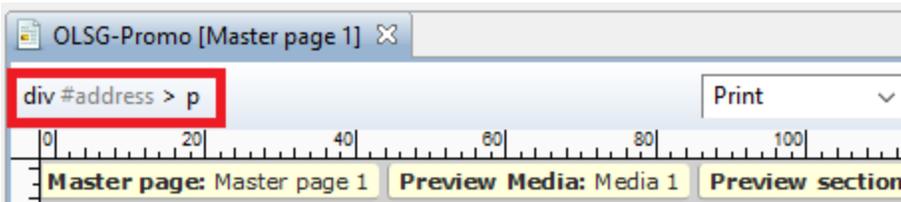
To select an element in the content, you can of course click on it, but this isn't always as easy as it seems, especially when the element has elements inside it.

## Tip

Click the **Edges** button on the toolbar to make borders of elements visible on the Design tab. The borders will not be visible on the Preview tab.

There are two more ways to select an element in the content:

- Use the *Breadcrumbs* at the top of the workspace.



Breadcrumbs show the HTML tag of the clicked element, as well as the HTML tags of 'parent elements': elements inside of which the clicked element is located. The clicked element is at the end of the line.

Elements with classes or IDs show these details next to them, for instance `div #contents > ol.salesitems > li`. Click any of the elements in the Breadcrumbs to select that element.

If an element is selected in the Breadcrumbs and the Backspace key is pressed, that element is deleted.

- Use the **Outline** pane. You can find this pane next to the **Resources** pane. It displays a tree view of the elements in the file. Click an element in the tree view to select it.

## Styling and formatting an element

### Format elements directly

Images and other graphical elements can be resized by clicking on them and dragging the resize handles. There are toolbar buttons to color, indent or style text. Other toolbar buttons can left-align, right-align, or rotate graphical elements.

The toolbar buttons only represent a selection of the formatting options for each element. There are no toolbar buttons to change an element's margins, or to add a border to it, for example. To access all formatting properties of an element, you have to open the Formatting dialog. There are two ways to do this:

- Right-click the element and select the type of element on the shortcut menu.
- Select the element (see "Selecting an element" on page 107) and select the type of element on the **Format** menu.

See "Styling and formatting" on page 170 for more information about the formatting options.

### **Format elements via Cascading Style Sheets (CSS)**

It is highly recommended to use style sheets in templates right from the start. Even more so if the communications are going to be output to different output channels, or if they consist of different sections (for example, a covering letter followed by a policy). Using CSS with templates allows a consistent look and feel to be applied. A style sheet can change the look of multiple elements, making it unnecessary to format each and every element in the template, time and again, when the company's layout preferences change. See "Styling templates with CSS files" on page 171.

## **Barcode**

In PrintShop Mail Connect Designer, you can add a variety of barcodes to your template. The supported Barcode types include 1d barcodes (the striped ones) and 2d barcodes (encoded horizontally and vertically).

### **Adding a Barcode**

#### **Note**

When generating Print output, you can add extra barcodes and OMR marks. The reason why you would do this, is that at merge time more information is available about the actual output document. The page count, for example, is not available at design time.

To add barcodes and OMR marks on the fly when generating Print output, select **File > Print** and check the option **Add additional content** (see "Additional Content" on page 365) in the Print Wizard. To have this done automatically, save this and other output options in an Output Creation Preset: select **File > Print presets > Output Creation Settings** (see "Output Creation Settings" on page 410).

Before adding a Barcode, load data or at least a Data Model; see "Loading data" on page 204. You will need the field names when adding the Barcode. Then, to add a Barcode to a section, Master Page or snippet:

1. Select **Insert > Barcode** on the menu or click the **Barcode** toolbar button
2. Choose the desired barcode type. The list is divided between 1d and 2d barcodes.
3. An **ID** is required. You can change the given ID and, optionally, add a class.
4. Check the option **Absolute** to insert the barcode in an absolute-positioned box inside the `<body>` of the HTML, but outside other elements. Alternatively, use the **Location** drop-down to select where to insert the Barcode.
  - **At cursor position** inserts it where the cursor is located in the template.
  - **Before element** inserts it before the HTML element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be before the `<p>` tag.\*
  - **After start tag** inserts it within the current HTML element, at the beginning, just after the start tag.\*
  - **Before end tag** inserts it within the current HTML element, at the end, just before the end tag.\*
  - **After element** inserts it after the element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be after the end tag of the paragraph (`</p>`).\*

\* If the current element is located inside another element, use the **Elements** drop-down to select which element is used for the insertion location. The list displays every element in the breadcrumbs, from the current selection point until the root of the body.

5. Under **Script**, select the field that contains the barcode value. The barcode type dictates the length and exact format of the required value. For a detailed description or for background information on a specific barcode, please refer to the documentation provided by the individual barcode supplier. Note that some barcode readers may require specific parameters as well.

If it is necessary to concatenate fields to compose the barcode value, edit the script after adding the barcode; see "Barcode script" on the next page.

#### Note

For barcodes that require a Checksum, the Designer can calculate a Checksum if that isn't provided by your data. In that case the field should contain the required

value minus the Checksum. To include a calculated Checksum in the barcode value, edit the barcode properties after adding the barcode to the template; see below.

6. Click **OK** to close the dialog.

In the template the barcode shows up as a gray box. The associated barcode script is added to the Scripts pane. To see the barcode script working, toggle to the **Preview** tab in the Workspace.

A barcode is always added with the barcode type's default properties and dimensions, but they can easily be changed; see "Barcode type and properties" below.

## Changing a barcode

### Barcode script

The barcode script determines which value is fed to the barcode generator. Double-click the script on the **Scripts** pane to change which field or fields are added to the barcode value. When you select more than one field, the script puts the values of the selected fields in one string and passes that to the barcode generator.

### Tip

If you don't know which script matches the barcode, click the box that contains the barcode and check the ID of that box on the Attributes pane. Then take a look at the Scripts pane: the selector of the associated script is the same as the ID of the barcode box.

### Barcode type and properties

To change the barcode type or the barcode's properties such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select **Barcode** on the shortcut menu.

A barcode is always added with the barcode type's default properties and dimensions. Click the barcode type below for information about its properties.

- "Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5" on page 134
- "Code 39, Code 39 extended" on page 118
- "UPC-A, UPC-E, EAN-8, EAN-13" on page 148
- "OneCode, KIX Code, Royal Mail, Australia Post" on page 145
- "Code 128" on page 122
- "GS1-128" on page 127
- "Codabar" on page 115
- "MSI" on page 136
- "IMPB" on page 130
- "Postnet" on page 141
- "QR Code" on page 143
- "Data Matrix" on page 123
- "Royal Mail Mailmark" on page 147
- "PDF417" on page 139
- "Aztec Code" on the next page
- "MaxiCode" on page 136

### **OneCode, KIX Code, Royal Mail, Australia Post**

OneCode, KIX Code, Royal Mail and Australia Post are some of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

### **Barcode properties**

This topic lists the properties of the barcode types OneCode, KIX Code, Royal Mail and Australia Post. For the properties of other barcode types, see "Barcode type and properties" on the previous page.

## Height, width and spacing

The height, width and spacing of the barcode are all measured in pixels (38 dpi).

- **Bar height:** the height of the (shorter) bars
- **Extended bar height:** the total height of the extended bars
- **Bar width:** the width of the bars
- **Spacing:** the distance between the bars

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Aztec Code

Aztec is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type Aztec. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module size

Enter the size of the square modules in pixels

### Configuration type

Use the drop-down to select the format type used when creating the barcode: only full range format, only compact formats, or any format.

### Preferred configuration

Use the drop-down to select the preferred format for the barcode. Note that the barcode generator may choose a different format if the data cannot be represented by the preferred format.

### Encoding

Use the drop-down to select the encoding type:

- **Normal** can encode any character but is not very efficient for encoding binary values (above 128)
- **Binary** is to be used only if the data contains many bytes/characters above 128.

### Error Correction Level

This option reserves a percentage of the symbol capacity for error correction. The recommended percentage for this type of barcode is 23.

## Rune

When set to a value between 0 and 255, an Aztec Rune corresponding to the selected value is created. Set the Rune to -1 to disable this feature.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Codabar

Codabar is one of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the Codabar barcode. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### Start Char and Stop Char

Use the drop-down to select the start and stop character for the barcode, which defines the encoding mode. Available characters are A, B, C.

### Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

### Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

### Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example

[w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5

Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, and Matrix 2 of 5 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the following barcode types :

- Code 11
- Code 93
- Code 93 extended
- Industrial 2 of 5
- Interleaved 2 of 5
- Matrix 2 of 5

For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Code 39, Code 39 extended

Code 39 and Code 39 extended are two of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types Code 39 and Code 39 extended. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### Inter Character Gap

Two adjacent characters are separated by an inter-character gap. A value of 1 means that the separator will have the same length as the width of the narrow bars (in centimeters).

### Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5

Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, and Matrix 2 of 5 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the following barcode types :

- Code 11
- Code 93
- Code 93 extended

- Industrial 2 of 5
- Interleaved 2 of 5
- Matrix 2 of 5

For the properties of other barcode types, see "Barcode type and properties" on page 111.

### **Module width**

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### **Scale**

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### **Add Checksum**

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

### **Human Readable Message**

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

### **Color**

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

### **Output format**

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Code 128

Code 128 is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type Code 128. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### Code set

Set of characters to be used:

- **A:** ASCII characters 00 to 95 (0–9, A–Z and control codes), special characters, and FNC 1–4
- **B:** ASCII characters 32 to 127 (0–9, A–Z, a–z), special characters, and FNC 1–4
- **C:** 00–99 (encodes each two digits with one code) and FNC 1

In Auto mode, the barcode generator will automatically select the correct encoding mode (set A, B or C) according to the input data.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Data Matrix

Data Matrix is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the QR barcode. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Dots per pixel

Type the number of dots per pixel. To optimize barcode quality a Data Matrix symbol should not be printed with dots smaller than 4 pixels.

### Encoding

The data represented in the symbol can be compressed using of the following algorithms.

- **ASCII** is used to encode data that mainly contains ascii characters (0-127)
- **C40** is used to encode data that mainly contains numbers and uppercase characters.
- **Text** is used to encode data that mainly contains numbers and lowercase
- **Base256** is used to encode 8 bit values
- **Auto Detect** automatically detects the data content and encodes using the most appropriate method.
- **None** does not use any encoding.

### Preferred format

Use the drop-down to select the size of the Data Matrix.

### Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## UPC-A, UPC-E, EAN-8, EAN-13

UPC-A, UPC-E, EAN-8 and EAN-13 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types UPC-A, UPC-E, EAN-8 and EAN-13. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

## Show guardbars

Checking this option adds guardbars to the barcode. Guardbars are bars at the start, in the middle and at the end that help the barcode scanner to scan the barcode correctly.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Supplement

UPC-A, UPC-E, EAN-13, and EAN-8 may all include an additional barcode to the right of the main barcode.

- **Type:** The supplement type can be 2-digit (originally used to indicate the edition of a magazine or periodical) or 5-digit (used to indicate the suggested retail price for books). In case this option is set to None, and the data includes digits for the 2 or 5 supplement, the supplement data will be skipped and the additional barcode will not be rendered.

### Note

When the chosen supplement type doesn't match the data, the supplement data will be skipped and the additional barcode will not be rendered.

- **Height Factor:** This is the relative height of the supplement's bars compared to the normal bars.
- **Space Before :** Defines the space between the main symbol and the supplement, in cm.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## GS1-128

GS1-128 is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type GS1-128. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## OneCode, KIX Code, Royal Mail, Australia Post

OneCode, KIX Code, Royal Mail and Australia Post are some of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types OneCode, KIX Code, Royal Mail and Australia Post. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Height, width and spacing

The height, width and spacing of the barcode are all measured in pixels (38 dpi).

- **Bar height:** the height of the (shorter) bars
- **Extended bar height:** the total height of the extended bars
- **Bar width:** the width of the bars
- **Spacing:** the distance between the bars

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## IMPB

IMPB is one of the barcode types that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert** > **Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type IMPB. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

### **Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5**

Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, and Matrix 2 of 5 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

### **Barcode properties**

This topic lists the properties of the following barcode types :

- Code 11
- Code 93
- Code 93 extended
- Industrial 2 of 5
- Interleaved 2 of 5
- Matrix 2 of 5

For the properties of other barcode types, see "Barcode type and properties" on page 111.

### **Module width**

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### **Scale**

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

### Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

### Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

### Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

### Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5

Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, and Matrix 2 of 5 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the following barcode types :

- Code 11
- Code 93
- Code 93 extended
- Industrial 2 of 5
- Interleaved 2 of 5
- Matrix 2 of 5

For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None**: The barcode is rendered based on the module width.
- **Fit to box**: The barcode is stretched to fit the parent box in both width and height.
- **Proportionally**: The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, Matrix 2 of 5

Code 11, Code 93, Code 93 extended, Industrial 2 of 5, Interleaved 2 of 5, and Matrix 2 of 5 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the following barcode types :

- Code 11
- Code 93
- Code 93 extended

- Industrial 2 of 5
- Interleaved 2 of 5
- Matrix 2 of 5

For the properties of other barcode types, see "Barcode type and properties" on page 111.

### **Module width**

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### **Scale**

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### **Add Checksum**

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

### **Human Readable Message**

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

### **Color**

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

### **Output format**

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## MaxiCode

MaxiCode is one of the barcode types that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the MaxiCode barcode. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Resolution

Select the printer output definition for the barcode (200, 300, 400, 500 or 600 dpi).

## Mode

PrintShop Mail Connect supports several modes; for an explanation of these modes see the [MaxiCode page on Wikipedia](#).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## MSI

MSI is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type MSI. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### Add Checksum

When checked, PrintShop Mail Connect will calculate a Checksum character and add that to the result of the Barcode script. If the value to be encoded is longer than 10 digits, a second check character will be calculated.

### Checksum Type

The Checksum type can be MSI10, MSI11, MSI1010 or MSI1110; see [https://en.wikipedia.org/wiki/MSI\\_Barcode](https://en.wikipedia.org/wiki/MSI_Barcode).

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## OneCode, KIX Code, Royal Mail, Australia Post

OneCode, KIX Code, Royal Mail and Australia Post are some of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types OneCode, KIX Code, Royal Mail and Australia Post. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Height, width and spacing

The height, width and spacing of the barcode are all measured in pixels (38 dpi).

- **Bar height:** the height of the (shorter) bars
- **Extended bar height:** the total height of the extended bars
- **Bar width:** the width of the bars
- **Spacing:** the distance between the bars

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## PDF417

PDF417 is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type PDF417. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Mode

Use the drop-down to set the compaction mode:

- **Binary:** allows any byte value to be encoded
- **Text:** allows all printable ASCII characters to be encoded (values from 32 to 126 and some additional control characters)
- **Numeric:** a more efficient mode for encoding numeric data

### Error Correction Level

Use the drop-down to select the built-in error correction method based on Reed-Solomon algorithms. The error correction level is adjustable between level 0 (just error detection) and level 8 (maximum error correction). Recommended error correction levels are between level 2 and 5, but the optimal value depends on the amount of data, printing quality of the PDF417 symbol and decoding capabilities of the scanner.

### Nr. of Columns

The number of data columns can vary from 3 to 30.

### Nr. of Rows

A PDF417 bar code can have anywhere from 3 to 90 rows.

### Bar height

Defines the height of the bars for a single row measured in pixels drawn.

## Compact

Check this option to use Compact PDF417 instead of the PDF417 barcode. This shortened form of the PDF417 barcode is useful where the space for the symbol is restricted.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Postnet

Postnet is one of the barcode types that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type Postnet. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

### Bar height

You can set the height (in cm) of the short bars and the tall bars in the Postnet barcode.

### Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

### Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

### Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

### Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## QR Code

A QR Code is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

Instead of using the Script wizard (see "Adding a Barcode" on page 109) you could write your own script to get the data for the QR Code; see this how-to: [QR Codes in Designer](#).

## Barcode properties

This topic lists the properties of the QR barcode. For the properties of other barcode types, see "Barcode type and properties" on page 111.

### Module size

Enter the size of the square modules in pixels.

### Auto configure

When this option is checked, the barcode generator overwrites the selected Preferred version (see below) and defines the barcode version based on the supplied data.

### Preferred version

There are 40 sizes of QR codes. Select the preferred version for the QR code.

## Encoding

This option defines the encoding of the barcode. When **Auto** is selected, the barcode generator determines the encoding based on the supplied string. The other options are:

- **Numeric:** 10 bits per 3 digits, with a maximum of 7089 numerical characters.
- **Alphanumeric:** 11 bits per 2 characters, with a maximum of 4296 alphanumeric characters.
- **Byte:** 8 bits per character, with a maximum of 2953 characters.
- **Kanji:** 13 bits per character, with a maximum of 1817 characters.

## Extended Channel Interpretation (ECI)

This setting enables data using character sets other than the default set. Select **Latin-1**, **Shift JIS** or **UTF-8**, or select **None** to disable extended channel interpretation.

## Correction level

Part of the robustness of QR codes in the physical environment is their ability to sustain 'damage' and continue to function even when a part of the QR code image is obscured, defaced or removed. A higher correction level duplicates data within the QR Code to that effect, making it larger.

## FNC

Use the drop-down to either disable FNC or select a FNC option:

- **First:** This mode indicator identifies symbols encoding data formatted according to the UCC/EAN Application Identifiers
- **Second:** This mode indicator identifies symbols formatted in accordance with specific industry or application specifications previously agreed with AIM International. You must then set a value for the Application Indicator property.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.

- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Barcode Data

QR Codes can have many different types of data, which determines how the code will be generated. On top of just straightforward data, special data structures are used to trigger actions on the device that reads them. This can include contact cards, phone numbers, URLs, emails, etc.

To learn more about the specifications of the different QR code types, see the ZXing Project [barcode contents](#) page.

## OneCode, KIX Code, Royal Mail, Australia Post

OneCode, KIX Code, Royal Mail and Australia Post are some of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types OneCode, KIX Code, Royal Mail and Australia Post. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Height, width and spacing

The height, width and spacing of the barcode are all measured in pixels (38 dpi).

- **Bar height:** the height of the (shorter) bars
- **Extended bar height:** the total height of the extended bars
- **Bar width:** the width of the bars
- **Spacing:** the distance between the bars

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Royal Mail Mailmark

Royal Mail Mailmark is one of the types of barcodes that can be added to a template; see "Barcode" on page 109.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode type Royal Mail Mailmark. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

The recommendation is to print these barcodes with a module size of 0.5 mm, which equates to 6 dots when printed at 300dpi. The maximum module size for printing is 0.7 mm.

## Preferred version

Use the drop-down to select the size of the barcode, in a number of modules. The actual size of the barcode can be 12 mm x 12 mm up to 22.4 mm x 22.4 mm, depending on the preferred version and the module width.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example [w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG**: Vector format. This is smaller in size, but not compatible with Email output.
- **PNG**: Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## UPC-A, UPC-E, EAN-8, EAN-13

UPC-A, UPC-E, EAN-8 and EAN-13 are a few of the barcode types that can be added to a template.

The barcode can be added either using the Barcode toolbar button or through selecting **Insert > Barcode** on the menu; see "Adding a Barcode" on page 109.

Initially the barcode will have the barcode type's default properties. To change those properties, such as the scale and color, open the Barcode properties dialog: right-click the barcode (on the **Design** tab in the Workspace) and select the barcode type on the shortcut menu.

## Barcode properties

This topic lists the properties of the barcode types UPC-A, UPC-E, EAN-8 and EAN-13. For the properties of other barcode types, see "Barcode type and properties" on page 111.

## Module width

Specifies the width of the narrow bars in centimeters. Changing this value to a higher value will make the barcode bigger when Scale is set to None.

## Show guardbars

Checking this option adds guardbars to the barcode. Guardbars are bars at the start, in the middle and at the end that help the barcode scanner to scan the barcode correctly.

## Scale

Defines if and how the rendered barcode is scaled in relation to the parent element:

- **None:** The barcode is rendered based on the module width.
- **Fit to box:** The barcode is stretched to fit the parent box in both width and height.
- **Proportionally:** The barcode is stretched up to where it fits either the width or height of the parent box, whichever requires the less stretching.

## Supplement

UPC-A, UPC-E, EAN-13, and EAN-8 may all include an additional barcode to the right of the main barcode.

- **Type:** The supplement type can be 2-digit (originally used to indicate the edition of a magazine or periodical) or 5-digit (used to indicate the suggested retail price for books). In case this option is set to None, and the data includes digits for the 2 or 5 supplement, the supplement data will be skipped and the additional barcode will not be rendered.

### Note

When the chosen supplement type doesn't match the data, the supplement data will be skipped and the additional barcode will not be rendered.

- **Height Factor:** This is the relative height of the supplement's bars compared to the normal bars.
- **Space Before :** Defines the space between the main symbol and the supplement, in cm.

## Human Readable Message

When this option is checked, PrintShop Mail Connect shows a human readable text below or above the barcode, as defined using the Text Position, using the specified font and font size. The font size is given in points (pt).

## Color

The **Color** property allows you to choose a different **Barcode** color (instead of black) and **Background** color (instead of white), by typing a hexadecimal color value (see for example

[w3school's color picker](#)).

## Output format

Defines how the barcode is output on the page. There are two possible formats:

- **SVG:** Vector format. This is smaller in size, but not compatible with Email output.
- **PNG:** Binary rasterized format. This is slightly larger than SVG but will display properly in Email output.

## Boxes

Boxes are elements that are used to surround other elements, either to style them, to find them, or to place them in specific locations.

### Tip

Wrapping elements in a box (see "Boxes" above) or in a semantic HTML element makes it easier to target them in a script or a style sheet. Place the cursor in the element or select multiple elements. Then, on the menu, click **Insert > Wrap in Box**. You can now use the wrapper element as a script's or style's `selector`; see "Using the Text Script Wizard" on page 214 and "Styling and formatting" on page 170.

## Positioned Box

A Positioned Box is one that can be freely moved around the page and does not depend on the position of other elements. A positioned box is actually a `<div>` element that has an **absolute position**; in other words, it has its CSS property `position` set to `absolute`.

Positioned Boxes are suitable for use in Print templates only.

## Adding a Positioned Box

To insert a Positioned Box, use the  icon on the toolbar. Positioned Boxes can be moved by dragging the borders, and resized using the handles on the sides and the corners. They can be styled using the **Format > Box** menu item, through the CTRL+M keyboard shortcut or through the CSS files; see "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.

## Inline Box

An Inline Box is one that is placed within the text flow, where other elements (including text) can wrap around it. An inline box is actually a <div> element that is **floating**; in other words, it has its CSS property `float` set to `left`, `right` or `no float`.

Inline Boxes can be used in Print context and in Web pages. It is common to do entire web layouts using the float property. In Email templates, it is best to use Tables to position elements.

## Adding an Inline Box

To insert an inline box, use the  icon on the toolbar. Inline Boxes can be resized using the handles on the sides and corner. They can be styled using the **Format > Box** menu item, through the CTRL+M keyboard shortcut or through the CSS files; see "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.

## Positioning an Inline Box

Initially an Inline Box will float to the left. Use the  (Float left),  (No float) and  (Float right) icons on the toolbar to change the position of an Inline Box within the text.

- The **Float left** button aligns the Inline Box to the left. The text is positioned to the right of it and is wrapped around the box.
- The **Float right** button aligns the Inline Box to the right, with the text wrapped around it to the left.
- The **No float** button positions the Inline Box where it occurs in the text.

It is not possible to move an Inline Box using drag and drop. To move the Inline Box to another position in the text, you have to edit the HTML on the Source tab in the Workspace, moving the <div> element using cut and paste. To open the Source tab, click it (at the bottom of the Workspace) or select **View > Source View**.

## Span

The Span element (<span> in HTML code) is used to group inline elements, such as text in a paragraph. A Span doesn't provide any visual change by itself, but it provides a way to target its content in a script or in a style sheet.

To wrap content in a span, select the text and other inline elements and click **Insert > Wrap in Span** on the menu. Give the span an ID, if you are going to add a style rule or script for it that is

unique to this span; or give the span a class, if this span can be targeted by a style or script along with other pieces of content. Now you can use the wrapper's ID or class as a script's or style's `selector`; see "Using the Text Script Wizard" on page 214 and "Styling and formatting" on page 170.

## Div

The Div is the element used to create both Positioned Boxes and Inline Boxes. By default, a Div element reacts pretty much like a paragraph (<p>) or an inline box set to 'no float' except that it can be resized directly. Just like Positioned Boxes and Inline Boxes, Div elements can be styled using the **Format > Box** menu item, through the CTRL+M keyboard shortcut or through the CSS files; see "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.

### Adding a Div element

To add a Div, select **Insert > Structural Elements > Div** on the menu. For an explanation of the options, see "Inserting an element" on page 107.

### HTML tag: div, span

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file. It is possible to edit the source of the HTML file directly in the Designer; see "Editing HTML" on page 105.

In HTML, boxes are <div> elements. Spans are <span> elements. To learn how to change the attributes of elements, see "Attributes" on page 106.

## Business graphics

Business graphics display variable data, originating from one record, in a graphical way. Three types of business graphics are available: Pie Charts, Bar Charts and Line Charts.

### Adding a business graphic

To add a business graphic to the template:

1. Place the cursor where the graphic should be added.
2. Click the toolbar button of the type of chart you want to add, or select **Insert > Business graphic** and choose the chart type.

3. An **ID** is required. You can change the given ID and, optionally, add a class.
4. Use the **Location** drop-down to select where to insert the graphic:
  - **At cursor position** inserts it where the cursor is located in the template.
  - **Before element** inserts it before the HTML element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be before the <p> tag.\*
  - **After start tag** inserts it within the current HTML element, at the beginning, just after the start tag.\*
  - **Before end tag** inserts it within the current HTML element, at the end, just before the end tag.\*
  - **After element** inserts it after the element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be after the end tag of the paragraph (</p>).\*

\* If the current element is located inside another element, use the **Elements** drop-down to select which element is used for the insertion location. The list displays every element in the breadcrumbs, from the current selection point until the root of the body.

5. Use the **Input format** drop-down to select the source of the data for the Chart:
  - With **Static Labels** the chart has the same number of items for each record. Select the data fields (with a numerical value) and type a label for those fields.

All other options fill the chart dynamically, using data from a detail table. At least one detail table must be available in the Data Model for this option to be functional.

- **Dynamic labels** (Pie Charts only). The chart uses **one color per record** in the detail table. Select the data field under **Values**.
- **Columns are series** (Bar Charts and Line Charts only). The chart uses **one color per data field** selected under **Columns**. For each data field, a series of bars or lines in one colour is added to the chart.
- **Rows are series** (Bar Charts and Line Charts only). The chart uses **one color per record** in the detail table. For each data field selected under **Columns**, a series of bars or lines in different colours is added to the chart.

With the options to fill a chart dynamically you also have to select a **detail table** and a **(row) label**: a data field of which the value appears near the parts in a pie chart or under the bars or points of the line in a bar chart or line chart. The label is also used for the

legend. Note that initially the legend is not visible. To make it visible, check the option **Show legend** in the chart's properties (see "Business graphic properties" below).

6. Close the dialog and go in Preview mode in the Workspace to view the result.

## Changing a business graphic

### Chart script

The script related to a business graphic determines which values are used to generate the graphic. Double-click the script on the **Scripts** pane to change this.

#### Tip

To find the script that fills the chart, hover over the names of the script in the Scripts pane. That script will highlight the chart in the template and its selector is the same as the ID of the business graphic (preceded by #).

For an explanation of the options in the Script wizard, see **step 5** of "Adding a business graphic" on page 152.

### Business graphic properties

A chart is always added with the chart's default properties. To change those properties, such as the colors and the legend, open the Chart properties dialog: right-click the business graphic (on the **Design** tab in the Workspace) and select the respective business graphic on the shortcut menu.

For an overview of all properties, see:

- "Dialogs" on page 256
- "Line Chart Properties dialog" on page 271
- "Pie Chart Properties dialog" on page 279.

## Date

The Date element inserts the current system date, optionally making it dynamic so that it updates whenever the template is viewed or produces output.

## Adding a date

To add a Date element, use the **Insert > Date** option in the "Menus" on page 320. A dialog appears with the following controls:

- **Language:** Use the drop-down to select which language the date should be displayed in.
- **Update Automatically:** Check to update the date automatically when the template is viewed or produces output. When this option is checked, a placeholder is inserted in the template and a script is created to update it automatically, otherwise a static text with the date is inserted.
- **Available Formats:** Select the date/time format in which to display the date.

Click OK to insert the date or Cancel to close the dialog.

### Tip

If you are looking to add a date, originating from a record set, to a template, see: "Variable Data" on page 211. To insert a date you could use either the drag and drop method or the Text Script Wizard, however the latter lets you set the date/time format.

## Changing the date

Once inserted, a date can be modified directly in the template (if it does not update automatically) or through the **date** script (if it does update automatically). To modify the date in the script:

1. Double-click the **date** script in the **Scripts** pane.
2. Between the round brackets after Date, enter the desired date in the following order: year, month, day, and optionally hours, minutes, seconds, milliseconds (see [http://www.w3schools.com/js/js\\_dates.asp](http://www.w3schools.com/js/js_dates.asp) and [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date).) When the time is omitted, it defaults to 12:00:00 AM.

## Formatting an automatically updating date

The script added to automatically update the date uses the short date format. To change this:

1. Double-click the **date** script in the **Scripts** pane.
2. Delete the first line of the script.
3. On the second line, delete what comes after `format` and change `format` to `formatter` (see "formatter" on page 456).
4. Now type a dot after `formatter`, press **Ctrl + space** and choose one of the functions to format a date and time; see "Date, date/time and time functions" on page 445.

#### Note

The Locale, set in the **Edit > Locale** dialog, has an influence on the formatting of a date. The Locale can be the system's locale, a specific locale, or it can depend on the value of a data field; see "Locale" on page 201.

## Hyperlink and mailto link

Links can be added to any template but they only work in electronic output (web pages, email and PDF files). They can be a regular hyperlink pointing to a web page or a mailto link that will open the default email client when clicked.

### HTML element: a

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file. It is possible to edit the source of the HTML file directly in the Designer; see "Editing HTML" on page 105.

The HTML tag of a hyperlink or mailto-link is `<a>`. This is sometimes called an `anchor` tag. For a list of attributes, see [http://www.w3schools.com/tags/tag\\_a.asp](http://www.w3schools.com/tags/tag_a.asp).

### Adding a hyperlink or mailto link

1. Select text or an image.

#### Note

Although it is possible, it is not advisable to add a Hyperlink to other elements, such

as a Paragraph or Div. HTML 4 specifies that hyperlinks and mailto-links may only contain inline elements. Block elements, such as a Div, may not appear inside a link. HTML 5 states that the link "may be wrapped around entire paragraphs, lists, tables, and so forth, even entire sections, so long as there is no interactive content within (e.g. buttons or other links)"; see <https://www.w3.org/TR/html5/text-level-semantics.html>.

2. Click the **Insert hyperlink** button on the toolbar, or on the menu, select **Format > Hyperlink > Insert**.
3. Select URL to create a regular hyperlink pointing to a web page, or select Email to create a mailto-link that will open the default email client when clicked.
4. For a **URL**:
  - **URL**: enter a valid, well-formed URL to link to. It must start with the protocol, such as `http://` or `https://`.
  - **Target**: use the drop-down or type in the target for the link. When the target is `_blank` the link will open in a new browser window or tab.

For a **mailto link**:

- **Email**: enter a valid email address that appears by default in the To: field of the email client.
- **Subject**: type a default subject that appears in the Subject: field of the email client.
- **Message**: type a message that appears by default in the Message field of the email client.

Note that all these can be changed within the email client once the link is clicked.

### **Dynamically inserting or modifying a hyperlink**

You may wish to adjust a hyperlink depending on a value in a record that is merged to the template when generating output, for example, to provide a different mailto link for different customers.

How to add or modify a hyperlink is described in a how-to; see [How to dynamically insert a hyperlink](#). This implies writing a script. For information about scripts, see "Writing your own scripts" on page 223.

## Adding a personalized link

Personalized URLs (pURLs) are links that are tailor-made for a specific purpose, generally for individual clients. Typically, a pURL in a Connect template takes the user to a personalized landing page, for example, to download an invoice or get access to specific products or services. For more information, see [Personalized URL](#).

# Images

Images are a powerful ingredient in all of your templates. This topic explains how to add and use them.

## Ways to use images

In templates, both **imported** images and **external** images can be used (see ["Adding images"](#) on the next page and ["Resources"](#) on page 45). Once added to the content of a template, an image can be resized (see ["Resizing an image"](#) on page 161) and alternate text can be linked to it (see ["Setting an alternate text"](#) on page 162).

## Dynamic images

Images can be switched dynamically, so that a letter or email can include one image or another, depending on a value in the data set. Read ["Dynamic Images"](#) on page 222 to find out how to add such switching images.

## Background images

Several parts of templates, such as sections and media, and elements such as positioned boxes, can have a background image. Right-click the element and click the **Background** tab to select an image to be used as the element's background image. See ["Background color and/or image"](#) on page 186 and ["Using a PDF file as background image"](#) on page 65.

### Tip

Editing PDF files in the Designer is not possible, but when they're used as a section's background, you can add text and other elements, such as a barcode, to them. To create a new Print template from a PDF file, use the PDF-based Print template (see ["Creating a Print template with a Wizard"](#) on page 54). To use a PDF file as background image for an existing section, see ["Using a PDF file as background image"](#) on page 65.

## Filling optional whitespace

Conditional content and dynamic tables, when used in a Print section, may or may not leave an empty space at the bottom of the last page. To fill that space, if there is any, an image or advert can be used as a 'whitespace element'; see "Images" on the previous page.

### Tip

Using images in an Email template? See "Using images in email campaigns: tips" on page 88.

## HTML tag: img

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file. It is possible to edit the source of the HTML file directly in the Designer; see "Editing HTML" on page 105.

In the section's source file, images are `<img>` elements. The `<img>` tag has at least four attributes: `src`, `alt`, `width` and `height`. `src` specifies the URL of the image. `alt` contains the alternate text; see "Setting an alternate text" on page 162.

The value of the attributes can be changed via a script; see "Attributes" on page 106.

## Adding images

### Imported or external images

In templates, both **imported** images and **external** images can be used.

**Imported** images are images that are saved within the template file. To import images into a template and add them to the content, you can use the drag-and-drop method or the Select Image dialog (both are explained below).

**External** images are either located on a specific website (URL), or in a folder on a hard drive that is accessible from your computer. Note that external images need to be available at the time the template is merged with a record set to generate output, and that their location should be accessible from the machine on which the template's output is produced. External images are updated (retrieved) when the output is generated. External images can not be added via the drag-and-drop method. Use the Select Image dialog instead (see below).

For information about referring to images in HTML or in a script, see "Resources" on page 45.

## Via drag-and-drop

The drag-and-drop method is a quick way to import one or more images into a template.

1. Look up the image file or image files on your computer using the Windows Explorer.
2. Select the image (or images, using Shift+click or Ctrl+click) and drag the image file from the Explorer to the **Images** folder on the **Resources** pane at the top left.
3. To place an image in the content, drag it from the **Images** folder on the **Resources** pane to the content and drop it. The image will be inserted in the template at the position of the cursor.

## Via the Select Image dialog

To either import an image into a template or use an external image in a template, the Select Image dialog can be used:

1. Position the cursor in the content where you want the image to be inserted.
2. On the **Insert** menu, click **Image**. Or, click the **Insert Image** button on the toolbar. The Select Image dialog appears.
3. Click **Resources**, **Disk** or **Url**, depending on where the image is located.
  - **Resources** lists the images that are present in the **Images** folder on the **Resources** pane.
  - **Disk** lists image files that reside in a folder on a hard drive that is accessible from your computer. Click the **Browse** button to select a folder (or an image in a folder). As an alternative it is possible to enter the path manually. The complete syntax is: file://<host>/<path>. Note: if the host is "localhost", it can be omitted, resulting in file:///<path>, for example: file:///c:/resources/images/image.jpg.
  - **Url** lists image files from a specific web address. Select the protocol (**http** or **https**), and then enter a web address (for example, <http://www.mysite.com/images/image.jpg>).
4. With an external image, you can check the option **Save with template**. If this option is checked, the file will be inserted in the **Images** folder on the **Resources** pane at the top left.

If not saved with the template, the image will remain external. Note that external images need to be available when the template is merged with a record set to generate output, and that their location should be accessible from the machine on which the template's

output is produced. External images are updated (retrieved) at the time the output is generated.

5. Select the image from the list.
6. If the image is a PDF file that consists of more than one page, select the desired page.
7. Click **Finish**. The image will be inserted at the position of the cursor.

### Using one file that contains a collection of images

When a template that contains lots of images is merged with a large record set, the many file requests may slow down the process of output generation. The solution is simple: combine the images into a single image file and display the part that holds the image. This reduces the number of file requests and can improve the output speed significantly.

For an explanation of how to do this, see "Optimizing a template" on page 494.

### Styling an image

Images can be styled using the **Format > Image** menu item. They have a border, margin and padding; see "Border" on page 187 and "Spacing" on page 196. Images can also be left-aligned or right-aligned, just like boxes.

### Resizing an image

There are three ways to resize an image after inserting it in the content of a template.

- Click the image and drag the handles to resize it. Press the **Shift** key while dragging, to scale the image proportionally.
- Select the image (see "Selecting an element" on page 107) and type the desired width and height in the respective fields on the **Attributes** pane.
- Select the image and select **Format > Image**, on the menu. On the **Image** tab, change the **width** and **height** of the image.

### Positioning an image

Initially an image will float to the left. Select the image (see "Selecting an element" on page 107) and use the  (Float left),  (No float) and  (Float right) icons on the toolbar to change the position of an image within the text.

- The **Float left** button aligns the image to the left. The text is positioned to the right of it and is wrapped around the box.
- The **Float right** button aligns the image to the right, with the text wrapped around it to the left.
- The **No float** button positions the image where it occurs in the text.

To position an image using the menu, select the image and then select one of the options in **Format > Float**.

Alternatively, select the image and on the menu, select **Format > Image** and on the **Image** tab, set the **Float** property. This is equivalent to the `float` property in CSS.

### Rotating an image

To rotate an image, select the image (see "Selecting an element" on page 107) and select **Format > Image**, on the menu. On the **Image** tab, set the rotation angle of the image in clockwise degrees. This is equivalent to the `transform: rotate` property in CSS.

### Setting an alternate text

Once an image has been inserted in the content of a template, it can have an alternate text. The alternate text will be shown in emails at the position of the image while the image is loading and when the image is not found.

To set an alternative text, click the image and enter the alternate text in the **Alternate text** field on the **Attributes** pane at the top right.

### Using a CSS gradient to create an image

CSS gradients are a new type of image added in the CSS3 Image Module. CSS gradients let you display smooth transitions between two or more specified colors, while repeating gradients let you display patterns. This way, using images for these effects can be avoided, thereby reducing download time and bandwidth usage. In addition, objects with gradients look better when zoomed in a browser, and you can adjust your layout with much more flexibility.

For more information about the various types of CSS gradients and how to use them, see [https://developer.mozilla.org/docs/Web/CSS/CSS\\_Images/Using\\_CSS\\_gradients](https://developer.mozilla.org/docs/Web/CSS/CSS_Images/Using_CSS_gradients).

### Note

When CSS repeating gradients are displayed in a PDF reader, artifacts, like very thin lines may occur. When this happens, try setting the gradient's position a little bit different.

## Table

Tables serve two different purposes: they are a way to display data in a tabular format, and they are also a way to position elements on a page.

In HTML email, Tables are the most reliable way to position text and images; see "Designing an Email template" on page 86. In web pages, on the other hand, Inline Boxes are the preferred way to position elements. Tables should only be used to display data in a tabular format, not to position text and images. Tables used in web pages to position elements make those pages less accessible to users with disabilities and to viewers using smaller devices.

In print, Tables can be used for both purposes.

### HTML element: table

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file. It is possible to edit the source of the HTML file directly in the Designer; see "Editing HTML" on page 105.

The HTML tag of a Table is `<table>`. Tables are divided into table rows with the `<tr>` tag. Table rows are divided into table data with the `<td>` tag. A table row can also be divided into table headings with the `<th>` tag.

The tags `<thead>`, `<tbody>` and `<tfoot>` can be used to group the header, body, or footer content in a table, respectively.

For information about HTML tables and a list of attributes, see [http://www.w3schools.com/html/html\\_tables.asp](http://www.w3schools.com/html/html_tables.asp).

### Inserting a Table

1. On the toolbar, click the **Insert table** button, or on the menu select **Insert > Table > Standard**.

2. Enter the table's desired attributes:

- **ID**: a unique identifier for the table. IDs are used to access the Table from scripts and as CSS selectors for style rules.
- **Class**: A class identifier for the table. Classes can be shared between elements and are used to access the table from scripts and as CSS selectors for style rules.
- The number of **rows** for the header, body and footer of the table.
- The number of **columns**
- The **width** of the table.

3. Check the option **Absolute** to give the Table an absolute position or use the **Location** drop-down to select where to insert the table:

- **At cursor position** inserts it where the cursor is located in the template.
- **Before element** inserts it before the HTML element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be before the <p> tag.\*
- **After start tag** inserts it within the current HTML element, at the beginning, just after the start tag.\*
- **Before end tag** inserts it within the current HTML element, at the end, just before the end tag.\*
- **After element** inserts it after the element in which the cursor is currently located. For example if the cursor is within a paragraph, the insertion point will be after the end tag of the paragraph (</p>).\*

\* If the current element is located inside another element, use the **Elements** drop-down to select which element is used for the insertion location. The list displays every element in the breadcrumbs, from the current selection point until the root of the body.

#### Note

Tables on a Master Page have to have an absolute position, unless they are located inside another element with an absolute position.

4. Click **Next** and select which fields should show up in the Dynamic Table.

The order of the fields indicates in which order columns are displayed in the dynamic table, from left to right. Select a line and then use the **Up** and **Down** buttons to change the order of the columns.

You could change the placeholder for each data field as well; just click a placeholder to edit it.

5. Click **Next** and use the drop-down to select the desired table style.
6. Uncheck the box **Resizable columns** if the columns should not be resizable from the Design and Preview modes in the workspace. This is useful if the column size is determined in the Source mode or in a style sheet.
7. Click **Finish** to add the table to the section.

## Header and footer

### Adding a header or footer

To add a header or footer to an existing table, right-click the table and then select **Table > Insert thead** or **Insert tfoot**, on the shortcut menu.

Alternatively, click in one of the cells and select **Insert > Table > Insert thead** or **Insert tfoot**, on the menu.

### Deleting a header or footer

To delete a header or footer, simply right-click the header or footer and select **Row > Delete** on the shortcut menu.

## Rows and columns

### Adding a row or column

To add a row or column to an existing table, click in a cell. Then click the black triangle next to the **Insert Row Above** button on the toolbar, and click one of the **Insert** buttons, or select one of the options in the **Insert > Table Elements** menu.

Alternatively, right-click the table and on the shortcut menu, select **Row > Insert Above** or **Insert Below**, or select **Column > Insert Before** or **Insert After**.

## Deleting a row or column

To delete a row or column, simply right-click the row or column and select **Row > Delete** or **Column > Delete** on the shortcut menu.

## Styling a Table

Tables can be styled using the **Format > Table** menu item, while individual selected cells can be styled using the **Format > Table Cell** menu item.

## Resizing and moving a Table

Before you can resize or move a Table:

- Make sure that the position of the Table is absolute. If it's not, right-click the Table and on the shortcut menu, select **Convert to absolute**. (This option isn't available for Tables on a Master Page, as they must always have an absolute position, or be located inside another element with an absolute position.)
- Select the Table (see "Selecting an element" on page 107) and then, on the **Attributes** pane, check the option **Allow resizing**.

## Resizing a Table

- Click in the table and drag the handles to resize it. Press the **Shift** key while dragging, to scale the table proportionally.
- Select the Table (see "Selecting an element" on page 107) and type the desired width and height in the respective fields on the **Attributes** pane.
- Select the Table and select **Format > Table**, on the menu. On the **Table** tab, change the **width** and **height** of the Table.

## Moving a Table

- Click in the table and then drag the border to move the Table.
- Select the Table (see "Selecting an element" on page 107) and type the desired Y-offset and X-offset in the respective fields on the **Attributes** pane.
- Select the Table and select **Format > Table**, on the menu. On the **Table** tab, change the **Y-offset** and **X-offset** of the image.

## Hiding the border

When using a Table to position other elements, you will want to hide the borders of the table. To do this, set the width of the border to 0; see "Border" on page 187.

## Text and special characters

The vast majority of templates for personalized customer communications contain, of course, text. While the most common text element is a <p> or paragraph, other elements such as Headings (<h1> through <h6>) are also considered text elements. Text elements can be present within other types of elements such as table cells (<td>), boxes (<div>), etc.

### Adding text

To add text, simply type in the workspace in the middle.

- Press **Enter** to insert a new paragraph.
- Press **Shift+Enter** to insert a line break.

Alternatively, copy-paste text into a template, or use the **Insert Lorem Ipsum** toolbar button to insert dummy text.

Text that precedes or follows the value of a **data field** can be added by the Text Script Wizard; see "Using the Text Script Wizard" on page 214.

Note: it is not possible to open a Word file in the Designer. When you copy text from a Word document, however, basic style characteristics travel with the content to PrintShop Mail ConnectDesigner. Formatting options like bold, italic and formats like Heading 1, Heading 2 are maintained.

### Extra spaces

In HTML, extra spaces are generally removed. In Designer templates this is the same, because they are HTML files. In some cases however, you want extra spaces to be shown in your output. Read this how-to to learn how to maintain extra spaces in the text: [Maintain extra spaces in text](#).

### Adding special characters

To add special characters:

1. Position the cursor where the character should be inserted.
2. On the **Insert** menu, point to **Special Characters** click **Symbols, Dashes and Spaces**, **Arrows**, or **Geometric Shapes**, and click one of the available special characters.

## Adding page numbers

Page numbers can only be used in the Print context. See "Page numbers" on page 72.

## HTML element: p, h, li and others

When adding elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file; see "Editing HTML" on page 105.

In HTML text can be contained in many different elements: paragraphs, span elements, line items and table cells, for example.

The HTML tag of a paragraph is `<p>`. The paragraph should be followed by a closing tag: `</p>`.

A line break looks like this in HTML: `<br>`.

## Formatting text

Text can be styled, colored, centered, indented etc. It can even be displayed so that it reads from right to left. See "Styling text and paragraphs" on page 182.

In all templates you can use the fonts that are provided with the Designer, as well as imported fonts; see "Fonts" on page 194.

# Snippets

A snippet is a small, ready-to-use piece of content in a file. Snippets can be re-used within the same template, in all contexts and sections. They can contain any contents that a section can have, such as text, images, variable data, dynamic tables, etc.

When a snippet is added to different sections or contexts, it is displayed according to the section's or context's stylesheet. This means that the same content can appear differently depending on the styles applied to the section or context, but it still has the exact same contents.

## Adding a snippet

Before adding a snippet:

- Import the resource files that are related to the snippet, such as image files and CSS files, into the template file. Drag and drop the files to the corresponding folders (**Images** and **Stylesheets**, respectively) on the **Resources** pane. If you want to use external images, see "Images" on page 158.
- Drag the snippet itself to the **Snippets** folder on the **Resources** pane, or create a new snippet from an existing piece of content in the template (see "Creating a snippet" below).

To add the snippet to the content of a section, drag the snippet from the **Snippets** folder on the **Resources** pane to the desired location in a section.

Check the option **Insert as shared content** to insert a reference to the original snippet in the template, rather than a copy of the original snippet. When a snippet is being used as shared content, the contents of the snippet itself are not added to the page. Modifying the snippet on the page actually modifies the snippet's source. If a snippet is used in multiple locations (such as different contexts and sections), modifying one instance will modify all of them at once.

It is also possible, and often useful, to insert a snippet or part of it, using a script. See "Loading a snippet via a script" on page 238.

### Tip

To export a snippet from your template, drag or copy/paste it out of the Snippets folder to a folder on the local hard drive.

## Creating a snippet

To turn a parts of a letter or email into a snippet for reuse in the content of a template:

1. Select the part that should be saved in a snippet.
2. Right-click the selection, point to **Snippet** and click **Create**.
3. Right-click the new snippet on the **Resources** pane in the **Snippets** folder and rename it.

## JSON Snippets

JSON Snippets are snippets that contain pieces of JSON data instead of HTML. Just like HTML snippets, JSON snippets are stored in the **Snippets** folder on the **Resources** pane, but their file names should end in '.json'.

JSON Snippets cannot be inserted into the content directly, but they can be accessed via a script using the function **loadjson()**:

```
var json_data = loadjson("snippets/snippet.json");
results.html(json_data.field1);
```

See also: "Writing your own scripts" on page 223.

## Styling and formatting

In the Designer you have everything at hand to make your templates look good: colors, fonts and all the tools to position, align and embellish elements in your designs. This topic informs about the ways to style a template.

### Local formatting versus style sheets

There are in general two ways to style elements:

- Using **local formatting**. Local formatting means styling an element directly, using a toolbar button or one of the formatting dialogs.
- Using **Cascading Style Sheets (CSS)**. Style sheets can determine the appearance of individual elements, as well as the appearance of elements that have the same class or HTML tag.

Whether applied through style sheets or through local formatting, behind the scenes all layout properties in the Designer are CSS properties. When you format an element locally, an **inline** style rule is added to the element.

Note that where local formatting conflicts with a formatting rule for the same element in one of the style sheets, the local formatting rule gets priority; the rule in the style sheet will be ignored.

It is highly recommended to use style sheets in templates right from the start. Even more so if the communications are going to be output to different output channels, or if they consist of different sections (for example, a covering letter followed by a policy). Using CSS with templates allows a consistent look and feel to be applied. A style sheet can change the look of multiple elements, making it unnecessary to format each and every element in the template, time and again, when the company's layout preferences change. See "Styling templates with CSS files" below.

## Layout properties

Colors and fonts make an important contribution to the look and feel of your template. See "Colors" on page 190 and "Fonts" on page 194.

Text and paragraphs have a number of formatting options that are not available for other elements: font styles and line height, for example. See "Styling text and paragraphs" on page 182.

Boxes and a number of other elements can have a background color and/or background image; see "Background color and/or image" on page 186.

Several elements, such as boxes, images, paragraphs, and tables, can have a border; see "Border" on page 187.

Boxes, images, tables, text and other elements can be rotated; see "Rotating elements" on page 200.

Spacing (padding and margin) helps to position elements relative to other elements in the template; see "Spacing" on page 196.

The best way to position elements depends on the output channel for which the template is intended; see "How to position elements" on page 179.

The locale setting influences how dates, numbers and amounts of money are displayed; see "Locale" on page 201.

## Styling templates with CSS files

The Layout toolbar and the Format menu offer many possibilities to style every piece of a template. However, styling every single element, one after another, is a lot of work and, more

importantly, can result in a template with a messy mix of styles that isn't easy to maintain and lacks consistent design. Therefore the preferred way to style templates is with CSS: Cascading Style Sheets.

The basic idea behind CSS is to separate the structure and contents of a (HTML) document as much as possible from the presentation of that document.

Cascading Style Sheets were originally designed for use with web pages, or HTML files. Since every template in the Designer is constructed in HTML, CSS files can also be used in the Designer.

Instead of setting the font size, line height, color etc. for each and every paragraph in the template itself, you can define a layout for all paragraphs, and for all output channels, in a CSS file.

The benefit of this is that you can quickly and easily change the look and feel of all contexts in one template, without having to change the contents. In the event that your company chooses to use another font or to adjust its corporate colors, you only have to change the style sheets.

## You are writing HTML

When you add elements, such as text, images or a table, to the content of a template, you are actually constructing an HTML file.

To see this, toggle to the **Design** tab in the workspace. Click anywhere in the content. Take a look at the *breadcrumbs* at the top of the workspace. The breadcrumbs show the HTML tag of the clicked element, as well as the HTML tags of other elements to which the clicked element belongs. The clicked element is at the end of the line.

To edit the HTML text directly:

- In the workspace, toggle to the **Source** tab.

On this tab you can view and edit the content of the template in the form of plain text with HTML tags (note the angle brackets: <>). You may add and edit the text and the HTML tags, classes, ID's and other attributes.

To learn more about HTML, see for example <https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Introduction> and <http://www.w3schools.com/html/default.asp>.

Many video courses and hands-on courses about HTML (and CSS) are offered on the Internet as well, some for free. Go, for example, to [www.codeschool.com](http://www.codeschool.com) or [www.codecademy.com](http://www.codecademy.com) and look for HTML (and CSS) courses.

## Included Cascading Style Sheets

When you create a template, a number of style sheets is automatically included:

- One style sheet that applies to all document types: `context_all_styles.css`.
- One or more style sheets specific to the context (Print, Email). For example, when you create an action email using the Wizard, the files `context_htmlmail_styles.ccs` and `basic_email_action.css` are automatically added to the **Stylesheets** folder on the **Resources** pane.
- A style sheet that defines default styles for tables: `default.css`. It contains the styles that you can choose from when you insert a table via the **Insert** menu or the **Insert table** toolbar button.

### Note

Do not change the `default.css` style sheet. Use the global style sheet or the style sheet for the relevant context to define your own styles for tables.

## Adding CSS files

To add a CSS file of your own, open an Explorer window, **drag** the file to the **Resources** pane and drop it on the **Stylesheets** folder.

To create a new CSS file, right-click the **Stylesheet** folder on the **Resources** pane and select **New Stylesheet**.

### Tip

To export a CSS file from your template, drag or copy/paste it out of the Stylesheets folder to a folder on the local hard drive.

## Note

The order in which style sheets are executed, can affect the actual output. This sequence can be set per section; see "Determining the order in which style sheets are read" on page 179.

## Using a remote style sheet

A remote style sheet is not located within your template but is rather hosted on an external web server (generally called a **CDN**). When generating Web output, these files are referenced in the web page's header and are served by the remote server, not by the PrintShop Mail Connect Server module.

To add a remote style sheet:

1. Right-click the **Stylesheet** folder on the **Resources** pane, and click **New Remote Stylesheet**.
2. Enter a name for the file as it appears in the Stylesheet resources. For better management, it's best to use the same filename as the remote resource.
3. Enter the **URL** for the remote resource. This must be a full URL, including the http:// or https:// prefix, domain name, path and filename.
4. Optionally, for a Capture OnTheGo Form, you can check **Use cached Capture OnTheGo resource**, to prevent downloading a remote style sheet again if it has been downloaded before. The file should be available on a publicly accessible location, for example: a folder location on a corporate website, hosted by a CDN (Content Delivery Network) or shared via a Workflow process.

There are a few advantages to remote resources:

- These resources are not served by your server, saving on space, bandwidth and processing.
- Using a popular CDN takes advantage of caching - a client having visited another website using that same CDN will have the file in cache and not re-download it making for faster load times for the client.

## Styling your templates with CSS files

### Note

Email clients do not read CSS files and some even remove a `<style>` tag when it is present in the email's header. Nevertheless, CSS files can be used with the Email context in the Designer. When generating output from the Email context, the Designer converts all CSS rules that apply to the content of the email to inline style tags, as if local formatting was applied.

## Step 1: edit CSS

### Editing CSS using a property sheet

1. Select **Edit > Stylesheets**.
2. Click the downward pointing arrow next to **Global** and select the context that you want to edit styles for, or select the Global CSS file to edit CSS rules that apply to all contexts.
3. Click **New**, or click one of the selectors that are already listed and click **Edit**.
4. Type a CSS selector. This can be:
  - A class: `.class`. Class rules apply to all HTML elements with that class. When you create a class, choose a name that indicates what the class is used for, e.g. 'small' for a class that gives elements the font size 'small'. The class name has to be preceded by a dot, e.g. `.small`.
  - An ID: `#id`. An ID is always preceded by `#`, e.g. `#sender`. When you create an ID, choose a name that indicates what the ID is used for, e.g. `#sender` would refer to the HTML element with information about the sender.

### Note

Each ID should be unique and can only be used once in each section.

- An HTML element: `p`, `h1`, `table`, etc. Type the tag name without the angle brackets.
- A combination of HTML elements, separated by a comma. The CSS rule will apply to all HTML elements that are listed in the selector. For instance, a CSS rule with the selector "`h1, p`" applies to first level headings as well as paragraphs.

- HTML elements inside other HTML elements. For instance, a rule for all paragraphs inside a div element has the selector: div p.
  - Etcetera. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for more CSS selectors and combinations of CSS selectors.
5. Select the layout options that should apply to selected elements; see "Styling and formatting" on page 170. Note: where a width can be set as a percentage, it is a percentage of the space between the margins.
  6. Click **OK**.
  7. In the Stylesheets dialog, click the selector that you chose. All CSS rules for that selector will become visible in a box below the list of selectors.

## Edit plain CSS

- Click the button **Advanced** in any property sheet to open a CSS property editor. Type CSS properties at the left and values at the right.
- In the **Resources** pane at the left, double-click the global stylesheet or the stylesheet for the relevant context. The file opens in the workspace in the middle.

A list of all CSS properties and their possible values can be found here:  
<http://www.w3schools.com/cssref/>.

## Step 2: apply CSS to the content

After editing the CSS file(s), make sure that the CSS rules actually apply to one or more elements in the template.

CSS rules for HTML elements, such as paragraphs, are automatically applied to all elements with the corresponding HTML tag.

To make a CSS rule for a certain class or ID work for an element in your document, you have to add the class or ID to that HTML element.

### Note

Classes may be reused throughout one section, but a specific ID should not be used more than once in each section. CSS layout rules for an element with a certain ID only apply to the first element with that ID in each section. If you have two sections inside of a Print context, then you can have

the same ID on two sections; they will both be affected by the CSS rules for the element with that ID.

## Adding a class or ID to an HTML element

1. Select the element (see "Selecting an element" on page 107).
2. On the **Attributes** pane, type the **ID** and/or **class**. Type the ID **without** the preceding # and class names **without** a dot.

### Note

Note: Elements can have multiple classes. Separate the class names with a space (eg. "red small").

Alternatively, after selecting an element, you can click the **Source** tab at the bottom of the workspace. The selected element will be highlighted in the source. Add the class or classes and/or the ID to the opening tag of the HTML element, for example: `<p class="intro">`.

## How to determine which styles are applied

To see which styles are applied to an element, select the element (see "Selecting an element" on page 107) and take a look at the Styles pane that sits next to the Attributes pane.

The Styles pane shows which CSS style rules apply to the currently selected element. A link next to a style rule will open the file where that particular style is defined. This can be either a CSS file or the source file of a section if local formatting was used (see "Styling and formatting" on page 170).

A crossed-out style rule signals that it was overruled by another style rule. This happens when:

- A more specific, and therefore more important rule, is encountered for the same element. See "Using a more specific CSS rule" on the facing page to learn more about the specificity of style rules.
- A rule with the same importance is read after the first rule. Not only is the order of the rules in a CSS file important, but also the order in which the style sheets are read. The style

sheets that are included with a section are read in the specified order; see "Determining the order in which style sheets are read" on the next page.

## Using a more specific CSS rule

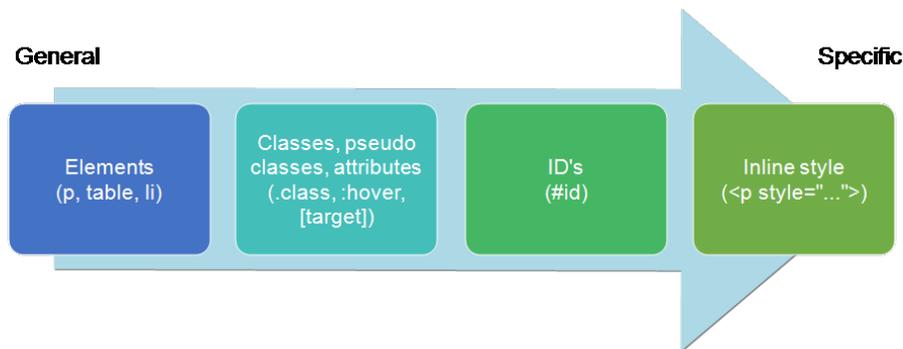
By default, many CSS properties of an HTML element also apply to the elements inside that element. For example, a CSS rule that specifies a certain font-type for a box is also applied to paragraphs in that box. In this example the box is the 'parent' element and the paragraphs are the 'child' elements that inherit the font-type property of the box.

### Note

Although the background color property seems to be inherited, it isn't. Most elements are transparent; therefore the background color of the parent element shines through.

To replace inherited style properties, you need to add a more specific CSS rule for that (type of) element. In case of a conflict between a general rule and a more specific rule, the more specific rule will be applied.

The following diagram shows the order of specificity.



Rules for HTML elements (p, table, li etc.) are general rules. Rules for classes, pseudo classes, and elements with a certain attribute (.class, :hover, [target]) are more specific. Rules for elements with a certain ID are even more specific. The most specific are inline styles.

### Example

A more specific rule for cells in a table that has the CSS property "color: red" (which colors text in the cells red) could be, for example:

- A rule for the text color of all table cells (td elements), for example:`td { color: green; }`.
- A rule for the text color of table cells with a certain class, for example `.green { color: green; }`
- A rule for the text color of a table cell with a certain ID, for example:`#greentext { color: green; }`
- An inline style rule (local formatting) added to the HTML tag of a particular table cell, for example:`<td style="color: green;">...</td>`

Each of these rules is more specific than the previous rules. All of these rules are more specific than the rule that applies to the table as a whole.

### Determining the order in which style sheets are read

For each section, the style sheets are applied in a certain order. The styles in each following style sheet add up to the styles found in previously read style sheets. When style sheets have a conflicting rule for the same element, class or ID, the last style sheet 'wins' and overrides the rule found in the previous style sheet.

The order in which style sheets are applied, can be changed per section:

1. On the **Resources** pane, expand the **Contexts** folder, expand the folder of the corresponding context and then right-click the template.
2. Click the tab **Includes**.
3. Click a CSS file and use the **Up** and **Down** buttons to change the order in which the style sheets are read.
4. Note: Moving a style sheet up in the list gives it **less** weight, because style sheets read later will override previous ones in case of conflicting rules.

## How to position elements

To position elements in relation to each other in a template, wrap those elements in a Table or Box (see "Table" on page 163 and "Boxes" on page 150) and/or use the Spacing property of the elements. The Spacing property can also be used to indent elements or create a hanging paragraph or image; see "Spacing" on page 196. Guides help to align elements as well; see below.

## Where to use Tables and Boxes

Tables, Positioned Boxes and Inline Boxes can help position elements in relation to other elements. It depends on the context which element is best to use.

In the Email context, Tables are the most reliable way to position text and images; see "Designing an Email template" on page 86 and "Table" on page 163.

In the Web context, Inline Boxes are the preferred way to position elements; see "Boxes" on page 150. Tables should only be used to display data in a tabular format, not to position text and images. Tables used in web pages to position elements (and often, Positioned Boxes) make those pages less accessible to users with disabilities and to viewers using smaller devices.

In the Print context, Tables can be used to position elements, as well as both types of Boxes; see "Table" on page 163 and "Boxes" on page 150.

## Spacing

Boxes, tables, paragraphs and many other elements have a **margin** and **padding**.

The margin is the white space around an element, outside the border. It is used to position an element in relation to the other elements, by putting more space between the element and its surrounding elements.

The padding is the space between an element's content and its border. It is used to position the content of the element inside the border.

To learn how to set an element's spacing properties, see "Spacing" on page 196.

### Tip

Use a negative left margin to create a hanging paragraph or image.

## Guides

Guides are horizontal and vertical lines used to help in designing templates. They can only be used in Print sections. Positioned Boxes (and any other objects that have their **position** set to **absolute**) will snap to guides when moved within a few pixels of them.

To add a guide, press the **Insert Horizontal Guide** or **Insert Vertical Guide** buttons on the Toolbar.

To move a guide, click and drag it to a new location. Double-clicking the guide brings up its Edit dialog where its exact position can be adjusted.

To delete a guide, double-click on it and press the **Delete** button.

## Using the CSS `position` property

An element can be positioned independently of the text flow by changing its `position` property to `absolute`. When the element is placed inside another element, such as a Box, changing its `position` property to `absolute` positions the element absolutely inside its 'parent'.

With the `position` property of an element set to `absolute`, the `top` or `bottom` and `left` or `right` properties position the element inside its parent with exact values: pixels (px), centimeters (cm), etc. Negative values are allowed.

For an explanation of all values that the `position` property can possibly have, see [http://www.w3schools.com/css/css\\_positioning.asp](http://www.w3schools.com/css/css_positioning.asp).

## Where to use it

In Print sections, setting the `position` property to `absolute` can be very useful in Print sections. It takes the element out of the text flow, so that the element stays where it is on the page. On Master Pages (which are only used in Print sections) elements are always positioned absolutely; if not, they must be located inside an element that has an absolute position.

In Web sections, this property can be useful for elements inside a Div element. A Div element that is not inside another element should not be positioned absolutely: designs for the Web should be flexible so that they display nicely on a variety of devices and screen sizes.

In Email sections, do not use this property. Use Tables instead (see "Designing an Email template" on page 86 and "Table" on page 163).

## How to use it

In the Formatting dialog the `position` property can be found on the **Image** tab, under **Positioning**.

This property isn't present in one of the tab menus of the style rule editor, but you can add it

after clicking the Advanced button in the style rule editor (see "How to position elements" on page 179).

## Styling text and paragraphs

There are numerous ways to format text in a template. You can apply a certain font, make text bold, transform it to uppercase, center it, color it, etc.

This topic explains how to apply local formatting to text. It is recommended though, to format text using style sheets; see "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.

### Formatting text and paragraphs locally

An intuitive way of formatting text locally is by using the toolbar buttons: select some text, or an element that contains text (see: "Selecting an element" on page 107) and click one of the toolbar buttons to make it bold, center it, create a numbered or bulleted list, etc.

To quickly change a paragraph into a Heading, Address or Pre element, select the paragraph (see: "Selecting an element" on page 107) and on the **Format** menu, select the appropriate element.

More local formatting options are available in the Formatting dialogs; see below.

### Formatting text

To open the Text Formatting dialog, select some text and then select **Format > Text**. In the Text Formatting dialog you can set:

- The font, font size, color and background color:
  - **Font:** see also: "Fonts" on page 194. This is equivalent to setting the `font-family` property in CSS.
  - **Font size.** Enter the size in a measure, named size or percentage. This is equivalent to setting the `font-size` property in CSS.
  - **Color:** this the color of the text. Select a named font color as defined in the Edit Colors dialog (see "Colors" on page 190) or click the colored square to create a new color or to enter a color manually. The color value must be a valid HTML color name or hexadecimal color code. This setting is equivalent to the `color` property in

CSS.

- **Background color:** this is the background color of the text. Select a named font color as defined in the Edit Colors dialog (see "Colors" on page 190) or click the colored square to create a new color or to enter a color value manually. a valid HTML color name or hexadecimal color code. This setting is equivalent to the `background-color` property in CSS.
- The spacing between letters and words and the way the text is wrapped:
  - **Letter Spacing:** The space between characters in a text in measure or percentage. This is equivalent to the letter-spacing property in CSS.
  - **Word Spacing:** Set the space between each word in a text in measure or percentage. This is equivalent to the `word-spacing` property in CSS.
  - **Whitespace:** Specify how the text wraps. See [CSS White-Space](#) for details. This is equivalent to the `white-space` property in CSS.
- The style of the text:
  - **Bold:** Sets the `font-weight` to 700.
  - **Italic:** Sets the `font-style` to `italic`.
  - **Underline:** Sets the `text-decoration` to `underline`.
  - **Strikethrough:** Sets the `text-decoration` to `line-through`.
  - **Subscript:** Sets the `vertical-align` to `super`.
  - **Superscript:** Sets the `vertical-align` to `sub`.
  - **Capitalize:** Sets the `text-transform` to `capitalize`.
  - **Uppercase:** Sets the `text-transform` to `uppercase`.
  - **Lowercase:** Sets the `text-transform` to `lowercase`.
  - **Small-caps:** Sets the `font-variant` to `small-caps`.

#### Note

All settings in the Text Formatting dialog are in fact CSS style rules. When you change one or more settings, the selected text gets wrapped in a Span element that has an **inline** style tag containing the selected setting(s). Click the **Advanced** button to add CSS properties and values to the inline style tag of the Span directly. For more information about CSS, see "Styling and formatting" on page 170.

## Formatting a paragraph

Through the Paragraph Formatting dialog you can set the line height and first indent, among other things. It also lets you add spacing and a border; see "Spacing" on page 196 and "Border" on page 187.

To open the Paragraph Formatting dialog, select a paragraph (see: "Selecting an element" on page 107) or place the cursor in a paragraph, and then select **Format > Text**.

On the **Formats** tab you can set:

- **Line-height:** Specify the height of each line in the paragraph's text, in a measure or percentage. Note that this is not the spacing between lines, but rather the complete height of the line itself including the text. This is equivalent to the `line-height` property in CSS.
- **Align:** Select how text should be aligned, such as left, center, right or justify. Equivalent to the `align` property in CSS.
- **First Indent:** Specify the indentation of the first line of the paragraph. Equivalent to the `text-indent` property in CSS.
- **Display:** Select how to display the element. This can also be used to hide an element completely using the none option. See CSS Display. Equivalent to the `display` property in CSS.
- **Direction:** Select in which direction text should be displayed (left to right, right to left, or auto). Useful for certain languages such as Arabic, Hebrew, etc. This is equivalent to the `dir` HTML attribute.
- **(Page) breaks:** these settings are only useful in Print sections, as only Print sections have pages.
  - **Before:** Sets whether a page break should occur **before** the paragraph. This is equivalent to the `page-break-before` property in CSS; see [CSS page-break-before property](#) for an explanation of the available options.
  - **Inside:** Sets whether a page break is allowed inside the paragraph. Equivalent to the `page-break-inside` property in CSS; see [CSS page-break-inside property](#) for an explanation of the available options.
  - **After:** Sets whether a page break should occur **after** the paragraph. Equivalent to the `page-break-after` property in CSS; see [CSS page-break-after property](#) for an explanation of the available options.

- **Widows and orphans:** Keeps lines of text together; see "Preventing widows and orphans" on page 73 for an explanation.

#### Note

For more information on page breaks, widows and orphans, see the [W3 Paged Media reference](#).

Click the **Advanced** button to add CSS properties and values to the inline style tag directly.

### Remove local formatting from text

Layout buttons and options on the Format menu add **inline** style tags to the text. Style tags can look like this: `<b>...</b>` or like this: `<p style= "color: red;" >`.

Inline style tags have priority over styles defined in a CSS file. For example, when a formatting rule in a style sheet colors all paragraphs green, a paragraph with an inline style tag to color it red would still stay red. So, when a rule in a style sheet doesn't seem to work, an inline style tag can be the culprit. In that case you might want to remove the local formatting.

To remove local formatting:

- Select the formatted text and click the toolbar button **Remove Formatting**. Doing this removes inline style tags from the selection.
- Alternatively, click the **Source** tab at the bottom of the workspace (or select **View > Source View**) to manually remove style tags.

#### Tip

When you select an element in the template, the **Styles** pane will show which styles are applied to that element. The link behind the style will take you to the place (the Source tab, or a CSS file) where that style is defined.

# Background color and/or image

In any type of template, boxes, tables and table cells can have a background color and/or a background image.

To select a background image or color:

1. Right-click the box and click **Box** on the shortcut menu.
2. Alternatively, select the box (see "Selecting an element" on page 107; note that a Box is a <div> element) and on the **Format** menu click **Box**.
3. Click the **Background** tab.

To select a **background color**: click the downward pointing arrow next to **Color** to select a color from the list of predefined colors (see "Defining colors, spot colors and tints" on page 190), or click the colored rectangle to open the Color Picker dialog; see "Color Picker" on page 262. In this dialog you can select a color from the color wheel or using the eye dropper tool, set RGB or CMYK color values or enter a hexadecimal color code.

To select a **background image**:

1. Click the **Select Image** button.
2. Click **Resources**, **Disk** or **Url**, depending on where the image is located.
  - **Resources** lists the images that are present in the **Images** folder on the **Resources** pane.
  - **Disk** lists image files that reside in a folder on a hard drive that is accessible from your computer. Click the **Browse** button to select a folder (or an image in a folder). As an alternative it is possible to enter the path manually. The complete syntax is: file://<host>/<path>. Note: if the host is "localhost", it can be omitted, resulting in file:///<path>, for example: file:///c:/resources/images/image.jpg.
  - **Url** lists image files from a specific web address. Select the protocol (**http** or **https**), and then enter a web address (for example, <http://www.mysite.com/images/image.jpg>).
3. With an external image, you can check the option **Save with template**. If this option is checked, the file will be inserted in the **Images** folder on the **Resources** pane. If not saved with the template, the image will remain external. Note that external images need to be available when the template is merged with a record set to generate output,

and that their location should be accessible from the machine on which the template's output is produced. External images are updated (retrieved) at the time the output is generated.

4. Select an image from the list.
5. If the image is contained in a PDF file that consists of more than one page, select the desired page.
6. Click **OK**.
7. Set the size of the image. The options are explained here:  
[http://www.w3schools.com/cssref/css3\\_pr\\_background-size.asp](http://www.w3schools.com/cssref/css3_pr_background-size.asp).
8. Set the position of the image in the box.
9. Finally, click **OK**.

#### Note

It is also possible to set an element's background in a style sheet; see "Styling templates with CSS files" on page 171. When referring to images or fonts from a CSS file, refer to a path that is relative to the current path, which is `css/`. For example: **`#header { background-image: url('../images/image.jpg'); }`**

## Border

In any type of template, boxes, tables and table cells, paragraphs and other elements can have a border.

Elements have a rectangular shape, so their border has four sides. Each side of the border can have a different layout.

### Adding a border

1. Right-click the element and click the respective element on the shortcut menu. Alternatively, select the element (see "Selecting an element" on page 107) and on the **Format** menu click the respective element.
2. Click the **Border** tab.
3. Uncheck the option **Same for all sides** to be able to style each side of the border separately.

4. Specify the width of the border (side). This is equivalent to the `border-width` property in CSS.
5. Specify the style of the border (side), such as solid, dashed or dotted. This is equivalent to the `border-style` property in CSS.
6. Specify the color of the border (side): click the downward pointing arrow next to **Color** to select a color from the list of predefined colors (see "Defining colors, spot colors and tints" on page 190), or click the colored rectangle to open the Color Picker dialog. In this dialog you can select a color from the color wheel, set RGB or CMYK color values or enter a hexadecimal color code. This setting is equivalent to the `border-color` property in CSS.

#### Note

It is also possible to set an element's border in a style sheet; see "Styling templates with CSS files" on page 171.

## Rounding corners

Any element in a template can have rounded corners. For boxes and images, this option is available in the Formatting dialog. For other elements, you have to create a CSS rule to set the `border-radius` of the element (or class of elements).

## Boxes, images and tables

To round the corners of a box, image or table:

1. Select a Box, Image or Table element (see "Selecting an element" on page 107) and on the **Format** menu click the respective element. Alternatively, right-click the element and click the respective element on the shortcut menu.
2. On the first tab in the Formatting dialog (the **Box**, **Image** or **Table** tab respectively) specify the **corner radius** in a measure (10mm, 5px, 0.5in) or percentage (0 - 90%).
3. For a Box or Image, click **Apply** to see the effect without closing the dialog or **OK** to close the dialog.

For a Table, you have to take yet another step. Tables can't have rounded corners and collapsed borders at the same time. All built-in table styles in the Designer have collapsed

borders. For the rounded corners to show, you must create a CSS rule that sets the table's `border-collapse` property to `separate` instead of `collapse`.

1. Click the **Advanced** button at the bottom of the Formatting dialog.
2. Under **Property**, type **border-collapse**.
3. Under **Value**, type **separate**.
4. Add a padding to keep the table cells from sticking out of the rounded corners: under **Property** type **padding** and under **Value** type a measure for the padding.
5. Click OK, and click OK again to close the Formatting dialog.

If the table's rounded corners are still not (fully) visible, check the styles for table cells. Table cells can have their own background color and by that, hide the table's background color - including the rounded corners. Table cells can have rounded corners as well, just as any other elements; see below.

### Other elements

To round the corners of elements other than boxes and images, or to have different roundings on different corners, you have to make use of the CSS property: `border-radius`; see [http://www.w3schools.com/css/css3\\_borders.asp](http://www.w3schools.com/css/css3_borders.asp).

This is, for example, how you could round the corners of a paragraph:

1. Select the paragraph (see "Selecting an element" on page 107) and then select **Format > Paragraph** on the menu, or right-click the paragraph and select **Paragraph** on the shortcut menu.
2. Click the **Advanced** button at the bottom of the Formatting dialog.
3. Under **Property**, type **border-radius**.
4. Under **Value**, type the value of the corner radius in a measure (10mm, 5px, 0.5in) or percentage (0 - 90%).
5. Click OK, and click OK again to close the Formatting dialog.

### Using a CSS file

Of course you could also add this rule to a CSS file; see "Styling templates with CSS files" on page 171. The following rule sets the border-radius of the corners of all paragraphs to 5 pixels:

```
p { border-radius: 5px; }.
```

To make this rule apply to one specific paragraph, first give the paragraph an ID (select the

paragraph and type the **ID**, for example **rounded**, on the **Attributes** pane). Then add the ID to the selector of the CSS rule, for example `p#rounded { border-radius: 5px; }`.

To make the CSS rule apply to a set of paragraphs with the same class, first give the paragraphs the same **class** (for example **rounded**). Then add that class to the selector of the CSS rule, for example `p.rounded { border-radius: 5px; }`.

## Colors

Colors make an important contribution to the look and feel of your templates. This topic explains how to define and apply colors and how to keep them consistent in different output channels.

### Defining colors, spot colors and tints

Color selectors, such as the drop-down list on the toolbar, initially contain a small set of colors. Add your own colors so that they can be used throughout the templates, in all contexts and in color selector dialogs as well as with their names in style rules (see "Styling and formatting" on page 170).

### Defining colors

To do this:

1. Select **Edit > Colors** on the menu.
2. Add a color. There are two ways to do this:
  - Click the **New** button (the green plus).
  - Select an existing color from the list and copy it using the **Duplicate** button . (The Filter drop-down limits the list to colors of a certain type.) Select the new color and click the **Edit** button .
3. In the Edit color dialog, type a name for the color (or let the Designer create a name based on the values that you select). The color's name can be used in style sheets. This name should not contain spaces or special characters.

### Tip

Working with style sheets? Choose a name that informs about the purpose of the color, rather than a name that describes the color. This way you won't have to change the color's name in the style sheets when you change the color.

4. Click **Color**. (Tint is used for transparent colors.)
5. Select the color type: **CMYK** or **RGB**.  
The letters **CMYK** stand for Cyan (a greenish-blue color), Magenta (reddish-purple), Yellow and Key (black). In color printing, these are the usual primary colors.  
**RGB** stands for Red, Green and Blue. In the RGB color model, red, green, and blue light are added together in various ways to reproduce a broad array of colors. This model is typically used for electronic devices.  
If applicable, check **Spot color**. Note that spot colors can only be used on certain printers.
6. Drag the slider bars to set the values for the color and click OK or Apply.

### Defining a spot color

A spot color is any color generated by an ink (pure or mixed). If your printer can use spot colors and you want a spot color to be used in a Print context, you can define the color as described above, with two differences:

- Check the option **Spot color** instead of **Color**.
- Make sure that the color's name matches that of the spot color used in the printer.

### Defining a tint

A tint is a transparent color, based on another color in the template. To define a tint:

1. Select **Edit > Colors** on the menu.
2. Click the **New** button (the green plus) to add the tint.
3. Click the Type drop-down and select **Tint**.
4. In the Edit color dialog, type a name for the color (or let the Designer create a name based on the values that you select). The color's name can be used in style sheets. This name should not contain spaces or special characters.

5. Select one of the existing colors in the template as t the **Source** of the color. The tint or opacity will be applied to this color.
6. Check **Use opacity** if you want to set the Tint slider to use Opacity instead.
7. Use the slider to set the percentage of the tint or opacity, or type the percentage directly in the input box and finally click OK.

## Applying a color

Colors can be applied to elements in your templates locally or through style sheets.

## Using colors in style sheets

It is highly recommended to use style sheets in templates right from the start. Even more so if the communications are going to be output to different output channels, or if they consist of different sections (for example, a covering letter followed by a policy). Using CSS with templates allows a consistent look and feel to be applied. A style sheet can change the look of multiple elements, making it unnecessary to format each and every element in the template, time and again, when the company's layout preferences change. See "Styling templates with CSS files" on page 171.

In style sheets, you can color every type of element that has a CSS color property, such as **color**, **background-color** or **border-color**. Use the color's name as it is defined in the Designer, or any legal color value: a valid color name (see [color names on w3schools](#)), hexadecimal color code (see [w3school's color picker](#)), RGB color value, for example `rgb(216,255,170)` or CMYK color value, for example `cmyk(15%, 0%, 33%, 0%)`.

The following CSS rule applies `MyColor`, which is a custom color (see "Defining colors, spot colors and tints" on page 190), to the text of all paragraphs:

```
p {  
color: MyColor;  
}
```

## CMYK colors

You may use the custom `cmyk()` CSS function to assign a CMYK color to any element, or a series of elements. The following example assigns a steel blue color as a background for all H1 elements:

```
h1 {  
background-color: cmyk(33%, 17%, 0%, 20%);  
}
```

## Coloring text

Instead of using a style sheet (see above), you can color text locally:

1. Select text or an HTML element that contains text (see "Selecting an element" on page 107).
2. On the menu, select **Format > Color**, or click the black triangle on the **Text color** toolbar button.
3. Select one of the colors in the list, or click **Other** to set all aspects of the text style, including text color and/or background color.

## Coloring backgrounds and borders

Instead of using a style sheet (see above), you can color a background or border locally. This is how:

1. Select an HTML element (see "Selecting an element" on page 107).
2. On the **Format** menu, click the element. For a **div** element, click **Box**. The Formatting dialog opens up.
3. Click the **Border** or **Background** tab.
4. Click the downward pointing arrow next to **Color** to select a color from the list of predefined colors (see "Defining colors, spot colors and tints" on page 190).  
Alternatively, click the small rectangle to the right of the color list to open the Color Picker dialog. In this dialog you can select a color from the color wheel. You can also choose the color mode: RGB or CMYK. For an explanation of these two modes, see "Defining colors, spot colors and tints" on page 190; for an explanation of the other options in this dialog, see "Color Picker" on page 262.  
You could also type a name or value in the Color field directly. It must be a valid color name (see [color names on w3schools](#)), a hexadecimal color code (see [w3school's color picker](#)), RGB color value, for example `rgb(216, 255, 170)` or CMYK color value, for example `cmyk(15%, 0%, 33%, 0%)`.
5. Click **OK** or **Apply**.

## Color management

Color profiles can keep colors consistent across different outputs. To manage color profiles, select **Edit > Color settings**; for an explanation of the options in the Color settings dialog, see "Color Settings" on page 264.

# Fonts

In templates for personalized customer communications you can use the fonts that are provided with the Designer, as well as imported fonts.

## Applying a font

To apply a particular font to a piece of text, you can:

- Select some text, or an element that contains text (see: "Selecting an element" on page 107) and select a font from the **Fonts** drop-down on the toolbar.
- Use the name of the font in a CSS rule, for example:

```
body {  
font-family: Verdana, Arial, sans-serif;  
}
```

Instead of the body tag, any element that can have the CSS property 'font-family' can be used.

Make sure that the rule is applied to the text that you wanted to apply the font to; see [Styling with CSS](#).

Note: The reason for specifying more than one fonts in a style sheet for web pages is that the font might not be available on the device on which the web page is viewed. Order the font names by preference. The last one should be generic font family (either serif or sans-serif).

## Importing a font

To import a font into a template:

- Drag the appropriate font files into the **Fonts** folder on the **Resources** pane.

When text is displayed in an imported font, the Designer can mimic the bold and italic versions of that font. If you have separate files for the bold, italic and possibly other versions of a font, you can make the Designer use the appropriate files to style text. To do this:

1. Import the files for the bold, italic and/other versions of the font into the Fonts folder.
2. On the **Edit** menu, click **Fonts**, to open the Font Manager.
3. Select the normal version of the imported font and duplicate it using the **Duplicate** button, once for each version of the font.

4. For each of the duplicates, combine a font effect with a file:
  - Click a duplicate and click the button Edit. Note: don't change the duplicate's name!
  - Select the appropriate font effect (font-weight and/or font-style).
  - Check the file or files the Designer should use for that effect. Per file type, one file can be checked.
5. Close the Font Manager.

The Designer currently supports 4 font types: TTF, OTF, WOFF, EOT and SVG.

When you are creating a **Web** template, keep in mind that the different font types are not supported by all clients; for instance, EOT and SVG are used only by Explorer and Safari, respectively.

If you're creating an **Email** template, it's better to import several types of the same font, in order for any client to see the appropriate fonts.

In the case of a **Print** context you do not need to provide alternative fonts, because the output is not displayed using a font from the device on which the output is read.

#### Note

Font software may have specific restrictions for copying and redistribution. Please consult the license agreement for each font vendor before using it in a template. It is your responsibility to comply with the requirements of third-party agreements.

### Applying an imported font

Once a font is imported, it is automatically added to the Fonts drop-down on the toolbar.

It can also be used in the style sheets, even in combination with other fonts, for example:

```
body {  
font-family: 'MyWebFont', Arial, sans-serif;  
}
```

# Spacing

Boxes, tables, paragraphs and many other elements have a **margin** and **padding**.

The margin is the white space around an element, outside the border. It is used to position an element in relation to the other elements, by putting more space between the element and its surrounding elements.

The padding is the space between an element's content and its border. It is used to position the content of the element inside the border.

Elements have a rectangular shape, so they have four sides. The margin and padding have be different on all sides.

## Tip

Use a negative left margin to create a hanging paragraph or image.

To set the spacing:

1. Right-click the element and click the respective element on the shortcut menu. Alternatively, select the element (see "Selecting an element" on page 107) and on the **Format** menu click the respective element.
2. Click the **Spacing** tab.

## Note

All settings in the Formatting dialog are in fact CSS style rules. Click the **Advanced** button to manually add CSS properties (at the left) and values (at the right). For more information about CSS, see "Styling and formatting" on page 170.

It is also possible to set an element's border in a style sheet; see Styling templates with CSS files.

3. Set the value for the **padding** in measure or percentage. You can do this for each side separately, which is equivalent to the **padding-top**, **padding-bottom**, **padding-left** or **padding-right** property in CSS. To set the same padding for all sides, check the option **Same for all sides**. This is equivalent to the **padding** property in CSS.

4. Set the value for the margin in measure or percentage. You can do this for each side separately, which is equivalent to the **margin-top**, **margin-bottom**, **margin-left** or **margin-right** property in CSS. To set the same margin for all sides, check the option **Same for all sides**. This is equivalent to the **margin** property in CSS.
5. Click **OK**, or click **Apply** to apply the changes without closing the dialog.

## Styling a table

Just as other elements, tables can be styled in two ways:

- With **local formatting**. This means styling the table directly, using the Formatting dialog.
- Via **Cascading Style Sheets (CSS)**. In a style sheet, style rules are declared for elements with different HTML tags, ID's and classes.

These two methods are described below. See "Styling and formatting" on page 170 for background information about these two methods.

### Selecting a table, row or cell

There are several ways to select a table or row:

- Click in the table or row. Then, in the **breadcrumbs** (see "Selecting an element" on page 107) click **table** to select the table, or **tr** to select the row.
- Right-click a cell and from the shortcut menu, choose **Table > Select** or **Row > Select**.
- Click in a cell and then use the toolbar: click the **Select Table** button or click the black triangle next to that button and then click **Select Table** or **Select Row**.

Selecting one cell is easy: just click in it.

#### Tip

Use the Styles pane to see which styles apply to the currently selected table, row or cell.

### Via the Formatting dialog

The Formatting dialog allows you to change the font, font size and color (see "Fonts" on page 194), the borders (see "Border" on page 187), the cell padding (the distance between the

edge of the cell and its content, see "Spacing" on page 196), and the background color or image of the table and its cells ("Background color and/or image" on page 186).

To open the Formatting dialog for **one cell** or for the **table as a whole**:

- Click in a cell and choose **Format > Table** or **Format > Table Cell**.
- Right-click it and choose **Cell...** or **Table...** from the shortcut menu.

Note that in this case **Table** styles the table as a whole. When you choose **Table** and change the border, for example, the borders of the cells inside it will not be changed.

To style **all cells** in a table or row at the same time via the Formatting dialog, you have to select the table or row first; see "Selecting a table, row or cell" on the previous page

Next, to open the Formatting dialog, choose **Format > Table Cell**. The settings that you make now will be applied to all cells in the selected row or table.

### **Via a style sheet**

Cascading Style Sheets (CSS) offer more ways to style a table and its contents, than the Formatting dialog does.

How to use style sheets is explained in another topic; see "Styling templates with CSS files" on page 171.

Note that to make a style rule apply to a **specific** table, row or cell, you have to add an ID or class to that table, row or cell.

### **Adding an ID or class to a table, row or cell**

A style sheet contains a bunch of style rules for different elements, that are identified via a CSS **selector**. This can be the element's HTML tag (without the angle brackets), ID or class.

When used as a CSS selector, the HTML tag for a table is **table**. For a row, it is **tr** and for a cell, **td**. A style rule that uses one of these, however, would apply to **all** tables, rows, or cells. For a rule to be more specific you need to add an ID (for a unique element) or a class (for a set of similar elements) to the table, row or cell, and use that as the style rule's selector.

Before you can add an ID or class to a table, row or cell, you have to select that table, row or cell (see "Selecting a table, row or cell" on page 197). After selecting the cell, row or table, type the ID or class in the respective field on the **Attributes** pane.

In CSS, refer to the table, row or cell with `#ID` (where ID should be replaced with the actual ID) or with `.class` (where class should be replaced with the actual class).

## Styling the first, last and nth rows

The CSS pseudo-classes `:first-child`, `:last-child` and `:nth-child()` are very useful for styling table rows .

A CSS **pseudo-class** follows a selector to specify a special state of that selector. It always starts with a colon.

The pseudo-classes `:first-child`, `:last-child` and `:nth-child()` select an element only if it is the first, last or nth child element respectively. (In HTML and CSS, the word **child** refers to an element inside another element.)

The following CSS style rule selects the table row (tr) that comes first (`:first-child`) in its parent (which naturally is a table), and colors its background red:

```
tr:first-child {
    background: red;
}
```

## Selecting a specific row, odd or even rows, or every nth row

The pseudo-class `:nth-child()` lets you select a specific row, all odd or even rows, or every nth row.

Between the round brackets in `:nth-child()` you can fill in a number, `odd` or `even`, or a formula: `an+b`. In the formula, `a` represents a cycle size (every...), `n` is a counter (for the child elements), and `b` is an offset value ('start at b'). The following examples will make this clear.

`:nth-child(3)` matches just one element: the third child element.

`:nth-child(odd)` matches child elements 1, 3, 5, 7, etc. The keyword `odd` substitutes the expression `2n+1`, which in other words says: 'take every second element, starting at 1'.

`:nth-child(even)` matches child elements 2, 4, 6, 8, etc. The keyword `even` substitutes the expression  $2n+0$ , or simply  $2n$ .

`:nth-child(3n)` matches child elements 3, 6, 9, 12 etc.

`:nth-child(3n+1)` matches child elements 1, 4, 7, 10 etc., so every third element, starting at 1.

### Via script (based on a data field value)

To style a table, row or cell based on a data field value, you have to write a script (see "Writing your own scripts" on page 223).

First add an ID or class to the table, row or cell that needs to be styled: select the element (see "Selecting a table, row or cell" on page 197) and add an ID on the **Attributes** pane. Then create a script, using that ID or class as the script's selector. The script can be very simple:

```
if (record.fields.COUNTRY == 'CANADA') {
    results.css('color', 'green');
}
```

The Designer Scripts API provides several functions to style elements, for example `css()`, `hasClass()` and `addClass()` (see "Designer Scripts API" on page 429).

## Rotating elements

In any type of template, boxes, images, tables, text and other elements can be rotated.

The toolbar buttons **Rotate Clockwise** and **Rotate Counter Clockwise** rotate the element in which the cursor is located 90 degrees at a time.

To rotate an element into another angle position, use the 'angle' CSS property of the element. In most cases, this can be done in the element's Formatting dialog. In other cases, such as with text, you have to enter the CSS property and value manually. Both methods are explained in the following procedure.

1. Right-click the element and click the respective element on the shortcut menu. Alternatively, select the element (see "Selecting an element" on page 107) and click the respective element on the **Format** menu.

2. On the first tab, look for the **angle** property. If it is available, type the number of degrees the element should be rotated. A positive number will rotate the element clockwise, a negative number rotates it counter-clockwise. Skip steps 3 to 6. If the angle property is not available, proceed with the following step.
3. Click the **Advanced** button to open the Advanced Formatting dialog.
4. Click in the first blank field under **Property** and type **transform**.
5. Click in the field next to it, under **Value** and type **rotate(**, followed by the number of degrees the element should rotate, and then **deg)**, for example: rotate(20deg). A positive number will rotate the element clockwise, a negative number rotates it counter-clockwise.
6. Close the Advanced Formatting dialog.
7. Close the Formatting dialog, or click the Apply button to see the effect without closing the dialog.

#### Note

It is also possible to rotate elements by creating a style rule in a style sheet; see "Styling templates with CSS files" on page 171.

## Locale

The locale is a setting that can affect date, time and currency output, and other formatting that depends on location and language. This setting is specific to each template, so changing it for one template will not affect other templates.

Assume that a record set has a `Date` field that contains the following date: 4/11/12, and that this field has been added to the template using the Text Script Wizard with the Long Date format (see "Using the Text Script Wizard" on page 214 and "Formatting variable data" on page 217). If the locale is set to `en-US`, the date appears on the page as **April 1, 2016**. Setting the locale to `fr-CA` makes this text appear as **1 avril 2016**. Setting it to `zh-CN` will print **2016年4月1日**.

The locale can also be used in scripts; see "Writing your own scripts" on page 223 and "Designer Scripts API" on page 429.

## Changing the locale

By default, the locale is the same as the operating system's locale setting. To change this setting for the currently open template:

1. On the menu, select **Edit > Locale**.
2. Use the drop-down to select how the locale is to be set for the current template:
  - Select **System Locale** to use the operating system's locale settings. The operating system's locale is set in the **Region** settings of the control panel. Note that when output is generated on a different operating system, that operating system's locale will be used.
  - Select **Explicit Locale** to specify a static locale which will remain static for this template, whichever server the template is used on. Use the **Locale** drop-down to select a specific locale. The locales comprise a language code followed by a 2-letter country code (*de-DE*, *zh-CN*, *fr-CA*, *fr-FR*, etc), as defined by the international standards ISO-639-1 and ISO 3166.
  - Select **Data Field** to use a data field from the record. The locale will be record-specific in this case. Use the drop-down to select a field within the current Data Model that contains the locale. This field must be a string and contain the exact locale to be used, such as "en" or "fr-CA". It cannot be an alias such as "english" or "french". The locale supports language codes (*en*, *fr*, etc), as well as language codes followed by a 2-letter country code (*de-DE*, *zh-CN*, *fr-CA*, *fr-FR*, etc). The language codes are defined by ISO-639-1. The 2-letter country code as defined by ISO 3166.
3. Click **OK** to apply the setting. The setting will be saved with the template.

## Personalizing content

Variable-data printing is a form of digital printing in which elements such as text and graphics may be changed using information from a database or data file. It prints unique documents with customized messages for each customer. This is exactly what you can do with Connect: using variable data you can personalize your company's communications.

Before you can start personalizing the content of a template, you must open a data file or database; see: "Loading data" on page 204.

The most common ways to personalize templates are listed below.

## Variable data

Variable data are data from a database or data file that are used to personalize documents for each customer. Variable data fields can be inserted in the text directly. For example, if a person's last name can be found in your data, the field that holds the last name can be used in the text of a letter or email. Scripts in PrintShop Mail Connect Designer are the basis of Variable Data Printing.

The easiest, quickest and most direct way to add customer data to content is via drag and drop; see "Variable Data" on page 211.

The drag-and-drop method results in a Text Script. Another way to create a Text Script is to use the Text Script Wizard. Often it is better to use the Text Script Wizard than the drag-and-drop method.

The Text Script Wizard gives you more control over the way data is displayed. It can insert one or more data fields, each with an optional prefix and suffix. For blocks of data, such as addresses, the Text Script Wizard definitely is the better choice. See "Using the Text Script Wizard" on page 214.

## Conditional content

In a template you may want to reveal content - text or images - to one group of recipients, but hide it from others. You can use a Conditional Script Wizard to achieve this, if you have a data field in your data on the basis of which a condition can be set. See "Showing content conditionally" on page 219.

## Dynamic images

Dynamic Images are dynamic in the sense that they are replaced by another image when a data field contains a certain value. Think of a signature image being swapped based on the sender's name, for example. You can use the Dynamic Image Script Wizard to make this happen; see "Dynamic Images" on page 222.

## Snippets

Snippets are pieces of content that can be re-used within the same template, in all contexts and sections. Snippets can contain any contents that a section can have, such as text, images, variable data, dynamic tables, etc. They are often very useful to personalize content, especially

in combination with variable data and scripts. See "Snippets" on page 168 and "Loading a snippet via a script" on page 238.

## Scripts

### Self-made scripts

As soon as you want to do more than what can be done with the available (Text, Conditional) Script Wizards, self-made scripts are the solution. You could, for example, combine data of two or more data fields in a condition for conditional text. Or you could load a part of a snippet depending on the value of a data field. With a self-made script you can achieve anything that can be done by any of the Script Wizards, and much more. For an introduction on this, see "Writing your own scripts" on page 223.

### Control Scripts

When output is generated from a template, Control Scripts run **before** all other scripts, when a record is merged with a context. They determine how different sections of the context are handled. They can, for example, make the page numbering continue over all Print sections, split Email attachments, or omit Print sections from the output.

Some knowledge of JavaScript is needed to edit Control Scripts, just as for any other self-made scripts, because there is no Control Script Wizard; see "Writing your own scripts" on page 223.

See "Control Scripts" on page 240.

## Loading data

Before you can add variable data fields to a template in the Designer, you need to have a Data Model and a sample of customer data. At the design stage the Designer doesn't have to have access to all data; it just needs to know which data fields exist in your data and it needs some data to be able to display a preview of the output.

To get access to a Data Model and data, you can open:

- a data file, see "Adding data from a data file" on the next page
- a database, see "Adding data from a database" on page 206.

After opening a data file or database, the **Data Model** pane at the right hand bottom shows the data fields that occur in the data.

The **Value** column displays data from the first record in the data file. Use the **First**, **Previous**, **Next** and **Last** buttons to browse through the records.

### Tip

If you have no data at hand, download a demo from <http://demo.objectiflune.com> and open a dummy data file to test with.

## Adding data from a data file

1. Click **File**, select **Add Data** and then click **From file data source**. Browse to the location of the file and select it.

The Designer can open the following types of data files:

- CSV files (.csv)
  - Microsoft Access Database (.mdb, .accddb)
  - XML files (.XML).
  - AFP files (optional).
2. Review the options presented, to ensure that the data will be interpreted correctly. The options available depend on the type of data file (see below).

## CSV file options

- **Encoding**: the Designer can not infer from a CSV file what encoding it is in. The default is right in the large majority of cases, but when it isn't, it can be very difficult to figure out the correct encoding. Ask your source what the encoding of the file is.
- **Field separator**: choose the character that separates the fields in the file.
- **Comment delimiter**: if there are comment lines in the file, type the character that starts a comment line.
- **Text Delimiter**: type the character that surrounds text fields in the file. Other delimiters will not be interpreted within these text delimiters.
- **Ignore unparsable lines**: when checked, any line that does not correspond to the above settings will be ignored.

- **First row contains field names:** check this option to use the first line of the CSV as headers. This option automatically names all extracted fields.

### **MDB file options**

- **File:** Include the full path to the file.
- **Password:** If the file isn't password protected, you can click **Next** without filling out this field.
- **Table name:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
- **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.

### **XML File options**

Select what level of XML elements defines a record.

The **Trigger** is what triggers the creation of a new record. It can be set to:

- **On element:** this defines a new record when a new element occurs on the selected XML level.
- **On change:** this defines a new record when a specific field under the chosen XML level has a new value. After selecting this option, you have to select the field that triggers the creation of a new record.

### **Adding data from a database**

1. Click **File**, select **Add Data** and then click **From database data source**. Browse to the location of the file and select it.

The Designer can open databases from the following types of data sources:

- MySQL
- Microsoft Access Database (.mdb, .accddb)
- SQL Server
- ODBC DataSource
- JDBC
- Oracle.

2. Review the options presented. The options available depend on the type of database data source; see below.

## MySQL

1. Enter the appropriate information to connect to the database:
  - **Server:** Enter the server address for the MySQL database.
  - **Port:** Enter the port to communicate with the MySQL server. The default port is 3306.
  - **Database name:** Enter the exact name of the database from where the data should be extracted.
  - **User name:** Enter a user name that has access to the MySQL server and specified database. The user only requires *Read* access to the database.
  - **Password:** Enter the password that matches the username above.
2. Click **Next** and enter the information for the source table.
  - **Connection string:** Displays the full path to the database.
  - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
  - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.
3. Click **Finish** to open the database.

## Microsoft Access

1. Enter the appropriate information to connect to the database:
  - **File name:** Browse to your Microsoft Access database file (.mdb)
  - **Password:** Enter a password if one is required.
2. Click **Next** and enter the information for the source table.
  - **Connection string:** Displays the full path to the database.
  - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
  - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.

3. Click **Finish** to open the database.

## SQL Server

1. Enter the appropriate information to connect to the database:
  - **Server:** Enter the server address for the SQLServer database.
  - **Port:** Enter the port to communicate with the SQLServer. The default port is *1433*.
  - **Database name:** Enter the exact name of the database from where the data should be extracted.
  - **User name:** Enter a username that has access to the SQLServer and specified database. The user only requires *Read* access to the database.
  - **Password:** Enter the password that matches the username above.
2. Click **Next** and enter the information for the source table.
  - **Connection string:** Displays the full path to the database.
  - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
  - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.
3. Click **Finish** to open the database.

## ODBC DataSource

1. Select the ODBC system data source. Note: Only 32-bit data sources are currently shown in this dialog, even if your system is 64-bits.
2. Click **Next** and enter the information for the source table.
  - **Connection string:** Displays the full path to the database.
  - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
  - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.
3. Click **Finish** to open the database

## JDBC

1. Enter the appropriate information to connect to the database:
  - **JDBC Driver:** Use the drop-down to select which JDBC Driver to use for the database connection.
  - **JAR file path:** Enter a path to the JAR file that contains the appropriate driver for the database below.
  - **Server:** Enter the server address for the database server.
  - **Port:** Enter the port to communicate with the server.
  - **Database name:** Enter the exact name of the database from where the data should be extracted.
  - **User name:** Enter a username that has access to the server and specified database. The user only requires *Read* access to the database.
  - **Password:** Enter the password that matches the username above.
  - **Advanced mode:** check to enable the Connection String to manually enter the database connection string.
  - **Connection string:** Type or copy in your connection string.
2. Click Next and enter the information for the source table.
  - **Connection string:** Displays the full path to the database.
  - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
  - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.
3. Click **Finish** to open the database.

## Oracle

1. Enter the appropriate information to connect to the database:
  - **Server:** Enter the server address for the Oracle database.
  - **Port:** Enter the port to communicate with the Oracle server.
  - **Database name:** Enter the exact name of the database from where the data should be extracted.

- **User name:** Enter a username that has access to the Oracle server and specified database. The user only requires *Read* access to the database.
  - **Password:** Enter the password that matches the username above.
2. Click **Next** and enter the information for the source table.
    - **Connection string:** Displays the full path to the database.
    - **Table:** Use the drop-down to select the appropriate table or stored query to retrieve the appropriate data set.
    - **Encoding:** Use the drop-down to select the encoding with which to read the data in the table.
  3. Click **Finish** to open the database.

After adding data from a database, the **Data Model** pane at the right hand bottom shows the data fields that occur in the data.

The **Value** column displays data from the first record in the data file. Use the **First**, **Previous**, **Next** and **Last** buttons to browse through the records.

### **Add a counter using the Generate Counter Wizard**

Generating a counter is useful for numbered tickets or any other template requiring sequential numbers but no variable data.

The Generate Counter Wizard creates a record set with a Counter field and in that field, the current counter value for each record. The Counter starts and stops at set values and is incremented by a set value as well.

1. To open the Generate Counter Wizard, select **File > Add data > Generate counters**.
2. Adjust the settings:
  - **Starting value:** The starting number for the counter. Defaults to 1.
  - **Increment value:** The value by which to increment the counter for each record. For example, an increment value of 3 and starting value of 1 would give the counter values of 1, 4, 7, 10, [...]
  - **Number of records:** The total number of counter records to generate. This is not the end value but rather the total number of actual records to generate.

- **Padding character:** Which character to add if the counter's value is smaller than the width.
- **Width:** The number of digits the counter will have (prefix and suffix not included). If the width is larger than the current counter value, the padding character will be used on the left of the counter value, until the width is equal to the set value. For example for a counter value of "15", a width of "4" and padding character of "0", the value will become "0015".
- **Prefix:** String to add before the counter, for example, adding # to get #00001. The prefix length is not counted in the width.
- **Suffix:** String to add after the counter. The suffix length is not counted in the width.

3. Click **Finish** to generate the Counter record set.

### Tip

While the Generate Counter script is really useful for things like raffle tickets, it's unusable in combination with a data file or database, as it cannot complement that data automatically. This can only be done with a script. A script that adds a counter to data, using the current record index to calculate the current counter value, can be found in this how-to: [Manual counter in designer](#).

## Variable Data

Variable data are data from a database or data file that are used to personalize documents for each customer. Variable data fields can be inserted in the text directly. For example, if a person's last name can be found in your data, the field that holds the last name can be used in the text of a letter or email. Scripts in PrintShop Mail Connect Designer are the basis of Variable Data Printing.

After loading data from a data file or database (see "Loading data" on page 204), you can add variable data fields to the contents of your template. You can do this via the drag-and-drop method, or using the Text Script Wizard.

Use the **Text Script Wizard** when there are empty fields in the data, and the value of a data field needs to be preceded or followed by a space, line break or text in the template. Otherwise, empty data fields will cause empty lines and superfluous white spaces to show up in the text. You should also use this method for blocks of data, such as address blocks, and when you want to format data differently, for example, when you want a number to be displayed as a currency.

You can use the **drag-and-drop** method for simple fields that do not need to be preceded or followed by a space, line break or text.

#### Note

Web templates are personalized just like any other template. There are a few extra possibilities, though; see Using variable data on a Web page.

### Inserting variable data directly (drag-and-drop)

An easy, quick and direct way to insert variable data in the content is via drag and drop:

1. Open the section you want to add the data field to.
2. Drag and drop a data field from the **Data Model** pane at the bottom right into the content of your template.  
To select and insert multiple data fields at the same time, press **Shift** or **Ctrl**, whilst selecting fields in the **Data Model** pane.

What happens is that:

- A **placeholder** for the value of the data field shows up in the text. It looks as follows: @FIELDNAME@.
- A **text script** appears in the **Scripts** pane at the bottom left.

A **text script** replaces placeholders in the content with the value of a data field in the current record.

Switch to the **Preview** tab at the bottom of the workspace to see the script in operation. The value of the corresponding data field in the first record appears instead of the placeholder, everywhere where the placeholder is found in the text. This value will be refreshed when you browse through the records in the Data Model pane.

When the output (the letter, email, etc.) is generated, the text script executes for each record in the record set, and each time it replaces the placeholders by the value of the field in the current record.

In the **Scripts** pane you can see that the script has a **name** and a **selector**.

The drag-and-drop method automatically generates a script that is named after the data field (see the first column of the **Scripts** pane).

The **selector** (in the second column in the **Scripts** pane) is the text that the script will replace. The selector that the drag-and-drop method generates for a script, is the same as the placeholder that is placed in the text.

When you drag the same field to the content again, a second placeholder appears in the text, but no new script is added. The existing script will find and replace all placeholders that match its selector.

### Tip

Press the **Alt** key while dragging, to wrap the placeholder in a **span**, give the span an **ID** and have that ID used as the script's selector.

Press the **Ctrl** key while dragging, to wrap the placeholder in an absolute positioned box (a **div**) at the cursor position. A unique ID is assigned to the box and used as the script's selector. This method is particularly useful when the document mainly consists of a PDF used as the background image of a section (see "Using a PDF file as background image" on page 65).

### Tip

Drag the data field directly to the **Scripts** pane to create a script without adding a placeholder to the template.

### Note

Looking for text in a text is a less optimized operation and may impact output speeds in longer documents. To speed up the output process, put the placeholder(s) in a Box or Span (see "Boxes" on page 150), give that Box or Span an ID and use that ID as the script's selector. See "Using the Text Script Wizard" on the facing page for an explanation about the various types of selectors. For more tips to make a template generate output faster, see "Optimizing scripts" on page 234.

## Using the Text Script Wizard

The Text Script Wizard can insert one or more data fields into your template, each with an optional prefix and suffix. It is recommended to use the Text Script Wizard for blocks of data, such as address blocks, and when data fields can be empty or need to be formatted differently.

1. Create a new text script and open the Text Script Wizard. There are two ways to do this:
  - On the **Scripts** pane at the bottom left, click the black triangle on the **New** button and click **New Text Script**. A new script appears in the list. Double-click the new script to open it.
  - Select a word in the content. Right-click the selection and on the shortcut menu, choose **Text Script**.

The Text Script Wizard appears.

2. Change the name of the script to make clear what it does.
3. The **selector** states the text to be found in the template. The results can be replaced by the script.

### Tip

Hover over the name of a script in the **Scripts** pane to highlight parts of the template that are affected by the script.

- **Text**, for example: @lastname@, or {sender}. The text doesn't have to have any special characters, but special characters do make it easier to recognize the text for yourself. In the Text Script Wizard, click **Text** and type the text to find.

### Note

A script made with the Text Script Wizard for a block of data already runs faster than a series of individual scripts, because it only has one selector. However, searching for text can be a lengthy operation, compared to searching for an element with an ID. When speed matters, select one of the two remaining options: **Selector** or **Selector and Text**. See also: "Testing scripts" on page 231 and "Optimizing scripts" on page 234.

- An **HTML/CSS selector**:
  - HTML elements, such as a paragraph. In the Text Script Wizard, click **Selector** and type the HTML tag without the angle brackets, for example: **p**.
  - HTML elements with a specific class. In the Text Script Wizard, click **Selector** and type the class name, including the preceding dot, for example: **p.green** for all paragraphs with the class 'green' or **.green** for all kinds of HTML elements that have the class 'green'. See "Styling and formatting" on page 170 for an explanation about CSS (Cascading Style Sheets).
  - An HTML element with a specific ID. In the script Wizard, click **Selector** and type the ID, including the preceding #, for example: **#intro**.

**Note**

Each ID should be unique. An ID can be used once in each section.

- Etcetera. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for more selectors and combinations of selectors.
  - A **selector and text**. This is text inside an HTML element (or several HTML elements) with a specific HTML tag, class or ID. In the Text Script Wizard, click **Selector and text** and type the selector and the text in the respective fields.
4. Click the the downward pointing arrow in the first row in the column **Field**. Select a data field from the list that appears.
  5. Add a **Prefix** and/or a **Suffix**. The prefix and suffix can contain text and/or HTML tags. If a field is empty, the prefix and suffix will be ignored, which means you can add line returns and static text, such as:
    - with a Number field, Prefix: Your invoice (one space at the end), Suffix: is now ready to be viewed!
    - with a field LastName, Suffix <br/> (which adds a line break)
    - with a field State, Prefix: , (comma then space).

For a comma between fields, use the Prefix of the second field, if you don't want a comma when the second field has no value.
  6. The Wizard allows you to reformat the data (for example, apply uppercase, apply thousand separators to numbers, etc.). Click the column **Format**, click the downward

pointing arrow and select one of the formats. See "Formatting variable data" on the next page.

7. Add as many data fields as you need, following the same procedure.
8. Optionally, you can click **Options** to specify where and how the script inserts its results:
  - As **HTML**. HTML elements in the results are processed and displayed as HTML elements. For instance, `<b>this is bold</b>` will be displayed as **this is bold**. This is the default setting.
  - As **text**. This inserts the results as-is, meaning HTML tags and elements are displayed as text in the output. In this scenario, "`<br>`" shows up in the text and does not insert a line break.
  - As the value of an **attribute** of an HTML element. The selector of the script should be an HTML element. Which attributes are available depends on the selected HTML element. If the script's selector is an image (`<img>` element) for example, and the attribute is `src`, the script will modify the image's source. The script's results should be a valid value for the chosen attribute.

#### Note

When checked, the option **Convert fields to JSON string** writes the results from the script into an attribute or text as a JSON string. This is useful for web contexts where a front-end script can read this value easily.

9. Close the Text Script Wizard and type the placeholder for the results of the script in the content of your template, or make sure that there is at least one element that matches the selector of the script.
10. Hover over the name of the script in the **Scripts** pane. In the workspace you will see which parts of the template are affected by the script. If the script produces an error, the error message will be displayed in a hint on the **Scripts** pane.

#### Tip

When one of the included data fields is empty, the respective line, including the prefix and suffix, is skipped. The result of the script will be shorter, causing the rest

of the content to move up or down. If, in a Print context, you don't want the result of the script to be part of the text flow (for example, when a letter is going to be sent in an envelope with a window), put the placeholder for the script in a positioned box (see "Boxes" on page 150 and "How to position elements" on page 179).

### Tip

An example of how to create an address block using the Text Script Wizard is described in a how-to; see [How to create an Address Block](#).

## Formatting variable data

When a Text Script, made with the Text Script Wizard (see "Using the Text Script Wizard" on page 214) adds variable data to a template, it can easily change the way the data are formatted as well. This is done in the Text Script Wizard through a special formatting modifier or a format mask for each field that the script adds to the template.

You could also format data in a script using the `formatter` ; see "Designer Scripts API" on page 429.

### Date

Dates in variable data can be displayed as long, medium and short dates with different time displays. There are quite a few presets, but you can also enter a custom format mask.

1. Open the Text Script Wizard: double-click to open an existing script in the Scripts pane or create a new Text Script using the Text Script Wizard; see "Using the Text Script Wizard" on page 214.
2. Click a data field that contains text, or add such a data field to the script with the Add field button on the right.
3. Under **Format** you can choose one of the following options:
  - **Short Date** displays the day, month and year in two digits each, for example **01.04.16**.

- **Medium Date** displays the day and month in two digits each and the year in four digits, for example **01.04.2016**. (This is also the value of the Default Date.)
- **Long Date** displays the day as a number, the month's full name and the year in four digits, for example **1. April 2016**.
- **Short Time** displays a time in hours and minutes in two digits each, for example **00:00**.
- **Medium Time** displays a time in hours, minutes and seconds in two digits each, for example **00:00:00**. (This is also the value of the Default Time.)
- **Long Time** displays a time in hours, minutes and seconds in two digits each, and adds a time zone, for example **00:00:00 EDT**.
- **Short Date/Time** displays the date as a short date and the time as a short time, for example **01.04.16 00:00**.
- **Medium Date/Time** displays the date as a medium date and the time as a medium time, for example **01.04.2016 00:00:00** (This is also the value of the Default Date/Time.)
- **Long Date/Time** displays the date as a medium date and the time as a medium time, for example **1. April 2016 00:00:00 EDT**.

#### Note

The locale influences the way dates, times, numbers and currencies are formatted; see "Locale" on page 201.

4. Close the Script Wizard. For a new script, don't forget to add the selector to the template.

### Font style

Text originating from variable data can be displayed in uppercase, lowercase or proper case.

1. Open the Text Script Wizard: double-click to open an existing script in the Scripts pane or create a new Text Script using the Text Script Wizard; see "Using the Text Script Wizard" on page 214.
2. Click a data field that contains text, or add such a data field to the script with the Add field button on the right.

3. Under **Format** choose the correct setting:
  - **Uppercase** transforms all characters to uppercase.
  - **Lowercase** displays transforms all characters to lowercase.
  - **Propercase** transforms the first character of each word to uppercase and all other characters to lowercase.
  - **None** leaves the text as is.
4. Close the Script Wizard. For a new script, don't forget to add the selector to the template.

## Numbers and currencies

Numbers, and strings existing of digits, can be displayed as a number with a certain formatting or as an amount of money.

1. Open the Script Wizard: in the Scripts pane, double-click the script, or create a new Text Script using the Text Script Wizard; see "Using the Text Script Wizard" on page 214.
2. Click the data field that contains the numeric value that you want to display differently, or add the data field to the script with the Add field button on the right.
3. Under **Format** choose one of the following settings:
  - **Grouped** displays a number with three decimal places and sets the thousands separator for the value based on the current locale; see "Locale" on page 201.
  - **Currency** displays a number as an amount of money, with a thousands separator and rounded to two decimal places, based on the current locale; see "Locale" on page 201.
  - **Currency no symbol** does the same as Currency, but omits the currency symbol.
4. Close the Script Wizard. For a new script, don't forget to add the selector to the template.

## Showing content conditionally

One way to personalize content is to show or hide one or more elements depending on a field's value. For example, a paragraph written for Canadian customers could be hidden when the recipient of the letter is not living in Canada, if that can be derived from the data.

Use the Conditional Script Wizard to show or hide one element – a paragraph, image or other HTML element - based on the value of a data field.

## Showing or hiding elements using the Conditional Script Wizard

1. Right-click the element and click **Make Conditional**. Alternatively click the black triangle on the **New** button on the **Scripts** pane at the bottom left of the window, and click **Conditional Content Script**. The Conditional Script Wizard opens.
2. Rename the script so that it reflects what the script does.
3. If you have started creating the script from the **Scripts** pane, you have to type a **Selector**. The selector selects one or more pieces of text or elements from the template, so that the conditional content script can hide or show those pieces. An ID (for example: #conditional-script) is best if you want to show or hide one element only. Use a class selector (for example: .conditional) if the script should show or hide more than one element. See "Using the Text Script Wizard" on page 214 for further explanation on selectors.  
If you have started the Conditional Script Wizard by right-clicking an element, you don't have to set a selector. If the element didn't have an ID, a new ID has been generated automatically. The new ID functions as the selector of the script.  
You can change the selector after closing and reopening the script (double-click the name of the script in the **Scripts** pane).
4. Set the **Action**: use the drop-down to select whether to **Show** or **Hide** the element when the condition below is true.
5. Click the downward pointing arrow next to **Field**, to select the data field that should be evaluated.
6. Click the downward pointing arrow next to **Condition** to expand the list of conditions with which the data field can be evaluated. The options are: **Equal to**, **Not equal to**, **Contains**, **Does not contain**, **Begins with**, **Ends with**.
7. Type the **Value** that should be used for the conditional check.  
For example, you could check whether the data field **Gender** is 'Equal To' the value 'M', in order to show a paragraph or an image applying to male customers only.  
If the condition evaluates to **true**, the selected action will be performed. If, conversely, the condition evaluates to **false**, and the option **Toggle Visibility** is checked, the opposite action will be performed. By default, this option is checked.

### Note

To combine the values of two or more data fields, you have to click **Expand** and

edit the code of the script. See "Writing your own scripts" on page 223.

8. Click **Apply** or **OK**.
9. To see the result, toggle to the **Preview** tab at the bottom of the workspace (or select **View > Preview View** on the menu).

### Showing or hiding several elements with one conditional script

To apply one conditional content script to several elements, you have to use a CSS **class** or HTML element as the selector of the script. When using a CSS class, apply that class to the elements in question:

1. Double-click the conditional script in the **Scripts** pane to reopen it, or create a new conditional content script and follow the actions described in "Showing or hiding elements using the Conditional Script Wizard" on the previous page.
2. Change the selector to a CSS class (for example, `.male`) or to an HTML element with a certain CSS class (for example, `p.male`). See "Using the Text Script Wizard" on page 214 for further explanation on selectors.
3. Apply the same CSS class to all elements that should be shown or hidden under the condition that you have set in the conditional script. Click each element and type the class (without the preceding dot) in the **Class** field.

### Showing or hiding a text selection

When you right-click on an element and make it conditional, the element as a whole will be made conditional. This happens even when you select a few words in a paragraph and right-click those words; the paragraph as a whole will be made conditional.

It is, however, possible to partially show or hide a paragraph or a line item in a list. Before you can do that, you have to select the text that you want to be shown or hidden and wrap it in a span element first:

1. Select the part of the text that you want to make conditional.
2. Right-click the selected text and click **Wrap in span**.
3. Type an **ID** and/or a **class**. An ID is fine if this is the only thing that should be shown or hidden on a given condition. Use a class if there is more that should be shown or hidden on the same condition.

4. Start creating a conditional content script from the **Scripts** pane. Use the ID or class as the selector of the script. See "Showing or hiding elements using the Conditional Script Wizard" on page 220.

## Dynamic Images

Dynamic images are called dynamic because they are switched, depending on the value of a data field. This way, a template can be adjusted to different customers.

### Adding dynamic images

Dynamic images can be added to the template using the Dynamic Image Script Wizard only if you have:

- One or more data fields that contain values on the basis of which the images can be switched.
- An appropriate image for each group of customers. All files should be of the same type and they need to be stored in one folder (the **Images** folder on the **Resources** pane, or an external folder). It is important that they are named after the various possible values of the related data field. Adding dynamic images that are not named after a data field value requires a self-made script.

To use the Dynamic Image Script Wizard:

1. Add one image to the template. See "Adding images" on page 159.
2. Right-click the image and click **Dynamic Image**. The Dynamic Image Script Wizard opens.

The image's ID is used as the script's selector. If the image did not have an ID, it is automatically generated.

The Dynamic Image Script Wizard composes a file name (including the path) based on the value of a data field, a prefix and a suffix:

- The prefix shows the path of the image.
- The suffix states the file extension of the image.
- The file name is the value of the data field(s) in the **Field** column.

The prefix and suffix are derived from the current image.

3. If necessary, enter another **Prefix** and/or **Suffix**.

4. Click the first field in the column **Field**, and then click the downward pointing arrow. Select the data field to be evaluated. Click the button **Add**, to add more fields if you want the file name to be composed of the value of several data fields. Note that only the suffix of the last data field should hold the file extension. The resulting file name, including the path and file extension, is assigned to the **src** (source) attribute of the image. You can click **Options** to verify this.
5. Click **Apply** or **OK**. Now click the **Preview** tab and browse through the records to verify that the script works as expected.

### Tip

The dynamic images feature can be used to insert dynamic signatures, as described in this how-to: [Dynamic signatures](#).

How to insert dynamic images if there are no data fields with the actual names of the images is described in another how-to: [Dynamic image that doesn't contain the data field value](#).

## Editing a Dynamic Image

To edit dynamic images added to the template earlier, right-click the image, or the space reserved for the dynamic images. Then click **Dynamic Image** to open the Dynamic Image Script Wizard again.

# Writing your own scripts

Personalization can be taken a lot further than just inserting names and addresses, and hiding or showing text or images. Every bit of information in your communications can be made entirely personal, using scripts.

Most scripts can be made using one of the script Wizards. For a block of variable data, such as an address, the Text Script Wizard is a perfect fit. Paragraphs can be made conditional with a Conditional Script Wizard. For dynamic images, you can use the Dynamic Image Script Wizard. In an Email context, you are provided with a number of script Wizards to set the sender, the recipients and the subject of the email.

However, when you want to do something that goes beyond what you can do with a Wizard, like creating a conditional paragraph with a condition that is based on a combination of data fields, you have to write the script yourself.

This topic explains how scripts work and how you can create and write a script.

## How scripts work

A script is a small set of instructions to the program, written in JavaScript.

When Connect generates the actual output – letters, web pages or emails -, it opens a record set and merges it with the template. It takes each record, one by one, and runs all scripts for it (Control Scripts first).

All scripts, except Control Scripts, must have a **selector**. The selector can be text, an HTML element and/or a CSS selector. Running a script starts with looking for pieces of content in the template that match the script's selector.

The results of this query can vary from one occurrence of a simple text (for example: @EMAIL@) to a large collection of HTML elements. For example, when the selector is **p**, the HTML tag for a paragraph, all paragraphs will be collected and passed to the script.

### Tip

Hover over the name of a script in the **Scripts** pane to highlight parts of the template that are affected by the script.

Next, the script can modify the selected pieces of content, using values from the record that is merged to the template at the time the script runs. It can, for example, hide, replace or add text or change the style of those pieces of content. This is how scripts personalize documents.

## Creating a new script

Writing a script starts with this procedure:

1. On the **Scripts** pane at the bottom left, click **New**. A new script appears in the list. Double-click on it to open it.

2. Change the name of the script, so that it reflects what the script does.
3. Choose which kind of **selector** you want to use. Running a script starts with searching the template for pieces of content that match the script's selector. The collected pieces of content are passed on to the script, so that the script can modify them.

The selector can be:

- **Text**, for example: @lastname@, or {sender}. The text doesn't have to have any special characters, but special characters do make them easier to recognize for yourself. In the Script Wizard, click **Text** and type the text to find.
- A **selector** (HTML/CSS):
  - HTML elements of a certain type, such as a paragraph: <p>. In the script Wizard, click **Selector** and type the HTML tag without the angle brackets: p.
  - HTML elements with a specific CSS class (.green). In the script Wizard, click **Selector** and type the class name, including the preceding dot: .green.
  - An HTML element with a specific ID (#intro). In the script Wizard, click **Selector** and type the ID, including the preceding #: #intro. In an HTML file, each ID should be unique. This means that a particular ID can be used only once in each section.
  - Etcetera. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for more selectors and combinations of selectors.
- A **selector and text**. This is text inside an HTML element (or several HTML elements) with a specific HTML tag, CSS class or ID. In the script Wizard, click **Selector and Text**.

### Tip

When output speed matters, choose **selector** or **selector and text**. Searching text is a rather lengthy operation, compared to searching for HTML elements and/or CSS selectors. See also profiling scripts.

There is a shorter route to create a script for an element with a specific ID:

1. In the template, click the element for which you want to create a script.
2. On the **Attributes** pane at the top right, type an ID. (In HTML, IDs start with #, but in this field you should type it without the preceding #).

3. Click the label to the left of the ID input field (ID) to make a new script with the ID that you typed as a selector.

## Writing a script

1. Create a new script (see: "Creating a new script" on page 224), or double-click an existing script in the **Scripts** pane on the bottom left.  
If the script was made with a Script Wizard, you have to click the **Expand** button before you can start writing code. This will change the Script Wizard into an editor window.

### Warning

When you change an expanded text script and save it, it becomes impossible to edit the script using the Script Wizard again.

2. Write the script. Click **Apply** from time to time to see if the script works as expected. This will be visible on the **Preview** tab in the main workspace.

### Syntax rules

Every script in the Designer must follow JavaScript syntax rules. For example, each statement should end with ; and the keywords that can be used, such as **var** to declare a variable, are JavaScript keywords. There are countless tutorials available on the Internet to familiarize yourself with the JavaScript syntax. For a simple script all that you need to know can be found on the following web pages: [http://www.w3schools.com/js/js\\_syntax.asp](http://www.w3schools.com/js/js_syntax.asp) and [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp).

### Tip

In the editor window, press **Ctrl+Space** to see the available features and their descriptions.

Use the arrow keys to select a function or object and press enter to insert it in the script.

Type a dot after the name of the function or object and press **Ctrl+Space** again to see which features are subsequently available.

## Two basic code examples

Writing a script generally comes down to modifying the piece(s) of content collected from the template with the script's selector, using values, or depending on values of the record that is being merged to the template at the moment the script runs.

### Modifying the template

To access and change the results of the query that is carried out with the selector (in other words: to modify the output), use the object **results**.

The following script (with the selector **p**) changes the text-color of all paragraphs to red with a single line of code:

```
results.css('color', 'red')
```

It does this for each and every customer, because it does not depend on a value from the record that is being merged to the template.

### Using values from the record in a script

To access the record that is being merged to the template when the script runs, use the object **record**.

Suppose you want to display negative amounts in red and positive amounts in green.

Assuming that there is an AMOUNT field in your customer data, you could write the following script (with the selector: **td.amount**, that is: table cells with the class 'amount').

```
var amount = record.fields.AMOUNT;
if (amount >= 0)
    {results.css('color', 'green');}
else if (amount < 0) {
    results.css('color', 'red');
}
```

When this script executes, it stores the value of the AMOUNT field from the current record in a variable and evaluates it. If the value is zero or higher, the color of text in the **results** - in this case they are cells with the CSS class 'amount' - will be set to green; if the value is below zero, the text color will be set to red.

#### Tip

For more examples of using conditions, see this how-to: [Combining record-based conditions](#).

## Designer API

Features like **results** and **record** do not exist in the native JavaScript library. These are additional JavaScript features, designed for use in Connect scripts only. All features designed for use in the Designer are listed in the Designer's API, with a lot of examples; see "Designer Scripts API" on page 429.

# Managing scripts

## Order of execution

When a record set is merged with a template to generate output, all scripts are executed once for every record in the record set, in the order in which they appear in the **Scripts** pane at the bottom left.

The order in which scripts are executed is particularly important when one script produces content that contains a selector for another script. If the other script has already been executed, it will not run again automatically. So, scripts that produce content that contains one or more selectors for other scripts, need to come first.

To change the order in which scripts are executed:

- Click a script or a folder in the **Scripts** pane at the bottom. Drag it up or down and drop it.

### Note

Control scripts are always executed first, regardless of where they are in the Scripts pane. They can not be excluded from execution for a specific context or section, using the execution scope of a folder; see "Execution scope" on the next page. What you can do is disable the script or the containing folder; see "Enable/disable scripts" on page 230.

## Script folders

Scripts can be organized in folders. Why would you do that? For three reasons:

- Folders have an execution scope. You can specify for which contexts and sections the scripts in a folder have to run.
- Folders provide a better overview than a long unorganized list of scripts.

- Folders make it easier to change the order of execution for a bunch of scripts (see: "Order of execution" on the previous page to learn why the order of execution is important). Dragging a folder up or down will cause all the scripts in that folder to be executed earlier or later, respectively.

To make a new folder on the **Scripts** pane:

1. In the **Scripts** pane, click the black triangle on the **New** button.
2. Click **Folder**. The folder will appear in the list of scripts.
3. Change the name of the new folder: right-click the folder and click **Rename**.
4. Drag scripts to the folder.

#### **Tip**

It may be helpful to put scripts that have an effect on the same context or section in one folder, because you can set the execution scope of scripts per folder (see: "Execution scope" below).

#### **Note**

Control scripts are always executed first, regardless of where they are in the Scripts pane. They can not be excluded from execution for a specific context or section, using the execution scope of a folder; see "Execution scope" below. What you can do is disable the script or the containing folder; see "Enable/disable scripts" on the facing page.

### **Execution scope**

A particular script may be used in one context or section, but not in other contexts or sections. Nevertheless, when processing the template, the Designer tries to find the selector of each script in all contexts and sections – unless the script is located in a scripts folder for which the execution scope has been set to the relevant contexts or sections. So, setting the execution scope of a folder saves processing time.

To change the execution scope of a script:

1. Put the script in a folder; see "Managing scripts" on page 228.
2. Right-click the folder, and then click **Properties**.
3. Check the contexts and sections for which the scripts in this folder should run.

#### Note

Control scripts are always executed first, regardless of where they are in the Scripts pane. They can not be excluded from execution for a specific context or section, using the execution scope of a folder; see "Execution scope" on the previous page. What you can do is disable the script or the containing folder; see "Enable/disable scripts" below.

#### Tip

For more ways to optimize scripts, see "Optimizing scripts" on page 234.

### Enable/disable scripts

A disabled script will not run at all when the template is merged with a record set to generate output. Disabling script execution in certain contexts or sections helps with performance, since scripts normally run, whether or not their placeholder or selector is present in your template. It is highly recommended to disable any script that is not relevant to specific sections or contexts.

When you disable a folder, all scripts in the folder will be disabled.

To enable or disable a script or a folder:

- On the **Scripts** pane, right-click the script or the folder and click **Disable** (if the script or folder was enabled) or **Enable** (if the script or folder was disabled).

#### Tip

For more ways to optimize scripts, see "Optimizing scripts" on page 234.

### Import/export scripts

Scripts can be exported - one at a time - for use in other templates. To do this:

1. On the **Scripts**pane, click on a script, and then click the **Export**button, or right-click a script and select **Export**.
2. Give the script a name and click **OK**.

To import a script in a template:

- On the **Scripts**pane, click the **Import**button. Find the script and click **OK**

Files that a script may refer to, such as images, snippets and fonts, are not exported or imported together with a script.

Test the script to make sure that all files are present in the template and that the script's selector matches something in the content of the template; see "Testing scripts" below.

## Testing scripts

The quickest way to test that scripts work as expected, is to click the **Preview** tab at the bottom of the workspace.

You can even do this while creating a new script, either with a Script Wizard or in the expanded script editor. Click **Apply** at the bottom of the script editor to see the effect of the script on the **Preview** tab of the Designer.

Note that scripts that use values of data fields can only be effective when a data file is open. See "Loading data" on page 204.

### Test for errors

Another way to 'test' a script is to take a look at the Scripts pane.

#### Tip

Hover over the name of a script in the **Scripts** pane to highlight parts of the template that are affected by the script.

Icons on the name of scripts in the **Scripts** pane can show a warning, information or error icon.

 Spa Location	#spa-location
 Promo	#promo1
 Date	@Date@
 Year	@Year@
  Address	
 Employee	@EMP@

- The information icon  (i) shows that the selector of the script does not produce a result in the current section.
- The warning icon  (!) appears, for example, when a script refers to an unknown field in the record set, or when ; is missing after a statement.
- The error icon  (x) displays when the script results in an error, for example, when it uses an undeclared variable.

In addition to the icons and messages in the Scripts pane, there is another way to see if your scripts function as expected before generating output:

- On the **Context** menu, click **Preflight**.

**Preflight** executes the template without actually producing output and it displays any issues once it's done.

It will tell, for example, which selectors were not encountered in the template.

### Test for speed issues

To measure the time that the execution of scripts will take:

- On the **Context** menu, click **Profile scripts**.

Profiling means running the scripts in the template, to see how fast scripts in the **Scripts** pane execute. It helps greatly in troubleshooting performance issues caused by scripts.

After running the Script Profiler you can see in which sections the script has run:

- Hover the mouse over a value in the column **Count** to see the number of times that the script has run, per section.

You can also see the breakdown of the execution time across different execution stages:

- Hover the mouse over a value in the column **Elapsed** to see the time elapsed (in milliseconds) since the start of the session. In the Scripts Profiler, the scripts are by default sorted based on the values in the **Elapsed** column, from high to low.
- Hover the mouse over a value the column **Delta** to see the difference between the time elapsed (in milliseconds) in the previous session and in the current session.

The script execution stages are:

**Query:** the time it takes to find the selector in the template.

#### **Tip**

Looking for text is a rather lengthy operation. Use an ID (possibly in combination with a text) instead of a text selector to make the query faster. For more tips, see "Optimizing scripts" on the facing page.

**Execution:** the time it takes to execute the script. If you are an experienced JavaScript coder you may be able to optimize the code to speed up the execution of the script.

#### **Tip**

Functions that actually change the content of the template (for example, **append()**) are comparatively time consuming. Avoid using such functions in a loop. For more tips, see "Optimizing scripts" on the facing page.

Note that the times vary slightly per run of the Script Profiler. Run the Script Profiler a number of times and calculate an average from the results, before trying to speed up the execution of a script.

## **Script Profiler settings**

### **Number of runs**

By default, the Script Profiler runs on 1000 instances of all the scripts. To test on a higher or lower number of instances:

1. On the menu, select **Window > Preferences**.
2. Click **Scripting**.
3. Set a number of iterations (maximum one billion) and click OK.

## Sorting

In the Scripts Profiler, the scripts are by default sorted based on the values in the **Elapsed** column, from high to low. Click any of the columns to sort the scripts according to the values in that column.

## Optimizing scripts

In the process of output generation, the execution of scripts may take up more time than necessary. To optimize a template, it helps to disable scripts that don't have an effect on the output; see "Managing scripts" on page 228.

This topic presents a number of other ways to speed up script execution by optimizing the scripts.

### Use an ID as selector

Scripts (except control scripts) start with a query. The **selector** in the second column in the **Scripts** pane is what a script looks for in the template. If you've used the drag-and-drop method (without pressing the Alt or Ctrl key) to insert a data field in a template, the selector is a small text: the name of the data field surrounded by @ signs, @firstname@ for example.

Looking for text in a text is a less optimized operation and may impact output speeds in longer documents. To speed up the output process, point the script to the element that contains the placeholder, by using its ID as selector. This narrows the scope of the search and results in a very fast query, as elements with an ID are indexed by Connect Designer's layout engine.

To learn how to put a placeholder or placeholders inside an element that has an ID, see "Boxes" on page 150. To use that ID as the script's selector: double-click the script in the Scripts pane and change the Find method to **Selector and Text**, or to **Selector** if the placeholder is the only content of the container. Enter the ID of the wrapper element in the **Selector** field, preceded by #, for example: #firstname.

## Tip

When using the drag-and-drop method to insert data fields in a template:

- Press the **Alt** key while dragging, to wrap the placeholder in a **span**, give the span an **ID** and have that ID used as the script's selector.
- Press the **Ctrl** key while dragging, to wrap the placeholder in an absolute positioned box (a **div**) at the cursor position. A unique ID is assigned to the box and used as the script's selector.

## Avoid DOM manipulations

The Scripting API of the Designer is a very powerful tool to manipulate and personalize your document. But keep in mind that DOM manipulation commands like `append()`, `prepend()`, `before()` and `after()` are resource intensive.

Try avoiding DOM modifications, especially within loops. Storing the content in a variable and appending the information after the loop is more efficient: this way, the template will be touched only once.

## Example

The following example loads a snippet into a variable and uses the `find()` and `text()` commands of the Designer scripting API.

```
var labelElm = loadhtml('snippets/label.html');
for(var i = 0; i < record.tables.products.length; i++) {
    var label = labelElm.clone();
    label.find('@ProductLabel@').text(record.tables.products
[i].ProductDescription);
    results.after(label);
}
```

What's wrong with this code is that it inserts the personalized information **within** the loop. The `after()` command runs as many times as there are records in the detail table 'products'.

The script below is much more efficient: it adds the personalized content to a string called `labelStr` and only calls `after()` after the `for` loop.

```

var labelElm = loadhtml('snippets/label.html');
var labelStr = "";
for( var i = 0; i < record.tables.products.length; i++) {
    var label = labelElm.clone();
    label.find('@ProductLabel@').text(record.tables.products
[i].ProductDescription);
    labelStr += label;
}
results.after(labelStr);

```

## Use replace()

When personalizing HTML fragments retrieved from a snippet or from the template itself, JavaScript's `replace()` method shows the best performance.

`Replace()` can only be used on Strings, while the commands `loadhtml()` and `query()` return or a `QueryResult`, which is a set of strings, like the `results` object.

A `QueryResult` allows you to perform DOM manipulations like adding and removing elements, adding and removing CSS classes etc. When the required manipulations are limited to find/replace actions, you could change the `QueryResult` into a string. This allows you to replace text using the `replace()` method.

For this, you could use `toString()`:

```
var labelSnippet = loadhtml('snippets/label.html').toString();
```

Or you could copy the HTML of the `QueryResults` to a variable:

```
var block = results.html();
```

## Example

```

var labelSnippet = loadhtml('snippets/label.html').toString();
var labelStr = "";
for( var i = 0; i < record.tables.detail.length; i++) {
    var label = labelSnippet;
    label = label.replace('#', i);
    label = label.replace('@product@', record.tables.detail[i].fields
['product']);
    label = label.replace('@notes@', record.tables.detail[i].fields
['notes']);
}

```

```
        label = label.replace('@netweight@', record.tables.detail
[i].fields['netweight']);
        labelStr += label;
    }
results.after(labelStr);
```

### Tip

The `replace()` method as used in the above example replaces only the first occurrence of the search string. To **replace every occurrence** of a search string in a given string, use a **regular expression**. In the following line of code, the regular expression `/@product@/g` makes `replace()` search for all occurrences of the string `@product@` in the `label` string:

```
label = label.replace(/@product@/g, record.tables.detail
[i].fields['product']);
```

In this example, `@product@` is a pattern (to be used in a search) and `g` is a modifier (to find all matches rather than stopping after the first match). For more information about possible regular expressions, see [http://www.w3schools.com/js/js\\_regexp.asp](http://www.w3schools.com/js/js_regexp.asp).

## Replace several placeholders in one script

Suppose there are 20 different placeholders in a postcard (for the address, account and customer details, a promo code, the due date, discounts, a link to a personalized landing page etc.). Typically this would require 20 queries. Even after optimizing these scripts by using an ID as selector for those scripts, there are still 20 scripts, 20 queries to run.

If there was only one query, one single script to do all the work, the output could be generated much faster. Reducing the number of scripts improves the performance of the template. How to do this?

First, wrap the content that contains all of the placeholders in one (inline) Box and give that Box or Span an ID (on the Attributes pane). Next, create a script that uses that ID as selector. Then replace all placeholders in the script and put the content back in the template.

This is similar to working with snippets, but in this case the element is extracted from the actual template.

## Example

The following script replaces all of the placeholders on a postcard. It takes advantage of the JavaScript `replace()` command. Assuming that the ID of the block that requires personalization is `promoblock`, the script has to have its selector set to `#promoblock`.

```
var block = results.html();
var data = record.fields;
block = block.replace('@name@', data.first + ' ' + data.last);
block = block.replace('@address@', data.address);
block = block.replace('@zip@', data.zip);
block = block.replace('@city@', data.city);
block = block.replace('@country@', data.country);
block = block.replace('@saldo@', data.saldo);
block = block.replace('@promo@', data.promo);
block = block.replace('@customercode@', data.customercode);
...
results.html(block);
```

The first line retrieves the HTML of the promo block and stores it in a variable called `block`. To make the code more readable, the fields from the record are stored in a variable named `data`. After replacing the placeholders by values, the script replaces the HTML of the `promoblock` with the personalized string.

## Loading a snippet via a script

Instead of dragging it into the content directly, it is possible, and often very useful, to load a snippet dynamically. Create a script (see "Writing your own scripts" on page 223) and in the code, use the following function:

```
results.loadhtml('snippets/nameofthesnippet.html')
```

This function will insert the snippet in the content at any position where the script's selector is encountered.

For more examples, see "loadhtml()" on page 459.

### Note

Make sure that the file name is exactly the same as the file in the **Snippets** folder. If the

file name isn't correct, the snippet will not appear in the template.

### Loading part of a snippet

When a snippet contains a part that can be identified by a selector, that selector can be used to load that part of the snippet into a template.

In script, use the following code:

```
results.loadhtml('snippets/nameofthesnippet.html', 'selector')
```

See "loadhtml()" on page 459 for more information about this function.

### Loading a snippet, depending on the value of a data field

To load a snippet depending on the value of a data field, you have to add a condition to the script.

#### Example

The following script evaluates if the value of the LANGUAGE field in the record is 'En'. If so, the snippet is added to the content.

```
if (record.fields.LANGUAGE == 'En') {  
  results.loadhtml('snippets/nameofthesnippet.html');  
}
```

Another example is given in a how-to; see [Load a snippet based on a data field value](#).

### Loading part of a snippet, based on the value of a data field

When a snippet contains a part that can be identified by a selector, that selector can be used to load that part of the snippet into a template. It is possible to do this, based on the value of the data field. This is easiest when the selector matches the value of a data field.

#### Example

The following script reads the value of the LANGUAGE field in the record and uses that value as the selector in the function loadhtml(). If the snippet contains an HTML element with this ID (for example, <p ID="En">), that HTML element will be added to the content:

```
var language = record.fields.LANGUAGE;
results.loadhtml('snippets/nameofthesnippet.html', '#' + language)
```

Another example is given in the following how-to: [Using a selector to load part of a snippet](#).

See also: "Designer Scripts API" on page 429.

### Tip

An easy way to group content in a snippet is putting each part in a container and giving that container an ID, for example:

```
<div ID="EN"><p>This is text for English customers.</p></div>
```

Use the function `.children()` to load the contents of the container, and not the container itself. For example:

```
results.loadhtml('Snippets/myfooter.html', '#EN').children()
```

This script loads the paragraph of the example (`<p>`), but not the container itself (`<div>`).

## Load a snippet and insert variable data into it

The following script loads part of a snippet based on the value of a field, and then finds/replaces text by the value of a field before inserting the content into the document.

```
var promoTxt = loadhtml('snippets/promo-en.html', '#' +
record.fields['YOGA']);
promoTxt.find('@first@').text(record.fields['FIRSTNAME']);
results.html(promoTxt);
```

## Control Scripts

When output is generated from a template, Control Scripts run **before** all other scripts, when a record is merged with a context. They determine how different sections of the context are handled. They can, for example, make the page numbering continue over all Print sections, split Email attachments, or omit Print sections from the output.

Some knowledge of JavaScript is needed to edit Control Scripts, just as for any other self-made scripts, because there is no Control Script Wizard; see "Writing your own scripts" on page 223.

This topic explains how to add a Control Script and it gives an overview of what Control Scripts can do. It will also tell you where you will find information about each feature, including examples.

## What Control Scripts are

Control Scripts are a special kind of Designer script. They can manipulate the way output is generated from a template. They allow you, for example, to change the page numbering in Print output, to split one generated Print document into multiple Email attachments, or to set a Print section's background dynamically. (These are only a few examples; for more uses of Control Scripts see "What to use a Control Script for" on the facing page.)

Control Scripts differ from template scripts in two ways:

- Control Scripts run before all other scripts. When a template consists of several contexts, and these contexts are combined in the output - for example, when an Email is generated with the Print context as attachment - all scripts run once before each context; Control Scripts first.
- Control Scripts don't have a selector, like the other scripts do. A selector selects parts of the content of a section and stores them in the results object, so that they can be modified in the script. As Control Scripts don't have a selector, the results object can't be used there. Control Scripts don't touch the content - meaning, the text flow - of the sections.

## Adding a Control Script

To add a Control Script:

1. On the **Scripts** pane at the bottom left, click the black triangle on the **New** button and click **New Control Script**. A new script appears in the list.
2. Double-click the new script to open it. The script editor appears.
3. Change the name of the script so that it reflects what the script does.
4. Write the script; see the "Control Script API" on page 477. If you are not familiar with scripting, also see "Writing your own scripts" on page 223.

### Tip

New Control Scripts added to the template contain code to continue the page numbering over all print sections, and two examples: one to select different sections

of a Print context for email and print output, and one to select a web section.

## What to use a Control Script for

Control Scripts let you change the way a template is merged, by giving access to the template with all its contexts and sections in a script. A Control Script may, for example, omit, group and clone sections; add a background to a Print section; or add a header to an email. A number of the things that you can do with them is listed in the table below, with a link to a topic that explains how to do them and that shows what the script should look like.

Control Scripts differ from template scripts in two ways:

- Control Scripts run **before** all other scripts. When a template consists of several contexts, and these contexts are combined in the output - for example, when an Email is generated with the Print context as attachment - all scripts run once for each context, but Control Scripts always go first.
- Control Scripts **don't** touch the content - meaning, the text flow - of the sections. They don't have a selector, like the other scripts do. A selector selects parts of the content of a section and stores them in the `results` object, so that they can be modified in the script. As Control Scripts don't have a selector, the `results` object can't be used there.

In a Control Script, `section` usually is the most important object. To get a quick overview and lots of examples, see "section" on page 480. For help on specific tasks, see the table below.

Task	See topic	Field/function of <code>section</code> object
Change the page numbering of Print sections	"Control Script: Page numbering" on the next page	<code>restartPageNumbering</code>
Set the background image of a Print section	"Control Script: Setting a Print section's background" on page 248	<code>background.source</code> , <code>background.url</code> , <code>background.position</code>
Split Email attachments and rename	"Parts: splitting and	<code>part</code>

Task	See topic	Field/function of section object
them	renaming email attachments" on page 246	
Dynamically set a password on PDF attachments	"Control Script: Securing PDF attachments" on page 252	password, ownerPassword
<p>Include/exclude sections:</p> <ul style="list-style-type: none"> <li>• Conditionally omit sections</li> <li>• Output one section or another, based on the value of a data field</li> <li>• Select one print section as PDF attachment if the output is to be emailed, and another print section if the output is to be printed.</li> </ul>	"section" on page 480, take a look at the examples.	enabled
Add sections dynamically	"Dynamically adding sections (cloning)" on page 250.	clone()
Add a header to an email	"section" on page 480, example: "Adding custom ESP handling instructions" on page 515.	headers

### Control Script: Page numbering

This topic explains how to write a Control Script that changes the page numbering in Print sections. Note that when you add a Control Script, it already contains a script to make the page numbering continue over all Print sections.

For information about Control Scripts in general, see "Control Scripts" on page 240 and "Control Script API" on page 477. If you don't know how to write scripts, see "Writing your own scripts" on page 223.

## How to change page numbering in a control script

A Control Script can make the page numbering continue over all Print sections or let it restart on a section. This is done by setting the `restartPageNumber` field on a section to `true` or `false`. For example: `merge.template.contexts.PRINT.sections['Section 2'].restartPageNumber = true;` (Also see "section" on page 480 and "Control Script API" on page 477.)

Page numbering starts with page 1 for each section. If for a section `restartPageNumber` is set to `false`, that section will start with the page number following the last page of the previous section.

Note that even if a section is not enabled (so it will not be outputted), its `restartPageNumber` flag is still taken into account for composing the page number sequences.

By default, each section has `restartPageNumber = false` when the first control script runs.

### Tip

If you are looking to create a table of contents, add a template script that uses the `pageRef()` function. For an example, see "pageRef()" on page 462.

## Examples

### Restarting the page numbers several times

Assume that a template has four sections (of 1 page each) in the Print context and a Control Script sets the page numbering as follows:

1. Section A (1 page) `restartPageNumber = true`
2. Section B (1 page) `restartPageNumber = true`
3. Section C (1 page) `restartPageNumber = false`
4. Section D (1 page) `restartPageNumber = true`

The code would look like this:

```
if (merge.context.type == ContextType.PRINT) {
    merge.context.sections['Section A'].restartPageNumber = true;
    merge.context.sections['Section B'].restartPageNumber = true;
    merge.context.sections['Section C'].restartPageNumber = false;
    merge.context.sections['Section D'].restartPageNumber = true;
}
```

The page numbering in the output will be:

1. Section A page 1
2. Section B page 1
3. Section C page 2
4. Section D page 1

### Disabled section

When a section is disabled, it will not be outputted, but its `restartPageNumber` flag will still be taken into account for composing the page number sequences. So, if the `restartPageNumber` flags are set as follows:

1. Section A (1 page) `restartPageNumber = true`
2. Section B (2 pages) `restartPageNumber = false`
3. Section C (3 pages) `restartPageNumber = true, enabled = false`
4. Section D (4 pages) `restartPageNumber = false`

In code:

```
if (merge.context.type == ContextType.PRINT) {
    merge.context.sections['Section A'].restartPageNumber = true;
    merge.context.sections['Section B'].restartPageNumber = false;
    merge.context.sections['Section C'].restartPageNumber = true;
    merge.context.sections['Section C'].enabled = false;
    merge.context.sections['Section D'].restartPageNumber = false;
}
```

The page numbering in the output will be:

1. Section A page 1
2. Section B page 2

3. Section D page 1 (page numbering is restarted due to section C's restartPageNumber = true)

## Parts: splitting and renaming email attachments

In a Control Script, **parts** can be defined to determine which sections should be output to the same file. This way it is possible to split the Print context or the Web context into multiple email attachments. This topic shows how to do that.

For information about Control Scripts in general, see "Control Scripts" on page 240 and "Control Script API" on page 477. If you don't know how to write scripts, see "Writing your own scripts" on page 223.

### Defining parts

Defining parts is done by setting the `part` field on a `section`, for example:

```
merge.template.contexts.PRINT.sections['Section 2'].part = "PDF_Attachment2";. (Also see "section" on page 480 and "Control Script API" on page 477.)
```

- If a part name is given, then that delimits the start of a new part (even if the part name is the same as the previous one). Following sections that don't define a part name, will be added to the previous part.
- A part ends at the last enabled\* section or at the last section before the start of a new part.  
\*When a Control Script has set the `enabled` field of a `section` to `false`, it will not be outputted.

If no part name is set on any section, it is assumed that there is only one part, consisting of the default section (for Web output) or of all sections (for Print output). The attachment(s) will be named after the email subject.

### Examples

#### No parts defined

Assume there are three Print sections: sections A, B and C. When generating Email output with the Print context as attachment, all three Print sections will be put together in one file and attached to the email. If the email's subject is 'Take action', the name of the attached file will be 'Take action.PDF'.

## Splitting and renaming a Print attachment

Assume there are three Print sections: sections A, B and C. In a Control Script a part name is defined for section C:

```
var section = merge.template.contexts.PRINT.sections['Section C'];
section.part = 'Part2';
```

When generating Email output with the Print context as attachment, the email will have two attachments:

- attachment 1: Section A, Section B
- attachment 2: "part2", which is Section C. The file name of this attachment is the part name.

### **Note**

For Web sections, a part always consists of only the given section. Web pages cannot be appended to form a single part. It is however possible to attach multiple Web pages to one email; see "Control Script API" on page 477 for an example.

## Controlling multiple Email attachments

The following script attaches the following sections to an email:

- Print section 3 + 4 as attachment with continued page numbers
- Print section 6 as separate attachment
- Web sections A and B as separate attachments

```
if (channel == Channel.EMAIL) { // only when generating Email
output
if (merge.context.type == ContextType.PRINT) {
    merge.context.sections['Section 1'].enabled = false;
    merge.context.sections['Section 2'].enabled = false;
    merge.context.sections['Section 3'].enabled = true;
    merge.context.sections['Section 3'].part = "PDFAttach1";
    merge.context.sections['Section 4'].enabled = true;
    merge.context.sections['Section 4'].restartPageNumber = false;
    merge.context.sections['Section 5'].enabled = false;
    merge.context.sections['Section 6'].enabled = true;
    merge.context.sections['Section 6'].part = "PDFAttach2";
```

```
} else if (merge.context.type == ContextType.WEB) {
    merge.context.sections['default Section'].enabled = false; //
disable whatever is the default section
    merge.context.sections['Section A'].enabled = true;
    merge.context.sections['Section A'].part = "WebPartA";
    merge.context.sections['Section B'].enabled = true;
    merge.context.sections['Section B'].part = "WebPartB";
}
}
```

### Note

For another example, see this how-to: [Output sections conditionally](#).

## Control Script: Setting a Print section's background

In the Print context, a PDF file can be used as a Print section's background. To learn how to do this without a Control Script, see "Using a PDF file as background image" on page 65. With a Control Script, a Print section's background can be set dynamically. You could for example specify a particular PDF file as a section's background depending on the value of a field in the current record. This topic shows how.

For information about Control Scripts in general, see "Control Scripts" on page 240 and "Control Script API" on page 477. If you don't know how to write scripts, see "Writing your own scripts" on page 223.

### Setting a background in script

The Control Script should first enable a background on the section, in case an initial background wasn't set via the user interface. This is done by setting the source type for the background of the section to either DataMapper PDF or Resource PDF (see "BackgroundResource" on page 488). For example:

```
merge.template.contexts.PRINT.sections['Policy'].background.source
= BackgroundResource.RESOURCE_PDF;
```

A DataMapper PDF is, as you would expect, a PDF generated by the DataMapper. A Resource PDF is a PDF from another source.

For a DataMapper PDF, nothing else has to be done to set the background. For a PDF from another source, the Control Script should specify a path, for example:

```
var resourceUrl = 'images/policy-' + record.fields.policy + '.pdf';
merge.template.contexts.PRINT.sections['Policy'].background.url =
resourceUrl;
```

## Positioning the background

After that, the background can be positioned, setting the section's `background.position`:

```
activeSection.background.position = MediaPosition.FIT_TO_MEDIA;
```

For all possible positions, see "MediaPosition" on page 491.

## Setting a page range in script

When a PDF that serves as a dynamic section background has multiple pages, you can specify a range of pages to be used, in a control script.

Put the number of the first page in the range in the section's `background.start` field and the last page in `background.end`.

This requires you to set the `background.allPages` option to `false`, first. This option is `true` by default and takes precedence, so when it is `true`, the entire PDF will be used, even if a page range has been set.

The following script sets the page range from 2 to 5:

```
merge.template.contexts.PRINT.sections
['Policy'].background.allPages = false;
merge.template.contexts.PRINT.sections['Policy'].background.start =
2;
merge.template.contexts.PRINT.sections['Policy'].background.start =
5;
```

## Example

This script sets a background on a Print section using absolute positioning.

```
var activeSection = merge.template.contexts.PRINT.sections['Section
1'];
activeSection.background.source = BackgroundResource.RESOURCE_PDF;
activeSection.background.url = "images/somepage.pdf";
activeSection.background.position = MediaPosition.ABSOLUTE;
```

```
activeSection.background.left = "10mm";
activeSection.background.top = "10mm";
```

You could replace the last three lines of the previous script by the following line to scale the Print section background to Media size:

```
activeSection.background.position = MediaPosition.FIT_TO_MEDIA;
```

## Dynamically adding sections (cloning)

This topic explains how to clone a section in a Control Script, Print sections can be cloned, so that a document can have a dynamic number of sections, based on data. This is particularly useful when the record set defines one or more PDFs (e.g. insurance policies) per recipient. Via a Control Script, for each PDF a section can be cloned and each clone can be given one of the PDFs as background (see "Control Script: Setting a Print section's background" on page 248). For each page in the PDF, a page will be added to the section.

For information about Control Scripts in general, see "Control Scripts" on page 240 and "Control Script API" on page 477. If you don't know how to write scripts, see "Writing your own scripts" on page 223.

## Cloning a section

To clone a section, first use the `clone()` function and then add the clone to the Print context before or after a specific section, using `addAfter()` or `addBefore()`:

```
var clone = printSections["Section 1"].clone();
printSections["Section 1"].addAfter(clone);
```

Cloned sections have the same properties as normal sections, but they cannot call `section` functions.

### Note

Due to resource constraints, the number of clones that can be created is limited to 60.

## Renaming a clone

By default, clones receive the name of their source section with a "Clone {sequence}" suffix, for example:

Source: "Section 1"

Clone Name: "Section 1 Clone 1"

Use the `name` property to assign the cloned section another name, for example:

```
clone.name = "my_section_clone";
```

Note that section names must be unique.

### Targeting elements in a cloned section

As each clone receives a unique section name, one could use CSS style sheets (see "Styling and formatting" on page 170) and personalization scripts (see "Variable Data" on page 211 and "Writing your own scripts" on page 223) to further personalize the cloned sections.

The following CSS style rules target the `<h1>` element in a number of clones and assigns the respective text a different color:

```
[section="my_section_clone_0"] h1 { color: red; }
[section="my_section_clone_1"] h1 { color: green; }
[section="my_section_clone_2"] h1 { color: blue; }
```

The same selectors could be used in personalization scripts:

**Selector:** `[section="my_section_clone_0"] h1`

**Script:** `results.css('color', 'red');`

In a template script, cloned sections can be found using `merge.section`:

```
if (merge.section == "my_section_clone_0") {
    results.html("Clone!");
} else {
    results.html("Original.");
}
```

Note that in a Control Script, `merge.section` is only defined when the output channel is WEB; see "merge" on page 479.

## Examples

### Cloning a section based on the number of records in a detail table

This script creates as many clones of a section as there are records in a detail table. It assigns the new sections a unique name.

```
var printSections = merge.template.contexts.PRINT.sections;
var numClones = record.tables['detail'].length;
for( var i = 0; i < numClones; i++){
    var clone = printSections["Section 1"].clone();
    clone.name = "my_section_clone_" + i;
    printSections["Section 1"].addAfter(clone);
}
```

### Cloning a section based on data and assigning a background PDF

This script clones a section based on data fields. It disables the source section first and then calls the `addPolicy` function. `addPolicy` clones the section, renames it and sets a PDF from the resources as its background. It explicitly enables the clone and then adds it to the Print context.

```
var printSections = merge.template.contexts.PRINT.sections;
merge.template.contexts.PRINT.sections["Policy"].enabled = false;
if(record.fields.policy_a == 1) {
    addPolicy('a');
}
if(record.fields.policy_b == 1) {
    addPolicy('b');
}
function addPolicy(policy){
    var resourceUrl = 'images/policy-' + policy + '.pdf';
    var clone = printSections["Policy"].clone();
    clone.name = "policy_" + policy;
    clone.background.url = resourceUrl;
    clone.enabled = true;
    printSections["Policy"].addAfter(clone);
}
```

### Control Script: Securing PDF attachments

The Print context can be attached to an email in the form of a PDF file and secured with a password. This can be done without a Control Script, see "Email attachments" on page 511 and "Email PDF password" on page 101.

With a Control Script, you can do the same, and more: the attachment can be split into multiple

attachments (see Parts). Each attachment may have a different (or no) set of passwords, so you could mix secured and unsecured attachments in a single email. This topic shows how.

For information about Control Scripts in general, see "Control Scripts" on page 240 and "Control Script API" on page 477. If you don't know how to write scripts, see "Writing your own scripts" on page 223.

### Setting passwords in script

To set a password on a Print section in a Control Script, the script should first retrieve the Print section/s using `merge.template.contexts.PRINT.sections` or `merge.context.sections` (also see the example below).

Next, the script can split the attachments, if needed (see "Parts: splitting and renaming email attachments" on page 246), and it can set a password on each `section`. For example:

- `merge.template.contexts.PRINT.sections['Section 2'].password = 'secret';`
- `merge.template.contexts.PRINT.sections['Section 2'].ownerPassword = 'secret';`

When producing a **single** attachment, the password(s) should be set on the first Print section. When producing **multiple** attachments, it should be set on the first section of each part.

### Password types

PDF allows for two types of passwords to be set on a secured PDF file: a user password and owner password. The user password allows a limited access to the file (e.g. printing or copying text from the PDF is not allowed). The owner password allows normal access to the file. The Email PDF password script sets both the user and owner password to the same value, so that when the recipient provides the password, he can manipulate the file without limitations.

In a Control Script:

- `password` is used to set the user password and owner password for a PDF attachment to the same value.
- `ownerPassword` is used to set the owner password for a PDF attachment. Setting only the owner password creates a secured PDF that can be freely viewed, but cannot be manipulated unless the owner password is provided. Note that the recipient needs Adobe Acrobat to do this, because the Acrobat Reader does not allow users to enter the owner password.

## Removing a password

Passwords set in the Control Script override the password set through the Email PDF password script (see "Email PDF password" on page 101). This allows you to change or remove the password from a specific part. Removal is done by setting the `password` field to `null` or an empty string ("").

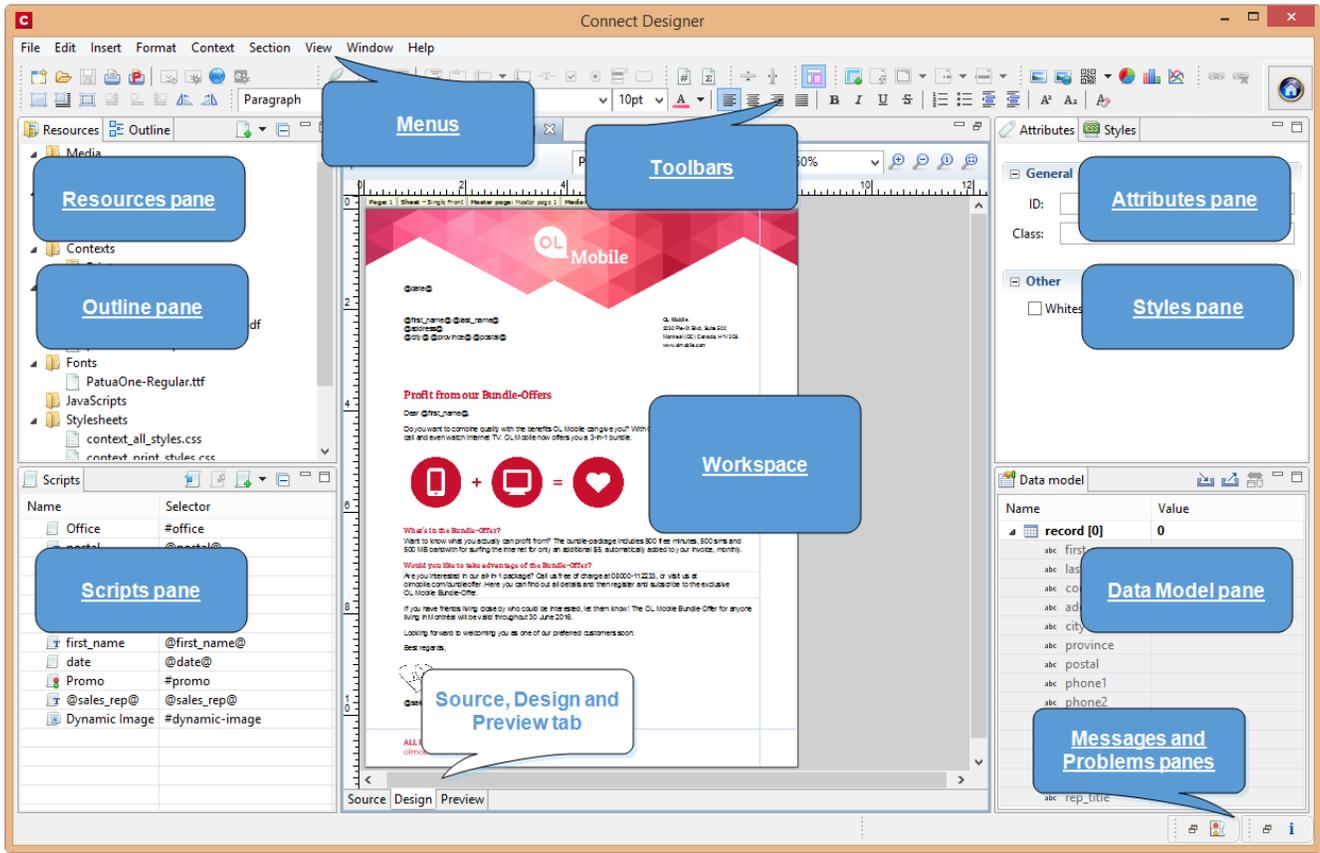
### Example

This scripts splits the Print output into two PDF attachments and sets a password for the second attachment.

```
var printSections;
if (channel == Channel.EMAIL) { // only when generating Email
output
if (merge.context.type == ContextType.PRINT) {
printSections = merge.template.contexts.PRINT.sections;
    printSections['Section 1'].part = 'PDFAttach1';
    printSections['Section 2'].part = 'PDFAttach2'
    printSections['Section 2'].password = 'secret';
    }
}
```

## Designer User Interface

The Designer's user interface gives you several options to work with.



See:

- "Menus" on page 320
- "Toolbars" on page 343
- "Resources Pane" on page 334
- "Outline Pane" on page 334
- "Attributes Pane" on page 328
- "Styles pane" on page 341
- "Workspace" on page 341
- "Data Model Pane" on page 330
- "Scripts pane" on page 339
- "Problems and messages" on page 332

# Dialogs

## Bar Chart Properties dialog

The Bar Chart dialog appears when a Bar Chart object is right-clicked and the Bar Chart... option is clicked. It determines how the Bar Chart is displayed when generating output and in Preview mode.

### General Tab

- **General Group:**
  - **Display grid above graph:** Check to display the grid on top of the bars so that it is always visible.
  - **Rotate:** Check to rotate the graph 90 degrees so that the bars are horizontal starting from the left.
  - **Stack Series:** [TBD]
- **Text Group:** Determines how text is displayed in labels and legends.
  - **Font:** Type in the font-face to use to display text. The font must be installed on the system and defaults to Verdana if the font is not found. Equivalent to the `font-family` property.
  - **Size:** Type in the size of the font. For example, 12pt or 20px. Defaults to 11px. Equivalent to the `font-size` property.
  - **Color:** Type in the color in which to display text. The color value must be a valid [HTML Hex Color](#). Equivalent to the `color` property.
- **Fill Group:**
  - **Color:** Enter a color for the bars. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). This color replaces all initial colors.
  - **Opacity:** Enter the percentage for the opacity of the bars. Does work on the initial colors, if no fill color is entered in this dialog.
- **Line Group:**
  - **Show Line:** Adds a line around each bar (or fills the bar if the bar has no fill color).
  - **Color:** Enter a color for the line. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the percentage for the opacity of the line.

## Value Axis Tab

- **Title group:**
  - **Label:** Enter a label for the Y axis (X axis if the graph is rotated).
  - **Bold:** Check if you want the label to be in bold style.
  - **Color:** Enter a custom color for the label (Default is Black). The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Font Size:** Enter a font size for the label, in pt.
- **Grid group:**
  - **Show Grid:** Displays a grid behind the bars.
  - **Color:** Enter a color for the grid that displays in the graph. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity percentage of the grid. Default is 15%.
  - **Thickness:** Enter a thickness for the grid, in pixels. Default is 1px.
  - **Tick Length:** The distance between each vertical line in the grid.
- **Axis group:**
  - **Show Axis:** Check to show the value axis (the line between the chart and the values).
  - **Color:** Enter a color for the value axis. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity in percentage for the axis.
  - **Thickness:** Enter the thickness, in pixels, for the axis.

## Category Axis Tab

- **Title group:**
  - **Label:** Enter a label for the X axis (Y axis if the graph is rotated).
  - **Bold:** Check if you want the label to be in bold style.
  - **Color:** Enter a custom color for the label (Default is Black). The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Font Size:** Enter a font size for the label, in pt.

- **Grid group:**
  - **Color:** Enter a color for the grid that is displays in the graph. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity percentage of the grid. Default is 15%.
  - **Thickness:** Enter a thickness for the grid, in pixels. Default is 1px.
  - **Position:** Choose Middle to centre the grid over the graph, or choose Start to make the first vertical grid lign match the value axis.
  - **Tick Length:** The distance between each vertical line in the grid.
- **Axis group:**
  - **Show Axis:** Check to show the value axis (the line between the chart and the values).
  - **Color:** Enter a color for the value axis. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity in percentage for the axis.
  - **Thickness:** Enter the thickness, in pixels, for the axis.

## Chart Tab

- **3D group:**
  - **Apply 3D Effect:** Check to apply a 3D effect to the bars.
  - **Depth:** Enter a thickness of the 3D effect, in pixels.
  - **Angle:** Enter angle for the 3D effect, in degrees.

## Legend tab

- **Show Legend:** Check to show the legends in the chart object.
- **Legend Group:** Defines how the legends are shown.
  - **Equal label widths:** Check so that all labels are of equal width in the Legends box. The Legend's width will accommodate the largest value.
  - **Position:** Use the drop-down to select where the labels are shown: Right, Left, Top or Bottom.

- **Align:** Use the drop-down to select how to align the text in the labels: Left, Middle or Right.
- **Horizontal Space:** When multiple columns appear, enter a numerical value (in pixels) to define horizontal spacing between the columns.
- **Vertical Space:** Enter a numerical value (in pixels) to define vertical spacing between legends.
- **Max Columns:** Enter a numerical value to define the maximum number of columns allowed in the Legends box.
- **Values Group:** Defines if and how values are shown in the Legends box.
  - **Show Values:** Check to show values besides the Legend's label.
  - **Text:** Enter the text used to display the values. Variables can be used to display specific data, <br> can be used to create a new line:
    - `[[percents]]` : Contains the percentage of the chart the value represents.
    - `[[value]]` : Contains the numerical value of the field.
    - Any Text: Adding text (such as a dollar sign or column, etc) will make it appear in each label.
- **Markers Group:** Defines how the Legends Markers. Markers are icons with a color matching the Legend with its corresponding bar.
  - **Type:** Use the drop-down to select in which shape the Markers are displayed. Select None to hide the Markers completely.
  - **Size:** Enter the size (in pixels) for the Markers to be displayed.
  - **Label Gap:** Enter the distance (in pixels) between the Markers and the Legends text.
  - **Border Width:** Use the drop-down to define the thickness of the border added to the Markers. Default is 0pt.
  - **Border Color:** Enter a valid HTML Hex Color for the border's color.
  - **Border Opacity:** Enter a numerical value between 0 and 100 to define the opacity (in percentage) of the border.

### Box Formatting dialog

The Text Formatting dialog is accessible by clicking inside a positioned box in the template and then selecting **Format > Box** in the menu.

## Box Tab

- **General group:**
  - **Width:** Set the width of the box in measure or percentage. Equivalent to the CSS `width` property.
  - **Height:** Set the height of the box in measure or percentage. Equivalent to the CSS `height` property.
  - **Angle:** Set the rotation angle of the box in clockwise degrees. Equivalent to the CSS `transform: rotate` property.
  - **Corner radius:** Set the radius of rounded border corners in measure or percentage. Equivalent to the CSS `border-radius` property.
  - **Display:** Use the drop-down or type in the value for how to display the box. Equivalent to the CSS `display` property.
  - **Overflow:** Use the drop-down or type in the value for how to handle overflow (text that does not fit in the current size of the box). Equivalent to the `overflow` property.
- **Text wrap group:**
  - **Float:** Use the drop-down or type in the value for how to float the box, if the box is not in an absolute position. Equivalent to the CSS `float` property.
  - **Clear:** Use the drop-down or type in the value for clearing pre-existing alignments. Equivalent to the CSS `clear` property.
- **Positioning:**
  - **Position:** Use the drop-down or type in the value for the type of positioning for the box. Equivalent to the CSS `position` property.
  - **Top:** Set the vertical offset between this box and its parent's top position. Equivalent to the CSS `top` property.
  - **Left:** Set the horizontal offset between this box and its parent's left position. Equivalent to the CSS `left` property.
  - **Bottom:** Set the vertical offset between this box and its parent's bottom position. Equivalent to the CSS `bottom` property.
  - **Right:** Set the horizontal offset between this box and its parent's left position. Equivalent to the CSS `right` property.
  - **Z-index:** Set the z-index of the box. The z-index defines in which order elements appear. Equivalent to the CSS `z-index` property.

## Background Tab

- **General group:**
  - **Color:** Specify the color of the box background. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `background-color` property.
- **Background image group:**
  - **Source:** click the **Select Image** button to select an image via the "Select Image dialog" on page 308. Equivalent to the CSS `background` property.
  - **Size:** select `auto`, `cover` or `contain` (for an explanation see [http://www.w3schools.com/cssref/css3\\_pr\\_background-size.asp](http://www.w3schools.com/cssref/css3_pr_background-size.asp)), or type the width and height of the image in a measure (e.g. `80px 60px`) or as a percentage of the box size (e.g. `50% 50%`). Equivalent to the CSS `background-size` property.
  - **Position:** select the position for the background-image. Equivalent to the CSS `background-position` property.

## Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:
  - **All sides:** Check to set all padding to use the Top value. Equivalent to the CSS `padding` property.
  - **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the CSS `padding-left`, `padding-top`, `padding-right` and `padding-bottom` properties.
- **Margin group:** Defines margins (spacing outside the element) in measure or percentage:
  - **All sides:** Check to set all margins to use the Top value. Equivalent to the CSS `margin` property.
  - **Top, Left, Bottom, Right:** Set the margin for each side. Equivalent to the CSS `margin-left`, `margin-top`, `margin-right` and `margin-bottom` properties.

## Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the CSS `border` property.

- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the CSS `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the CSS `border-style` property.
  - **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `border-color` property.

## Color Picker

The Color Picker dialog appears when creating a color in the formatting dialogs of certain elements, for example border colors in boxes and paragraphs.

The dialog consists of two main parts. On the left is the color wheel that can be used to select a color **hue** by clicking anywhere on that wheel. To the right of the color wheel there is a vertical bar used to select the color **saturation**. At the top-right, two colors are shown: the **New** box displays the currently selected color, while the **Original** shows the color currently attributed to the element.

The rest of the dialog has various options for choosing colors:

- **Color Mode:** Use the drop-down to select whether the color is set as **RGB** or **CMYK**. This determines how the color is saved in the formatting properties, and how they are printed or output; see "Colors" on page 190.
- **RGB group:** Enter the Red, Green and Blue color values from 0 to 255.
- **HEX:** Enter a valid [HTML Hex Color](#).
- The **eye dropper** lets you select a color from anywhere on your desktop. To open it, click the eye dropper button  next to the HEX color field.
- **CMYK group:** Enter the Cyan, Magenta, Yellow and Black color values from 0 to 100 percent.

## Note

Whenever one value within this dialog is changed, all the other values are adjusted to their equivalent.

## Colors Properties

The Colors Properties defines and sets named colors used in the template; see "Colors" on page 190. Named colors can be used throughout the templates, in all contexts. They are visible in color selector dialogs and useable with their names in style sheets; see "Styling and formatting" on page 170.

- **Color Type Selector:** Click and use the drop-down to display which color types to show in the list: All, RGB, CMYK or Spot colors.
- **Color List:** Displays the colors, filtered using the Color Type Selector.
- **New:** Create a new color using the **Edit Color** dialog (see below and see "Colors" on page 190).
- **Edit:** Edit the currently selected color using the **Edit Color** dialog (see below and see "Colors" on page 190).
- **Delete:** Delete the currently selected color.
- **Duplicate:** Duplicate the currently selected color using the name [color]CopyX.

## Edit color

You can edit the following color properties.

- **Name:** Enter the name of the color. This name should not contain spaces or special characters.
- **Create name based on values:** Check so that the name is automatically based on the color slider values below.
- **Type:** Use the drop-down to specify which type of color this should be: either a Tint or a Color.
- **Option group:** contains the options for the chosen type. Options change depending on the selected type.

- **Color:**
  - **Model:** This can be either CMYK or RGB.
  - **Spot Color:** Check to set the color as Spot Color. When Spot Colors are used, the Name must match that of the spot color used in the printer.
  - **Cyan/Magenta/Yellow/Black (CMYK):** Each slider sets a percentage for the color. Set the values using the sliders, or type in the percentage directly in the input boxes.
  - **Red/Green/Blue (RGB):** Each slider sets the values of 0-255 for the color. Set the value using the sliders or type in the value directly in the input boxes.
  - **Color Preview:** Box displaying the preview of the color (converted to RGB when relevant).
- **Tint:**
  - **Source:** Select an existing *Color* in the template. The *tint* or *opacity* will be applied to this color.
  - **Tint/Opacity:** The slider sets the percentage of tint or opacity. Set the value using the slider, or type the percentage directly in the input box.
  - **Use Opacity:** Check to set the Tint slider to use Opacity instead.

## Color Settings

Color Management can keep colors consistent across different outputs by using Color Profiles. When producing output to a new device, color adjustments are made to present the color as accurately as possible on this new device.

- **Enable Color Management:** Check to disable color management and ignore embedded color profiles when importing images (with the exception of imported PDF files as it might contain a multiple tagged sub images).
- **Working Space Group:** Defines the color profiles for the current template.
  - **RGB:** Use the drop-down to select a color profile for RGB colors. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\rgb".
  - **CMYK:** Use the drop-down to select a color profile for CMYK colors. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\cmyk".
  - **Gray:** Use the drop-down to select a color profile for Grayscale. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\gray".

- **Untagged Images Group:** Defines color profiles for any image that does not specifically have color profiles or color settings enabled.
  - **RGB:** Use the drop-down to select a color profile for RGB colors. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\rgb".
  - **CMYK:** Use the drop-down to select a color profile for CMYK colors. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\cmyk".
  - **Gray:** Use the drop-down to select a color profile for Grayscale. The list displays ICC profiles located in "%USERPROFILE%\Connect\color-profiles\gray".
- **Options Group:**
  - **Rendering intent:** Use the drop-down to specify how colors are converted that are out of range of a profile. For example, you may use tricks like reducing the saturation of the entire print so that a color that is out of range still appears a bit more vibrant than ones that are in range. Rendering intents use different methods to trick the eye into believing that the print can reproduce irreproducible colors.

## Context Properties dialog

Which properties are available in the Context Properties dialog depends on the context: Email or Print.

To open this dialog, right-click the context on the **Resources** pane and select **Properties**.

## Email Context Properties

For the Email context, the Context Properties dialog defines options that are used when generating email output (see "Generating Email output" on page 509).

- **Print Context Image Compression:** Defines the properties of the PDF when attaching the Print context to email output.
  - **Lossless:** Enables maximum quality in the PDF. Note that this will produce a larger PDF.
  - **Quality:** Disabled when Lossless is checked. Determines the quality (aka compression) of the attached PDF.
  - **Tile Size:** Use the drop-down to select the size of the tiles used in the image. When low Quality values are used to optimize images smaller than 1024 x 1024 pixels, using the largest tile size will produce better results.

## Print Context Finishing Options

For the Print context, the Context Properties dialog defines finishing options for when it is printed (see "Generating Print output" on page 495). These options affect the context as a whole including all sections.

For an explanation of all Binding and Hole making options, see "Finishing Options" on page 403.

## Edit Label Properties

The Edit Label Properties defines how a Pie Chart Label displays its title and data. It contains two options:

- **Label:** Enter a title for Labels and Legends when they are shown (see "Pie Chart Properties dialog" on page 279).
- **Value:** Use the drop-down to select which Value to use as data within the Pie Chart as well as for Label and Legend values.

## Find/Replace Dialog

The Find/Replace dialog can replace text within the current template. The scope of the replacement depends on the currently selected tab in the Workspace. If the Source tab is selected, the replace will affect the HTML source code. If the Design tab is selected, the replace will affect the text on the page. If the Preview tab is selected, the Replace feature is inactive.

### Note

When replacing text in the Design tab, formatting in the replaced text will be removed. If formatting is necessary in the new text, select the Source tab before opening the Find/Replace dialog and include the required HTML tags in the replacement text.

Here are the options available in this dialog

- **Find:** The source string to find.
- **Replace with:** The string to replace the source with.

- **Direction**

- **Forward:** Look forward from the current position of the pointer in the template or source.
- **Backward:** Look backward from the current position of the pointer in the template or source.

- **Scope**

- **All:** Searches in the complete text of the template or source.
- **Selected lines:** Searches in the currently selected text or source.

- **Options**

- **Case sensitive:** Use a case-sensitive search, which differentiates **TEXT** from **text** or **Text**.
- **Wrap search:** Loop back from the end of the template or selection to its beginning, when the Search is at the end of the template or the selection.
- **Whole word:** Searches for the source string as a whole word.
- **Incremental:** With this option selected, each letter you type in the Find field causes the editor focus to move to the first complete occurrence of the text you are typing.
- **Regular expressions:** Enables regular expressions for a search in the **Source** view of the workspace. After checking this option, you can type Ctrl + Space in either text box to view a list of regular expressions.

**Tip**

The Find/Replace dialog can fill in regular expressions in the Find field by itself. Open the dialog, check the option Regular expressions and close the dialog again. Select the text you want to search for and reopen the dialog: the Find field will now contain the regular expression for the text to find.

- **Find:** Click to find the next instance of the source string.
- **Replace/Find:** Click to replace the current instance with the replacement text and go to the next instance of the source string.
- **Replace:** Click to replace the current instance with the replacement text.
- **Replace All:** Click to replace all instances of the source string with the replacement text.

- **Close:** Close the dialog.

## Fonts Manager

The Fonts Manager contains the fonts that were added to the template manually. It essentially lists the fonts located in the **Fonts** folder of the Resources pane (see "Fonts" on page 194).

Fonts with the same file name with a different extension are considered variations of the same font. For example, if there are three files, named gotham-book-webfont.eot, gotham-book-webfont.ttf, gotham-book-webfont.woff, only "gotham-book-webfont" appears in the Name column of this dialog.

The following buttons appear to the right of the list of fonts:

- **New:** Click to open the Edit Font dialog to add a new font.
- **Edit:** Click to open the Edit Font dialog to edit the currently selected font.
- **Remove:** Click to delete the currently selected font entry.
- **Duplicate:** Click to create a copy of the currently selected font entry.

## Edit Font

The Edit Font dialog appears when clicking New or Edit from the Fonts Dialog.

- **Name:** Enter the name that should be used to refer to the font. This is equivalent to the `font-family` property of the `@font-face` CSS rule (see [http://www.w3schools.com/cssref/css3\\_pr\\_font-face\\_rule.asp](http://www.w3schools.com/cssref/css3_pr_font-face_rule.asp)).
- **Font Weight:** Use the drop-down to select the default font weight (the thickness, see [http://www.w3schools.com/cssref/pr\\_font\\_weight.asp](http://www.w3schools.com/cssref/pr_font_weight.asp)):
  - **None:** Does not define the property.
  - **Normal:** Defines font-weight as normal
  - **Bold:** Defines the font-weight as bold (equivalent to a numerical value of 700).
  - **Numerical values:** Defines the line thickness; 400 is normal, 700 is bold.
- **Font Style:** Use the drop-down to select the font style (see [http://www.w3schools.com/cssref/pr\\_font\\_font-style.asp](http://www.w3schools.com/cssref/pr_font_font-style.asp)):
  - **None:** Does not define the property.
  - **Normal:** Defines font-style as normal

- **Italic:** Makes the font italic.
- **Oblique:** Makes the font oblique (this is generally the same as italic but does not require a special italic version of the font).
- **Name:** Check the fonts in the list to include them in the font definition.

## Image Formatting dialog

The Image Formatting dialog is accessible by selecting an image in the template and then selecting **Format > Image** in the menu.

## Image Tab

- **General group:**
  - **Width:** Set the width of the image in measure or percentage. Equivalent to the CSS `width` property.
  - **Height:** Set the height of the image in measure or percentage. Equivalent to the CSS `height` property.
  - **Angle:** Set the rotation angle of the image in clockwise degrees. Equivalent to the CSS `transform: rotate` property.
  - **Corner radius:** Set the radius of rounded border corners in measure or percentage. Equivalent to the CSS `border-radius` property.
  - **Display:** Use the drop-down or type in the value for how to display the image. Equivalent to the CSS `display` property.
  - **Overflow:** Use the drop-down or type in the value for how to handle overflow (the part of the image that does not fit in the current size of the box). Equivalent to the CSS `overflow` property.
  - **Source:** Enter the web address or local file address of the image. Equivalent to the HTML `src` attribute.
  - **Alternate text:** Enter an alternate text for the image. This is displayed in browsers and email clients when the image is loading or if the image cannot be displayed. It is also used for accessibility. Equivalent to the HTML `alt` attribute.
- **Text wrap group:**
  - **Float:** Use the drop-down or type in the value for how to float the image, if the image is not in an absolute position. Equivalent to the CSS `float` property.

- **Clear:** Use the drop-down or type the value to clear pre-existing alignments. Equivalent to the CSS `clear` property.
- **Positioning:**
  - **Position:** Use the drop-down or type in the value for the type of positioning for the image. Equivalent to the CSS `position` property.
  - **Top:** Set the vertical offset between this image and its parent's top position. Equivalent to the CSS `top` property.
  - **Left:** Set the horizontal offset between this image and its parent's left position. Equivalent to the CSS `left` property.
  - **Bottom:** Set the vertical offset between this image and its parent's bottom position. Equivalent to the CSS `bottom` property.
  - **Right:** Set the horizontal offset between this image and its parent's left position. Equivalent to the CSS `right` property.
  - **Z-index:** Set the z-index of the image. The z-index defines in which order elements appear. Equivalent to the CSS `z-index` property.

## Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:
  - **All sides:** Check to set all padding to use the Top value. Equivalent to the CSS `border` property.
  - **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the CSS `border-left`, `border-top`, `border-right` and `border-bottom` properties.
- **Margin group:** Defines margins (spacing outside the element) in measure or percentage:
  - **All sides:** Check to set all margins to use the Top value. Equivalent to the CSS `margin` property.
  - **Top, Left, Bottom, Right:** Set the margin for each side. Equivalent to the CSS `margin-left`, `margin-top`, `margin-right` and `margin-bottom` properties.

## Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the CSS `border` property.

- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the CSS `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the CSS `border-style` property.
  - **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `border-color` property.

## Line Chart Properties dialog

The Line Chart dialog appears when a Line Chart object is right-clicked and the Line Chart... option is clicked. It determines how the chart is displayed when generating output and in Preview mode.

### General Tab

- **General Group:**
  - **Display grid above graph:** Check to display the grid on top of the lines so that it is always visible.
  - **Rotate:** Check to rotate the graph 90 degrees so that the lines are vertical starting from the top.
  - **Stack Series:** Stack the lines so that lines representing the same value do not overlap.
- **Text Group:** Determines how text is displayed in labels and legends.
  - **Font:** Type in the font-face to use to display text. The font must be installed on the system and defaults to Verdana if the font is not found. Equivalent to the `font-family` property.
  - **Size:** Type in the size of the font. For example, 12pt or 20px. Defaults to 11px. Equivalent to the `font-size` property.
  - **Color:** Type in the color in which to display text. The color value must be a valid [HTML Hex Color](#). Equivalent to the `color` property.
- **Fill Group:**

- **Color:** Enter a color for the lines. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). This color replaces all initial colors.
- **Opacity:** Enter the percentage for the opacity of the bars. Does work on the initial colors, if no fill color is entered in this dialog.
- **Line Group:**
  - **Show Line:** Adds a line around each line part (or fills the line part if it has no fill color).
  - **Color:** Enter a color for the line. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the percentage for the opacity of the line.

## Value Axis Tab

- **Title group:**
  - **Label:** Enter a label for the Y axis (X axis if the graph is rotated).
  - **Bold:** Check if you want the label to be in bold style.
  - **Color:** Enter a custom color for the label (Default is Black). The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Font Size:** Enter a font size for the label, in pt.
- **Grid group:**
  - **Color:** Enter a color for the grid that is displays in the graph. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity percentage of the grid. Default is 15%.
  - **Thickness:** Enter a thickness for the grid, in pixels. Default is 1px.
  - **Tick Length:** The distance between each vertical line in the grid.
- **Axis group:**
  - **Show Axis:** Check to show the value axis (the line between the chart and the values).
  - **Color:** Enter a color for the value axis. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity in percentage for the axis.
  - **Thickness:** Enter the thickness, in pixels, for the axis.

## Category Axis Tab

- **Title group:**
  - **Label:** Enter a label for the X axis (Y axis if the graph is rotated).
  - **Bold:** Check if you want the label to be in bold style.
  - **Color:** Enter a custom color for the label (Default is Black). The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Font Size:** Enter a font size for the label, in pt.
- **Grid group:**
  - **Color:** Enter a color for the grid that is displays in the graph. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity percentage of the grid. Default is 15%.
  - **Thickness:** Enter a thickness for the grid, in pixels. Default is 1px.
  - **Position:** Choose Middle to centre the grid over the graph, or choose Start to make the first vertical grid lign match the value axis.
  - **Tick Length:** The distance between each vertical line in the grid.
- **Axis group:**
  - **Show Axis:** Check to show the value axis (the line between the chart and the values).
  - **Color:** Enter a color for the value axis. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#).
  - **Opacity:** Enter the opacity in percentage for the axis.
  - **Thickness:** Enter the thickness, in pixels, for the axis.

## Legend tab

- **Show Legend:** Check to show the legends in the chart object.
- **Legend Group:** Defines how the legends are shown.
  - **Equal label widths:** Check so that all labels are of equal width in the Legends box. The Legend's width will accommodate the largest value.
  - **Position:** Use the drop-down to select where the labels are shown: Right, Left, Top or Bottom.

- **Align:** Use the drop-down to select how to align the text in the labels: Left, Middle or Right.
- **Horizontal Space:** When multiple columns appear, enter a numerical value (in pixels) to define horizontal spacing between the columns.
- **Vertical Space:** Enter a numerical value (in pixels) to define vertical spacing between legends.
- **Max Columns:** Enter a numerical value to define the maximum number of columns allowed in the Legends box.
- **Values Group:** Defines if and how values are shown in the Legends box.
  - **Show Values:** Check to show values besides the Legend's label.
  - **Text:** Enter the text used to display the values. Variables can be used to display specific data, <br> can be used to create a new line:
    - `[[percents]]` : Contains the percentage of the chart the value represents.
    - `[[value]]` : Contains the numerical value of the field.
    - Any Text: Adding text (such as a dollar sign or column, etc) will make it appear in each label.
- **Markers Group:** Defines how the Legends Markers. Markers are icons with a color matching the Legend with its corresponding line.
  - **Type:** Use the drop-down to select in which shape the Markers are displayed. "none" hides the Markers completely.
  - **Size:** Enter the size (in pixels) for the Markers to be displayed.
  - **Label Gap:** Enter the distance (in pixels) between the Markers and the Legends text.
  - **Border Width:** Use the drop-down to define the thickness of the border added to the Markers. Default is 0pt.
  - **Border Color:** Enter a valid HTML Hex Color for the border's color.
  - **Border Opacity:** Enter a numerical value between 0 and 100 to define the opacity (in percentage) of the border.

## Locale Settings

The Locale dialog box sets the locale used inside the template. The Locale can affect time, currency output, and other formatting that depends on location and language (see "Locale" on page 201).

- **Use:** Use the drop-down to select how the Locale is set for the current template.
  - **System Locale:** Select this to use the operating system's locale settings. This is set in the Region settings of the control panel.
  - **Explicit Locale:** Select this option to specify a static locale which will remain static for this template, whichever server the template is used on.
  - **Data Field:** Select this to use a data field from the record. The locale will be record-specific in this case.
- **Locale:** Use the drop-down to select a specific locale. Only enabled when **Explicit Locale** is selected above.
- **Data Field:** Use the drop-down to select a field within the current data model that contains the locale. This field must be a string and contain the exact locale to be used, such as "en" or "fr-CA". It cannot be an alias such as "english" or "french". The locale supports both ISO-639-1 alone ("en", "fr", etc) or ISO-639-1 followed by a 2-letter country code ("de-DE", "zh-CN", "fr-CA", "fr-FR", etc).

## Master Page Properties

Master Pages can only be used in a Print context; see "Master Pages" on page 76.

The following properties are available for Master Page resources:

- **Name:** The name of the master page, displayed in all drop-downs where the Master Page is shown as well as in the "Resources Pane" on page 334.
- **Margins** group:
  - **Header:** The space at the top of the Master Page where no content will print, when this Master Page is used in a Section.
  - **Footer:** The space at the bottom of the Master Page where no content will print, when this Master Page is used in a Section.

## Media Properties

Media can only be used in a Print context; see "Media" on page 80.

Media are not printed, unless you want them to; see "Printing virtual stationery" on page 84.

The following properties are available for Media resources.

## Properties Tab

- **Name:** The name of the media, displayed in all drop-downs where the Media is shown as well as in the [Resources Pane](#).
- **Size group:** This group is read-only and only used to display the size selected in the linked Print section's properties (see "Print Section Properties" on page 300).
  - **Page Size:** The named page size.
  - **Width:** The width of the page.
  - **Height:** The height of the page.
  - **Orientation:** Whether the page is portrait or landscape.

## Virtual Stationery Tab

- **Front/Back group:** Defines the preprinted media used for the front and back of the Virtual Stationery.
  - **PDF:** Click the Select Image button to open the "Select Image dialog" on page 308 and select which PDF (and optionally, which page of the PDF) to display as a background for the page.
  - **Position:** Use the drop-down to select how the PDF is displayed on the page:
    - **Fit to Media:** Select to stretch the PDF to fit the media size.
    - **Centered:** Select to center the PDF on the page, vertically and horizontally.
    - **Absolute:** Select to place the PDF at a specific location on the page. Use the Top and Left options below to specify the positioning of the PDF.
      - **Top:** The distance between the top side of the page and the top side of the PDF.
      - **Left:** The distance between the left side of the page and the left side of the PDF.
- **Front side:** Select the image that is shown as a background for all "front" sides in the template.
- **Back side:** Select the image that is shown as a background for all "back" sides in the template.

## Characteristics tab

The characteristics define the type of paper on which the Print context is meant to be printed on.

- **Media Type:** The type of paper, such as *Continuous, Envelope, Labels, Stationery, etc.*
- **Weight:** The weight of the media in grammage (g/m2).
- **Front Coating:** The pre-process coating applied to the front surface of the media, such as *Glossy, High Gloss, Matte, Satin, etc.*
- **Back Coating:** The pre-process coating applied to the front surface of the media.
- **Texture:** The texture of the media, such as *Antique, Calenared, Linen, Stipple or Vellum.*
- **Grade:** The grade of the media, such as *Gloss-coated paper, Uncoated white paper, etc.*
- **Hole Name:** Pre-defined hole pattern that specifies the pre-punched holes in the media, such as *R2-generic, R2m-MIB, R4i-US, etc.*

## Package File dialog

The Package File dialog saves the template with the print presets as a package file. Package files can be sent to other users of the Connect Designer.

- **Files to Package group:**
  - **Template:** Select the template to send. By default the currently active template is listed. Click **Browse** to select another template. In version 1.3 you may select more than one template in the Browse dialog, and each of them is sent to Workflow or added to a package file.
  - **Job Creation Preset:** Use the drop-down to select a Job Creation Preset to send. Click **Browse** to select a preset that is not in the default save location.
  - **Output Creation Preset:** Use the drop-down to select an Output Creation Preset. Click **Browse** to select a preset that is not in the default save location.

## Paragraph Formatting dialog

The Paragraph formatting controls how the selected paragraph is formatted. It is accessed by placing the cursor within a paragraph then going in the Format menu, then Paragraph...

## Formats Tab

- **General group:**
  - **Line-height:** Specify the height of each line in the element's text, in measure or percentage. Note that this is not spacing between lines, but rather the complete height of the line itself including the text. Equivalent to the `line-height` property.

- **Align:** Select how text should be aligned, such as `left`, `center`, `right` or `justify`. Equivalent to the `align` property.
- **First Indent:** Specify the indentation of the first line of each paragraph in the element. Equivalent to the `text-indent` property.
- **Display:** Select how to display the element. This can also be used to hide an element completely using the `none` option. See [CSS Display](#). Equivalent to the `display` property.
- **Direction:** Select in which direction text should be displayed (`ltr`, `rtl`, `auto`). Useful for certain languages such as arabic, hebrew, etc. Equivalent to the `dir` HTML attribute.
- **Breaks group:**
  - **Before:** Specifies how to handle page breaks before the element. Equivalent to the `page-break-before` property.
  - **Inside:** Specifies whether to accept page breaks within the paragraph. Equivalent to the `page-break-inside` property.
  - **After:** Specifies how to handle page breaks after the element. Equivalent to the `page-break-after` property.
  - **Widows:** Specifies how to handle widows within the paragraph (lines appearing alone on the next page if the paragraph does not fit on the current one). Equivalent to the `widows` property. Widows and orphans are ignored if the `page-break-inside` property is set to `avoid`.
  - **Orphans:** Specifies how to handle orphans within the paragraph (lines appearing alone at the end of a page if the paragraph does not fit on the current one). Equivalent to the `orphans` property.

#### Note

For more information on page breaks, widows and orphans, see the [W3 Paged Media reference](#).

## Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:

- **All sides:** Check to set all padding to use the Top value. Equivalent to the CSS `padding` property.
- **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the CSS `padding-left`, `padding-top`, `padding-right` and `padding-bottom` properties.
- **Margin group:** Defines margins (spacing outside the element) in measure or percentage:
  - **All sides:** Check to set all margins to use the Top value. Equivalent to the `margin` property.
  - **Top, Left, Bottom, Right:** Set the margin for each side. Equivalent to the `margin-left`, `margin-top`, `margin-right` and `margin-bottom` properties.

## Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the `border` property.
- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the `border-style` property.
  - **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the `border-color` property.

## Pie Chart Properties dialog

The Pie Chart dialog appears when a Pie Chart object is right-clicked and the **Pie Chart...** option is clicked. It determines how the Pie Chart is displayed when generating output and in Preview mode (see "Business graphics" on page 152).

## General tab

- **Text Group:** Determines how text is displayed in labels and legends.
  - **Font:** Type in the font-face to use to display text. The font must be installed on the system and defaults to Verdana if the font is not found. Equivalent to the `font-family` property.

- **Size:** Type in the size of the font. For example, 12pt or 20px. Defaults to 11px. Equivalent to the `font-size` property.
- **Color:** Type in the color in which to display text. The color value must be a valid [HTML Hex Color](#). Equivalent to the `color` property.
- **Slice Colors Group:** Determines which colors are used to display the Pie Chart.
  - **Apply:** select which set of colors to use for the chart: **standardColors**, **baseColor** (a set of colors based on the color defined in the Base Color option) or **colors** (a set of colors defined in the Color Array option).
  - **Base Color:** Enter a valid HTML Hex Color. When a Base Color is set, it will be the color of the first slice and the colors of all other slices are based on this color.
  - **Brightness Step:** Enter the amount of brightness to change on each new slice. Positive values increase brightness (max: 100), minimum values decrease brightness (minimum: -100). Default is 10.
  - **Color Array:** Enter a comma separated list of hex colors to specify the colors of the slices. If there are more slices than colors in this list, the chart picks random colors. Example: `#FF3300, #FFFF00, #33CC33, #FFCC00`.
  - **Gradient Ratio:** Enter a start and end point gradient to be applied to each slice (for example: `-0.5, 0.5`).
- **Slice Outline Group:** Determines whether an outline should be added to each slice of the chart.
  - **Width:** Use the drop-down to select the width of the outline for each pie slice. Values are 0pt, 0.5pt, 1pt, 1.5pt, 2pt or 3pt.
  - **Color:** Enter a valid HTML Hex Color for the outline to appear.
  - **Opacity:** Enter the opacity of the outline. 100 is fully opaque, 0 is transparent.

## Pie tab

- **Pie Group:** Defines how the pie chart is displayed in the template.
  - **Automatically calculate radius:** Check to automatically calculate the radius of the Pie Chart, determined by the size of the object it is contained in. The radius, by default, is 50% of the shortest length of the containing `<div>` object.
  - **Radius:** Enter the radius of the Pie Chart in percentage of the shortest length of the containing `<div>` object.

- **Hole Radius:** Enter the radius of the center of the Pie Chart to remove, between 0% and 100%. The hole radius removes the center of the chart, creating a doughnut hole pie chart.
- **Start Angle:** Enter the starting angle of the first slice of the chart, between 0 and 360. This essentially rotates the Pie Chart. Note that if a 3D effect is added to the chart, the only accepted values are 90 or 270 degrees.
- **3D Group:** Defines 3D effects of the Pie Chart.
  - **Apply 3D effect:** Check to enable the Pie Chart to be displayed in a 3D fashion.
  - **Depth:** Enter a numerical value for the thickness of the Pie Chart. Must be in steps of 10 (0, 10, 20, etc).
  - **Angle:** Enter the angle at which the Pie Chart is rotated to create the 3D effect. Default is 20 degrees of rotation.

## Labels tab

- **Hide Labels:** Check to disable the label's display.
- **Labels Group:** Defines how the label text is shown.
  - **Custom label text:** Check to enable custom text for the labels. The default display is `[[title]]: [[percents]]`.
  - **Text:** Enter the text to use to display labels. Variables can be used to display specific data, `<br>` can be used to create a new line:
    - `[[title]]` : Contains either the contents the Label column if Static Labels are used, or the Field Name if Dynamic Labels are used.
    - `[[percents]]` : Contains the percentage of the Pie Chart the value represents.
    - `[[value]]` : Contains the numerical value of the field.
    - Any Text: Adding text (such as a dollar sign or column, etc) will make it appear in each label.
  - **Radius:** Enter a numerical value representing the percentage of the Pie Chart's radius to add as a space between the pie and the labels. The value can be negative, in which case labels are shown within the Pie Chart. If a positive value is used, a line (called a "tick") from each slice of the pie to its label is added.

- **Tick Group:** Defines how ticks (line between the Pie Chart and its labels) is shown.
  - **Color:** Enter a valid HTML Hex Color for the color of the tick.
  - **Opacity:** Enter a percentage of opacity for the tick to be displayed. Default 20 (20% opacity).
- **Grouping Group:** Defines how smaller percentage are grouped together into an individual "Other" category.
  - **Apply slice grouping:** Check to enable grouping.
  - **Less than %:** Enter a percentage below which values are placed within the "Other" category.
  - **Slice Title:** Enter a name for the label of the "Other" category. Defaults to "Other".
  - **Color:** Enter a valid HTML Hex Color for the slice. Defaults to the colors set in the General tab.

## Legend tab

- **Show Legend:** Check to show the legend in the Pie Chart object.
- **Legend Group:** Defines how the legend is shown.
  - **Equal label widths:** Check so that all labels are of equal width in the Legend box. The Legend's width will accommodate the largest value.
  - **Position:** Use the drop-down to select where the labels are shown: Right, Left, Top or Bottom.
  - **Align:** Use the drop-down to select how to align the text in the labels: Left, Middle or Right.
  - **Horizontal Space:** When multiple columns appear, enter a numerical value (in pixels) to define horizontal spacing between the columns.
  - **Vertical Space:** Enter a numerical value (in pixels) to define vertical spacing between legends.
  - **Max Columns:** Enter a numerical value to define the maximum number of columns allowed in the Legends box.
- **Values Group:** Defines if and how values are shown in the Legends box.
  - **Show Values:** Check to show values besides the Legend's label.
  - **Text:** Enter the text used to display the values. Variables can be used to display specific data, <br> can be used to create a new line:

- `[[percents]]` : Contains the percentage of the Pie Chart the value represents.
- `[[value]]` : Contains the numerical value of the field.
- Any Text: Adding text (such as a dollar sign or column, etc) will make it appear in each label.
- **Markers Group**: Defines how the Legends Markers. Markers are icons with a color matching the Legend with its corresponding Pie Chart slice.
  - **Type**: Use the drop-down to select in which shape the Markers are displayed. "none" hides the Markers completely.
  - **Size**: Enter the size (in pixels) for the Markers to be displayed.
  - **Label Gap**: Enter the distance (in pixels) between the Markers and the Legends text.
  - **Border Width**: Use the drop-down to define the thickness of the border added to the Markers. Default is 0pt.
  - **Border Color**: Enter a valid HTML Hex Color for the border's color.
  - **Border Opacity**: Enter a numerical value between 0 and 100 to define the opacity (in percentage) of the border.

## Preferences

The Preferences dialog is used to modify the general software preferences. Changes made in this dialog affect the software globally, not individual templates and data mapping configurations.

The Preferences dialog is separated into individual tabs, where each tab controls certain aspects of the software.

The General Preferences are as follows:

- **Always run in background**: This option correlates with the "Always run in background" option selectable in the "Document Boundaries Refresh" dialog and "Print via Print Server" dialog. When either of these dialogs is used and the option is checked, it will also be checked here. To prevent the refresh boundaries and print via print server dialogs to automatically run as background, uncheck this option.

## Clean-up Service preferences

The Clean-up Service defines how the Connect database and the "managed" temporary files (referenced within said database) created during Connect production runs are cleaned up after the production run has finished. The "managed" files include temporary data files, configuration files and intermediate files created during the production process. They are longer be needed, once the run has completed.

The values below define when the specified targets are to be *set* as ready for deletion, not *when* they are actually deleted. The actual deletion occurs only when PrintShop Mail Connect is started (if **Run at application start up** is selected), or when the **Run Now** button is pressed, or as per the cron job scheduling.

### Tip

The more items present in the database, and the larger they are, the more time and processing power (CPU) that is required in cleaning them up. Thus a regular clean-up of the database (as often as possible) is recommended. This is especially the case if items are not going to be retrieved from the database at a later date. i.e. If the Connect job is not going to be re-run.

The clean-up can always be set to run outside of business hours (see the **Run according to the cron schedule** option below), to reduce impact upon Production systems.

- **Enable clean-up service:** Check to enable the clean-up services. When checked, either or both of the *Database clean-up* and *File clean-up* services can be set individually. If the box is not checked, then no clean-up will occur.
- **Run at application start up:** Click to start the clean-up service when the Designer module is opened.
- **Run according to the cron schedule:** Enter the interval at which the clean-up service runs.  
To understand how to write a cron job schedule, please refer to the excellent [Quartz Scheduler](#) reference page.

### Note

If the **Product managing the service** is set to Designer, then the Designer *must be running* at the time that the cron job is scheduled, for the clean-up to run.

- **Product managing the service:** Select which of the applications will run the service. For PrintShop Mail Connect the Designer is the only choice.

### Note

The **Server Engine** is set as the default as it is generally considered the best option.

This is particularly the case when using a scheduled cron job, as the Sever Engine is always running, whilst the Designer might well not be at the scheduled time (in which case the clean-up will fail to run).

- **Database Clean-up Service:**

- **Allow database clean-up service:** Select this checkbox to enable the database clean-up settings, and enable the actual clean-up.
- **Threads to use for database deletions:** The number of Threads to be used in the clean-up. PrintShop Mail Connect is a multi-threaded application, and the clean-up is likewise.

### Tip

The default number of threads is considered the best compromise for running both clean-up and production jobs simultaneously. If experience suggests that the clean-up is not running efficiently, then upping the number of threads here would be recommended. Conversely, if production appears to be suffering courtesy of the clean-up process, then reduce the number of threads here.

In general, higher end machines (those with multiple cores) will allow a higher

numbers of threads, whilst low end machines will perform better with a lower number of threads.

- **Number of entities in each deletion batch:** The number of entities to be deleted at a time. This is done to break the clean-up into smaller chunks. This improves PrintShop Mail Connect clean-up responsiveness, whilst the clean-up is occurring. The number selected here applies to all the following settings.  
i.e. a selection of 1,000 would delete 1,000 data records within a **Data Set**, 1,000 content items within a **Content Set**, and so on.
- **Minimum time to retain Data Sets:** The minimum time a Data Set (and all the records it contains) is retained within the database before being set for deletion.
- **Minimum time to retain Content Sets:** The minimum time a Content Set (and all the content items it contains) is retained within the database before being set for deletion.
- **Minimum time to retain Job Sets:** The minimum time a Job Set (and all the jobs information it contains) is retained within the database before being set for deletion.
- **Minimum time to retain Managed Files:** The minimum time file references (to files such as data mapping configurations and templates) are retained within the database before being set for deletion.
- **Minimum time to retain other entities:** The minimum time any orphaned data (such as Finishing tables, Media tables, DataModels and Properties tables) are retained within the database before being set for deletion.
- **File Clean-up Service:**
  - **Allow file clean-up service:** Check to automatically detect orphan files and set them for deletion. Orphan files could be resources and internal files used by Connect, but which are not needed by any running job.
  - **Minimum time to retain orphaned files:** The minimum time during which orphaned files are kept in the database before being set for deletion.

## Editing preferences

These preferences define different editing options in the Designer module.

- **Object Resizing for <div> elements:** This defines in which contexts to enable the resizing of <div> elements (including Positioned and Inline boxes). Resizing <div> elements may

cause layouts to produce undesirable results especially when using Foundation templates.

- **Enable for Print Context:** Check to enable <div> resizing in the Print contexts.
- **Enable for Email Context:** Check to enable <div> resizing in the Email contexts.

## Images Preferences

- **Transparent PDF image preview:** Check this option so that PDF resources added to the template (including in the Master Page and Media) display using transparency. Note that this can affect display performance (showing transparent PDFs is slower) but will not affect output speed.

## **Email Preferences**

### Email (General) Preferences

- **Default From Group:**
  - **Name:** Enter the name that is set by default in the "From name" field in the "Send Email" on page 306 and "Send Test Email" on page 307 dialogs.
  - **Email Address:** Enter the email that is set by default in the "From Email" field in the "Send Email" on page 306 and "Send Test Email" on page 307 dialogs.
- **Litmus account Group:**
  - **Email Test address:** If you have a Litmus account, enter the test address to use when using the "Send Test Email" on page 307 dialog. For more information on Litmus, please see <http://litmus.com/>

### Email (SMTP) Preferences

SMTP server presets can be selected when sending emails using either the "Send Email" on page 306 or "Send Test Email" on page 307 dialogs. See also: "Email header settings" on page 98. For all presets, the password is not saved and must be re-entered when sending emails.

- The **Add**, **Edit** and **Delete** buttons let you create and manage the presets.
- **SMTP Host Settings:** These settings can be made or edited after clicking the Add or Edit button.

- **Name:** The name of the preset. This will show up in the Send Email dialog.
- **Host:** The SMTP server through which the emails are to be sent. Can be a host (mail.domain.com) or an IP address.
- **Port:** The specified port number will be added to the host name, for example: smtp.mandrillapp.com:465.
- **Use authentication:** Check if a user name and password are needed to send emails through the host.
- **Start TLS:** Enabled if authentication is checked. Sends emails through Transport Layer Security (TLS), which is sometimes referred to as SSL.
- **User:** Enter the user name used to connect to the SMTP server.
- **Restore Defaults:** There are three default presets, each for working with a different Email Service Provider (ESP): Mandrilapp.com, Sendgrid and Mailgun (see "Using an ESP with PrintShop Mail Connect" on page 513).
- **Apply:** Apply the new settings without closing the Preferences dialog.

## Emmet Preferences

Emmet is a framework that enables the lightning-fast creation of HTML code through the use of a simple and effective shortcut language resembling CSS Selectors (see "Emmet" on page 87). The Emmet functionality is available in the HTML and CSS source editors of Connect Designer. Emmet transforms abbreviations for HTML elements and CSS properties to the respective source code.

This is, for example, the abbreviation for a <div> element with the class `row`:

```
div.row
```

On pressing the Tab key, this abbreviation is transformed to:

```
<div class="row"></div>
```

To learn more about Emmet itself, please see their website [Emmet.io](http://Emmet.io) and the [Emmet.io documentation](http://Emmet.io/documentation).

## Note

Emmet is a plugin. All options listed below are Emmet's default options. They are not specifically adjusted for Connect.

## Common Emmet preferences

- **Expand abbreviations by Tab key:** Check to enable the [Expand Abbreviation](#) function.
- **... in files with extension:** Enter a comma-separated list of all file extensions in which expand abbreviation will work.
- **Upgrade web editors:** This Emmet option doesn't affect how Emmet works in Connect Designer.
- **Extensions Path:** Choose a folder where to put json and js files to extend Emmet. This includes custom snippets, preferences and syntax profiles. For more information see [Customization](#).

## Emmet Abbreviation Preferences

This Preferences tab lets you add and manage custom abbreviations. All standard abbreviations can be found in Emmet's documentation: [Abbreviations](#).

If there is no need to transform the text while expanding it, create an Emmet snippet instead (see below).

- **New:** Add a new abbreviation.
  - **Name:** The name of the abbreviation is also its trigger.
  - **Context:** The context in which the abbreviation is enabled (HTML, CSS, etc.).
  - **Description:** A short description of the abbreviation .
  - **Pattern:** This defines what an abbreviation expands to. Since Emmet is mostly used for writing HTML/XML tags, abbreviation definition uses XML format to describe elements; see [Abbreviation types](#).
  - **Automatically insert:** This standard option doesn't affect how Emmet works in Connect Designer.
- **Edit:** Edit the currently selected abbreviation.
- **Remove:** Remove the currently selected abbreviation.

- **Import:** Click to open a browse dialog to import an XML file containing exported abbreviations. The imported abbreviations are added to the current list.
- **Export:** Click to open a Save as dialog to export all the abbreviations in an XML file that can be shared and re-imported.
- **Preview box:** Shows what the selected abbreviation is expanded to.
- **Restore Defaults:** clear all custom abbreviations.
- To temporarily disable an abbreviation, uncheck the checkbox next to the name of the abbreviation in the list.

### Emmet Output Preferences

The Output Preferences dialog is used to control how the expanded (output) code behaves when expanding abbreviations and snippets. There are 6 different dialogs to control output and, while they all have identical options, they control different output types: CSS, HAML, HTML, XML, XSL and the "Default" one controlling the rest of the types.

These options are equivalent to [Emmet's syntaxProfiles.json feature](#).

### Emmet Snippets Preferences

Emmet Snippet are similar to abbreviations in that they are expanded when the Tab key is pressed, but they are just blocks of plain text. Anything in a snippet will be outputted "as is", without any transformation.

- **New:** Click to create a new snippet.
  - **Name:** The name of the abbreviation is also its trigger.
  - **Context:** The context in which the snippet is enabled (HTML, CSS, etc.).
  - **Description:** A short description of the snippet.
  - **Pattern:** The pattern defines what a snippet expands to.
  - **Automatically insert:** This option doesn't affect how Emmet works in Connect Designer.
- **Edit:** Modify the currently selected snippet.
- **Remove:** Remove the currently selected snippet from the list.
- **Import:** Click to open a browse dialog to import an XML file containing exported snippets. The imported snippets are added to the current list.

- **Export:** Click to open a Save as dialog to export all the snippets in an XML file that can be shared and re-imported.
- **Preview box:** Shows what the selected snippet is expanded to.
- To temporarily disable a snippet, uncheck the checkbox next to the name of the snippet in the list.

## Emmet Variables Preferences

Variables are placeholders used in Emmet snippets to output predefined data. For example, the `html:5` snippet of HTML syntax has the following definition:

```
<!doctype html>\n<html lang="${lang}">...</body>\n</html>
```

In the example above, `${lang}` is used to refer `lang` variable defined in `variables` below. If your primary language is, for example, Russian, you can simply override `lang` variable with `ru` value and keep the original snippets. Also, you can override variable values with inline abbreviation attributes: `html:5[lang=ru]`.

- **Name:** The name of the variable. This should be a single alphanumeric string with no spaces or special characters. For example, the `myVar` name is referred to as `${myVar}`.
- **Value:** The value of the variable when the snippet is expanded.
- **New:** Click to create a new variable and define its name and value.
- **Edit:** Click to modify the currently selected Variable.
- **Remove:** Click to delete the currently selected Variable.

## **Language Setting Preferences**

The Language Setting preferences are used for selecting the language used in the User Interface.

- **Language:** Select a language from the options available in the drop down list.

## **Print Preferences**

### Available Printers Preferences

The Available Printers preferences control which printer definitions are available when generating print output or creating Output Presets. Any printer that is unchecked in this dialog

will not be visible in the "Model" drop-down of the Print Options dialog; see "Print Options" on page 411.

Available Printer Preferences:

- **Selected Printers:** Lists the available Printer Definition Files in the system. Note that these are not installed Windows printers or printer queues, but PrintShop Mail Connect Printer Definition Files.
- **Printer checkbox:** This checkbox selects/deselects all printers in the list. Click to check all, click again to uncheck all.

### General Print Preferences

The General Print Preferences are used to set communication settings with the PrintShop Mail Connect Server module that does the actual generation of print output. The Server module can be located on the same computer (hostname: localhost) or on a different machine. Multiple Designer modules can use a single Server module to generate Print output, as long as the appropriate hostname, username and password are provided. In essence, this can be used to create a single Print Server.

- **Print Server Settings group:**
  - **Protocol:** Use the drop-down to select whether to use the HTTP or the secure HTTPS protocol to connect to the Print Server.
  - **Hostname:** Enter the IP, machine name or URL of the Print Server. Default is `localhost`.
  - **Port:** Enter the port through which to communicate with the Print Server. Default is `9340`.
  - **Username:** Enter the username to authenticate to the Print Server. Default is `o1-admin`. This is set on the server's Server Security Settings.
  - **Password:** Enter the password to authenticate to the Print Server. Default is `secret`.
  - **Confirm Password:** Re-enter the password above.
- **External sort command timeout (seconds):** Enter the number of seconds to wait for an external sort command before giving up. External sort commands are set up in the [Sorting Options](#) page of the Output module.

## Print Measurements Preferences

- **Units:** Use the dropdown to specify the default measurements system used for dimensions of the template and boxes. In addition it defines the coordinates/position of box elements.
- **Flip insert guide axis:** Check this option to flip the axis on which guides are inserted. Normally, dragging a guide from a horizontal ruler inserts a horizontal guide (see "Guides" on page 180). With this option checked, dragging a guide from a horizontal ruler inserts a vertical guide.

## **Saving Preferences**

The saving preferences are a way control if and how often PrintShop Mail Connect saves your work in the background, and if how many backup files it creates when you save the template or data mapping configuration. See also: "Saving a template" on page 43.

### Auto Save

**After** a template or data mapping configuration has been saved for the first time, Connect Designer can auto save it with a regular interval.

- **Enable:** activate the Auto Save function.
- **Interval (minutes):** enter a number of minutes, e.g. 3 to auto-save the template or data mapping configuration every 3 minutes.

### Auto Backup

Connect Designer can automatically create a backup file when you **manually** save a template or data mapping configuration. The Auto Save function does **not** cause backup files to be created.

- **Enable:** activate the Auto Backup function.
- **Revisions to keep:** Enter the maximum number of backup files. When the maximum is reached, Auto Backup will overwrite the oldest file.
- **Destination:** Select the directory in which the backups should be stored.
  - **Original:** the directory in which the original file is stored.
  - **Other directory:** use the **Browse** button to select another directory.

Backup files have the same name as the original file with two underscores and a progressive number (without leading zeros) at the end: **originalname\_\_1.OL-template**, **originalname\_\_2.OL-template**, etc.

## Scheduling Preferences

The scheduling preferences are a way to control precisely how the PrintShop Mail Connect services work in the background.

### Scheduling options

This preference page defines what is considered a small or large job (anything in between is considered "medium" jobs). For a detailed description of all options, see Scheduling Preferences.

### Scheduling - Merge engine

This preferences page defines how different instances and speed units are attributed to different jobs when creating output documents. For a detailed description of all options, see Merge Engine Scheduling.

### Scheduling - Weaver engine

This preference page determines the number of engines launched, as well as their speed, when generating Print Output of any type. For a detailed description of all options, see Weaver Engine Scheduling.

## Scripting Preferences

The Scripting preferences define different options related to scripting with the PrintShop Mail Connect interface.

- **Designer scripting profiling group:**
  - **Number of iterations:** Enter the number of times to run scripts when running the [Profile Scripts](#) dialog. The default is 1000. Accepted values are 1 to 1000000000. Yes, that's 1 billion - which would take a *long* time to run!

### Profile Scripts dialog

The Script Profiler is accessible through the **Context > Profile Scripts** menu option. It runs the scripts in the template in order to verify the speed at which scripts in the [Scripts Pane](#) execute. It

helps greatly in troubleshooting performance issues caused by scripting (see also: "Testing scripts" on page 231).

When the dialog opens, the script profiler runs automatically, on 1000 instances of all the scripts by default (this can be changed through the "Scripting Preferences" on the previous page).

#### Note

The script profiler can take a while, so please be patient.

The results are shown as follows (the first in the line is indicated as **Total** and represents the totals of all the scripts underneath, representing a good overview of the scripts performance in the template):

- **Name:** The name of the script being executed.
- **Count:** As the profiler runs, Count shows the current number of iterations that have been run. This goes up to the total number of set instances and then stops. Hover with your mouse to display a tooltip indicating in which sections the scripts has run (and in which contexts).
- **Elapsed:** Displays the total elapsed time since the start of the session. The table entries are initially sorted based on the values in this column, from high to low. Hovering the mouse over it will display a tooltip that indicates the breakdown of the execution time across different execution stages.
- **Delta:** Displays the estimated difference in performance between the current session and the previous session. Uses average values, so should still work if the previous session was stopped after a different number of iterations. Will be empty if no previous data is available. Hover with your mouse to display a tooltip indicating the breakdown of the execution time across different execution stages.

## Script wizards

### Wizard types

Script wizards are simplified interfaces for common scripts in templates:

- **Text Script:** This is the default script that is created when a data field is dragged from the Data Model onto the page. See "Variable Data" on page 211.
- **Dynamic Image Script:** Provided that its selector refers to an image, this script dynamically changes the image for each record. See "Dynamic Images" on page 222.
- **Email Scripts.** The Email To Script is automatically added to any new Email context; it defines where the email should be sent for each record. Other Email scripts define other recipients, the subject of the email that is sent, and the PDF password. See "Email header settings" on page 98.
- **Barcode Script:** This script controls the contents of a Barcode. It is automatically added when a barcode is added to a template. See "Barcode" on page 109
- **Business Graphic Script:** This script controls the contents of a Pie Chart, Bar Chart or Line Chart.

The result of the script can be either text appearing on the page, an email address or subject, the barcode data, or a JSON string that is written to the attribute of an HTML element.

## Options

Here are the options visible in Script wizards:

- **Name:** The name of the script, making it easier to identify it.
- **Find:** The Selector or Text to apply the result of the script to.
- **Selector:** Uses CSS selectors to find the element to which the script applies
- **Text:** Uses text as a trigger for the script. The script applies to all instances of the text found in the template.
- **Wizard Results:** Displays a list of the data that is sent to replace the content that matches the script's selector:
  - **Prefix:** Static text to use before the set field. For example in Dynamic Image scripts, the default prefix is **images/**.
  - **Field:** A drop-down to select which field contents to use in the script. The field should contain a valid value. For an email script, for example, the field would have to contain an email address.
  - **Format:** A special formatting modifier applied to the Field; see "Formatting variable data" on page 217.

- **Suffix:** Static text to use after the set field. For Dynamic Image Scripts, the default suffix is `.jpg` and refers to the file extension.
- **[+]:** Adds a new line to the Wizard Results. Note that by default there is no line return between fields in the list. Adding `<br>` in the Suffix or Prefix field can establish a line return.
- **[-]:** Removes the currently selected line in the Wizard Results list.
- **Arrow Up:** Moves the currently selected line up one position.
- **Arrow Down:** Moves the currently selected line down one position.
- **Options** (only available in the Text Script wizard and the Dynamic Image wizard): specifies where and how the script inserts its results:
  - As **HTML**. HTML elements in the results are processed and displayed as HTML elements. For instance, `<b>this is bold</b>` will be displayed as **this is bold**. This is the default setting.
  - As **text**. This inserts the results as-is, meaning HTML tags and elements are displayed as text in the output. In this scenario, "`<br>`" shows up in the text and does not insert a line break.
  - As the value of an **attribute** of an HTML element. The selector of the script should be an HTML element. Which attributes are available depends on the selected HTML element. If the script's selector is an image (`<img>` element) for example, and the attribute is `src`, the script will modify the image's source. The script's results should be a valid value for the chosen attribute.
  - When checked, the option **Convert fields to JSON string** writes the results from the script into an attribute or text as a JSON string. This is useful for Web contexts where a front-end script can read this value easily.
- **OK:** Click to save any changes made to the script, apply the changes in the template, and close the dialog.
- **Cancel:** Click to close the dialog without saving changes.
- **Expand:** Click to convert the script generator to a regular script. Note that this action is not reversible once the regular script has been saved.
- **Apply:** Saves changes made to the script and applies the changes in the template without closing the dialog.

## Expanded Script window

When expanded, the Script window replaces all parts of the wizard below the Selector by a box in which the script can be typed. See "Writing your own scripts" on page 223.

## Chart Script dialog

These are the options in the Chart Script dialog:

- **Name:** The name of the script, making it easier to identify it.
- **Find:** The Selector or Text to apply the result of the script to.
  - **Selector:** Uses CSS selectors to find the element to which the script applies.
  - **Text:** Uses text as a trigger for the script. The script applies to all instances of the text found in the template.
  - **Selector and Text:** Uses text as a trigger for the script but only applies to text within the specified Selector.
- **Input Type:** Use the drop-down to select the source of the data to add to the Chart. The selection changes the options below:
  - **Static Labels:** Select to use a static number of data lines below. The chart will always have the same number of items.
    - **Data List:** Lists the data lines that are part of the Chart. Each line represents a segment of the pie as well as a label if they are shown.
      - **Labels:** The text of the label to display next to the Chart or within the legends.
      - **Values:** The value that will be used to create the Chart. This is the name of a field within the Data Model.
    - **Add:** Click to add an entry to the Data List. Opens the [Edit Label Properties](#) dialog.
    - **Delete:** Click to delete the currently selected line in the Data List.
    - **Move Up:** Click to move the currently selected line up one position.
    - **Move Down:** Click to move the currently selected line down one position.
  - **Dynamic Labels:** Select to use data from a detail table to fill the Chart dynamically. At least one detail table must be available in the [Data Model Pane](#) for this option to be functional.

- **Details:** Use the drop-down to select which detail table provides the data for the Chart.
- **Labels:** Use the drop-down to select which field within the detail table contains the text for the labels shown in the Chart.
- **Values:** Use the drop-down to select which field within the detail table contains the numerical values used to build the Chart.

### Conditional script dialog

Conditional script generators can show or hide elements on the page depending on certain conditions and values. They can be added by right-clicking any element in a template and clicking **Make Conditional**. If the current element does not have an ID, one will be automatically generated. See "Showing content conditionally" on page 219.

The options in the Conditional Script wizard are:

- **Name:** The name of the script, making it easier to identify it.
- **Selector:** The Selector or Text to apply the result of the script to.
  - **Selector:** Uses CSS selectors to find the element to which the script applies.
  - **Text:** Uses text as a trigger for the script. The script applies to all instances of the text found in the template.
  - **Selector and Text:** Uses text as a trigger for the script but only applies to text within the specified Selector.

For more information about Selectors see "Using the Text Script Wizard" on page 214.

- **Action:** Use the drop-down to select whether to **Show** or **Hide** the element when the condition below is true.
- **Data Field:** Use the drop-down to select which data field in the record the condition will be based on.
- **Condition:** Select which kind of condition is applied. Possible options are: **Equal to**, **Not equal to**, **Contains**, **Does not contain**, **Begins with**, **Ends with**.
- **Value:** The value used for the conditional check.

For example, you could check whether the value in the data field "Gender" is "Equal To" the value `Mr`, in order to show a paragraph or an image applying only to male customers.

## Section properties dialogs

### Email Section Properties

#### Properties Tab

The properties for an Email section are minimal and contain the following options:

- **Name:** Enter the name of the Section in the Email Context. This has no effect on output.
- **Subject:** Enter a name for the default Subject of any email sent out. This is superseded by an Email subject script, when it is present (see "Email header settings" on page 98).

#### Includes Tab

This tab defines what other resources are included in the output

- **Stylesheets:** What CSS stylesheets to use in producing the output. Stylesheets are loaded in the order shown, and styles in later Stylesheets overwrites earlier ones when the same selector is used.
- **JavaScript** (Web and Print only): Which JavaScript resources are included in the HTML header of the web output.
- **Up:** Move the selected StyleSheet or JavaScript up in priority.
- **Down:** Move the selected StyleSheet or JavaScript down in priority.

### Print Section Properties

The Section Properties dialog is separated in a few separate tabs depending on the Context in which it resides:

#### General tab (all Contexts)

- **Section group:**
  - **Name:** The name of the Section.
  - **Show PDF data mapping input as background image:** Check this option to display each page of a PDF data source when using a PDF data mapping configuration. Each page of the PDF is shown, separated by the appropriate records. Note that as many pages as there are in the PDF will be created in the section.

- **Pagegroup:**
  - **Size:** The named page size.
  - **Width:** The width of the page.
  - **Height:** The height of the page.
  - **Portrait:** Whether the page is portrait (otherwise, it's landscape).
- **Margins** group: Defines the margins where contents will not appear on the page.
  - **All sides:** Check to use the same margin setting for all sides. Note that this will copy the *Top* value to all margins, overwriting existing values.
  - **Top:** The top margin.
  - **Left:** The left margin.
  - **Bottom:** The bottom margin.
  - **Right:** The right margin.
- **Bleed group:** This group defines the bleeds for the sections added within the print context.
  - **All sides:** Check to define all bleeds identically using the **Top** value.
  - **Top:** The bleed at the top of the page.
  - **Left:** The bleed at the left of the page.
  - **Bottom:** The bleed at the bottom of the page.
  - **Right:** The bleed at the right of the page.

### Includes Tab

This tab defines what other resources are included in the output

- **Stylesheets:** What CSS stylesheets to use in producing the output. Stylesheets are loaded in the order shown, and styles in later Stylesheets overwrites earlier ones when the same selector is used.
- **JavaScript** (Web and Print only): Which JavaScript resources are included in the HTML header of the web output.
- **Up:** Move the selected StyleSheet or JavaScript up in priority.
- **Down:** Move the selected StyleSheet or JavaScript down in priority.

## Finishing tab

This tab defines finishing options for this section when it is printed.

- **Binding group:**
  - **Style:** What type of binding to request on the printer. This includes Stapled, Glued, Stitched, etc.
  - **Side:** On which side the binding occurs, such as Bottom, Left, top, etc.
  - **Location:** The location of the binding, such as Saddle, Side, Corner, etc.
  - **Angle:** If the binding should be done horizontally, vertically, or at an angle.
  - **Item count:** When certain binding styles are selected which require multiple items (such as Stapled or Sticked), use the Count option to specify the number of items the printer should use, or select Default to let the printer decide how many items to use.
  - **Area:** The area where the binding can be applied.
- **Hole making group:**
  - **Number of holes:** When certain binding styles are selected which require the printer to create holes in the paper, use this drop-down to select the number of holes that should be created, or use Default to let the printer decide.
  - **Pattern Catalog ID:** When the Number of Holes is not the default in the option above, use the drop-down to select the Pattern Catalog ID for the hole making.

## Sheet Configuration Tab

This tab defines how different Print Context Sections output on different Media and using different Master Pages.

There are multiple groups, each defining the settings for individual *position* within the section as it outputs: **First**, **Middle** and **Last** sheet, as well as a group for **Single** sheets.

This tab contains the following options:

- **Duplex:** Check to enable content to be printed on the back of each sheet. Your printer must support duplex for this option to work.
- **Tumble:** Check to enable tumble mode so pages are duplexed as in a notebook (on Portrait output, this would be equivalent to short-edge duplex).

- **Same for all positions:** Check to enable a single group below, which defines the same options for all positions in the document. If unchecked, individual position options are available.

The next options are identical for all positions, but of course only affects the position where you change the options:

- **Allow content on:** Selects on which face of the sheet content is allowed. If "Front only" or "Back Only" is selected, the other page may contain a Master Page, but no contents will be printed on it. As such it does not count in the "Content Page Number" and "Content Page Count" markers which can be inserted via the [Insert menu](#).
- **Media:** Defines the media that is used. If the Media has *Preprinted Media* defined, the selected preprinted media image is shown as a background to each page that correspond to the media's sheet position.
  - **Edit Script:** Click to open a Script Editor dialog. The script defines what Media is used, so it can be dynamically defined using data from the source record or the extracted record.
- **Master Page Front:** Defines the Master Page used for the front of the selected sheet's position. Disabled if "Back Only" is selected under Allow content on.
- **Master Page Back:** Defines the Master Page used for the front of the selected sheet's position. Disabled if "Back Only" is selected under Allow content on, or if Duplex is unchecked.

### Background Tab

This tab defines the background for the current Print Context Section. It contains the following options:

- **General group:**
  - **PDF:** Select the PDF to use as the section's background: a PDF Datamapper Input or a PDF Resource.
  - **Path:** The path to a PDF Resource. Enter a path and file name or click the Browse button to open the Select Image dialog; see "Select Image dialog" on page 308.
  - **Position:**
    - **Centered:** The PDF will be centered on the page and will not be resized.
    - **Fit to Media:** The PDF will be resized to fit the Media.

- **Absolute.** Set a position for the top left corner of the PDF. The PDF will not be resized.
- **Top:** the distance between the top of the page and the top of the PDF
- **Left:** the distance between the left side of the page and the left side of the PDF.
- **Pages group:**
  - **All:** Use all pages in the PDF file as the section's background. For each page in the PDF file one page will be added to the Print section.
  - **Pages:** Select a number of pages from the PDF file. For each page in the range, one page will be added to the Print section.

## Web Section Properties

The Web Section Properties defines some of the web page properties, especially details appearing in the header.

### Properties Tab

- **Section Group:**
  - **Name:** Enter the name of the Section in the Web Context. This has no effect on output.
  - **Page Title:** Enter the title for the page. This is the contents of the <title> HTML tag.
  - **Shortcut Icon:** Enter the path to the favicon.ico file, for instance *images/favicon.ico*. If a valid favicon image is dragged to the Web Section, it will automatically be set as a shortcut icon.
- **Meta Information Group:** This lists all <meta> tags that are added to the header of the HTML file generated in the output. For more information on <meta> tags, see [W3Schools - HTML meta tag](#).
  - **Type:** Select the type of <meta> tag. This is either *name* or *http-equiv*.
  - **Value:** Enter the value of the <meta> tag, for instance when *name* is selected, this could be *keywords* or *description*.
  - **Content:** Enter the desired contents of the <meta> tag.
  - **Add:** Click to add a new <meta> tag. to the list.
  - **Delete:** Click to delete the currently selected <meta> tag.

- **Move up:** Click to move the currently selected <meta> tag up one position.
- **Move down:** Click to move the currently selected <meta> tag down one position.

### Includes Tab

This tab defines what other resources are included in the output

- **Stylesheets:** What CSS stylesheets to use in producing the output. Stylesheets are loaded in the order shown, and styles in later Stylesheets overwrites earlier ones when the same selector is used.
- **JavaScript** (Web and Print only): Which JavaScript resources are included in the HTML header of the web output.
- **Up:** Move the selected StyleSheet or JavaScript up in priority.
- **Down:** Move the selected StyleSheet or JavaScript down in priority.

### **Arrange Sections**

The Arrange dialog is used to change the order of sections within a context. To access the Arrange dialog, right-click on any section or the context containing them, and click Arrange.

- **Name:** Displays the name of each section within the context.
- **Move Up:** Click to move the currently selected section up one position.
- **Move Down:** Click to move the currently selected section down one position.

### **Section Includes**

This dialog defines what other resources are included in the output

- **Stylesheets:** What CSS stylesheets to use in producing the output. Stylesheets are loaded in the order shown, and styles in later Stylesheets overwrites earlier ones when the same selector is used.
- **JavaScript** (Web and Print only): Which JavaScript resources are included in the HTML header of the web output.
- **Up:** Move the selected StyleSheet or JavaScript up in priority.
- **Down:** Move the selected StyleSheet or JavaScript down in priority.

## Send Email

The Send Email dialog is used to generate mail output and send it to each recipient in the Record Set.

Options for this dialog:

- **From** group:
  - **Name**: Enter the name that should appear when sending emails. The name is optional.
  - **Email**: Enter the email address that will appear as a Sender to the email recipient. A single email address should be written.
- **Records** group:
  - **All**: Select to send to all records in the Record Set.
  - **Selection**: Select this open and enter a range of records that should be sent.
- **Attachments**:
  - **Print context as PDF**: If a Print Context exists in the template, its output will be generated and a PDF version of it will be attached to the outgoing email.
  - **Web Page context as HTML**: If a Web Page Context exists in the template, its output will be generated as a single HTML file with all required resources embedded in the file. This HTML file is then added as an attachment to the outgoing email.
- **Outgoing mail settings**:
  - **Presets**: Use the drop-down to select a preset. These presets are configured in the Email (SMTP) preferences; see "Email SMTP settings" on page 98.

### Note

It is recommended to use an Email Service Provider to get access to tools that give you full control over your mailings, like open rates, click through rates etc. See "Using an ESP with PrintShop Mail Connect" on page 513.

- **Host:** The SMTP server through which the emails are to be sent. Can be a host (mail.domain.com) or an IP address. You can specify a port number as part of the host name, for example: smtp.mandrillapp.com:465.
- **Use authentication:** Check if a username and password are needed to send emails through the host.
- **Start TLS:** Enabled if authentication is checked. Sends emails through Transport Layer Security (TLS), which is sometimes referred to as SSL.
- **User:** Enter the username used to connect to the SMTP server.
- **Password:** Enter the password for the above username.

## Send Test Email

The Send Test Email dialog is used to generate mail output of a certain number of records and send them to one recipient. To open this dialog, select **File > Send Test Email**, on the menu. For more information about the process of sending out email and the possible settings, see "Generating Email output" on page 509.

Options for this dialog:

- **Fromgroup:**
  - **Name:** Enter the name that should appear when sending emails. The name is optional.
  - **Email:** Enter the email address that will appear as a Sender to the email recipient. A single email address should be written.
- **Togroup:**
  - **Email address(es):** Enter one or more email addresses where the test emails are sent. Multiple emails can be separated by semicolons (;), and can be in the same format as above. Note that every email here will receive all the emails for the record-range below.
  - **Use Litmus:** Check to also send the emails to the Litmus test email set in the Email Preferences (to go to the Email Preferences, select **Window > Preferences**, click the arrow next to **Email**, and then click **General**). Disabled if no Litmus email is set. Also see this how-to: [Test your emails with Litmus](#).

- **Records:**
  - Select **All**, or click **Selection** and enter the range of records that should be sent. Removing the range disables the selection and sends emails to all records in the record set.
- **Attachments:**
  - **Print context as PDF:** If a Print context exists in the template, its output will be generated and a PDF version of it will be attached to the outgoing email.
  - **Web Page context as ZIP archive:** If a Web context exists in the template, its output will be generated as an HTML file. A ZIP archive, containing the HTML file and all required resources, will be attached to the outgoing email.
- **Outgoing mail settings:**
  - **Host:** The SMTP server through which the emails are to be sent. Can be a host (mail.domain.com) or an IP address. You can specify a port number as part of the host name, for example: smtp.mandrillapp.com:465.
  - **Use authentication:** Check if a username and password are needed to send emails through the host.
  - **Start TLS:** Enabled if authentication is checked. Sends emails through Transport Layer Security (TLS), which is sometimes referred to as SSL.
  - **User:** Enter the username used to connect to the SMTP server.
  - **Password:** Enter the password for the above username.

## Select Image dialog

The Select Image dialog lets you select an image, depending on where the image is located.

**Resources:** lists the images that are present in the Images folder on the Resources pane. A preview of the selected image will be shown at the right.

**Disk:** lets you select an image file that resides in a folder on a hard drive that is accessible from your computer. A preview of the selected image will be shown below.

- **Path.** The complete syntax of the path is: file://<host>/<path>. Note: if the host is "localhost", it can be omitted, resulting in file:///<path>, for example: file:///c:/resources/images/image.jpg.
- **Browse:** opens an explorer window to browse folders and select an image.

**Url** lets you select an image file from a specific web address. Select the protocol and then enter the URL (for example, <http://www.mysite.com/images/image.jpg>). A preview of the selected image will be shown below.

- **Protocol:** **http** or **https**.

The option **Save with template**, available when choosing an image from disk or by URL, inserts the file in the **Images** folder on the **Resources** pane. If not saved with the template, the image will remain external.

#### Note

External images need to be available when the template is merged with a record set to generate output, and their location should be accessible from the machine on which the template's output is produced. External images are updated (retrieved) at the time the output is generated.

### Sheet Configuration dialog

The Sheet Configuration dialog defines how different Print context sections output on different Media (see "Master Pages" on page 76) and using different Master Pages (see "Media" on page 80).

There are multiple groups, each defining the settings for pages grouped by their position within the section as it outputs: **First**, **Middle**, **Last** and **Single** sheets.

The first option defines **Duplex** printing, which also enables or disable the **Back** side of each sheet.

If Duplex is enabled, you can also check **Tumble** to duplex pages as in a calendar, and **Facing pages** to have the margins of the section switch alternately, so that pages are printed as if in a magazine or book.

If the option **Same for all positions** is checked, the same Master Page and Media will be applied to every page in the Print section.

Each group defines:

- **Allow content on:** Selects on which face of the sheet content is allowed. If **Front only** or **Back only** is selected, the page acts as a Simplex page even if Duplex printing is enabled.
- **Media:** Defines the media that is used. If the Media has Virtual Stationery defined, the selected image is shown as a background to each page that corresponds to the media's sheet position.
- **Master Page Front:** Defines the Master Page used for the front of the selected sheet's position.
- **Master Page Back:** Defines the Master Page used for the back of the selected sheet's position.

## Style sheets dialog

The Stylesheet editor dialog is used to edit CSS style sheet resources. For information on the use of style sheets, see "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.

This dialog lets you edit the Global style sheet (`context_all_styles.css`, which by default applies to all contexts), and the style sheet that applies to the context that is currently being edited in the workspace: Print (`context_print_style.css`) or Email (`context_email_style.css`).

To open this dialog, select **Edit > Stylesheets....**

- **Context:** Use the drop-down to select **Global** (all contexts) or the context that is open in the workspace, such as **Print**. Selecting a context shows all its CSS rules in the **Rule List**.
- **Show:** Use the drop-down to select whether to show all CSS rules or limit to certain types: **Class**, **ID** or **Element** rules.
- **Rule List:** Displays the list of rules in the currently selected style sheet.
- **Rule Display:** Displays the contents of the currently selected rule in the **Rule List**.
- **New:** Click to create a new rule with the Edit Rule dialog. See "New/Edit Rule dialog" on the next page.
- **Edit:** Click to edit the currently selected rule in the *Rule List* using the Edit Rule dialog. See "New/Edit Rule dialog" on the next page.
- **Delete:** Click to delete the currently selected rule in the *Rule List*.
- **Duplicate:** Click to create a duplicate of the currently selected rule in the *Rule List* using the Edit Rule dialog. The default name for the new rule is the name of the current one plus "-duplicated". See "New/Edit Rule dialog" on the next page.

- **Move Up:** Move the currently selected rule in the *Rule List* up one position in the list.
- **Move Down:** Move the currently selected rule in the *Rule List* down one position in the list.
- **Save:** Click to save all changes to the stylesheet and close the dialog.
- **Cancel:** Click to close the dialog without saving any changes.

## New/Edit Rule dialog

The New/Edit Rule dialog shows the properties for a specific CSS selector and how it affects all elements subject to that selector.

At any point you can click on the **Advanced** button to see the Advanced Stylesheet Rule. See "Advanced Stylesheet Rule" on page 314.

- **Name:** The CSS selector to which this rule applies. Since CSS selectors are not specific to PrintShop Mail Connect, any selector used in regular CSS can also be used here. See [CSS Selectors on W3Schools](#) for a simple reference page.

## Type Tab

- **General group:**
  - **Font:** Select the font used to display text. This is equivalent to the CSS `font-family` property.
  - **Size:** Enter the size in measure, named size or percentage. This is equivalent to the CSS `font-size` property.
  - **Color:** Select a named font color as defined in the [Colors Editor](#), create a new color or enter a color manually for text to be displayed. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). This is equivalent to the CSS `color` property.
  - **Background Color:** Select a named font color as defined in the [Colors Editor](#), create a new color or enter a color manually for the background color of the element. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). This is equivalent to the CSS `background-color` property.
- **Spacing group:**
  - **Letter Spacing:** Set the space between characters in a text in measure or percentage. This is equivalent to the CSS `letter-spacing` property.

- **Word Spacing:** Set the space between each word in a text in measure or percentage. This is equivalent to the CSS `word-spacing` property.
- **Whitespace:** Specify how to handle white spaces inside of an element. See [CSS White-Space](#) for details. This is equivalent to the CSS `white-space` property.
- **Style group:** Check any option to apply the selected style to text within the element. This list shows the CSS property and value for each of the options.
  - **Bold:** Sets the `font-weight` to `700`.
  - **Italic:** Sets `font-style` to `italic`.
  - **Underline:** Sets `text-decoration` to `underline`.
  - **Strikethrough:** Sets `text-decoration` to `line-through`.
  - **Subscript:** Sets `vertical-align` to `super`.
  - **Superscript:** Sets `vertical-align` to `sub`.
  - **Capitalize:** Sets `text-transform` to `capitalize`.
  - **Uppercase:** Sets `text-transform` to `uppercase`.
  - **Lowercase:** Sets `text-transform` to `lowercase`.
  - **Small-caps:** Sets `font-variant` to `small-caps`.

## Formats Tab

- **General group:**
  - **Line-height:** Specify the height of each line in the element's text, in measure or percentage. Note that this is not spacing between lines, but rather the complete height of the line itself including the text. Equivalent to the `line-height` property.
  - **Align:** Select how text should be aligned, such as `left`, `center`, `right` or `justify`. Equivalent to the `align` property.
  - **First Indent:** Specify the indentation of the first line of each paragraph in the element. Equivalent to the `text-indent` property.
  - **Display:** Select how to display the element. This can also be used to hide an element completely using the `none` option. See [CSS Display](#). Equivalent to the `display` property.
- **Breaks group:**
  - **Before:** Specifies how to handle page breaks before the element. Equivalent to the `page-break-before` property.

- **Inside:** Specifies whether to accept page breaks within the paragraph. Equivalent to the `page-break-inside` property.
- **After:** Specifies how to handle page breaks after the element. Equivalent to the `page-break-after` property.
- **Widows:** Specifies how to handle widows within the paragraph (lines appearing alone on the next page if the paragraph does not fit on the current one). Equivalent to the `widows` property. Widows and orphans are ignored if the `page-break-inside` property is set to `avoid`.
- **Orphans:** Specifies how to handle orphans within the paragraph (lines appearing alone at the end of a page if the paragraph does not fit on the current one). Equivalent to the `orphans` property.

### Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:
  - **All sides:** Check to set all padding to use the Top value. Equivalent to the `border` property.
  - **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the `border-left`, `border-top`, `border-right` and `border-bottom` properties.
- **Margin group:** Defines margins (spacing outside the element) in measure or percentage:
  - **All sides:** Check to set all margins to use the Top value. Equivalent to the `margin` property.
  - **Top, Left, Bottom, Right:** Set the margin for each side. Equivalent to the `margin-left`, `margin-top`, `margin-right` and `margin-bottom` properties.

### Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the `border` property.
- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the `border-style` property.

- **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the `border-color` property.

## Advanced Stylesheet Rule

The Advanced editor is used to manually input rules. Note that to use this dialog, basic knowledge of CSS rules is a pre-requisite, as no check is currently done to verify that properties and values are correct.

- **Property List:** Lists all the currently available properties for the selector.
  - **Property:** The name of the property. This must correspond exactly to a known property (see [CSS Reference](#)). An autocompletion drop-down displays to show possible values when typing.
  - **Value:** The value for the given property. The values must be valid for that property, see the CSS Reference link above and check the property for valid values.
- **New:** Click to create a new line and type in the property.
- **Delete:** Click to delete the currently selected property in the *Property List*.
- **Move Up:** Move the currently selected property in the *Property List* up one position in the list.
- **Move Down:** Move the currently selected property in the *Property List* down one position in the list.

## Table Formatting dialog

The Table Formatting dialog defines how the table looks and reacts.

### Table Tab

- **General group:**
  - **Width:** Set the width of the table in measure or percentage. Equivalent to the CSS `width` property.
  - **Height:** Set the height of the table in measure or percentage. Equivalent to the CSS `height` property.
  - **Angle:** Set the rotation angle of the table in clockwise degrees. Equivalent to the CSS `transform: rotate` property.

- **Corner radius:** Set the radius of rounded border corners in measure or percentage. Equivalent to the CSS `border-radius` property.
- **Display:** Use the drop-down or type in the value for how to display the table. Equivalent to the CSS `display` property.
- **Overflow:** Use the drop-down or type in the value for how to handle overflow (text that does not fit in the current size of the box). Equivalent to the CSS `overflow` property.
- **Text wrap group:**
  - **Float:** Use the drop-down or type in the value for how to float the table, if the table is not in an absolute position. Equivalent to the CSS `float` property.
  - **Clear:** Use the drop-down or type in the value for clearing pre-existing alignments. Equivalent to the CSS `clear` property.
- **Positioning:**
  - **Position:** Use the drop-down or type in the value for the type of positioning for the table. Equivalent to the CSS `position` property.
  - **Top:** Set the vertical offset between this table and its parent's top position. Equivalent to the CSS `top` property.
  - **Left:** Set the horizontal offset between this table and its parent's left position. Equivalent to the CSS `left` property.
  - **Bottom:** Set the vertical offset between this table and its parent's bottom position. Equivalent to the CSS `bottom` property.
  - **Right:** Set the horizontal offset between this table and its parent's left position. Equivalent to the CSS `right` property.
  - **Z-index:** Set the z-index of the table. The z-index defines in which order elements appear. Equivalent to the CSS `z-index` property.
- **Breaks group:**
  - **Before:** Specifies how to handle page breaks before the table. Equivalent to the CSS `page-break-before` property.
  - **Inside:** Specifies whether to accept page breaks within the table. Equivalent to the CSS `page-break-inside` property.
  - **After:** Specifies how to handle page breaks after the table. Equivalent to the CSS `page-break-after` property.

- **Widows:** Specifies how to handle widows within the table (rows appearing alone on the next page if the table does not fit on the current one). Equivalent to the CSS `widows` property. Widows and orphans are ignored if the `page-break-inside` property is set to `avoid`.
- **Orphans:** Specifies how to handle orphans within the tables (rows appearing alone at the end of a page if the table does not fit on the current one). Equivalent to the CSS `orphans` property.

## Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:
  - **All sides:** Check to set all padding to use the Top value. Equivalent to the CSS `padding` property.
  - **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the CSS `padding-left`, `padding-top`, `padding-right` and `padding-bottom` properties.
- **Margin group:** Defines margins (spacing outside the element) in measure or percentage:
  - **All sides:** Check to set all margins to use the Top value. Equivalent to the CSS `margin` property.
  - **Top, Left, Bottom, Right:** Set the margin for each side. Equivalent to the CSS `margin-left`, `margin-top`, `margin-right` and `margin-bottom` properties.

## Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the CSS `border` property.
- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the CSS `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the CSS `border-style` property.

- **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `border-color` property.

## Background Tab

- **General group:**
  - **Color:** Specify the color of the table background. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `background-color` property.
- **Background image group:**
  - **Source:** click the **Select Image** button to select an image via the "Select Image dialog" on page 308. Equivalent to the CSS `background` property.
  - **Size:** select `auto`, `cover` or `contain` (for an explanation see [http://www.w3schools.com/cssref/css3\\_pr\\_background-size.asp](http://www.w3schools.com/cssref/css3_pr_background-size.asp)), or type the width and height of the image in a measure (e.g. `80px 60px`) or as a percentage of the parent element's size (e.g. `50% 50%`). Equivalent to the CSS `background-size` property.
  - **Position:** select the position for the background-image. Equivalent to the CSS `background-position` property.

## Table Cell Formatting dialog

### Cell Tab

- **Width:** Set the width of the table in measure or percentage. Equivalent to the CSS `width` property.
- **Height:** Set the height of the table in measure or percentage. Equivalent to the CSS `height` property.
- **Vertical Align:** Specify how text is vertically aligned in the cell: `top`, `middle`, `bottom` or `baseline`. With the `baseline` value all the table data share the same baseline. Often this has the same effect as the `bottom` value. However, if the fonts are in different sizes, `baseline` looks better.

## Type Tab

- **General group:**
  - **Font:** Select the font used to display text, equivalent to the CSS `font-family` property.
  - **Size:** Enter the size in measure, named size or percentage, equivalent to the CSS `font-size` property.
  - **Color:** Select a named font color as defined in the [Colors Editor](#), create a new color or enter a color manually for text to be displayed. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `color` property.
- **Spacing group:**
  - **Letter Spacing:** Set the space between characters in a text in measure or percentage. Equivalent to the CSS `letter-spacing` property.
  - **Word Spacing:** Set the space between each word in a text in measure or percentage. Equivalent to the CSS `word-spacing` property.
  - **Whitespace:** Specify how to handle white spaces inside of an element. Equivalent to the CSS `white-space` property. See [CSS White-Space](#) for details.
- **Style group:** Check any option to apply the selected style to text within the element:
  - **Bold:** Sets the `font-weight` to 700.
  - **Italic:** Sets the `font-style` to `italic`.
  - **Underline:** Sets the `text-decoration` to `underline`.
  - **Strikethrough:** Sets the `text-decoration` to `line-through`.
  - **Subscript:** Sets the `vertical-align` to `super`.
  - **Superscript:** Sets the `vertical-align` to `sub`.
  - **Capitalize:** Sets the `text-transform` to `capitalize`.
  - **Uppercase:** Sets the `text-transform` to `uppercase`.
  - **Lowercase:** Sets the `text-transform` to `lowercase`.
  - **Small-caps:** Sets the `font-variant` to `small-caps`.

## Spacing Tab

- **Padding group:** Defines padding (spacing inside the element) in measure or percentage:
  - **All sides:** Check to set all padding to use the Top value. Equivalent to the CSS `border` property.
  - **Top, Left, Bottom, Right:** Set padding for each side. Equivalent to the CSS `padding-left`, `padding-top`, `padding-right` and `padding-bottom` properties.

## Border Tab

- **Same for all sides:** Defines the border properties for all sides using the Top properties. Equivalent to the `border` property.
- **Top, Left, Bottom, Right:** Each group defines the following properties:
  - **Width:** Specify the thickness of the border. Equivalent to the CSS `border-width` property.
  - **Style:** Specify the style of the border such as `solid`, `dashed` or `dotted`. Equivalent to the CSS `border-style` property.
  - **Color:** Specify the color of the border. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `border-color` property.

## Background Tab

- **General group:**
  - **Color:** Specify the color of the table cell background. The color value must be a valid [HTML Color Name](#), or a valid [HTML Hex Color](#). Equivalent to the CSS `background-color` property.
- **Background image group:**
  - **Source:** click the **Select Image** button to select an image via the "Select Image dialog" on page 308. Equivalent to the CSS `background` property.
  - **Size:** select `auto`, `cover` or `contain` (for an explanation see [http://www.w3schools.com/cssref/css3\\_pr\\_background-size.asp](http://www.w3schools.com/cssref/css3_pr_background-size.asp)), or type the width and height of the image in a measure (e.g. `80px 60px`) or as a percentage of the

parent element's size (e.g. 50% 50%). Equivalent to the CSS `background-size` property.

- **Position:** select the position for the background-image. Equivalent to the CSS `background-position` property.

## Menus

The following menu items are shown in the Designer menu:

### File Menu

- **New...:** Opens the **New (Select a Wizard)** dialog. You can choose from the Email, Print or Web Template Wizards. See "Templates" on page 42.
- **Open:** Opens a standard File Open dialog. This dialog can be used to open Templates. See "Templates" on page 42 and Data Mapping Configuration.
- **Open Recent:** List the most recently opened Templates and configurations. Clicking on a template will open it in the Designer module.
- **Close:** Close the currently active Template. If the file needs to be saved, the appropriate Save dialog will open.
- **Close All:** Close any template. If any of the files needs to be saved, the Save Resources dialog opens.
- **Close Others:** Close all templates except the one that is currently active in the workspace.
- **Save:** Saves the current Template to its current location on disk. If the file has never been saved, the Save As dialog appears instead.
- **Save All:** Saves all open files. If any of the open files have never been saved, the Save As dialog opens for each new unsaved file.
- **Save As...:** Saves the current file to a new location on disk.
- **Revert:** Reverts all changes to the state in which the file was opened or created.
- **Add Data:** Adds data to the open template. See "Loading data" on page 204 .
  - **From File Data Source...:** Opens the dialog to add a new data file.
  - **From Database Data Source...:** Opens the Edit Database Configuration dialog.
  - **Generate Counters:** Opens the Generate Counter Wizard to create a custom counter as a data source.

- **Print:** Opens the "Print Options" on page 411 dialog.
- **Print Presets:** Selecting this option allows you to create or modify Printing Presets, which can be saved and used in print runs thereafter.
  - **Job Creation Presets:** Click to open the "Job Creation Presets" on page 401 dialog.
  - **Output Creation Presets:** Click to open the "Output Creation Settings" on page 410 dialog.
- **Proof Print:** Opens the "Print Options" on page 411 dialog as a Proof Print dialog which limits the number of records output. The options themselves are identical to the regular Print Output dialog.
- **Send Email:** Opens the "Send Email" on page 306 dialog; see "Generating Email output" on page 509.
- **Send Test Email:** Click to open the "Send Test Email" on page 307 dialog.
- **Exit:** Closes the software. If any of the files needs to be saved, the Save Resources dialog opens.

## Edit Menu

- **Undo <action>:** Undoes the previous action that was done.
- **Redo <action>:** Redoes an action that was previously undone.
- **Cut:** Cuts the currently selected text, object or element and puts it on the clipboard.
- **Copy:** Copies the the currently selected text, object or element to the clipboard.
- **Copy to snippet:** Creates a new snippet from the selected text, object or element.
- **Paste:** Takes the current clipboard content and pastes it at the pointer location.
- **Delete Browser Element:** Removes the currently selected element in the workspace.
- **Find/Replace:** Only active while inside the Workspace. Opens the [Find/Replace](#) dialog.
- **Stylesheets...:** Open the "Style sheets dialog" on page 310. See "Styling and formatting" on page 170 and "Styling templates with CSS files" on page 171.
- **Colors...:** Opens the [Colors Editor](#) dialog. See "Colors" on page 190.
- **Fonts...:** opens the "Fonts Manager" on page 268. See "Fonts" on page 194.
- **Locale...:** Opens the [Locale Settings](#) dialog. See "Locale" on page 201.
- **Color Settings...:** Opens the [Color Settings](#) dialog. See "Colors" on page 190.

## Insert Menu

- **Image:** Inserts an image using a resource that is local to the template, a resource on disk or a URL. See "Images" on page 158.
- **Text:**
  - **Wrap in span:** Wraps selected text in a <span> element. The ID or class of the span can be used as a selector for scripts and styles.
- **Special Characters:** Displays a categorized list of special HTML characters that can be inserted at the current pointer location. When a character is clicked, its HTML Entity is inserted. This includes:
  - **Symbols:** Use the list to insert a special symbol such as Copyright, Trademark, or Ellipsis.
  - **Markers:** Use the list to insert pagination markers that are replaced with specific page numbering:
    - **Page Number:** This marker is replaced by the current page number in the document. Even if the page number is not used on certain pages, those page are still added to the page count.
    - **Page Count:** This marker is replaced by the total number of pages in the document, including pages with no contents.
    - **Content Page Number:** This marker is replaced by the current page number (with contents) in the document.
    - **Content Page Count:** This marker is replaced by the total number of pages that have contents in them, in the document. A page with contents is a page that is part of a section that has variable data on it. A page with a Master Page but no contents (set in the Sheet Configuration tab of the "Print Section Properties" on page 300) is not included in the Content Page Count.
    - **Sheet Number:** This marker is replaced by the current sheet number (physical piece of paper with two sides, or pages) in the document. This is equivalent to half the page number, for example if there are 10 pages, there will be 5 sheets.
    - **Sheet Count:** This marker is replaced by the total number of sheets in the document, whether or not they have contents.
  - **Dashes and Spaces:** Use the list to insert special dashes, such as an em-dash, and spaces, such as non-breaking spaces or an en-space. (The HTML code inserted for the dash or space is visible on the Source tab of the workspace.)
  - **Arrows:** Use the list to insert directional arrows (in one of four directions).

- **Geometric Shapes:** Use the list to insert a special geometric shape, such as circles, triangles and squares.
- **Date:** Click to open the "Date" on page 154 dialog to add a date to the template based on the current system's date and time.
- **Wrap in box:** Puts the element in which the cursor is located in an inline box (a <div>).
- **Table**
  - **Thead, tbody, tfoot:** Insert a header, body or footer (if not already present) in the current table.
  - **Standard:** Inserts a table with a specific number of columns and rows through the Standard Table Wizard; see "Table" on page 163.
- **Table Elements:**
  - **Insert Row Above:** Inserts a row above the current one. The row configuration, such as merged cells and cell styles, is duplicated, but contents is not.
  - **Insert Row Below:** Inserts a row below the current one. The row configuration, such as merged cells and cell styles, is duplicated, but contents is not.
  - **Insert Column Before:** Inserts a column to the left of the current one. The column configuration, such as merged cells and cell styles, is duplicated, but contents is not.
  - **Insert Column After:** Inserts a column to the right of the current one. The column configuration, such as merged cells and cell styles, is duplicated, but contents is not.
- **Common Elements:**
  - **Paragraph...:** Click to open a dialog to add a <p> element; see "Text and special characters" on page 167.
  - **H1 through H6...:** Click to open a dialog to add a <h1> to <h6> element; see "Text and special characters" on page 167.
  - **Address...:** Click to open a dialog to add an <address> element.
  - **Preformatted...:** Click to open a dialog to add a <pre> element.
- **Structural Elements:**
  - **Div...:** Click to open a dialog to add a <div> element; see "Boxes" on page 150
  - **Span...:** Click to open a dialog to add a <span> element; see "Boxes" on page 150
  - **Article...:** Click to open a dialog to add an <article> element

- **Section...**: Click to open a dialog to add a <section> element (the HTML element, not a section in a context).
- **Header...**: Click to open a dialog to add a <header> element.
- **Footer...**: Click to open a dialog to add a <footer> element.
- **Nav...**: Click to open a dialog to add a <nav> element.
- **Aside...**: Click to open a dialog to add an <aside> element.

#### Note

Article, Section, Header, Footer, Nav and Aside are HTML5 semantic elements; see [http://www.w3schools.com/html/html5\\_semantic\\_elements.asp](http://www.w3schools.com/html/html5_semantic_elements.asp)

- **Business Graphic**: Displays a list of available business graphic object to be inserted:
  - **Insert Pie Chart**: Click to insert a new Pie Chart and open the Pie Chart script dialog.
  - **Insert Bar Chart**: Click to insert a new Bar Chart and open the Bar Chart script dialog.
  - **Insert Line Chart**: Click to insert a new Line Chart and open the Line Chart script dialog.
- **Barcode**: Displays a list of available barcodes. Click on one to insert it in the page. See "Barcode" on page 109.

#### Format Menu

- **Size**: When text is selected, choose a predefined or custom font size in this submenu to change the size of the selected text.
  - **Other...**: Opens the Text Formatting dialog for advanced style selection; see "Styling text and paragraphs" on page 182.
  - **7pt - 72pt**: Sets the size of the selected text to the chosen font size.
- **Style**: When text is selected, sets the text style by applying or removing the following attributes: Plain, Bold, Italic, Underline, Strikethrough, Subscript, Superscript, Capitalize, Uppercase, Lowercase, Small-caps. This is the same as opening the Text Formatting dialog (**Format > Text**) and checking the appropriate style. See "Styling text and paragraphs" on page 182.

- **Color:** When text is selected, sets the text color by applying the color attribute to the text. The color submenu lists all the colors in the [Colors Editor](#).
- **Text...:** Opens the Text Formatting dialog to modify the current text selection. See "Styling text and paragraphs" on page 182.
- **Align:** When an element is selected, determines how its contents is aligned inside the element. Options are Align Left, Align Right, Align Center and Justify.
- **Paragraph...:** Opens the "Paragraph Formatting dialog" on page 277 to modify the paragraph where the cursor is located. See "Styling text and paragraphs" on page 182.
- **Paragraph Format:** Displays a list of generic element types that can be used for a text element. Selecting one of them converts the element where the cursor is located into the appropriate element (for example `<p>` for Paragraph, `<h3>` for Heading 3, etc).
- **Float**
  - **Left:** Floats the current element to the left. This is equivalent to setting the CSS `float` property to `left`.
  - **Right:** Floats the current element to the right. This is equivalent to setting the CSS `float` property to `right`.
  - **None:** Removes any float style applied to the currently selected element.
- **Box...:** Opens the "Box Formatting dialog" on page 259 to modify the box where the cursor is located.
- **Image...:** Opens the "Image Formatting dialog" on page 269 to modify the image that is currently selected.
- **Table...:** Opens the "Table Formatting dialog" on page 314 to modify the table in which the cursor is located. If the cursor is within a table embedded within another, the innermost table's formatting is the one modified.
- **Table Cell...:** Opens "Table Cell Formatting dialog" on page 317 to modify the cell where the cursor is located.
- **Hyperlink**
  - **Insert...:** Creates a hyperlink on the currently selected text or element and opens its properties; see "Hyperlink and mailto link" on page 156.
  - **Edit...:** Opens the properties for the currently selected hyperlink; see "Hyperlink and mailto link" on page 156.
  - **Remove:** Removes the currently selected hyperlink. The text or element that was the hyperlink is not removed.

## Context Menu

- **Add:**
  - **Print Context:** Click to add a new Print context to the template if one does not exist.
  - **HTML Email Context:** Click to add a new Email context to the template if one does not exist.
- **Delete:** Click to delete the currently selected context. The last remaining context cannot be deleted.
- **Go to:** Click to open the first section in the selected context. This is the same as double-clicking on the first section of any context in the Resource Pane.
- **Properties:** Click to open the currently selected context's properties; see "Context Properties dialog" on page 265.
- **Preview HTML:** Click to preview the currently open section in the default system browser to preview it. This feature works in all contexts.
- **Profile Scripts:** Click to open the "Profile Scripts dialog" on page 294 to test script performance (see "Testing scripts" on page 231).
- **Preflight:** Click to preflight the document and open the Preflight dialog. Preflight verifies the template for common errors (see "Testing scripts" on page 231).

## Section Menu

- **Add:** Click to create a new section to the currently selected context.
- **Delete:** Click to delete the currently selected section.
- **Arrange:** Click to open the "Arrange Sections" on page 305 dialog.
- **Go to:** Click to list the sections in the currently selected context and open one by clicking it.
- **Properties...:** Click to open the appropriate section properties: Email or Print. See "Section properties dialogs" on page 300.
- **Includes...:** Click to open the "Section Includes" on page 305 dialog.
- **Finishing...** (Print Sections only): Click to open the Finishing tab in the "Print Section Properties" on page 300.
- **Sheet Configuration...** (Print Sections Only): Click to open the "Sheet Configuration dialog" on page 309.

- **Master Pages:** Click to list the available Master Pages in the template, and open one by clicking it (see "Master Pages" on page 76).
- **Master Page Properties...:** Click to open the currently selected Master Page's properties dialog; see "Master Pages" on page 76.

## View Menu

- **50/75/100/150/200%:** Click to zoom the [Workspace](#) at the selected level.
- **Source View:** Click to show the HTML source for the template, including CSS and HTML code.
- **Design View:** Click to show the template including all styles, text and images as well as the placeholders used for variable data.
- **Preview View:** Click to show the template as it will output with the current record, with the personalized content (see "Personalizing content" on page 202).
- **Show Edges:** Click to show or hide a colored border around elements on the page.
- **Rulers:** Click to show or hide the rulers in the [Workspace](#). Rulers only appear for Print contexts.
- **Margins and Guides:** Click to show or hide the margin lines and guides in the [Workspace](#).
- **Snap to Guides:** Click to enable or disable snapping to guides and to margins when moving objects.
- **Virtual Stationery:** Click to enable or disable the visibility of the PDF Background image set in the Media.
- **Highlight Master Page Items:** Click to enable or disable a yellow border around Master Page items in a section.
- **Object Resizing:** Click to enable or disable the ability to resize <div> elements on the page. See "Editing preferences" on page 286 for more fine-tuned control.

## Window Menu

- **Show View>:** Use the options in this menu to show or hide different panes of the UI.
  - **Properties > Attributes:** Shows the [Attributes pane](#)
  - **Messages:** Shows the Messages pane, see "Problems and messages" on page 332.

- **Problems:** Shows the Problems pane, see "Problems and messages" on page 332.
- **Resources:** Shows the [Resources pane](#)
- **Outline:** Shows the [Outline pane](#)
- **Data Model:** Shows the [Data Model pane](#)
- **Scripts:** Shows the [Scripts pane](#)
- **Reset Perspective:** Resets all toolbars and panes to the initial configuration of the module.
- **Preferences:** Click to open the [Preferences](#) dialog.

## Help Menu

- **Software Activation:** Displays the Software Activation dialog. See "Activating a License" on page 31.
- **Help Topics:** Click to open the help system in the default web browser.
- **Contact Support:** Click to open the [Objectif Lune Contact Page](#) in the default system web browser.
- **About PrintShop Mail Connect Designer:** Displays the software's About dialog.
- **Welcome Screen:** Click to re-open the Welcome Screen.

## Panes

Panes are windows containing user interface elements (such as information or properties), which can be docked and undocked, moved around and merged together through tabbed panes.

Here is a list of all panes:

---

### Attributes Pane

The Properties Pane displays all of the properties of the currently selected object in the Workspace. These properties vary greatly depending on the object that has been selected.

### General

These attributes are common to all elements in the template and will always appear.

- **ID:** A unique identifier for the selected element. Used for CSS selections as well as JavaScript expressions affecting single elements.
- **Class:** One or more classes that can be common to more than one elements. Used for CSS selections and JavaScript expressions that can affect multiple elements.

## Other

These attributes are available depending on the item selected (in parenthesis).

- **Whitespace element :** Check to make the element a whitespace element, meaning it will only appear on the page if there is enough space for it. This is useful for templates with variable height elements or conditional elements, to fill empty spaces with transpromotional material. Note that only top-level elements (a paragraph not inside a table or a div) will function at whitespace elements.
- **Source (image):** The location of the image file. For image resources in the template, the image path is often `images/<imagefile>.<extension>`  
When the source is a PDF, an addition button appears next to this box that opens the "Select Image dialog" on page 308.
- **Alternate text (image):** The "Alt" text used when hovering over the image in a browser. Also used for accessibility.
- **Height (image):** The specified height of the image. Defaults to the original image height in pixels.
- **Width (image):** The specified width of the image. Defaults to the original image width in pixels.
- **Cell Spacing (table only):** Defines the *cellspacing* attribute of the table which controls the spacing between cells in the table.
- **Cell Padding (table only):** Defines the *cellpadding* attribute of the table which controls the padding inside each cell of the table.
- **Column Resizing (table only):** Check to enable columns to be resized directly within the [Workspace](#).
- **Detail Table (table only):** Defines which detail table the repeat of the table is based on. The number of detail lines in the table is the number of the time the repeating row (see below) is repeated.
- **Title (table only):** Defines the title of the table. This has no impact on the table's displays, only on accessibility of HTML pages and screen readers.

- **Repeat (table row not in <tfoot> or <thead> only):** Defines if the row is affected by the detail table calculation. This row is the one repeated in a Dynamic Table.
- **Show Row (table row only):** Use the drop-down to determine when the selected row appears when a dynamic table overflows. This option is only available in a row manually added inside of a Dynamic Table.
  - **Before page break:** The row will appear on all pages except the last one.
  - **At end of table:** The row will appear only on the last page.
  - **Always:** The row will appear on every page of the table.
- **Subtotal Line (table row inside a <tfoot> only):** Defines the footer row as the place where the SubTotal is displayed. This is the row where a subtotal script is expected to display the result.

## Geometry

These attributes are available for certain elements that have position or size attributes such as images and boxes.

- **X-Offset:** The horizontal distance from the top-left of the object to the left position of its parent. This is used only for relative and absolute positioned elements.
- **Y-Offset:** The vertical distance from the top-left of the object to the top position of its parent.
- **Width:** The width of the element, by default in pixels.
- **Height:** The height of the element, by default in pixels.

## Page

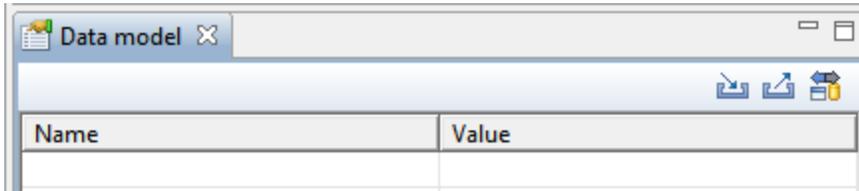
These attributes appear when selecting the *Page* node in the [Outline Pane](#).

- **Master Page:** Which of the "Master Pages" on page 76 to use for the template.

## Data Model Pane

The Data Model Pane displays a Data Model used to help design the template, along with optional Extracted Data generally resulting from the execution of a Data Mapping Configuration. The information shown is the extracted information for the current record within the Record Set. It is also used as a navigation tool between records and all tables.

Data is displayed as a tree view, with the root level being the Record table, levels below it being detail tables, and any level below being called Nested Tables.



## Pane Options

- **Minimize/Maximize:** Click to minimize or maximize the pane. See [Moving and Merging Panes](#).
- **Import Data Model:** Click to browse to a Data Model File and import it. See "Importing a Data Model File" below.
- **Export Data Model:** Click to browse to a location to save the Data Model File and save it.
- **Synchronize Data Model:** Click to synchronize the data model with the one currently loaded in the open Data Mapping Configuration. Disabled if no configuration is currently open. See "Synchronizing Data Models" below.

## Using the Data Model

When a Data Model is loaded inside of the Data Model Pane, it can be used to design templates by dragging the fields directly into the template; see "Variable Data" on page 211. If data is present (from a Data Model File or a Data Mapping Configuration), it is possible to preview the resulting data in the template using the Preview tab (see [Workspace](#)).

## Importing a Data Model

There are three different ways to import a Data Model into a Template to help designing it.

### Importing a Data Model File

Importing a Data Model File displays the file's data model structure into the Data Model Pane, with optional sample data for each field. To import a data model file, click the Import button at the top of the Data Model Pane.

## Synchronizing Data Models

If a Data Mapping Configuration is open and contains fields in its record, it is possible to use the *Synchronize data models* button in the pane's toolbar to retrieve the model currently stored in the DataMapper Module's Data Model Pane.

## Running a Data Mapping Configuration or Wizard

If executing a data mapping configuration or directly loading a data source, the resulting record set is loaded in the Data Model Pane. See "Loading data" on page 204.

## **Problems and messages**

### **Messages Pane**

The **Messages** pane displays any warnings and errors from the data mapping configuration or template.

To open it in the Designer module, click the Messages button at the bottom right of the window (see "Designer User Interface" on page 254).

### Buttons

- **Export Log** : Click to open a **Save As** dialog where the log file (.log) can be saved on disk.
- **Clear Log Viewer** : Click to remove all entries in the log viewer.
- **Filters** : Displays the Log Filter.
- **Activate on new events** : Click to disable or enable the automatic display of this dialog when a new event is added to the pane.

### Field Headers

- **Message**: The contents of the message, indicating the actual error.
- **Component**: Whether the entry is a warning or an error.
- **Source**: The source of the error.
- **Date**: The date and time when the error occurred.

## Problems Pane

The Problems pane displays any notifications or errors related to the template, its scripts, its code or output generation.

## Moving and merging panes

The PrintShop Mail Connect interface for the Designer module is highly configurable. Each panel in the application can be moved, with the exception of the "Workspace" on page 341 which area always in a static location. All panels can be minimized or maximized.

### To move a panel:

- Click and hold the left mouse button on the panel title (tab) to move and keep the button pressed.
- Start moving the mouse to the new location. A grey outline shows where the tab will show up:
  - A small grey outline next to a current panel tab indicates that both tabs will be at the same location and only the active tab will display its content.
  - A larger grey outline at one of the edges of the Workspace or Data Viewer indicates that the separate will be separate and always visible.
- When the grey outline displays the location where the panel should be, release the mouse button.

### To minimize a panel:

- Click the **Minimize panel** button at the top-right corner of the panel.

A minimized panel displays only as its icon wherever it was docked, generally on the left or right side, or the bottom.

### To restore a minimized or maximized panel:

- Click the **Restore** button next to the panel's display icon.

The restored panel will return to its original docked location.

### To temporarily display a minimized panel:

- Click the panel's display icon.

When another panel, menu or toolbar is clicked, the panel will be minimized again.

#### **To maximize a panel:**

- Click the **Maximize** button at the top-right corner of the panel.

A maximized panel takes the full available size for the panels. All other panels are minimized.

#### **Outline Pane**

The Outline Pane displays the current structure of the template, including all HTML tags present in each page.

- The display is in a treeview, the root being the *Pagenode*.
- At the top of the pane, a Text Filter box appears. Enter text in this box to only show elements which correspond to this inclusive filter. This can be class names, IDs, or element types (div, table, etc).
- Under the *Pagenode*, all top-level page elements are displayed. Each element under them is accessible by expanding (with the [+]) elements with children.
- Clicking on any element will select it in the [Workspace](#), whether it displays the Source, Design or Preview tab.
- Dragging an element inside the Outline Pane re-orders it in the actual HTML. Elements are executed top-to-bottom with lower elements appearing on top of previous elements (unless a CSS Z-Index is used).
- Right-clicking an element displays a contextual menu offering the following options:
  - **Delete Element:** Click to delete the element from the outline view. This also removes it in the template itself for the current section.

#### **Resources Pane**

The Resource Pane displays the resources that affect the template and its output.

### Tip

Images, fonts, stylesheets and snippets can be dragged or copied and pasted into the Resources Pane to add them to your template.

## Media

Media resources define paper handling configurations for Print output (see "Generating Print output" on page 495 and "Print Options" on page 411) including page size and paper type. See "Media" on page 80 for more information.

### Contextual menu

- **New Media:** Click to create a new media and open its [properties](#).
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Properties:** Click to open the media properties.

## Master Pages

Master Pages are layers of content that can be used by multiple Print Contexts to provide a reusable static background of content. Only one Master Page can be selected for each page position in the context. See "Master Pages" on page 76 for more information.

### Contextual menu

- **New Master Page:** Click to create a new Master Page and open its properties.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.
- **Properties:** Click to open the Master Page properties; see "Master Pages" on page 76 for more information.

## Contexts

Contexts hold the actual content of the template that is used to generate output. See "Contexts" on page 47 for more information.

### Contextual menu (Context folder or individual contexts)

- **New Print Context:** Click to create a new Print Context with a single section.
- **New Web Page Context:** Click to create a new Web Page Context with a single section.
- **New HTML Email Context:** Click to create a new HTML Email context with a single section.
- **Properties...** (Print and Email Contexts): Click to open the Context's properties. See "Contexts" on page 47 for more information.

## Sections

Sections hold part of the contents within a specific context. See "Sections" on page 48 for more information.

### Contextual menu

- **Set as Default** (Email and Web contexts only): Click to set the default section that is output if none is selected in the output generation.
- **New Section:** Click to add a new section within the context.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.
- **Properties...:** Click to open the appropriate section properties: Email, Print or Web. See "Section properties dialogs" on page 300.
- **Includes...:** Click to open the "Section Includes" on page 305 dialog.
- **Finishing...** (Print Sections only): Click to open the Finishing tab in the "Print Section Properties" on page 300
- **Sheet Configuration...** (Print Sections Only): Click to open the Sheet Configuration dialog; see "Master Pages" on page 76 and "Media" on page 80.

## Images

Images are graphical elements that can be added to the page for display, either statically or dynamically. See "Images" on page 158 for more information.

### Contextual menu

- **New Folder:** Click to create a new folder to organize resources more easily.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.

## Fonts

Font Resources included in a template are transported with it, so they can be accessed even if the template is moved to a different computer. Currently, fonts must be set through the CSS Stylesheet and do not appear in the fonts drop-down menu.

Currently supported font types: otf, woff, ttf, svg. Fonts must be set to *installable* to be useable in the output.

Please see the [Tips & Tricks post](#) for details on how to embed the fonts.

## JavaScripts

JavaScripts are scripted programs that can run on Web output when added to the page header. See Using JavaScript for more information.

### Contextual menu

- **New Javascript:** Click to create a new JavaScript resource.
- **New Remote Javascript:** Click to add a Remote JavaScript resource. See Using JavaScript for more information.
- **New Folder:** Click to create a new folder to organize resources more easily.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.

- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.

## Stylesheets

Stylesheets control how contents appears on the page. It defines spacing, color, size and other properties of elements on the page. See "Styling templates with CSS files" on page 171 for more information.

### Contextual menu

- **New Stylesheet:** Click to create a new Stylesheet resource. Adding a new stylesheet will automatically include it in the currently active section.
- **New Remote Stylesheet:** Click to add a Remote Stylesheet resource. See "Styling templates with CSS files" on page 171 for more information.
- **New Folder:** Click to create a new folder to organize resources more easily.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.

## Snippets

Snippets are pieces of HTML or JSON code that can be inserted within sections and master pages, dynamically or statically. See "Snippets" on page 168 for more information.

### Contextual menu

- **New HTML Snippet:** Click to create a new HTML Snippet resource.
- **New JSON Snippet:** Click to create a new JSON Snippet resource.
- **New Folder:** Click to create a new folder to organize resources more easily.
- **Rename:** Click to open the resource's Rename. This is the same as pressing the F2 key while the resource is selected.
- **Delete:** Click to delete the resource. This is the same as pressing the Delete key while the resource is selected.

## Scripts pane

The Scripts pane contains all of the scripts that are used to replace data in a template, or to modify its look. Scripts can be exported and imported via the buttons or through drag & drop between the Scripts pane and any location on the computer.

### Note

Scripts included on the Scripts pane are completely distinct from the JavaScript resources found in the "Resources Pane" on page 334 (see Using JavaScript). Think of scripts as server-side in the sense that they are executed through the Connect modules (Server and Content Creation especially). Scripts have access to the whole PrintShop Mail Connect JavaScript API (see "Designer Scripts API" on page 429), such as the `record` object. JavaScript resources, on the other hand, are only executed **after** the content creation is done, generally in a browser.

### Note

The scripts in the Scripts pane are always executed top-to-bottom. They can be dragged up or down in the pane to change their order of execution. For example, content loading scripts (snippets with variable data, for instance) must come before scripts that replace data within that loaded contents.

## Buttons

- **Import...:** Click to open a standard Open dialog to import a script. The script must have the .OL-script extension.
- **Export...:** Click to open a standard Save As dialog to save the currently selected scripts to disk. These scripts can be re-used in other templates. If more than one script is selected, they are all saved to a single file. If some scripts are inside folders, this folder structure is kept and will be restored when the scripts are imported.
- **New:** Displays a drop-down that shows the following options:
  - **Script:** Adds a new empty basic script.
  - **Text Script:** The default script that is created when adding variable data to a template. See "Variable Data" on page 211.

- **Dynamic Image script:** Provided that its selector refers to an image, this script dynamically changes the image for each record. See "Dynamic Images" on page 222.
- **Email scripts:** Email scripts define the sender, recipients, subject etc. of the email that is sent, and the PDF password. See "Email header settings" on page 98.
- **Control script:** A Control script affects the output of a template per record as a whole, instead of parts of the content. See "Control Scripts" on page 240 and "Control Script API" on page 477.
- **Conditional Content Script:** This script can conditionally show or hide any element in the template. See "Showing content conditionally" on page 219 and "Conditional script dialog" on page 299.
- **Folder:** Adds a folder in which scripts can be placed for easier management. See "Script folders" on page 228.
- **Collapse All:** Collapses all the folders, hiding the scripts inside of them.

### Scripts Pane column

- **Name:** The name added to better identify the script.
- **Selector:** Displays the initial text or selector that the script applies to.

#### Note

Fields from the Data Model pane can be dragged directly into the Scripts pane to create a Text Script. Additionally, Text scripts can be dragged into any section to add the script's placeholder at the insert location. See "Variable Data" on page 211.

### Contextual menu options

- **Duplicate:** Click to create an exact copy of the script.
- **Delete:** Click to delete the selected script. This does not delete any element or text in the template itself.
- **Rename:** Click to open a dialog to rename the script. This is the same as changing the **Name** field in the Edit Script window, which can be opened by double-clicking the script.

- **Enable/Disable:** Click to trigger the script to be enabled or disabled. Disabled scripts are greyed out and italic and will not be executed. See "Enable/disable scripts" on page 230
- **Import:** load a script from a Scripts file (\*.OL-script).
- **Export:** save the script to a Scripts file (\*.OL-script).
- **Properties** (Script folders only): edit the **name** and **execution scope** of the folder. See "Execution scope" on page 229.

## Styles pane

The Styles pane shows which CSS style rules apply to the currently selected element. A link next to a style rule will open the file where that particular style is defined. This can be either a CSS file or the source file of a section if local formatting was used (see "Styling and formatting" on page 170).

A crossed-out style rule signals that it was overruled by another style rule. This happens when:

- A more specific, and therefore more important rule, is encountered for the same element. See "Using a more specific CSS rule" on page 178 to learn more about the specificity of style rules.
- A rule with the same importance is read after the first rule. Not only is the order of the rules in a CSS file important, but also the order in which the style sheets are read. The style sheets that are included with a section are read in the specified order; see "Determining the order in which style sheets are read" on page 179.

## Workspace

The Workspace pane is where everything comes together. It is the contents of the page, the WYSIWYG editor that shows what the output will look like.

The Workspace contains three tabs. To switch between the tabs, click on the tab at the bottom or select **View > Design View, Preview View** or **Source View** on the menu.

## Source Tab

The source tab displays the HTML source for the template, including HTML Headers, CSS and HTML code. The source is displayed in a color-coded text editor, to quickly visualize the code. In this tab changes and adjustments can be made to the code.

To the left of the Source tab, a bar helps visually identify the start and stop of an element. For example when clicking on the opening `<table>` element, this bar marks the whole `<table>` and all its contents, until the ending `</table>` tag.

The top of the Design tab contains an area with the following options:

- **Breadcrumbs:** Displays the element type where the cursor is located and any of its parent elements. Elements with classes or IDs show these details next to them, for instance `div #contents > ol.salesitems > li`. Click on an element in the Breadcrumbs to select it. If an element is selected in the breadcrumbs and the Backspace key is pressed, that element is deleted.
- **Context Selector:** Displays the current context. The drop-down lists available contexts. Clicking on a context switches to that context.
- **Section Selector:** Displays the currently active section. Clicking on another section switches to that section.

## Design Tab

The design tab show the template including all styles, text and images as well as the placeholders used for variable data. In this tab, the template's scripts are not executed and only placeholders are shown.

The top of the Design tab contains an area with the following options:

- **Breadcrumbs:** Displays the element type where the cursor is located and any of its parent elements. Elements with classes or IDs show these details next to them, for instance `div #contents > ol.salesitems > li`. Click on an element in the Breadcrumbs to select it. If an element is selected in the breadcrumbs and the Backspace key is pressed, that element is deleted.
- **Context Selector:** Displays the current context. The drop-down lists available contexts. Clicking on a context switches to that context.
- **Section Selector:** Displays the currently active section. Clicking on another section switches to that section.
- **Media Selector** (Master Page editor only): Displays a list of Media resources. Clicking on a media will display its Virtual Stationery background while in Preview mode.
- **Zoom Level:** Displays the current zoom level and drops-down to change the level.
- **Zoom in:** Zooms in by 25%

- **Zoom out:** Zooms out by 25%
- **Actual Size:** Zooms to 100%.
- **Fit Width:** Adjusts zoom to fit the exact width of the template to the available workspace.
- **Responsive Design View:** Use the drop-down to select a specific screen width, to test the design for different devices. Not available in Print contexts.

## Preview Tab

The preview tab shows the template as it will output with the current record (see "Loading data" on page 204), with the personalized content (see "Personalizing content" on page 202). Although it is possible to edit the template in Preview mode to a certain extent, it is recommended to do all editing in the Design mode.

## Toolbars

In the Designer module, the following buttons are available in the top toolbar:

- **File Manipulation**
  - **New:** Displays the New Wizard where a new data mapping configuration or a new template can be created.
  - **Open:** Displays the Open dialog to open an existing template.
  - **Save:** Saves the current template. If the template has never been saved, the Save As... dialog is displayed.
  - **Print:** Opens the Print Output dialog.
  - **Proof Print:** Opens the "Print Options" on page 411 dialog as a "Proof Print" which limits the number of records output. The options themselves are identical to the regular Print Output dialog.
- **Output**
  - **Send Email:** Opens the [Send Email](#) dialog.
  - **Send Test Email:** Opens the "Send Test Email" on page 307 dialog.
  - **Preview HTML:** Opens the current template's Preview in the system default browser. Useful for testing scripts and HTML output.

- **Pagination** (Print Context only)
  - **Page Number:** Inserts a placeholder for the current page number
  - **Page Count:** Inserts a placeholder for the total number of pages in the current section.
- **Guides**
  - **Insert Horizontal Guide:** Click to insert a new horizontal guide; see "How to position elements" on page 179.
  - **Insert Vertical Guide:** Click to insert a new horizontal guide; see "How to position elements" on page 179.
- **Miscellaneous**
  - **Insert Lorem Ipsum:** Inserts a paragraph of generic lorem ipsum text, useful for placeholder or template design.
  - **Show Edges:** Shows a colored border around elements on the page and the type of element that is highlighted.
- **Table Manipulation**
  - **Insert Standard Table...:** Inserts a table with a specific number of columns and rows through the "Table" on page 163 Wizard.
  - **Select**
    - **Select Table:** Selects the table where the cursor is located. If the cursor is within a table embedded within another, the innermost table is the one selected.
    - **Select Row:** Selects the innermost row where the cursor is located.
    - **Select Cell:** Selects the innermost cell where the cursor is located.
  - **Delete**
    - **Delete Table:** Deletes the innermost table where the cursor is located.
    - **Delete Row:** Deletes the innermost row where the cursor is located.
    - **Delete Column:** Deletes the innermost cell where the cursor is located.
  - **Insert**
    - **Insert Row Above:** Inserts a row above the current one. The row configuration, such as merged cells and cell styles, are duplicated, but

contents is not.

- **Insert Row Below:** Inserts a row below the current one. The row configuration, such as merged cells and cell styles, are duplicated, but contents is not.
- **Insert Column Before:** Inserts a column to the left of the current one. The column configuration, such as merged cells and cell styles, are duplicated, but contents is not.
- **Insert Column After:** Inserts a column to the right of the current one. The column configuration, such as merged cells and cell styles, are duplicated, but contents is not.

- **Objects**

- **Insert Image...:** Inserts an Image using a resource that is local to the template, at the current location of the pointer and opens its properties. See "Images" on page 158.
- **Insert Image from Address...:** Inserts an Image using a URL instead of a resource, at the current location of the pointer and opens its properties. See "Images" on page 158.
- **Insert Barcode:** Displays a list of available barcodes. Click on one to insert it on the page. See "Barcode" on page 109.
- **Insert Pie Chart:** Click to insert a new Pie Chart object and open the Chart Script wizard.
- **Insert Bar Chart:** Click to insert a new Bar Chart object and open the Chart Script wizard.
- **Insert Line Chart:** Click to insert a new Line Chart object and open the Chart Script wizard.

- **Hyperlinks**

- **Insert Hyperlink...:** Creates a Hyperlink or mailto link on the currently selected text or element and opens its properties. See "Hyperlink and mailto link" on page 156.
- **Remove Hyperlink:** Removes the currently selected hyperlink. The text or element that was the hyperlink is not removed.

- **Boxes**

- **Insert Positioned Box:** Inserts an absolute-positioned box on the page, which can be moved around freely.

- **Insert Inline Box:** Inserts an inline box that is set to float to the left, at the position of the cursor.
  - **Wrap in Box:** Takes the current selection and wraps it inside a new box.
  - **Float Left:** Floats the current element to the left using a *float:left* style.
  - **No Float:** Removes any *float* style applied to the currently selected element.
  - **Float Right:** Floats the current element to the right using a *float:right* style.
  - **Rotate Counter Clockwise:** Rotates the currently selected box 90° counter-clockwise.
  - **Rotate Clockwise:** Rotates the currently selected box 90° counter-clockwise.
- **Styles**
    - **Element Type:** Displays the element type of the selected element and drops down to show other element types in which it can be changed.
    - **Style:** Displays the style of the selected element and drops down to show other available styles which can be applied to it.
    - **Font Face:** Displays the font face of the selected text or element where the cursor is located and drops down to show other available font faces which can be applied to it.  
Fonts added to the Fonts folder of the Resources pane are shown automatically in the Fonts drop-down.
    - **Font Size:** Displays the font size of the selected text or element where the cursor is located and drops down to show other available sizes which can be applied to it.
    - **Font Color:** When text is selected, click to apply the shown color to the selected text, or use the drop-down to change the color and apply it.
  - **Alignment**
    - **Align Left:** Aligns the currently selected element to the left.
    - **Align Center:** Aligns the currently selected element to the center.
    - **Align Right:** Aligns the currently selected element to the right.
    - **Justify:** Aligns the currently selected element to stretch text lines to fill all available width.

- **Text Decoration**

- **Bold:** Makes the currently selected text **bold**.
- **Italic:** Makes the currently selected text *italic*.
- **Underline:** Makes the currently selected text underline.
- **Strikethrough:** Makes the currently selected text ~~strikethrough~~.

- **Indentation**

- **Create Numbered List:** Makes the selected text element a numbered list (<ol>). If multiple paragraphs are selected, each becomes a list item (<li class="Bullet">).
- **Create Bulleted List:** Makes the selected text element a bullet list (<ul>). If multiple paragraphs are selected, each becomes a list item (<li class="Bullet">).
- **Indent:** Increases indentation of the selected text element. If the element is a paragraph, it is wrapped in a <blockquote> element. If it is a list item, it is moved to a child level, creating a new list if necessary.
- **Outdent:** Decreases indentation of the selected text element. If the element is wrapped in a blockquote element, one blockquote is removed. If the element is a list item, it is removed from one surrounding list.

- **Position**

- **Superscript:** Makes the currently selected text a <sup>superscript</sup>.
- **Subscript:** Makes the currently selected text a <sub>subscript</sub>.
- **Remove Formatting:** Remove any and all styles, text decorations and other formatting from the selected text. Indentation is not affected.
- **Welcome Screen:** Click to re-open the Welcome Screen.

## Welcome Screen

The **Welcome Screen** appears when first starting up PrintShop Mail Connect. It offers some useful shortcuts to resources and to recent documents and data mapping configurations.

The Welcome Screen can be brought back in two ways:

- The **Welcome Screen** button in the "Toolbars" on page 343.
- From the Menus in **Help, Welcome Screen**.

## Contents

- **Activation:** Click to open the **Objectif Lune Web Activation Manager**.
- **Release Notes:** Opens the current **Release Notes** for PrintShop Mail Connect.
- **Website:** Opens the PrintShop Mail Connect website.
- **Take A Tour:** Click to open the YouTube Playlist giving you a tour of the software.
- **Use the DataMapper to...:**
  - **Create a New Configuration:** Opens the Creating a New Configuration screen.
  - **Open an Existing Configuration:** Click to open the standard **Browse** dialog to open an existing data mapping configuration.
  - **Recent Configurations:** Lists recently used configurations. Click any configuration to open it in the DataMapper module.
- **Use the Designer to...:**
  - **Create a New Template:** Lets you choose a Context to create a new template without a Wizard.
  - **Browse Template Wizards:** Displays a list of available Template Wizards, producing premade templates with existing demo content; see "Creating a template" on page 42.
  - **Open an Existing Template:** Click to open the standard **Browse** dialog to open an existing template.
  - **Recent Templates:** Lists recently used templates. Click any template to open it in the Designer module.
- **Other Resources:**
  - **Documentation:** Opens this documentation.
  - **Courses (OL Learn):** Opens the [Objectif Lune e-Learning Center](#).
  - **User Forums:** Opens the [Questions & Answer](#) forums.

## Print Options

The Print Options page is the first page of both the **Advanced Print Wizard** and the [Output Creation Settings](#) Preset .

This page is the most important of the Advanced Print Wizard.

The other pages that appear throughout the Wizard are determined by the selections made on this page.

The choices can be broken down as follows:

- **Printer section:**

- **Model:** Use the drop-down to select the printer language / output type that will be generated.

Connect output options cover a range of industry standard print output types.

These include AFP, IPDS, PCL, PDF and PostScript (including PPML, VIPP and VPS variants), with a range of quality settings available.

**Note**

By default, Connect displays only the PDF output option, but other print output types can be added to the Printer Model drop down list via the Gear button 

For more information on how to do this, see "Adding print output models to the Print Wizard" on page 499.

- **Output Options section:**

- **Output Local** checkbox: Select to have the output created using the local Print Server.

- **Output Type** choices:

- **Prompt for file name:** Select to output to a local file on the hard drive. When this option is selected, no other configuration is necessary. A Save As dialog will appear to allow selection of the folder and filename.
- **Directory:** Select to output to a local folder on the machine.
  - **Job Output Mask:** The name of the file that will output. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
  - **Job Output Folder:** The path on the disk where the file is produced. Please note that the folder must exist, or output will fail when produced through the server.

- **LPR Queue:** Select to send the print job to an LPR queue. It is assumed that the print technology is supported by the system receiving the LPR job.
  - **Local Printer:** The IP or host name of the printer or machine where the LPD is installed and will receive
  - **Queue Name:** The queue name that will accept the job on the LPD. Default is generally "auto".
  - **Job Owner Name:** Optional entry for adding the name of the job owner.
  - **Job Name:** The name of the output file. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
- **Windows Printer:** Select to send the Print Job to a Printer Queue. The job is rendered as a PDF before being printed through the Windows driver.
  - **Windows Printer:** Use the drop-down to select the windows printer queue where the job will be sent.
  - **Job Owner Name:** Optional entry for adding the name of the job owner.
  - **Job Name:** The name of the output file. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
- **PDF Rendering Options (PDF output only):**
  - **Auto-rotate and center:** Check to automatically select the page orientation that best matches the content and paper.
  - **Choose paper source by page size:** Check to use the PDF page size to determine the output tray rather than the page setup option. This option is useful for printing PDFs that contain multiple page sizes on printers that have different-sized output trays.
  - **Scale:**
    - **None:** Select to not scale any page, whether it fits or not.
    - **Expand to printable area:** Select to expand any page to fit the page area. Pages larger than the paper size are not resized.
    - **Shrink to printable area:** Select to shrink any page to fit the page area. Pages smaller than the paper size are not resized.

- **Production Options:**

- **Booklet Imposition** checkbox: Check to tell the printer to generate a booklet for the print output. Booklet options are set in the "Booklet Options" on page 417 page. This option is unselected by default unless selected in the Designer "Print Section Properties" on page 300.
- **Cut and Stack Imposition** checkbox: Check to enable Cut & Stack Imposition, which is set in the "Imposition Options" on page 418 page.
- **Add Inserter marks** checkbox: Check to enable inserter mark functionality, which is set in the "Inserter Options" on page 421 page.
- **Override Finishing options** checkbox: Check to configure custom "Finishing Options" on page 403, such as binding.
- **Print virtual stationery** checkbox: Check to enable virtual stationery in the output.
- **Use grouping** checkbox: Check to configure grouping of output into jobs, job segments or document sets. See "Grouping Options" on page 408.
- **Include meta data** checkbox: Check to add meta data to the output. This can be done at Job, Job Segment, Document, Document Set and Page level. See "Metadata Options " on page 409.
- **Separation:** Check to activate the "Separation Options" on page 415 page of the wizard.
- **Add additional content** checkbox: Check to activate the "Additional Content" on page 365 page of the wizard.

- **Records** section:

- **Record Range:** Allows selection of a range of records or a custom selection. You can specific individual records separated by semi-colons (;) or ranges using dashes.  
For example: 2;4;6-10 would print pages 2, 4, 6, 7, 8, 9 and 10.

- **Copies** section:

- **Copies:** Enter the number of copies to print, of each record.
- **Collate:** When printing multiple copies you can check this checkbox to have the record copies printed together.  
For example in a three record job the records would print out as 1-1-2-2-3-3, rather than 1-2-3-1-2-3.

- **Pure Color Thresholds** section:

This section is valid for AFP, IPDS and PCL only. It applies to elements within the record

that are shades of gray, rather than black or white.

- **Black Threshold Percentage:** The percentage of shading at which the element will appear as full black, rather than dark gray.
- **White Threshold Percentage:** The percentage at which the element will appear as full white, rather than light gray.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 348 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 348 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Printer Settings

The Printer Settings page defines options on the printer. It is available for PostScript (and the VIPP and VPS variants of PostScript) only.

- **Map media by** options: Select from following choices:
  - **Media Attribute** displays all Media details, except the Tray selection.
  - **Tray** displays just the Media name and Tray selections.
  - **Both** displays all Media details.
- Tray selection columns:
  - **Media:** Lists the Media name, as defined in the template.
  - **Tray:** Use the drop-down to select in which tray to send any page using the media.
  - **Position:** Enter a MediaPosition option on the printer to define the media to use.

- **Weight:** Enter a weight for the paper.
- **Type:** Use the drop-down to select which type of stock to use on the printer.
- **Color:** Use the drop-down to select which color the paper should be on the printer.

## Booklet Options

The Booklet Options page defines how to generate booklets in the output. It is used in conjunction with [Imposition](#) settings, which will appear after the Booklet entries have been made.

This page includes a handy illustration that displays how the final binding would look, based upon the current selections.

Options:

- **Configuration:** Use the drop-down to select the type of binding to use:
  - **Saddle Binding:** This binding places all the pages in a stack, binds the middle and folds the stack as one.
  - **Perfect Binding:** This binding type is often used for books. Pages are folded in the middle and then set side by side. The pages are then bound along the folded "spine".
  - **1 up Perfect Binding:** This binding does not contain any folding. The pages are lined up side by side and bound along one edge.
- **Booklet Binding Edge:** Use the drop-down to select the side on which to bind the booklet.

Optional **Cover Page** selections are available to Saddle Binding only.

- **Cover Page** checkbox: Check to enable cover pages to be created with the options below:
  - **Media** selections:
    - **Cover Media Size:** Use the drop-down to select the media size for the cover page, or use a Custom size and select **Width** and **Height** values.
  - **Front Cover** selections:

- **Blank:** Select to add no data to the front cover.
- **First page on outside and second page on inside:** Select to use the first 2 pages as the inside and outside of the front cover.
- **Back Cover selections:**
  - **Blank:** Select to add no data to the back cover.
  - **Last two pages on inside and outside:** Select to use the final 2 pages as the inside and outside of the back cover.

## Imposition Options

Imposition refers to the printing of multiple pages on a single sheet. This is also known as N-Up printing.

The options on this page allow for the setting of imposition repetition, order, margins and markings.

- **Sheet Size** group:
  - **Final Media Size:** Use the drop-down to select the size of the media where the output is printed. The size of the media should be equivalent to the initial Section size multiplied by the number of repetitions, added with the margins and spaces between the repetitions.  
If *Custom* media size is selected, enter the custom **Width** and **Height** values.

### Note

The Sheet Size cannot be altered if a Cover Page was selected in the "Booklet Options" on page 417 Page.

- **Sheet Rotations:** Select aspect ratio of media (Landscape or Portrait), or allow Connect to automatically determine the proper aspect ratio (Auto-Rotate).
- **Position:** Select from following options:
  - **Auto-positioned:** This option creates unscaled imposition-ed pages.
  - **Scale to fit:** Scales the imposition-ed pages so they fit on the N-Up stock. The scaled pages are then auto-positioned as usual.

- **Offset:** Allows for the selection of an offset position. The imposition-ed pages will be laid out so that the top left corner of the top left imposition-ed page is located at the selected offset.  
If Offset is chosen, then the **Left Offset** and **Top Offset** selection boxes become active.

#### Note

The offset measures from the top left of the physical N-Up sheet to the top left imposition-ed page. If Auto-Rotate is selected (causing the N-Up stock to be rotated to fit the imposition-ed pages) then the measurement becomes the top left position of the rotated stock. i.e. The top left corner does not rotate with the stock.

#### Note

If *Booklet Binding* has been selected, then the **Position** settings are pre-set and cannot be altered here.

- **Rotate final output Sheet 180 degrees (upside down):** Select to flip the output upside down.
- **Repetition** group:  
Allows selection of how many Sections are to be placed, both Horizontally and Vertically. This is the total number of items, not the number of additional items being placed.

#### Note

If *Booklet Binding* were selected, some of these settings will be determined by the options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Space Between** group:  
Allows selection of the amount of blank space to add between each repetition.

### Note

If *Booklet Binding* were selected, some of these settings will be determined by the options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Order group:**

### Note

If *Booklet Binding* were selected, some of these settings will be determined by the options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Page Order:** Select in which direction to go when adding sections to the output:
  - **Left to right, then top to bottom**
  - **Right to left, then top to bottom**
  - **Top to bottom, then left to right**
  - **Top to bottom, then right to left**
- **Stack Depth:** Enter a stack depth or use the arrows to increment or decrement.
- **Reverse Pages:** Select this option to reverse the order of pages.  
This would print the final record on the first page and the first record on the last page.
- **Force simplex:** Select this option to make the output Simplex, rather than the imposition default of Duplex.
- **Bleed Margins group:**
  - **Top, Bottom, Left, Right:** Enter the bleed margins for each side of the page.
- **CropMarks group:**
  - **Type:** Use the drop-down to select the type of crop marks to add to the page.
  - **Offset:** How much separation (if any) to leave between the vertical and horizontal corner markings.

- **Width:** Select the width of the crop mark lines.
- **Length:** Select the Length of the crop mark lines.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Inserter Options

The Inserter Options page allows the selection of a High Capacity Feeder (HCF) model. These machines are also commonly referred to as Inserters or Folder-Inserters.

The options available on this page are dependent upon the model selected.

The options selected on this page influence the position of the markings set on the next page:

**"Mark Position Options" on page 423.**

- **Model:** Use the drop-down to select from any previously loaded Inserter model, or use the Browse button to select a HCF file to load a new Inserter model.  
An image representing the chosen folder-inserter is displayed under the list, along with the HCF file details.
- **Options Group:**  
The options available here are all Inserter dependent, and thus will change based upon the Inserter model selection.  
To see how the selected Inserter markings would look on the printed page, click the Next button to move to the "Mark Position Options" on page 423 page, which has a preview of the page. You can move back and forward between these two pages until you are entirely satisfied with the selections made.

- **Mark Configuration:** Use the drop-down to select the type of markings to add. This selection basically equates to the amount of area the markings will take up on the printed page.
- **Fold Type:** Use the drop-down to select the type of fold to apply to the paper. This will impact upon where on the page the markings will be placed.
- **Collation level:** Select whether the markings will be made at Document level, or Document Set level.
- **Print marks on back:** Check to place the Inserter Marks on the rear of the page.
- **Selective Inserts:** If selective inserts are supported by the chosen Mark Configuration you can select what markings to include and whether those markings are to included based upon some conditional setting.  
For example, you could add a marking to the third page of a document by making the selection Conditional and then setting the Condition entry to "page.nr = 3".
- **Clear Background Area:** Check to add a white background to the OMR, preventing background colors or elements interfering with the OMR Markings when they are read by the Inserter.
- **Margins:**
  - **Same for all sides:** Check so that the Left margin selection is used to set all sides identically.
  - **Left, top, right, bottom:** Enter a measure for the margins on each side of the OMR Marks.
- **Custom OMR mark sizing:** If supported by the chosen Mark Configuration you can select a Custom OMR size.  
You can select from any of the following, or leave the entries blank to use default values:
  - **Line length:** Enter a value between 10.16mm and 20mm.
  - **Line thickness:** Enter a value between 0.254mm and 0.63mm.
  - **Gap distance:** Enter a millimeter value 2.91mm and 4.2mm.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.

- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Mark Position Options

This page displays a Preview of the output and the possible locations to place the inserter marks. The initial settings are determined by the selections made within the "Inserter Options" on page 421 page.

You can move back and forward between these two pages to perfect the settings, or you could move the inserter mark box to the desired location on the preview.

### Preview box:

- The *pink area* displays the areas of the page where inserter marks can be positioned.
- The *small checkered box* displays the current location of the inserter marks. This box is selectable and can be dragged to the desired location within the printable (pink) areas. If the box is placed outside the printable areas the page will display an error and prevent attempts at leaving the page.

**Below the Preview box** are buttons which allow control of the Preview box. The selections that can be made are:

-  **First Page**: Click to jump to the first page.
-  **Previous Page**: Click to move to the previous page.
-  **Next Page**: Click to move to the next page.
-  **Last Page**: Click to jump to the last page.
- **Show Page**: Use the up and down arrows or type a page number to display a specific page within the document.

-  **Zoom in/out:** Click to zoom in or out by 25%
-  **Zoom Level:** Use the drop-down to select a predefined level or enter a zooming percentage.

## Finishing Options

Use this dialog to force specific finishing options, instead of using finishing options that were set in the Template's Media and Section options.

This is only applied when producing print output. It does not modify the original finishing options in either the Section or the Template.

- **Ignore section level finishing:** Check to override finishing options at the document level only.
- **Section to edit:** Use the drop-down to select which Section to apply the options below. The Document level is also listed to edit document-level finishing.
  -  **Settings:** Click the settings button to bring up control options:
    -  **Reload:** Restores the current Section's properties to the default values set in the template for this Section
    -  **Reload All:** Restores all Section properties to the default values set in the template for each Section.
    -  **Apply finishing from:** Displays a list of available Sections. Clicking on a Section name loads that Section's properties into the current Section to edit.
    -  **Apply current finishing to all sections:** Applies the current properties to all Sections.
- **Binding group:**
  - **Style:** What type of Binding to request on the printer. This includes Stapled, Glued, Stitched, Ring, and various other options..
  - **Side:** Sets the side of the paper that the Binding is to occur.
  - **Location:** Sets where the binding is to occur, if applicable. The selections available here are dependent upon the selection made in the Binding **Style**. Only Stapled and Stitched bindings have a **Location** option available to them.
  - **Angle:** Set Stapling or Stitching binding either horizontally, vertically, or at an angle (as supported by printer).

- **Item count:** Select the amount of Staples or Stitches to use. The choice is between the default amount or selecting a specific number using the Count option.

### Tip

The options actually available to you at print time will be printer dependent, so you will need to know the capabilities of your printer, or leave the value set to Default.

- **Area:** The area where the binding can be applied.
- **Hole making group:**  
Hole making options are available only to Ring, Comb (wire and plastic) and Coil Binding **Styles**. The selections will need to be made at run-time based upon the types of binding options available that the printer supports.
  - **Number of holes:** The number of holes to punch for the selected Binding option.
  - **Style**The style of hole punches for the selected Binding option.
  - **Pattern Catalog ID:** The Catalog ID of the selected Binding option.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Grouping Options

The Grouping options separates the job output into multiple blocks that can then be physically separated using split sheets in the printer.

- **Grouping Tabs:** Jobs can be grouped at three different levels, each of which is contained in a tab in this area. The groups/tabs are:
  - **Job** Grouping Fields
  - **Job Segment** Grouping Fields
  - **Document Set** Grouping Fields

All the Fields available to be used for Grouping are contained within the *Available Fields* box in each tab. Fields that you want to use for Grouping need to be added to the *Selected Fields* box via the arrows between the two boxes. Simply select the Field(s) you want to move and click the arrow. Any fields that you decide don't need to be used in Grouping can be returned to the Available Fields box in the same fashion.

- **Page Break Grouping:** Check to enable page break grouping, which separates different groups by the number of pages they contain. For example, enabling the Document Set Grouping Level and creating a page range from 1-5 and 6 to Largest, will create two groups. The first will contain all document sets of 1 to 5 pages, the second will contain any document set of 6 or more pages.
  - **Grouping Level:** Use the drop-down to select which grouping level to use, between **Job**, **Job Segment** or **Document Set**. Only one grouping level can be selected.
  - **Grouping list:** Add  (or remove ) entries to this list to create new groups based upon the number of pages in the level selected above. All groups must be contiguous from 1 to Largest and they must not contain any gaps.
    - **Range Name:** Enter a name identifying the range. It must be unique, but otherwise bears no impact on the range feature.
    - **From:** Enter the starting page number of the range. The first range must start with 1, all other ranges must be contiguous (the "From" range must be one higher than the previous "To" value).
    - **To:** Enter the last page number for the range. The last range must end with a selection of "Largest".

- **Generate page break ranges in reverse order:** Reverses the order of the groups created. By default, grouping will be from smallest to largest. Checking this option creates groups from largest to smallest.
- **Generate page break range groups after normal grouping:** Check this option to first group using the levels above, following which page break grouping are applied. This creates two different levels of grouping, applied in order.

## Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Metadata Options

The Metadata Option page defines metadata tags that will be added to the output file when producing PDF and AFP output in the [Output Creation Presets](#). Metadata tags are ignored in all other output types. The tags are added to each of the levels, as indicated by the tabs on top: **Job**, **Job Segment**, **Document**, **Document Set**, and **Page** Tags.

In each of these levels, a list of tags is available:

- **Always create meta data for this level even when fields are selected:** Select to create a blank meta data entry if no fields are selected. Done to ensure that a meta data store is always available, if required.
- **Tag Name:** Name of the metadata tag added to this level. Once a tag has been added, its name can be edited by double-clicking on the Tag Name.
- **Source Type:** Displays the type of field being used - either Text or Data Field.

- **Source:** For Data Fields only. The Field name from the data mapping configuration whose value will be used for this tag.
- **+ Add Field:** Click to add a new tag to the current level. The Field Selection dialog appears. Select either **Add field meta data** or **Add text meta data**.  
When adding field meta data select a field name from the Field List and click OK to add it as a tag of the same name.
- **- Delete Field:** Click to delete the currently selected tag.
- **↑ Move Up:** Click to move the currently selected tag one position up.
- **↓ Move Down:** Click to move the currently selected tag one position down.

## Separation Options

This page defines how to separate the jobs using subsets, slip sheets, or jogging.

- **Sheet Count Splitting** group.  
This group allows for the splitting of output based upon a pre-determined number of pages
  - **Split:** Use the drop-down to select how to split.
    - **None:** Select to ignore sheet count splitting entirely.
    - **At exactly:** Select to create a split at a specific sheet number.
  - **Every:** Enter the number of sheets at which to split the output.
- **Separation Settings** group.  
This setting is only available if no Sheet Count Split were specified.
  - **Separation:** Use the drop-down to select when a job separation occurs, which is either **None** (no separation) or at the **Job**, **Job Segment**, **Document** or **Document Set** level.
- **Slip Sheets** group
  - **Add slip sheet:** Use the drop-down to select whether to add a slip sheet before or after a specific separation, or whether to use none.
  - **Every:** Use the drop-down to select at which separation to add a slip sheet, at the **Job**, **Job Segment**, **Document** or **Document Set** level.
  - **Media Size:** Use the drop-down to select the media size of the slip sheet.  
If a custom Media Size was chosen:

- **Width:** enter slip sheet page width.
- **Height:** enter slip sheet page height.
- **Jog group**
  - **Jog after every:** Use the drop-down to select when to jog the printer, which is either **None** (no forced jogging) or at the **Job, Job Segment, Document** or **Document Set** level.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Additional Content

There are four different types of additional content that can be added at print time. **Text, Images, Barcodes** and **OMR Marks**. They are used to add static or variable content when generating output. This is useful when driving custom processes on machines using either Barcodes or OMR Marks, as well as allowing the addition of last minute information through text and images.

### Additional Text

Text is added to the output at specific positions. This dialog displays all the text settings:

- **Left:** Displays the distance between the left margin of the page and the text.
- **Bottom:** Displays the distance between the bottom margin of the page and the text .

- **Orientation:** Displays the orientation of the text.
- **Text:** Displays the actual text.

#### Note

The entered text might have been entered over multiple lines, so not all of the text will be displayed here. You might consider this a text entry preview of the text, rather than the complete text entry.

- **Condition:** Displays the condition which is used to determine if text element is to be included or not.
-  **Add:** Click to open the [Additional Text Settings](#) dialog to add a new text entry.
-  **Delete:** Click to delete the currently selected entry.
-  **Edit:** Click to edit the currently selected entry using the [Additional Text Settings](#) dialog.
-  **Duplicate:** Click to create a copy of the entry.

### Additional Images

Images are added at specific positions, with optional dimension constrains. This dialog displays all the configured additional image settings:

- **Left:** Displays the distance between the left margin of the page and the image.
- **Bottom:** Displays the distance between the bottom margin of the page and the image .
- **Orientation:** Displays the orientation of the picture.
- **Filename:** Displays the selected image filename.
- **Condition:** Displays the condition which is used to determine if the image is to be included or not.
-  **Add:** Click to open the "Image Settings" on page 369 dialog to add a new text entry.
-  **Delete:** Click to delete the currently selected entry.
-  **Edit:** Click to edit the currently selected entry using the "Image Settings" on page 369 dialog.
-  **Duplicate:** Click to create a copy of the entry.

## Additional Barcodes

Barcodes are added at specific positions. This dialog displays all the configured additional Barcode settings:

- **Left:** Displays the distance between the left margin of the page and the Barcode .
- **Bottom:** Displays the distance between the bottom margin of the page and the Barcode .
- **Orientation:** Displays the orientation of the Barcode .
- **Type:** Displays the type of Barcode that's added.
- **Text:** Displays a preview of the Barcode contents.
- **Condition:** Displays a preview of the condition.
-  **Add:** Click to add a Barcode. Select from the list of Barcode types that appears. The [Additional Barcode Settings](#) page lists all the available Barcodes, and links to their options..
-  **Delete:** Click to delete the currently selected Barcode entry.
-  **Edit:** Click to edit the currently selected Barcode entry using the [Additional Barcode Settings](#) dialog.
-  **Duplicate:** Click to create a copy of the barcode entry.

## Additional OMR Marks

Optical Mark Recognition (OMR) marks are added at specific positions, with optional dimension constrains. This dialog displays all the configured additional image settings:

- **Left:** Displays the distance between the left margin of the page and the OMR mark.
- **Bottom:** Displays the distance between the bottom margin of the page and the OMR mark.
- **Orientation:** Displays the orientation of the OMR mark.
- **Condition:** Displays the condition which is used to determine if the OMR mark is to be included or not.
-  **Add:** Click to open the "OMR Mark Settings" on page 392 dialog to add a new text entry.
-  **Delete:** Click to delete the currently selected entry.

-  **Edit**: Click to edit the currently selected entry using the "OMR Mark Settings" on page 392 dialog.
-  **Duplicate**: Click to create a copy of the entry.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Additional Text Settings

The Additional Text Settings dialog displays the property of Text added in the "Additional Content" on page 365 page.

- **Position group**:
  - **Orientation**: Use the drop-down to select the orientation of the Text added to the page.
  - **Output once per sheet**: Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Text printed once per sheet rather than once per document page.

### Note

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Text, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Text, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Font group:**
  - **Font Name:** Use the drop-down to select which font type to apply to the Text. The drop-down displays all the fonts installed on the system.
  - **Font Size:** Enter the font size in points (pt).
  - **Bold:** Check to make the Text **bold**.
  - **Italic:** Check to make the Text *italic*.
  - **Color:** Select what **color** the Text will be.
- **Text:** Enter the actual Text to appear on the page in the selected location. The Text can be spread over multiple lines, but no additional formatting can be added within this edit box. The entire Text will be printed use the formatting options selected in the **Font group**.
  -  **Add:** Click to display a list of variable data that can be added to the Text. This includes metadata fields added in the [Metadata Options](#), as well as some document information fields.
- **Condition:** Enter the condition which determines whether or not the Text will be added to the document at print time.  
For details on how to create a conditional, see the Conditionals page.

### Image Settings

The Image dialog displays the properties of the image added in the "Additional Content" on page 365 page.

- **Position group:**
  - **Orientation:** Use the drop-down to select the orientation of the image.
  - **Layer:** Whether this image will appear behind the text (the text will print over the image) or in front of the text (the text behind will be blanked out by the image, as transparent images are not supported)
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Image printed once per sheet rather than once per document page.

#### Note

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the image, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the image, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Filename:** Use the browse button to select an image.

#### Note

Transparent images are not supported.

- **Scaling group:**

Scaling the image expands the image but keeps the aspect ratio. The amount of scale and specific limitations can be applied used a combination of the following options:

  - **Max Width:** Enter the absolute maximum width the image can be scaled to, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Max Height:** Enter the absolute maximum height the image can be scaled to, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

- **Scale:** What scale to apply to the image. The maximum scale is 10.0 to 1. Decimal values are allowed for this field.
- **Condition:** Enter the condition which determines whether or not the image will be added to the document at print time.  
For details on how to create a conditional, see Conditionals page.

## Barcode Options

When adding Barcodes in the "Additional Content" on page 365 page you can select from a series of predetermined Barcode types.

The options for each of these types is described on the following pages:

- "Codabar Settings" below
- "Code 128 Settings" on page 373
- "Code 39 Settings" on page 374
- "Data Matrix Settings" on page 376
- "EAN-128 Settings" on page 378
- "EAN-13 Settings" on page 379
- "EAN-8 Settings" on page 381
- "Interleaved 2 of 5 Settings" on page 383
- "PDF417 Settings" on page 384
- "QR Code Settings" on page 386
- "UPC-A Settings" on page 389
- "UPC-E Settings" on page 390

### Codabar Settings

Codabar barcodes support the following data: 0-9 - \$ : / . + plus the optional specification of start/stop characters. Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.

- **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### Note

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Codabar Properties** group:
  - **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
  - **Bar width ratio:** Set the Barcode bar width.
  - **Default start symbol:** Use the drop-down to select the optional Barcode start character, which defines the encoding mode.
  - **Default stop symbol:** Use the drop-down to select the Barcode stop character, which defines the encoding mode.
  - **Print human readable text:** Check to add a textual version of the Barcode data.
    - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
    - **Font name:** Use the drop-down to select the font with which to display the human readable text.
    - **Font size:** Enter a font size for the human readable text.
    - **Display start/stop symbols** check box: Adds the stop/start symbols to the Barcode text.

- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### Code 128 Settings

Code 128 is a high-density barcode, used for alphanumeric or numeric-only barcodes. It supports all 128 ASCII characters. Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

- **Code 128 Properties** group:
- **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
- **Print human readable text:** Check to add a textual version of the Barcode data.
  - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
  - **Font name:** Use the drop-down to select the font with which to display the human readable text.
  - **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### Code 39 Settings

Code 39 is a discrete, self-checking barcode that is also known as "Alpha39", "Code 3 of 9" (often abbreviated to "3 of 9"), "Code 3/9", "Type 39", "USS Code 39" and "USD-3".

Code 39 data should contain no more than 20 digits from within the following range: Numeric digits: (0-9), upper-case letters (A-Z), seven special characters (- . space \$ / + %) and the start/stop asterisk (\*) character.

If the Extended character set is chosen, then lower-case letters (a-z) and other special ASCII characters can also be included.

Use the following options to configure the output Barcode settings:

- **Position group:**

- **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
- **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

**Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Code 39 Properties group:**
    - **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
    - **Use extended character set:** Check to use the Code 39 Extended character set. This extends the range of supported data to include the full ASCII character set. This adds support for lower case letters (a-z) and the full range of ASCII punctuation and special characters.
    - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger. The smallest Module Width is 0.19mm (high density).
    - **Bar width ratio:** Set the Barcode bar width.
    - **Checksum:** Use the drop-down to select how to deal with the Barcode checksum:
      - **Ignore:** Ignore checksum calculations.
      - **Auto:** Add a checksum character to the Barcode if the initial value does not validate. This is the default value.
      - **Check:** Verify the Barcode has a valid checksum.

- **Add:** Calculate and add a checksum character to Barcode, regardless of current value.
- **Print human readable text:** Check to add a textual version of the Barcode data.
  - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
  - **Font name:** Use the drop-down to select the font with which to display the human readable text.
  - **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### Data Matrix Settings

A Data Matrix barcode is a high-density, two-dimensional (2D) matrix barcode which supports encoded text, numbers, files and digital data. Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### **Note**

If Imposition options such as auto-positioning and scaling were selected,

these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Datamatrix Properties g:**
  - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
  - **Encoding:** The data represented in the symbol can be compressed using one of the following algorithms:
    - **Auto:** Automatically detect the data content and encodes using the most appropriate method. This is the default option.
    - **ASCII:** is used to encode data that mainly contains ASCII alphanumeric characters (ASCII 0-127). Use where Barcode size is a concern and where the data is alphanumeric.
    - **Base 256:** used to encode 8-bit values.
    - **C40:** used for data that mainly consists of numbers and upper-case alphabetic letters.
    - **Text:** used for data that mainly consists of numbers and lower-case alphabetic letters.
    - **None:** Does not use any encoding.
  - **Format:** select the Barcode size format from the drop-down list .
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the Conditionals page.

-  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

## EAN-128 Settings

EAN128 is also known as "EAN/UCC 128", "UCC 128" and "GS1-128". This barcode type not only encodes data, but also provides a mechanism for defining the meaning (or format) of that data. It supports alphanumeric data and some predefined Function Codes. See the [Wikipedia GS1-128 entry](#) for more information.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation**: Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet**: Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left**: Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Bottom**: Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **EAN 128 Properties** group:
    - **Height**: Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
    - **Module Width**: Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.

- **Check Digit marker:** This character is used as a placeholder for the check digit, which will be calculated at runtime. The character must be expressed in Hex.
- **Group separator:** This character is used to define group separation points. The character must be expressed in Hex.
- **Template:** Specify an optional Barcode "template".

Examples:

- `n13` defines a numeric field with exactly 13 digits.
- `n13+cd` defines a numeric field with exactly 13 digits plus a check digit.
- `an1-9` defines an alpha-numeric field with 1 to 9 characters.

Elements can be combined using the '+' symbol.

- **Print human readable text:** Check to add a textual version of the Barcode data.
  - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
  - **Font name:** Use the drop-down to select the font with which to display the human readable text.
  - **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### EAN-13 Settings

EAN-13 barcodes are composed entirely of numerical data. The first 12 digits representing country/economic area, manufacturer and product codes + 1 following checksum digit. Use the following options to configure the output Barcode settings:

- **Position group:**

- **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
- **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

**Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **EAN 13 Properties group:**
    - **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
    - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger. The EAN-13 barcode employs a module width between 0.27mm and 0.66mm.
    - **Checksum:** Use the drop-down to select how to deal with the Barcode checksum:
      - **Ignore:** Ignore checksum calculations.
      - **Auto:** Add a checksum character to the Barcode if the initial value does not validate. This is the default value.
      - **Check:** Verify the Barcode has a valid checksum.
      - **Add:** Calculate and add a checksum character to Barcode, regardless of current value.
    - **Print human readable text:** Check to add a textual version of the Barcode data.
      - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.

- **Font name:** Use the drop-down to select the font with which to display the human readable text.
- **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### EAN-8 Settings

An EAN-8 barcodes are composed entirely of numerical data. It is comprised of 7 data digits containing the country/economic area code and an item reference code, with 1 following checksum digit. Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **EAN 8 Properties:**
  - **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger. The EAN-8 barcode employs a module width between 0.27mm and 0.66mm.
  - **Checksum:** Use the drop-down to select how to deal with the Barcode checksum:
    - **Ignore:** Ignore checksum calculations.
    - **Auto:** Add a checksum character to the Barcode if the initial value does not validate. This is the default value.
    - **Check:** Verify the Barcode has a valid checksum.
    - **Add:** Calculate and add a checksum character to Barcode, regardless of current value.
  - **Print human readable text:** Check to add a textual version of the Barcode data.
    - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
    - **Font name:** Use the drop-down to select the font with which to display the human readable text.
    - **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the Conditionals page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

## Interleaved 2 of 5 Settings

Interleaved 2 of 5 barcodes are also known as "ITF" and "2/5 Interleaved". It is a numeric only barcode whose data must contain an even number of digits, as the barcode uses sequences of two digits interleaved with each other to create a single symbol. If the numeric data contains an odd number of digits, then a leading zero must be added to the beginning of the data.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
  - **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Interleaved 2 of 5 Properties** group:
    - **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
    - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
    - **Bar width ratio:** Set the Barcode bar width.
    - **Print human readable text:** Check to add a textual version of the Barcode data.
      - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.

- **Font name:** Use the drop-down to select the font with which to display the human readable text.
- **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### PDF417 Settings

PDF417 is a two-dimensional, multi-row Barcode. It is used for encoding large amounts of data, with hundreds or even thousands of characters. It encodes alphabetic text, numbers, binary files and actual data bytes.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **PDF417 Properties** group:
  - **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
  - **Row height:** Defines the height of the bars for a single row, measured in pixels, points or metric.
  - **Width to height ratio:** Select the ratio of column width to row height.
  - **Mode:** Use the drop-down to set the compaction mode.
    - **Binary:** allows any byte value to be encoded.
    - **Text:** allows all printable ASCII characters to be end coded (ASCII values 32 to 126 and some additional control characters).
    - **Numeric:** more efficient mode for encoding numeric data.
    - **Auto:** Automatically detect the data content and encodes using the most appropriate method. This is the default option.
  - **Error Correction Level:** Enter the error correction level for the built-in error correction method based on Reed-Solomon algorithms. The error correction level is adjustable between level 0 (just error detection, without correction) and level 8 (maximum error correction). Recommended error correction levels are between level 2 and 5, but the optimal value depends on the amount of data, printing quality of the PDF417 symbol and decoding capabilities.
  - **Rows:** A PDF417 bar code can have anywhere from 3 to 90 rows.
  - **Columns:** The number of data columns can vary from 1 to 30.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the Conditionals page.

-  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

## QR Code Settings

QR Code (Quick Response Code) is a 2D Barcode format that supports alphanumeric, numeric, byte/binary, and Kanji (Japanese-Chinese character) data.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation**: Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet**: Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left**: Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom**: Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

## **QRCode Properties** group:

- **Size by**: Select size from the two options available:
  - **By area**: Connect will try to size the Barcode to fit the specified area by dynamically changing the module width to the **Size** selection. The lower module width limit is governed by the **Minimum module width** selection. Enter the sizes in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

- **By module width:** Connect will try to size the Barcode to the module width of the characters. Large Barcode values will result in larger Barcode and vice versa. Enter the **Module width** in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Encoding:** Define the encoding of the Barcode:
  - **Auto:** Automatically detect the data content and encodes using the most appropriate method. This is the default option.
  - **Numeric:** 7089 numerical characters.
  - **Alphanumeric:** 4296 alphanumerical characters.
  - **Byte:** 2953 characters.
  - **Kanji:** 1817 Japanese/Chinese characters.
- **Version:** Select the preferred QR code version (which sets the data length field) from the 40 available.

**Note**

The Encoding and Version fields work together to determine how many characters are encoded within a *length field*. The following table shows the number of bits in a length field, based upon the selections made:

<b>Encoding</b>	<b>Ver. 1-9</b>	<b>Ver. 10-23</b>	<b>Ver. 27-40</b>
<b>Numeric</b>	10	12	14
<b>Alphanumeric</b>	9	11	13
<b>Byte</b>	8	16	16
<b>Kanji</b>	8	10	12

- **Error Correction Level:** Part of the robustness of QR codes is their ability to sustain “damage” and continue to function even when a part of the QR code image is obscured, defaced or removed. A higher correction level duplicates data within the QR Code to allow for damaged areas. The higher the Error Correction Level, the larger the Barcode will be. The choices are (in order from lowest to highest): **Low**, **Medium**, **Quartile** and **High**.

- **Use ECI for encoding messages as bytes:** Selecting Extended Channel Interpretations (ECI) allows encoding multiple character sets (e.g. Arabic, Cyrillic, Greek, Hebrew) and other data interpretations, into one QR Code symbol.
- **Multi-part QR Code (structured append):** Select to append a QR Code symbol in a structured format.
  - **Part:** indicates the position of the QR Code symbol within the group of Structured Append symbols.
  - **of:** indicates how many Structured Append symbols exist.

#### Note

The Structured Append symbols Part number can never exceed the sum total of Structured Append symbols available (the "of" value). Thus selecting a Part number beyond the existing sum total will increase the sum total to the same value.

- **Use FNC1:** Check to enable Application Identifiers. These are often used to encode links to websites, or to encode production/batch details.
  - **Position:** Select between the two methods for encoding FNC1 characters within QR Codes:
    - **First Position** - uses the GS1 QR Code standard.
    - **Second Position** - uses the AIM QR Code standard. If this option is chosen then the appropriate Application Indicator will also need to be set.
      - **Application ID:** Enter the appropriate QR-Code Application Indicator in accordance with the specific industry or application specifications (as provided by AIM International).
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the Conditionals page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

## UPC-A Settings

The Universal Product Code (UPC-A) Barcode is widely used for tracking trade items in stores and at the point-of-sale. It consists of 12 numerical digits which are uniquely assigned to each trade item.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

### **Note**

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

## **UPC A Properties** group:

- **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
- **Checksum:** Use the drop-down to select how to deal with the Barcode checksum:

- **Ignore:** Ignore checksum calculations.
- **Auto:** Add a checksum character to the Barcode if the initial value does not validate. This is the default value.
- **Check:** Verify the Barcode has a valid checksum.
- **Add:** Calculate and add a checksum character to Barcode, regardless of current value.
- **Print human readable text:** Check to add a textual version of the Barcode data.
  - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
  - **Font name:** Use the drop-down to select the font with which to display the human readable text.
  - **Font size:** Enter a font size for the human readable text.
- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the Conditionals page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

### UPC-E Settings

The Universal Product Code (UPC-E) Barcode is widely used for tracking trade items in stores and at the point-of-sale. It consists of 6 numerical digits which are uniquely assigned to each trade item.

Use the following options to configure the output Barcode settings:

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the Barcode added to the page.

- **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the Barcode printed once per sheet rather than once per document page.

#### Note

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the Barcode, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.

#### UPC A Properties group:

- **Height:** Enter the Barcode height in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Module Width:** Specifies the width of the narrow bars. Changing this value to higher value will generally make the Barcode bigger.
- **Checksum:** Use the drop-down to select how to deal with the Barcode checksum:
  - **Ignore:** Ignore checksum calculations.
  - **Auto:** Add a checksum character to the Barcode if the initial value does not validate. This is the default value.
  - **Check:** Verify the Barcode has a valid checksum.
  - **Add:** Calculate and add a checksum character to Barcode, regardless of current value.
- **Print human readable text:** Check to add a textual version of the Barcode data.
  - **Placement:** Use the drop-down to select whether to place the human readable text above or below the Barcode.
  - **Font name:** Use the drop-down to select the font with which to display the human readable text.
  - **Font size:** Enter a font size for the human readable text.

- **Text:** Enter the text used to generate the Barcode.
  -  **Add** button: Click to display a list of variable data that can be added to the Barcode. This includes metadata fields added in the [Metadata Options](#), as well as some information fields.
- **Condition:** Enter the condition which determines whether or not the Barcode will be added to the document at print time. For details on how to create a conditional, see the [Conditionals](#) page.
  -  **Add** button: Click to display a list of metadata fields, information fields to add, or common expressions to the condition.

## OMR Mark Settings

The **Add OMR** dialog displays the properties of an OMR Mark that was added in the "Additional Content" on page 365 page.

These OMR marks differ from High Capacity Feeder (HCF) generated inserter marks. Those marks are specific to the inserter machine they were created for, whereas these additional OMR marks are completely independent and customizable. These custom OMR marks can be used to cater for inserter machines not currently support by a HCF, or they can be used for any non-inserter related post processing driven by OMR marks

- **Position** group:
  - **Orientation:** Use the drop-down to select the orientation of the OMR Mark added to the page.
  - **Page Side:** Select whether the OMR Mark will print on the front or back of page.
  - **Output once per sheet:** Option relates to [Imposition](#) (also known as N-Up) printing. Select this box to have the OMR Mark printed once per sheet rather than once per document page.

### Note

If Imposition options such as auto-positioning and scaling were selected, these options won't apply to the Additional Content added to the physical N-Up sheet.

- **Left:** Enter the distance between the left margin of the page and the OMR Mark, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Bottom:** Enter the distance between the bottom margin of the page and the OMR Mark, in either metric (cm/mm), inch (in), pixel (px) or point (pt) values.
- **Options Tab:**
  - **Collation Level:** Choices are:
    - **Document:** Treats each document as a group and the group and match marks will be set based upon the start and end of a document.
    - **Document Set:** Treats each document set as a group and the group and match marks will be set based upon the start and end of a document set.
  - **Draw Hot Spots:** This adds a red rectangle around the location of each individual mark in the output, allowing easier checking of the OMR mark logic.
  - **Line Options** group:
    - **Line Thickness:** Sets the thickness of each OMR mark line.
    - **Line Length:** Sets the length of each OMR mark line.
    - **Line Spacing:** Determines how the spacing between each OMR mark line will be determined. The associated control beneath the combination box will be enabled, based upon this selection.
      - **Line Per Inch:** If **Line Spacing** is set to *Lines Per Inch* this option will be enabled. It defines how many lines will print per inch.
      - **Gap Distance:** If **Line Spacing** is set to *Gap Distance* this option will be enabled. It defines the size of the gap between lines.  
i.e. the distance from the bottom of one OMR mark line to the top of the next.
      - **Line Distance:** If **Line Spacing** is set to *Line Distance* this option will be enabled. It defines the distance from the top one line to the top of the next.
  - **Sequence Number Range** group: Allows selection of Start and Stop points for the wrapping page sequence number in a group.  
For example, a range of 2-10 would cause the sequence numbers to iterate as follows: 2, 3, 4, 5, 6, 7, 8, 9, 10, 2, 3, 4 ...

### Note

The Sequence Number iterates per page within a group and is used to identify missing pages in a group.

- **Start:** The starting point for the range
- **Stop:** The end point of the range
- **Start number:** The number to start from (from within the selected range).
- **Match Number Range** group: Allows selection of Start and Stop points for the wrapping match number for a group.  
For example, a range of 1-6, with a Start number of 2 would cause the matched numbers to be as follows: 2, 3, 4, 5, 6, 2, 3, 4, 5, 6, 2, ...

### Note

The Match Number iterates per group and is used to identify missing groups

- **Start:** Start number
- **Stop:** Stop number
- **Condition:** Enter the condition which determines whether or not the OMR Mark will be added to the document at print time.  
For details on how to create a conditional, see the Conditionals page.
- **OMR Marks Tab:**
  - **#:** OMR Mark number (display only).
  - **Type:** Type of OMR Mark (display only).
  - **Value:** OMR Mark Value. These can be selected and altered for Sequence, Match and Parity marks, as described below.
  - **Add:** Add an OMR Mark entry to the table.  
Choices are between:
    - **On:** This represents a mark that is always printed
    - **Off:** This represents a mark that is never printed.  
i.e. it pads the marks out with an empty position

- **Group Start:** This represents a mark that is printed on the first page of a group
- **Group End:** This represents a mark that is printed on the last page of a group

#### Note

In a single page group both Group Start and End marks will print if defined since the page is both the start and end of the group.

- **Sequence:** This represents a mark that is printed when the specified bit is set in the sequence number of the page.  
For example, if the bit for the mark is set to 2 and the sequence number for the page is 5 then it will not print since the value 5 consists of the bits 1 and 4.  
Use the drop down box to select the entry.
- **Match:** This represents a mark that is printed when the specified bit is set in the match number of the group.  
For example, if the bit for the mark is set to 2 and the match number for the group is 3 then it will print since 3 consists of the bits 1 and 2.  
Use the drop down box to select the entry.

#### Note

The match number is the same for all pages in a group

- **Parity:** This mark prints in order to maintain the parity of the number of lines printed on the page. If set to *Even* then it will print if the total count of the other printed marks in the printed is odd.  
For example, by printing the parity mark it will create an even number of marks on the page. And vice versa with *Odd* parity - the parity mark will print if the total number of other printed marks on the page is even in order to keep the overall count odd.  
Use the drop down box to select the entry.
- **Conditional:** Enter the condition which determines whether or not this OMR Mark will be added to the document at print time.  
For details on how to create a conditional, see the Conditionals page

-  **Delete**: Delete an entry from the table
-  **Move up**: Move a entry up the table
-  **Move down**: Move a entry down the table
-  **Edit**: Edit a **Conditional** entry within the table.

#### Tip

You can also double click a **Conditional** entry within the table to edit it.

## PDF Options

The PDF Options page is shown only when a PDF Print output type is selected in the [Print Options](#) dialog.

### • PDF Options Group

- **PDF Type**: Use the drop-down to specify which format the PDF should be generated in. These options are standard PDF, archive format PDF (PDF/A-1b), graphics format PDF (PDF-X4 ) and variable data printing format PDF (PDF-VT).
- **Embed standard fonts**: Click to embed the 14 standard system fonts within the PDF output. This increases the output filesize but makes the PDF output truly portable. Such PDFs print as displayed on screen, regardless of whether the 14 standard fonts are present on the target printing system or not.

#### Note

This box is ignored for PDF/A and PDF-X4 output, as fonts are always embedded in those output types.

- **Add Digital Signature Group**: Check to enable the integration of a digital signature into the PDF.

A digital signature identifies the person signing a document, similarly to a conventional handwritten signature. Unlike a handwritten signature, a digital signature is difficult to forge as it contains encrypted information which is unique to the signer and which can be password protected and verifiable.

- **All Keystores**:

Here you can choose from existing digital signatures, or select new ones.

- **Name:** The user-defined name of the keystore.
- **File:** The file path and name to the keystore file.

This is where you select keystore values.

-  **New:** Click to open the [Key Store](#) dialog to add a new keystore to the list.
-  **Duplicate:** Click to make a copy of the currently selected keystore.
-  **Edit:** Click to edit the currently selected keystore in the [Key Store](#) dialog.
-  **Delete:** Click to delete the currently selected keystore.
-  **Move Up:** Click to move the currently selected keystore up.
-  **Move Down:** Click to move the currently selected keystore down.
- **All Signatures:** Displays a list of signatures to add to the PDF output.
  - **Name:** The user-defined name of the signature.
  - **File:** The file path and name to the signature file.
  - **Alias:** The user-defined alias for the signature.
  -  **New:** Click to open the [PDF Signature](#) dialog to add a new signature to the list.
  -  **Duplicate:** Click to make a copy of the currently selected signature.
  -  **Edit:** Click to edit the currently selected signature in the [PDF Signature](#) dialog.
  -  **Delete:** Click to delete the currently selected signature.
  -  **Move Up:** Click to move the currently selected signature up.
  -  **Move Down:** Click to move the currently selected signature down.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way

through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.

- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Keystore

The security certificate Keystore dialog appears when adding or editing a keystore from the "PDF Options" on page 424 page.

This dialog allows you to select a keystore with a private key.  
The keystores currently supported by Connect are:

- JKS (Java Key Store) format.
- PKCS#12
- PKCS#11

### Note

PKCS#11 requires an extra plug-in not included in the PrintShop Mail Connect installation.

These are the options available in this dialog:

- **Name:** Enter a name for the keystore to describe it within Connect.
- **File:** Enter the path to the keystore file, or use the Browse button to locate the file.
- **Keystore properties group:**
  - **Type:** Use the drop-down to select the appropriate type of the keystore format the file is: JKS, PKCS11, PKCS12.
  - **Provider:** Enter the provider of the keystore.
    - "SUN" for JKS
    - "SunJSSE" for PKCS#12

- "IAIK PKCS#11:1" for PKCS#11
- **Password:** Type in the password that secures the keystore, if the keystore is password protected.
- **Repeat Password:** Re-type in the password that secures the keystore. Once this is done the two Password entry boxes will no longer have the red cross icon (indicating incomplete or unselected) flag beside them.
- **Properties file group:**
  - **File:** Load optional keystore properties file. Could be used to store the password in a file.

## PDF Signature

The PDF Signature dialog appears when adding or editing a signature from the "PDF Options" on page 424 page.

- **Name:** Enter a name that describes the signature entry.
- **Keystore:** Use the drop-down to select which keystore the signature is pulled from. These keystores are set in the "Keystore" on page 427 dialog, called from the "PDF Options" on page 424 page.
- **Signature Properties group:** These are optional Metadata fields associated with the signature, which can be omitted.
  - **Location:** The CPU host name or physical location of the signing.
  - **Reason:** Records the reason for the signing.
  - **Contact:** Information to enable a recipient to contact the signer to verify the signature. For example: a phone number.
  - **Handler:** The PDF reader plugin used to interpret the signature data. It should be left at its default setting (Adobe.PPKLite) unless time-stamping is desired, in which case "Adobe.PPKMS" is likely the best option.
- **Key group:** Refers to a key from the keystore.
  - **Alias:** The user-friendly name of the key
  - **Password:** Enter the password for the key (the same password as was entered in [Key Store](#)).
  - **Repeat Password:** Re-enter the password for the key (same as previous).

- **Apply Time Stamping Authentication group:** Check to enable time stamping authentication.

#### Note

Not available for signatures set to use Adobe.PPKLite Handler.

- **URL:** Select the Time Stamp Authority (TSA) URL address.
- **Account:** Account name specific to the TSA server chosen.
- **Password:** Password specific to the TSA server chosen.
- **Repeat Password:** Repeat of password.
- **Visible Signature group:** Check to add a visible signature to the PDF file.
  - **X:** Enter the horizontal distance between the left side of the page and the left side of the signature, in points (pt).
  - **Y:** Enter the vertical distance between the top of the page and the top of the signature, in points (pt).
  - **Width:** Enter the desired width of the signature, in points (pt).
  - **Height:** Enter the desired height of the signature, in points (pt).

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Spot Color Settings

The Spot Color Settings page is used to match Spot Colors contained within a job to the color settings available on the printer.

This option is only available for AFP and IPDS print outputs, and requires a Printer Definition File (see "Preferences" on page 283) that supports Spot Colors.

- **SpotColor Mapping** table:
  - **SpotColor Name** column: This contains the names of the Spot Colors used within the job.  
If the Spot Color were added to the job within Connect Designer, then those Spot Colors will be automatically included in the table.
  - **SpotColor Number** column: This lists which Spot Color available on the printer is to be associated with the named Spot Color.
-  **Add**: Add a new association between a Spot Color in the job, and the Spot Colors options available on the printer.

### Note

If the Spot Color was not added to the job within Connect Designer itself but are instead included within an external file (such as a PDF) added into the job, then care must be taken to ensure the Name added here is identical to the name reference used within that included file.

-  **Delete**: Delete a Spot Color association.
-  **Edit**: Edit an existing Spot Color association.

## Job Creation Presets

The Job Creation Setting dialog displays a list of available presets and a summary of their settings. Presets can also be edited from this dialog.

- **Data Mapping Configuration**: Use the drop-down to select which data mapping configuration this job creation preset will be based on. The data mapping configuration's model is used for field names in sorting, etc.

- **Configuration Name:** Use the drop-down to select the presets saved in the default location. Click the Gear icon for more options:
  - Click the **Reload** option to look for new presets.
  - Click the **Import Configuration...** option to import one or more Job Presets using a Browse dialog.
- **Properties:** Displays a summary of the settings for this Job Creation Preset.
  - **Has Custom Job Creation Options:** Indicates if any job creation settings have been added. Becomes Yes if any setting in any of the below windows has been added:
    - **Has Data Selection Filter:** Becomes Yes if [Data Filtering Options](#) are set.
    - **Has Sorting:** Becomes Yes if any [Sorting Options](#) are set.
    - **Has Grouping:** Becomes Yes if grouping options are set in the [Grouping and Splitting Options](#).
    - **Page Count Splitting:** Becomes Yes if page count splitting is used in the [Grouping and Splitting Options](#).
    - **Slip Sheets:** Becomes Yes if a slip sheet is set in the [Grouping and Splitting Options](#).
- **Options Group:** These options are checked, or not, depending on the selected preset chosen in the Configuration name.
  - **Use Grouping:** Check to activate the [Grouping and Splitting Options](#) page of the wizard.
  - **Apply filtering and sorting to record selection:** Check to activate the [Data Filtering Options](#) page of the wizard.
  - **Include Metadata (PDF and AFP only):** Check to activate the [Metadata Options](#) page of the wizard.
  - **Override Finishing Options:** Check to activate the [Finishing Options](#).
- **Next:** Click to go to the next page of the Job Creation Wizard, [Data Filtering Options](#)
- **Finish:** At any point during the wizard, click to save the current configurations, whatever page you are on.
- **Cancel:** At any point during the wizard, click to exit the wizard without saving changes.

## Finishing Options

Use this dialog to force specific finishing options, instead of using finishing options that were set in the Template's Media and Section options.

This is only applied when producing print output. It does not modify the original finishing options in either the Section or the Template.

- **Ignore section level finishing:** Check to override finishing options at the document level only.
- **Section to edit:** Use the drop-down to select which Section to apply the options below. The Document level is also listed to edit document-level finishing.
  -  **Settings:** Click the settings button to bring up control options:
    -  **Reload:** Restores the current Section's properties to the default values set in the template for this Section
    -  **Reload All:** Restores all Section properties to the default values set in the template for each Section.
    -  **Apply finishing from:** Displays a list of available Sections. Clicking on a Section name loads that Section's properties into the current Section to edit.
    -  **Apply current finishing to all sections:** Applies the current properties to all Sections.
- **Binding group:**
  - **Style:** What type of Binding to request on the printer. This includes Stapled, Glued, Stitched, Ring, and various other options..
  - **Side:** Sets the side of the paper that the Binding is to occur.
  - **Location:** Sets where the binding is to occur, if applicable. The selections available here are dependent upon the selection made in the Binding **Style**. Only Stapled and Stitched bindings have a **Location** option available to them.
  - **Angle:** Set Stapling or Stitching binding either horizontally, vertically, or at an angle (as supported by printer).
  - **Item count:** Select the amount of Staples or Stitches to use. The choice is between the default amount or selecting a specific number using the Count option.

### Tip

The options actually available to you at print time will be printer dependent, so you will need to know the capabilities of your printer, or leave the value set to Default.

- **Area:** The area where the binding can be applied.
- **Hole making group:**  
Hole making options are available only to Ring, Comb (wire and plastic) and Coil Binding **Styles**. The selections will need to be made at run-time based upon the types of binding options available that the printer supports.
  - **Number of holes:** The number of holes to punch for the selected Binding option.
  - **Style**The style of hole punches for the selected Binding option.
  - **Pattern Catalog ID:** The Catalog ID of the selected Binding option.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Data Filtering Options

The data filtering page is used to filter records and prevent them from being printed. Conditions are evaluated on each record.

## Data Selection Filter

- **Grouping:** Displays the type of line, either a Rule or a rule Grouping. The root of each group of rule is a drop-down selector that defines how the rules inside the grouping work together, which is either to make *any of the rules* or *all of the rules* have to be true for the group to be true.
- **Field:** Use the drop-down to select the field on which to make the comparison.
- **Operator:** Use the drop-down to select the comparison operator for the condition.
- **Value:** Type in a value for the comparison.
- **Add:** Click to add a new line to list. Different options are available in this menu, such as filtering by field, media and finishing properties, or document length.
- **Add a new nested rule group:** Click to add a new grouping at the current level.
- **Delete:** Click to delete the currently selected rule or group. **Note:** deleting a group deletes all rules under it, and this action cannot be undone.
- **Group selected rules as nested rules:** Click to create a group with the currently selected rules.
- **Merge selected rules/ruleset to parent rules:** Click to move the currently selected rule (s) to the parent group.

## Preview

This box displays a textual representation of the conditions set in the data filtering.

## Sorting Options

The sorting options page is used to sort the records in the output. Sorting is done from the top to the bottom, one after the other.

## Sorting Settings

- **Use standard sort:** Sort using the fields below:
  - **Field Name:** Use the drop-down to select which field to sort on.
  - **Order:** Use the drop-down to choose Ascending or Descending.

- **Add:** Click to add a new row to the sort list. The list that appears contains all the fields in the Data Model, as well as a special <Document Length> option which is used to sort by the number of pages in each document.
- **Delete:** Click to delete the currently selected row in the list.
- **Move up:** Click to move the currently selected row up in the list.
- **Move down:** Click to move the currently selected row down in the list.
- **Use external sort:** Sort the records using an external sorting software. A CSV file is exported, sorted by the external application and the sorted CSV file is returned and integrated, with the records now sorted according to the new order in the CSV file.
  - **Command:** Enter either:
    - The full path to the executable that will sort the CSV file.
    - A valid Windows command line instruction to sort the records.  
This instruction should do the following:
      1. Do some processing of the input CSV file which PrintShop Mail Connect will pass through in the position of the `${input}` placeholder.
      2. Generate an output file that contains the sorted data and must be named according the file name PrintShop Mail Connect will pass through in the position of the `${output}` placeholder
 For example: `cmd /C sort /R ${input} ${output}`  
 This would reverse the order of the `${input}` file, and sent the output to the `${output}` file.
- **Separator:** Enter the field separator used in the CSV file, such as a comma (,), pipe (|), semicolon (;), etc.
- **Quote Character:** Enter the quoting character that wraps around any field that contains the separator.
- **Escape Character:** Enter the character use to escape the Quote character if it appears in the field value.
- **Line Ending:** Use the drop-down to select which line ending to use. The selections are: Windows *Carriage Return/Line Feed* combination (CRLF), Linux *Line Feed* (LF) or Apple Macintosh *Carriage Return* (CR).
- **Character Set:** Use the drop-down to select which character set to use when encoding the CSV file. This always defaults to UTF-8, as this caters for all possible characters, is relatively compact (in terms of Unicode character sets) and is compatible with standard ASCII.

- **Exported sort data** group:
  - **First row of sort data has field names** checkbox: select to have field names placed on the first line of the exported CSV file.
  - **Fields to export:** Lists the fields to export in the CSV file. The buttons to the right of the table provide the following functionality:
    -  Click to select from available datafields. The Field Selection dialog will appear, which allows selection of one or several fields from those available.
    -  Click to remove a field from the list.
    -  Click to move fields up or down in the order of output.
  - **Record ID Field:** The Record ID field is a database *Primary Key* field, which is automatically added to the exported data file. The Record ID field name defaults to *RecID*, but can be changed here as desired.
- **Processing returns data** group: This allows the external sort application to introduce new data for each record. This data can be embedded in the metadata and used as the source for additional content within PrintShop Mail Connect. An example usage would be generating a postcode or postal barcode data from address details, making it available for use in PrintShop Mail Connect.
  - **Processing returns data** checkbox: Select this if the sort processing will be returning data. This activates the whole optional **Processing returns data** subsection.
  - **First row of return data has field names** checkbox: select to have field names placed on the first line of the returning datafile.
  - **Return Fields:** Lists the fields available in the selected data mapping configuration that can be used to sort the records. Fields can be added or removed by use of the add datafield () and remove datafield () buttons, or re-arranged with the arrow buttons (). Field names can be altered by selecting the field in the table, and editing the name. Fields can be made available to PrintShop Mail Connect via the "Include in meta data" checkbox. Click the checkbox beside the field name to make that datafield available as meta data.
  - **Record ID Field** selection box: Select which return field is to be the Record ID field.

- **Sorting by** selection box: Select whether the sorting will be by the returned sort order or whether it is to be sorted on a selected datafield.
  - **Sequence Field** selection box: Select the datafield to be sorted on, if such was chosen in the **Sorting by** entry.

## Grouping Options

The Grouping options separates the job output into multiple blocks that can then be physically separated using split sheets in the printer.

- **Grouping Tabs:** Jobs can be grouped at three different levels, each of which is contained in a tab in this area. The groups/tabs are:
  - **Job** Grouping Fields
  - **Job Segment** Grouping Fields
  - **Document Set** Grouping Fields

All the Fields available to be used for Grouping are contained within the *Available Fields* box in each tab. Fields that you want to use for Grouping need to be added to the *Selected Fields* box via the arrows between the two boxes. Simply select the Field(s) you want to move and click the arrow. Any fields that you decide don't need to be used in Grouping can be returned to the Available Fields box in the same fashion.

- **Page Break Grouping:** Check to enable page break grouping, which separates different groups by the number of pages they contain. For example, enabling the Document Set Grouping Level and creating a page range from 1-5 and 6 to Largest, will create two groups. The first will contain all document sets of 1 to 5 pages, the second will contain any document set of 6 or more pages.
  - **Grouping Level:** Use the drop-down to select which grouping level to use, between **Job**, **Job Segment** or **Document Set**. Only one grouping level can be selected.
  - **Grouping list:** Add  (or remove ) entries to this list to create new groups based upon the number of pages in the level selected above. All groups must be contiguous from 1 to Largest and they must not contain any gaps.
    - **Range Name:** Enter a name identifying the range. It must be unique, but otherwise bears no impact on the range feature.

- **From:** Enter the starting page number of the range. The first range must start with 1, all other ranges must be contiguous (the "From" range must be one higher than the previous "To" value).
- **To:** Enter the last page number for the range. The last range must end with a selection of "Largest".
- **Generate page break ranges in reverse order:** Reverses the order of the groups created. By default, grouping will be from smallest to largest. Checking this option creates groups from largest to smallest.
- **Generate page break range groups after normal grouping:** Check this option to first group using the levels above, following which page break grouping are applied. This creates two different levels of grouping, applied in order.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Metadata Options

The Metadata Option page defines metadata tags that will be added to the output file when producing PDF and AFP output in the [Output Creation Presets](#). Metadata tags are ignored in all other output types. The tags are added to each of the levels, as indicated by the tabs on top: **Job**, **Job Segment**, **Document**, **Document Set**, and **Page** Tags.

In each of these levels, a list of tags is available:

- **Always create meta data for this level even when fields are selected:** Select to create a blank meta data entry if no fields are selected. Done to ensure that a meta data store is always available, if required.
- **Tag Name:** Name of the metadata tag added to this level. Once a tag has been added, its name can be edited by double-clicking on the Tag Name.
- **Source Type:** Displays the type of field being used - either Text or Data Field.
- **Source:** For Data Fields only. The Field name from the data mapping configuration whose value will be used for this tag.
- **+ Add Field:** Click to add a new tag to the current level. The Field Selection dialog appears. Select either **Add field meta data** or **Add text meta data**.  
When adding field meta data select a field name from the Field List and click OK to add it as a tag of the same name.
- **- Delete Field:** Click to delete the currently selected tag.
- **↑ Move Up:** Click to move the currently selected tag one position up.
- **↓ Move Down:** Click to move the currently selected tag one position down.

## Output Creation Settings

The Output Creation Settings dialog displays a list of available presets and a summary of their settings. Presets can also be edited from this dialog.

- **Configuration Name:** Use the drop-down to select the presets saved in the default location. Click the Gear icon for more options:
  - Click the **Reload** option to look for new presets.
  - Click the **Import Configuration...** option to import one or more Output Presets using a Browse dialog.
- **Properties:** Displays a summary of the settings for this Output Creation Preset.
  - Output Type: Displays the print technology used, as defined in the [Print Options](#)
  - Inserter: Indicates whether Inserter Marks have been added in the [Inserter Marks](#) dialog. Expand to see which High Capacity Feeder (HCF) model is loaded.
  - Imposition: Indicates if Imposition has been set in the "Imposition Options" on page 418 dialog. Expand to see the specific imposition settings.
  - Has custom printer settings: Indicates if custom printer settings have been set in the [Printer Settings](#) dialog. Expand to see the list of settings.

- Output to: Indicates where the output will be done, either to a file or a printer.
- Has Custom Finishing: Indicates that the output creation settings contain custom finishing overrides.

Click Next in this dialog to see the [Print Options](#) window where output creation settings are selected.

## Print Options

The Print Options page is the first page of both the **Advanced Print Wizard** and the [Output Creation Settings](#) Preset .

This page is the most important of the Advanced Print Wizard.

The other pages that appear throughout the Wizard are determined by the selections made on this page.

The choices can be broken down as follows:

- **Printer** section:

- **Model:** Use the drop-down to select the printer language / output type that will be generated.  
Connect output options cover a range of industry standard print output types. These include AFP, IPDS, PCL, PDF and PostScript (including PPML, VIPP and VPS variants), with a range of quality settings available.

### Note

By default, Connect displays only the PDF output option, but other print output types can be added to the Printer Model drop down list via the Gear button 

For more information on how to do this, see "Adding print output models to the Print Wizard" on page 499.

- **Output Options** section:

- **Output Local** checkbox: Select to have the output created using the local Print Server.

- **Output Type** choices:
  - **Prompt for file name:** Select to output to a local file on the hard drive. When this option is selected, no other configuration is necessary. A Save As dialog will appear to allow selection of the folder and filename.
  - **Directory:** Select to output to a local folder on the machine.
    - **Job Output Mask:** The name of the file that will output. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
    - **Job Output Folder:** The path on the disk where the file is produced. Please note that the folder must exist, or output will fail when produced through the server.
  - **LPR Queue:** Select to send the print job to an LPR queue. It is assumed that the print technology is supported by the system receiving the LPR job.
    - **Local Printer:** The IP or host name of the printer or machine where the LPD is installed and will receive
    - **Queue Name:** The queue name that will accept the job on the LPD. Default is generally "auto".
    - **Job Owner Name:** Optional entry for adding the name of the job owner.
    - **Job Name:** The name of the output file. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
  - **Windows Printer:** Select to send the Print Job to a Printer Queue. The job is rendered as a PDF before being printed through the Windows driver.
    - **Windows Printer:** Use the drop-down to select the windows printer queue where the job will be sent.
    - **Job Owner Name:** Optional entry for adding the name of the job owner.
    - **Job Name:** The name of the output file. You can use `${template}` as a variable for the name of the Designer Template used to generate the output.
  - **PDF Rendering Options (PDF output only):**
    - **Auto-rotate and center:** Check to automatically select the page orientation that best matches the content and paper.

- **Choose paper source by page size:** Check to use the PDF page size to determine the output tray rather than the page setup option. This option is useful for printing PDFs that contain multiple page sizes on printers that have different-sized output trays.
- **Scale:**
  - **None:** Select to not scale any page, whether it fits or not.
  - **Expand to printable area:** Select to expand any page to fit the page area. Pages larger than the paper size are not resized.
  - **Shrink to printable area:** Select to shrink any page to fit the page area. Pages smaller than the paper size are not resized.
- **Production Options:**
  - **Booklet Imposition** checkbox: Check to tell the printer to generate a booklet for the print output. Booklet options are set in the "Booklet Options" on page 417 page. This option is unselected by default unless selected in the Designer "Print Section Properties" on page 300.
  - **Cut and Stack Imposition** checkbox: Check to enable Cut & Stack Imposition, which is set in the "Imposition Options" on page 418 page.
  - **Add Inserter marks** checkbox: Check to enable inserter mark functionality, which is set in the "Inserter Options" on page 421 page.
  - **Override Finishing options** checkbox: Check to configure custom "Finishing Options" on page 403, such as binding.
  - **Print virtual stationery** checkbox: Check to enable virtual stationery in the output.
  - **Use grouping** checkbox: Check to configure grouping of output into jobs, job segments or document sets. See "Grouping Options" on page 408.
  - **Include meta data** checkbox: Check to add meta data to the output. This can be done at Job, Job Segment, Document, Document Set and Page level. See "Metadata Options " on page 409.
  - **Separation:** Check to activate the "Separation Options" on page 415 page of the wizard.
  - **Add additional content** checkbox: Check to activate the "Additional Content" on page 365 page of the wizard.

- **Records** section:
  - **Record Range:** Allows selection of a range of records or a custom selection. You can specify individual records separated by semi-colons (;) or ranges using dashes. For example: 2;4;6-10 would print pages 2, 4, 6, 7, 8, 9 and 10.
- **Copies** section:
  - **Copies:** Enter the number of copies to print, of each record.
  - **Collate:** When printing multiple copies you can check this checkbox to have the record copies printed together. For example in a three record job the records would print out as 1-1-2-2-3-3, rather than 1-2-3-1-2-3.
- **Pure Color Thresholds** section:
 

This section is valid for AFP, IPDS and PCL only. It applies to elements within the record that are shades of gray, rather than black or white.

  - **Black Threshold Percentage:** The percentage of shading at which the element will appear as full black, rather than dark gray.
  - **White Threshold Percentage:** The percentage at which the element will appear as full white, rather than light gray.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Printer Settings

The Printer Settings page defines options on the printer. It is available for PostScript (and the VIPP and VPS variants of PostScript) only.

- **Map media by** options: Select from following choices:
  - **Media Attribute** displays all Media details, except the Tray selection.
  - **Tray** displays just the Media name and Tray selections.
  - **Both** displays all Media details.
- Tray selection columns:
  - **Media**: Lists the Media name, as defined in the template.
  - **Tray**: Use the drop-down to select in which tray to send any page using the media.
  - **Position**: Enter a MediaPosition option on the printer to define the media to use.
  - **Weight**: Enter a weight for the paper.
  - **Type**: Use the drop-down to select which type of stock to use on the printer.
  - **Color**: Use the drop-down to select which color the paper should be on the printer.

## Separation Options

This page defines how to separate the jobs using subsets, slip sheets, or jogging.

- **Sheet Count Splitting** group.

This group allows for the splitting of output based upon a pre-determined number of pages

  - **Split**: Use the drop-down to select how to split.
    - **None**: Select to ignore sheet count splitting entirely.
    - **At exactly**: Select to create a split at a specific sheet number.
  - **Every**: Enter the number of sheets at which to split the output.
- **Separation Settings** group.

This setting is only available if no Sheet Count Split were specified.

- **Separation:** Use the drop-down to select when a job separation occurs, which is either **None** (no separation) or at the **Job, Job Segment, Document** or **Document Set** level.
- **Slip Sheets** group
  - **Add slip sheet:** Use the drop-down to select whether to add a slip sheet before or after a specific separation, or whether to use none.
  - **Every:** Use the drop-down to select at which separation to add a slip sheet, at the **Job, Job Segment, Document** or **Document Set** level.
  - **Media Size:** Use the drop-down to select the media size of the slip sheet.  
If a custom Media Size was chosen:
    - **Width:** enter slip sheet page width.
    - **Height:** enter slip sheet page height.
- **Jog** group
  - **Jog after every:** Use the drop-down to select when to jog the printer, which is either **None** (no forced jogging) or at the **Job, Job Segment, Document** or **Document Set** level.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Booklet Options

The Booklet Options page defines how to generate booklets in the output. It is used in conjunction with [Imposition](#) settings, which will appear after the Booklet entries have been made.

This page includes a handy illustration that displays how the final binding would look, based upon the current selections.

Options:

- **Configuration:** Use the drop-down to select the type of binding to use:
  - **Saddle Binding:** This binding places all the pages in a stack, binds the middle and folds the stack as one.
  - **Perfect Binding:** This binding type is often used for books. Pages are folded in the middle and then set side by side. The pages are then bound along the folded "spine".
  - **1 up Perfect Binding:** This binding does not contain any folding. The pages are lined up side by side and bound along one edge.
- **Booklet Binding Edge:** Use the drop-down to select the side on which to bind the booklet.

Optional **Cover Page** selections are available to Saddle Binding only.

- **Cover Page** checkbox: Check to enable cover pages to be created with the options below:
  - **Media** selections:
    - **Cover Media Size:** Use the drop-down to select the media size for the cover page, or use a Custom size and select **Width** and **Height** values.
  - **Front Cover** selections:
    - **Blank:** Select to add no data to the front cover.
    - **First page on outside and second page on inside:** Select to use the first 2 pages as the inside and outside of the front cover.
  - **Back Cover** selections:

- **Blank:** Select to add no data to the back cover.
- **Last two pages on inside and outside:** Select to use the final 2 pages as the inside and outside of the back cover.

## Imposition Options

Imposition refers to the printing of multiple pages on a single sheet. This is also known as N-Up printing.

The options on this page allow for the setting of imposition repetition, order, margins and markings.

- **Sheet Size** group:
  - **Final Media Size:** Use the drop-down to select the size of the media where the output is printed. The size of the media should be equivalent to the initial Section size multiplied by the number of repetitions, added with the margins and spaces between the repetitions.  
If *Custom* media size is selected, enter the custom **Width** and **Height** values.

### Note

The Sheet Size cannot be altered if a Cover Page was selected in the "Booklet Options" on the previous page Page.

- **Sheet Rotations:** Select aspect ratio of media (Landscape or Portrait), or allow Connect to automatically determine the proper aspect ratio (Auto-Rotate).
- **Position:** Select from following options:
  - **Auto-positioned:** This option creates unscaled imposition-ed pages.
  - **Scale to fit:** Scales the imposition-ed pages so they fit on the N-Up stock. The scaled pages are then auto-positioned as usual.
  - **Offset:** Allows for the selection of an offset position. The imposition-ed pages will be laid out so that the top left corner of the top left imposition-ed page is located at the selected offset.  
If Offset is chosen, then the **Left Offset** and **Top Offset** selection boxes become active.

### Note

The offset measures from the top left of the physical N-Up sheet to the top left imposition-ed page. If Auto-Rotate is selected (causing the N-Up stock to be rotated to fit the imposition-ed pages) then the measurement becomes the top left position of the rotated stock. i.e. The top left corner does not rotate with the stock.

### Note

If *Booklet Binding* has been selected, then the **Position** settings are pre-set and cannot be altered here.

- **Rotate final output Sheet 180 degrees (upside down):** Select to flip the output upside down.
- **Repetition group:**  
Allows selection of how many Sections are to be placed, both Horizontally and Vertically. This is the total number of items, not the number of additional items being placed.

### Note

If *Booklet Binding* were selected, some of these settings will be determined by the options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Space Between group:**  
Allows selection of the amount of blank space to add between each repetition.

### Note

If *Booklet Binding* were selected, some of these settings will be determined by the

options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Order group:**

**Note**

If *Booklet Binding* were selected, some of these settings will be determined by the options made within the "Booklet Options" on page 417 Page and they cannot be altered here.

- **Page Order:** Select in which direction to go when adding sections to the output:
  - **Left to right, then top to bottom**
  - **Right to left, then top to bottom**
  - **Top to bottom, then left to right**
  - **Top to bottom, then right to left**
- **Stack Depth:** Enter a stack depth or use the arrows to increment or decrement.
- **Reverse Pages:** Select this option to reverse the order of pages.  
This would print the final record on the first page and the first record on the last page.
- **Force simplex:** Select this option to make the output Simplex, rather than the imposition default of Duplex.
- **Bleed Margins group:**
  - **Top, Bottom, Left, Right:** Enter the bleed margins for each side of the page.
- **CropMarks group:**
  - **Type:** Use the drop-down to select the type of crop marks to add to the page.
  - **Offset:** How much separation (if any) to leave between the vertical and horizontal corner markings.
  - **Width:** Select the width of the crop mark lines.
  - **Length:** Select the Length of the crop mark lines.

## Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Inserter Options

The Inserter Options page allows the selection of a High Capacity Feeder (HCF) model. These machines are also commonly referred to as Inserters or Folder-Inserters.

The options available on this page are dependent upon the model selected.

The options selected on this page influence the position of the markings set on the next page:

**"Mark Position Options" on page 423.**

- **Model:** Use the drop-down to select from any previously loaded Inserter model, or use the Browse button to select a HCF file to load a new Inserter model.  
An image representing the chosen folder-inserter is displayed under the list, along with the HCF file details.
- **Options Group:**  
The options available here are all Inserter dependent, and thus will change based upon the Inserter model selection.  
To see how the selected Inserter markings would look on the printed page, click the Next button to move to the "Mark Position Options" on page 423 page, which has a preview of the page. You can move back and forward between these two pages until you are entirely satisfied with the selections made.
  - **Mark Configuration:** Use the drop-down to select the type of markings to add. This selection basically equates to the amount of area the markings will take up on the

printed page.

- **Fold Type:** Use the drop-down to select the type of fold to apply to the paper. This will impact upon where on the page the markings will be placed.
- **Collation level:** Select whether the markings will be made at Document level, or Document Set level.
- **Print marks on back:** Check to place the Inserter Marks on the rear of the page.
- **Selective Inserts:** If selective inserts are supported by the chosen Mark Configuration you can select what markings to include and whether those markings are to included based upon some conditional setting.  
For example, you could add a marking to the third page of a document by making the selection Conditional and then setting the Condition entry to "page.nr = 3".
- **Clear Background Area:** Check to add a white background to the OMR, preventing background colors or elements interfering with the OMR Markings when they are read by the Inserter.
- **Margins:**
  - **Same for all sides:** Check so that the Left margin selection is used to set all sides identically.
  - **Left, top, right, bottom:** Enter a measure for the margins on each side of the OMR Marks.
- **Custom OMR mark sizing:** If supported by the chosen Mark Configuration you can select a Custom OMR size.  
You can select from any of the following, or leave the entries blank to use default values:
  - **Line length:** Enter a value between 10.16mm and 20mm.
  - **Line thickness:** Enter a value between 0.254mm and 0.63mm.
  - **Gap distance:** Enter a millimeter value 2.91mm and 4.2mm.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way

through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.

- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Mark Position Options

This page displays a Preview of the output and the possible locations to place the inserter marks. The initial settings are determined by the selections made within the "Inserter Options" on page 421 page.

You can move back and forward between these two pages to perfect the settings, or you could move the inserter mark box to the desired location on the preview.

### Preview box:

- The *pink area* displays the areas of the page where inserter marks can be positioned.
- The *small checkered box* displays the current location of the inserter marks. This box is selectable and can be dragged to the desired location within the printable (pink) areas. If the box is placed outside the printable areas the page will display an error and prevent attempts at leaving the page.

**Below the Preview box** are buttons which allow control of the Preview box. The selections that can be made are:

-  **First Page**: Click to jump to the first page.
-  **Previous Page**: Click to move to the previous page.
-  **Next Page**: Click to move to the next page.
-  **Last Page**: Click to jump to the last page.
- **Show Page**: Use the up and down arrows or type a page number to display a specific page within the document.
-  **Zoom in/out**: Click to zoom in or out by 25%

-  **Zoom Level:** Use the drop-down to select a predefined level or enter a zooming percentage.

## PDF Options

The PDF Options page is shown only when a PDF Print output type is selected in the [Print Options](#) dialog.

- **PDF Options Group**

- **PDF Type:** Use the drop-down to specify which format the PDF should be generated in. These options are standard PDF, archive format PDF (PDF/A-1b), graphics format PDF (PDF-X4 ) and variable data printing format PDF (PDF-VT).
- **Embed standard fonts:** Click to embed the 14 standard system fonts within the PDF output. This increases the output filesize but makes the PDF output truly portable. Such PDFs print as displayed on screen, regardless of whether the 14 standard fonts are present on the target printing system or not.

### Note

This box is ignored for PDF/A and PDF-X4 output, as fonts are always embedded in those output types.

- **Add Digital Signature Group:** Check to enable the integration of a digital signature into the PDF.

A digital signature identifies the person signing a document, similarly to a conventional handwritten signature. Unlike a handwritten signature, a digital signature is difficult to forge as it contains encrypted information which is unique to the signer and which can be password protected and verifiable.

- **All Keystores:**

Here you can choose from existing digital signatures, or select new ones.

- **Name:** The user-defined name of the keystore.
- **File:** The file path and name to the keystore file.

This is where you select keystore values.

-  **New:** Click to open the [Key Store](#) dialog to add a new keystore to the list.
-  **Duplicate:** Click to make a copy of the currently selected keystore.

-  **Edit**: Click to edit the currently selected keystore in the [Key Store](#) dialog.
-  **Delete**: Click to delete the currently selected keystore.
-  **Move Up**: Click to move the currently selected keystore up.
-  **Move Down**: Click to move the currently selected keystore down.
- **All Signatures**: Displays a list of signatures to add to the PDF output.
  - **Name**: The user-defined name of the signature.
  - **File**: The file path and name to the signature file.
  - **Alias**: The user-defined alias for the signature.
  -  **New**: Click to open the [PDF Signature](#) dialog to add a new signature to the list.
  -  **Duplicate**: Click to make a copy of the currently selected signature.
  -  **Edit**: Click to edit the currently selected signature in the [PDF Signature](#) dialog.
  -  **Delete**: Click to delete the currently selected signature.
  -  **Move Up**: Click to move the currently selected signature up.
  -  **Move Down**: Click to move the currently selected signature down.

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## PDF Digital Signature Options

### PDF Signature

The PDF Signature dialog appears when adding or editing a signature from the "PDF Options" on page 424 page.

- **Name:** Enter a name that describes the signature entry.
- **Keystore:** Use the drop-down to select which keystore the signature is pulled from. These keystores are set in the "Keystore" on the next page dialog, called from the "PDF Options" on page 424 page.
- **Signature Properties group:** These are optional Metadata fields associated with the signature, which can be omitted.
  - **Location:** The CPU host name or physical location of the signing.
  - **Reason:** Records the reason for the signing.
  - **Contact:** Information to enable a recipient to contact the signer to verify the signature. For example: a phone number.
  - **Handler:** The PDF reader plugin used to interpret the signature data. It should be left at its default setting (Adobe.PPKLite) unless time-stamping is desired, in which case "Adobe.PPKMS" is likely the best option.
- **Key group:** Refers to a key from the keystore.
  - **Alias:** The user-friendly name of the key
  - **Password:** Enter the password for the key (the same password as was entered in [Key Store](#)).
  - **Repeat Password:** Re-enter the password for the key (same as previous).
- **Apply Time Stamping Authentication group:** Check to enable time stamping authentication.

#### Note

Not available for signatures set to use Adobe.PPKLite Handler.

- **URL:** Select the Time Stamp Authority (TSA) URL address.
- **Account:** Account name specific to the TSA server chosen.

- **Password:** Password specific to the TSA server chosen.
- **Repeat Password:** Repeat of password.
- **Visible Signature group:** Check to add a visible signature to the PDF file.
  - **X:** Enter the horizontal distance between the left side of the page and the left side of the signature, in points (pt).
  - **Y:** Enter the vertical distance between the top of the page and the top of the signature, in points (pt).
  - **Width:** Enter the desired width of the signature, in points (pt).
  - **Height:** Enter the desired height of the signature, in points (pt).

### Advanced Print Wizard navigation options

- **Load** button: Click to select a previously created Output Creation Preset. This will change the Advanced Print Options to match the entries contained within the Preset.
- **Preview** button: Click to launch a [Proof Preview](#) window, which displays how the printed output would look based upon the currently chosen selections.
- **Back** and **Next** buttons: Used to navigate back and forth through all the selected options within the Wizard. Up until the Print button is pressed, one can reverse all the way through the wizard to return to the main selection page (the "Print Options" on page 411 page) and add or remove printing options from the print run.
- **Print** button: Click to produce print output according to the current settings. This can be done at any point within the Wizard, whether or not the options selected in the the "Print Options" on page 411 page have been completed or not.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

### Keystore

The security certificate Keystore dialog appears when adding or editing a keystore from the "PDF Options" on page 424 page.

This dialog allows you to select a keystore with a private key.

The keystores currently supported by Connect are:

- JKS (Java Key Store) format.
- PKCS#12

- PKCS#11

#### Note

PKCS#11 requires an extra plug-in not included in the PrintShop Mail Connect installation.

These are the options available in this dialog:

- **Name:** Enter a name for the keystore to describe it within Connect.
- **File:** Enter the path to the keystore file, or use the Browse button to locate the file.
- **Keystore properties group:**
  - **Type:** Use the drop-down to select the appropriate type of the keystore format the file is: JKS, PKCS11, PKCS12.
  - **Provider:** Enter the provider of the keystore.
    - "SUN" for JKS
    - "SunJSSE" for PKCS#12
    - "IAIK PKCS#11:1" for PKCS#11
  - **Password:** Type in the password that secures the keystore, if the keystore is password protected.
  - **Repeat Password:** Re-type in the password that secures the keystore. Once this is done the two Password entry boxes will no longer have the red cross icon (indicating incomplete or unselected) flag beside them.
- **Properties file group:**
  - **File:** Load optional keystore properties file. Could be used to store the password in a file.

## Designer JavaScript API

In Designer templates, every bit of information can be tailor-made, using scripts. Most scripts can be made using one of the Script Wizards (see "Personalizing content" on page 202). However, when you want to do more than what you can do with a Wizard, you may write a

script yourself. If you are not familiar with writing scripts, please read "Writing your own scripts" on page 223 first.

All scripts in the Designer have to be written in JavaScript.

If you don't know JavaScript, the many examples given in this API will help you get started.

It is worth the effort, however, to familiarize yourself with the JavaScript syntax. For a simple script all you need to know can be found on the following web pages:

[http://www.w3schools.com/js/js\\_syntax.asp](http://www.w3schools.com/js/js_syntax.asp) and [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp).

## Designer API

The "Designer Scripts API" below describes the objects and functions that are available in **template scripts**, created inside the Scripts pane. Template scripts change the contents of sections in a template.

## Control Script API

**Control Scripts** are a special kind of Designer Scripts. They don't touch the content of the sections themselves, but they change the way a template is outputted, for example by selecting or omitting sections from the output.

For more information about Control Scripts and their use, see "Control Scripts" on page 240.

Features that are specific to Control Scripts are listed in the "Control Script API" on page 477.

## Designer Scripts API

This page lists the global objects and functions that are available in scripts, created inside the Scripts pane. Click through to an object or function to get a description and examples.

If you are not familiar with writing scripts, see "Writing your own scripts" on page 223.

### Control Scripts

Control Scripts are a special kind of Designer Scripts. They don't touch the content of the sections themselves, but they change the way a template is outputted, for example by selecting or omitting sections from the output. For more information about Control Scripts and their use, see "Control Scripts" on page 240. Features that are specific to Control Scripts are listed in the "Control Script API" on page 477.

## Objects

Object	Description
"results" on page 472	This object is used to manipulate the content of the template. It contains the HTML element or set of HTML elements that match the selector of the script, specified in the script editor. This object is not available in Control Scripts, because that type of script doesn't have a selector (see "Control Scripts" on page 240).
"record" on page 468	The record in the main data set that is currently being merged. To get the value of a field in the record, use <code>record.fields['fieldname']</code> or <code>record.fields.fieldname</code> .
"logger" on page 462	Global object that allows you to log messages.
locale	Defines which locale to use. See "Locale" on page 201.
"formatter" on page 456	Global object that allows you to format values (such as a date or number).
automation	This object encapsulates the properties of the Workflow process that triggered the current operation. Not available in PrintShopMail Connect.
"merge" on page 479	The <code>merge</code> object is mainly used in Control Scripts. It gives access to the template with all of its contexts and sections. It doesn't give access to the content of the sections. To change the content of a section, you would create a script with a selector and use the <code>results</code> object in the script (see "results" on page 472).

## Global functions

Function	Description
"loadhtml()"	Loads HTML data from a HTML (snippet). The returned HTML can be

Function	Description
on page 459	placed into a variable or into a set of HTML elements.
"loadjson()" on page 461	Loads json data from a URL. This is a simple way to retrieve content from external systems.
"query()" on page 467	Performs a query in the template's contents and creates a new result set containing the HTML elements that match the given CSS selector.

### Examples of iterator functions

Function	Description
"Each" on page 451	A generic iterator function, to iterate over the elements in the result set
"For...in" on page 455	Iterates over the enumerable properties of an object, in arbitrary order. For each distinct property, statements can be executed.

### add()

The add() function allows you to add elements to a set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467).

### add(content)

Returns the union of this result or result set and other content.

### content

*A query result. This can be an HTML string or a result set.*

### Examples

#### Add one result set to another

This script adds one query result to another and sets the background color to yellow.

```
query("#test1").add(query("#test2")).css("background", "yellow");
```

Note: the way the functions `add()` and `css()` are used in this script is called 'chaining'. Chaining is optional; the same could be achieved by storing the results of the queries in a variable:

```
var myResult = query("#test1");  
myResult.add(query("#test2");  
myResult.css("background", "yellow");
```

### Creating an empty result set and adding elements to it

The following script loads snippets in an iteration and adds their elements to an empty result set (using `query()`). Then it replaces a placeholder in the template with the new result.

```
var chapters = query();  
for ( var i = 1; i <= 4; i++) {  
chapters = chapters.add(loadhtml('snippets/Chapter' + i +  
' .html')));  
}  
results.replaceWith(chapters);
```

Selector	Matched element	Matched element after script execution
#chapters	<p id="chapters">{{chapters}}</p>	<h1>Chapter 1</h1> <p>Lorem ipsum...</p> <h1>Chapter 2</h1> <p>Lorem ipsum...</p> <h1>Chapter 3</h1> <p>Lorem ipsum...</p> <h1>Chapter 4</h1> <p>Lorem ipsum...</p>

### addClass()

Adds the specified class(es) to each element in a set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). This has no effect if the class is already present.

## **addClass(className)**

Adds the specified class(es) to each element in a result set. Has no effect if the class is already present.

### **classname**

*String, space separated list of class names.*

### Examples

This script adds a class name to a paragraph.

```
results.addClass("foo");
```

Selector	Matched element	Matched element after script execution
p	<p>Hello world</p>	<p class="foo bar">Hello world</p>

The following script adds two class names to a paragraph.

```
results.addClass("foo bar");
```

Selector	Matched element	Matched element after script execution
p	<p>Hello world</p>	<p class="foo bar">Hello world</p>

## **after()**

Insert content after each element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). See also: "before()" on page 439.

## **after(content)**

Insert content after each element in the set of HTML elements that match the selector of the script, or of another query in the template (see "query()" on page 467). After creates a new result set.

## **content**

*String, HTML string or result set to insert after the matched elements. In case a plain text string is provided, it is automatically wrapped in a `<span>` element to avoid orphan text nodes to appear in the `<body>` element.*

## Examples

This script looks up an element with the ID `#salesrep` and inserts a paragraph after it.

```
query("#salesrep").after("<p>Lorem ipsum</p>");
```

Matched element	Matched element after script execution
<code>&lt;p id="salesrep"&gt;Peter Parker&lt;/p&gt;</code>	<code>&lt;p id="salesrep"&gt;Peter Parker&lt;/p&gt;</code> <code>&lt;p&gt;Lorem ipsum&lt;/p&gt;</code>

This script looks up an element with the ID `#salesrep`, sets its text color to red and inserts a paragraph after it.

```
query("#salesrep").after("<p>Lorem ipsum</p>").css("color","red");
```

Matched element	Matched element after script execution
<code>&lt;p id="salesrep"&gt;Peter Parker&lt;/p&gt;</code>	<code>&lt;p id="salesrep" style="color: red;"&gt;Peter Parker&lt;/p&gt;</code> <code>&lt;p&gt;Lorem ipsum&lt;/p&gt;</code>

Note: the way the functions `after()` and `css()` are used in this script is called 'chaining'. Chaining is optional; the same could be achieved by storing the result of the query in a variable:

```
var salesrep = query("#salesrep");  
salesrep.after("<p>Lorem ipsum</p>");  
salesrep.css("color","red");
```

The following script inserts a paragraph after the elements in the `results` (the set of HTML elements that match the selector of the script).

```
results.after("<p>Lorem Ipsum</p>");
```

Matched element	Matched element after script execution
<code>&lt;p id="salesrep"&gt;Peter Parker&lt;/p&gt;</code>	<code>&lt;p id="salesrep"&gt;Peter Parker&lt;/p&gt;</code>

Matched element	Matched element after script execution
	<p>Lorem ipsum</p>

This script looks for the string "Lorem " in the `results` (the set of HTML elements that match the selector of the script) and inserts the string "ipsum" right after that text. The string is automatically enclosed in a span.

```
results.find("Lorem ").after("ipsum");
```

Matched element	Matched element after script execution
<p>Lorem dolor sit amet, consectetur adipiscing elit.</p>	<p>Lorem <span>ipsum</span> dolor sit amet, consectetur adipiscing elit.</p>

This script looks up an element with the ID #salesrep and inserts a string after it. The string is automatically enclosed in a span.

```
query("#salesrep").after("Lorem Ipsum");
```

Matched element	Matched element after script execution
<p id="salesrep">Peter Parker</p>	<p id="salesrep">Peter Parker</p> <span>Lorem Ipsum</span>

## append()

Insert content at the end of each element in the set of each element in a set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). See also: "prepend()" on page 464.

## append(content)

Insert content as the last element to each element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). [Append](#)

creates a new result set.

### content

*String, HTML string or result set to insert after the elements. In case a plain text string is provided, it is automatically wrapped in a `<span>` element to avoid orphan text nodes to appear in the `<body>` element.*

### Examples

This script appends a paragraph to the `results` (the set of HTML elements that match the selector of the script).

```
results.append("<p>Peter Parker</p>");
```

Selector	Matched element	Matched element after script execution
<code>#box</code>	<code>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt;</code>	<code>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</code>

This script appends a string to the `results` (the HTML elements that match the selector of the script). The string is added to the end of the matched element(s) and wrapped in a Span element.

```
results.append("Peter Parker");
```

Selector	Matched element	Matched element after script execution
<code>.name</code>	<code>&lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p class="name"&gt;&lt;b&gt;Name: &lt;/b&gt;&lt;/p&gt; &lt;/div&gt;</code>	<code>&lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p class="name"&gt;&lt;b&gt;Name: &lt;/b&gt;&lt;span&gt;Peter Parker&lt;/span&gt;&lt;/p&gt; &lt;/div&gt;</code>

This script's selector is `<div>`, so the script appends a paragraph to all Div elements in the template.

```
results.append("<p>Peter Parker</p>");
```

Selector	Matched element	Matched element after script execution
div	<pre>&lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt; &lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt;</pre>	<pre>&lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt; &lt;div&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

The following script appends a snippet to a Div element with the ID `box`.

```
var a = loadhtml('snippets/snippet_name.html');
results.append(a);
```

Selector	Matched element	Matched element after script execution
#box	<pre>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

This script looks for an element with the ID `box` and appends a paragraph to it.

```
query("#box").append("<p>Peter Parker</p>");
```

Matched element	Matched element after script execution
<pre>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

This script looks for an element with the ID `box`, appends a paragraph to it and colors all text inside the box red.

```
query("#box").append("<p>Peter Parker</p>").css("color","red");
```

Matched element	Matched element after script execution
<pre>&lt;div id="box"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box" style="color: red;"&gt; &lt;h1&gt;Personal information&lt;/h1&gt; &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

Note: the way the functions `append()` and `css()` are used in this script is called 'chaining'. Chaining is optional; the same could be achieved by storing the result of the query in a variable:

```
var box = query("#box");
box.append("<p>Peter Parker</p>");
box.css("color","red");
```

### **attr()**

Returns the value of the specified attribute of the first element in a result set, or sets the value of the specified attribute of each element in a result set.

#### **attr(attributeName) : String**

Returns the value of the specified attribute of the first element in a result set.

#### **attributeName**

*String; the name of the attribute.*

### Examples

This script - with the selector `img` - stores the source of the first image in a variable.

```
var src = results.attr("src");
```

The following script looks up an image with the ID `#image1` and stores its background color in a variable.

```
var imgURL = query("#image1").attr("src");
```

#### **attr(attributeName, value)**

Sets the value of the specified attribute of each element in a result set.

**attributeName**

*String; the name of the attribute.*

**value**

*String; value for the attribute.*

**Examples**

This script looks up an image in an element with the ID #calloutbox and sets its alternative text to a value from a data field

```
var altText = record.fields.FavHobby;
query("#callout img").attr('alt', altText);
```

The following script sets the background color of a specific table cell in an email to red if the value of the field TOTAL has a negative value in the current record.

```
if(record.fields.TOTAL<0) {
    query("#total").attr("bgcolor", "red");
}
```

**before()**

Insert content before each element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). See also: "after()" on page 433.

**before(content)**

Before(content) inserts content before each element in the set of elements that match the script's selector. Before() creates a new result set.

**content**

*String, HTML string or result set to insert after the elements. In case a plain text string is provided, it is automatically wrapped in a <span> element to avoid orphan text nodes to appear in the <body> element.*

**Examples**

This script looks for an element with the ID `salesrep` and inserts a paragraph before that element.

```
results.before("<p>Lorem Ipsum</p>");
```

Selector	Matched element	Matched element after script execution
#salesrep	<p id="salesrep">Peter Parker</p>	<p>Lorem ipsum</p> <p id="salesrep">Peter Parker</p>

This script does the same, but it uses the `query()` function to look up the element.

```
query("#salesrep").before("<p>Lorem ipsum</p>");
```

Matched element	Matched element after script execution
<p id="salesrep">Peter Parker</p>	<p>Lorem ipsum</p> <p id="salesrep">Peter Parker</p>

The following script looks for an element with the ID `salesrep`, inserts a paragraph before that element and colors that element red.

```
query("#salesrep").before("<p>Lorem ipsum</p>").css("color", "red");
```

Matched element	Matched element after script execution
<p id="salesrep">Peter Parker</p>	<p >Lorem ipsum</p> <p id="salesrep" style="color: red;">Peter Parker</p>

Note: the way the functions `before()` and `css()` are used in this script is called 'chaining'. Chaining is optional; the same could be achieved by storing the result of the query in a variable:

```
var salesrep = query("#salesrep");
salesrep.before("<p>Lorem ipsum</p>");
salesrep.css("color", "red");
```

The following script searches the results for the string "ipsum" and puts "Lorem " before it. "Lorem " is automatically wrapped in a `Span` element.

```
results.find("ipsum").before("Lorem ");
```

Matched element	Matched element after script execution
<p>ipsum dolor sit amet, consectetur adipiscing elit.</p>	<p><span>Lorem </span>ipsum dolor sit amet, consectetur adipiscing elit.</p>

The following script looks for an element with the ID salesrep and inserts the text "Lorem Ipsum" before that element. "Lorem Ipsum" is automatically wrapped in a Span element.

```
query("#salesrep").before("Lorem Ipsum");
```

Matched element	Matched element after script execution
<p>ipsum dolor sit amet, consectetur adipiscing elit.</p>	<span>Lorem Ipsum</span> <p id="salesrep">Peter Parker</p>

### children()

Returns the immediate children (inner HTML) of the elements in a result set.

### Examples

This script retrieves the inner HTML of an element selected from a snippet.

```
var snippet = loadhtml('snippets/snippet.html','#foobar').children();
results.append(snippet);
```

The following script retrieves the inner HTML of the elements and then performs a find/replace.

```
var snippet = loadhtml('snippets/snippet.html','#foobar').children();
snippet.find('@firstname@').text('foobar');
results.append(snippet);
```

### clone()

This function returns a new set containing a copy of each element in a set; see "Dynamically adding sections (cloning)" on page 250.

To duplicate an existing template element, clone it before calling `append()`; see "append()" on page 435.

## Examples

This script performs an iteration over the elements in the `results` (the elements that match the selector of the script).

```
var row = query("tbody tr", results).clone();
query("tbody", results).append(row);
```

The following script clones an existing table row to match the number of rows in a detail table. Afterwards it iterates over the rows to populate the fields.

```
// Create the number of rows based on the records in the detail
table
// We start at 1 so the boilerplate row is used too and there is no
need to delete that row
for(var r = 1; r < record.tables['detail'].length; r++) {
results.parent().append(results.clone());
}

// Iterate over the rows and populate them with the data from the
accompanying data row
query("#table_2 > tbody > tr").each(function(i) {
this.find('@ItemNumber@').text( record.tables['detail'][i].fields
["ItemNumber"]);
this.find('@ItemOrdered@').text( record.tables['detail'][i].fields
["ItemOrdered"]);
this.find('@ItemTotal@').text( record.tables['detail'][i].fields
["ItemTotal"]);
this.find('@ItemDesc@').text( record.tables['detail'][i].fields
["ItemDesc"]);
this.find('@nr@').text(i);
});
```

The following script clones and populates a boilerplate row. Once completed you will need to hide the boilerplate row.

## **closest()**

For each element in a set, this function gets the first parent element that matches a selector, by testing the element itself and traversing up through its ancestors in the DOM tree. (In HTML, a parent is an element that contains another element.)

To get a child element or all child elements, use `children()` (see "children()" on page 441).

The `closest()` command is based on the `closest()` command found in the jQuery library: <https://api.jquery.com/closest/>.

## **closest(selector)**

For each element in a set, this function gets the first element that matches the selector by testing the element itself and traversing up through its ancestors in the DOM tree.

### **selector**

*A String containing an HTML tag (without the angle brackets, <>).*

### Examples

The following script looks up all table rows in the template that contain an <input> element.

```
query("input").closest("tr");
```

This code gets the closest 'parent' row for each element that matches the selector of the script (collected in the `results` object):

```
results.closest("tr");
```

The rows could be coloured red within the same statement:

```
results.closest("tr").css('background-color', 'red');
```

## **css()**

Gets the value of a style property for the first element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467), or sets one or more CSS properties for every element in the set.

### **css(styleName) : String**

Returns the value of the specified CSS property.

**propertyName**

*String; the name of the CSS property.*

**Examples**

This script stores the text color of the `results` (the HTML elements that match the selector of the script) in a variable.

```
var textcolor = results.css("color");
```

The following script looks up an element with the ID `#calloutbox` and stores its background color in a variable.

```
var backgroundcolor = query("#calloutbox").css("background-color");
```

**css(styleName, value)**

Function to set a CSS property.

**propertyName**

*String; the name of the CSS property.*

**value**

*String; value for the CSS property or a map of property-value pairs to set.*

**Examples**

This script looks up an element with the ID `#calloutbox` and sets its text color to red.

```
query("#callout p").css('color' , 'red');
```

The following script does the same, but it only sets the text color to red if in the current record the value of the field 'accounttype' is 'PRO'.

```
if(record.fields.accounttype == "PRO") {  
  query("#callout p").css("color","red");  
}
```

This script sets the text color of the results to a hexadecimal color code.

```
results.css('color' , '#669900');
```

This script loads a snippet into a variable. Then it finds/replaces text in the snippet and applies a css property to the replacing text.

```
var mysnippet = loadhtml('snippets/snippet vars.html');
mysnippet.find('@var@').text('OL Connect').css('text-decoration', 'underline');
results.replaceWith(mysnippet);
```

### **css(properties)**

Function to set one or multiple CSS properties.

#### **properties**

*Array; map of property-value pairs to set.*

#### Examples

This script colors the text of the `results` (the set of HTML elements that match the selector of the script) red and makes it bold.

```
results.css({'color' : 'red', 'font-weight' : 'bold'});
```

### **Date, date/time and time functions**

- `date()`
- `dateLong()`
- `dateMedium()`
- `dateShort()`
- `dateTime()`
- `dateTimeLong()`
- `dateTimeMedium()`
- `dateTimeShort()`
- `time()`
- `timeLong()`
- `timeMedium()`
- `timeShort()`

## Note

The locale also influences the output of the different Date functions; see "Locale" on page 201.

### **date(value, pattern)**

Formats a date object using a custom pattern.

#### **value**

*A Date object. A Date can contain a date and time.*

#### **pattern**

*String. The custom pattern may consist of pattern letters, separating symbols and quoted text, for example: "MMMM dd, yyyy"; see "Date and time patterns" on page 449. Note that the repetition of pattern letters determines the exact presentation.*

### **dateLong(value)**

Formats a date as long string representation, for example **April 1, 2016**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **dateMedium(value)**

Formats a date as medium string representation, for example **01/04/16**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **dateShort(value)**

Formats a date as short string representation, for example **1-Apr-2016**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **dateTime(value, pattern)**

Formats a date and time object using a custom pattern.

#### **value**

*A Date object. A Date can contain a date and time.*

#### **pattern**

*String. The custom pattern may consist of pattern letters, separating symbols and quoted text, for example: "yyyy.MM.dd G 'at' HH:mm:ss z"; see "Date and time patterns" on page 449.*

*Note that the repetition of pattern letters determines the exact presentation.*

### **dateTimeLong(value)**

Formats a date and time as long string representation, for example **April 1, 2016 12:00:00 EDT AM**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **dateTimeMedium(value)**

Formats a date and time as medium string representation, for example **1-Apr-2016 12:00:00 AM**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **dateTimeShort(value)**

Formats a date and time as short string representation, for example **01/04/16 12:00 AM**.

#### **value**

*A Date object. A Date can contain a date and time.*

### **time(value, pattern)**

Formats a time using a custom pattern.

#### **value**

*A Date object. A Date can contain a date and time.*

### **pattern**

*String. The custom pattern may consist of pattern letters, separating symbols and quoted text, for example: "'at' HH:mm:ss z"; see "Date and time patterns" on the next page. Note that the repetition of pattern letters determines the exact presentation.*

### **timeLong(value)**

Formats a time as long string representation, for example **12:00:00 EDT AM**.

### **value**

*A Date object. A Date can contain a date and time.*

### **timeMedium(value)**

Formats a time as medium string representation, for example **12:00:00 AM**.

### **value**

*A Date object. A Date can contain a date and time.*

### **timeShort(value)**

Formats a time as short string representation, for example **12:00 AM**.

### **value**

*A Date object. A Date can contain a date and time.*

## **Examples**

### Creating a Date object from a string

When you open a data file or database in the Designer, all fields are text fields (fields of the type `string`). The `formatter` cannot be used to format a string with a particular date format. The solution is to store the string in a variable as a Date object, and use the `formatter` with that variable.

The following sample script demonstrates this solution. It splits a string into parts and then creates a new Date object with the parts in the correct order. To construct a Date, the parts of the date must be put in the following order: year, month, day, and optionally hours, minutes,

seconds, milliseconds (see [http://www.w3schools.com/js/js\\_dates.asp](http://www.w3schools.com/js/js_dates.asp) and [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date).) When the time is omitted, it defaults to 12:00:00 AM.

```
/* Convert the string 21-12-1997 into a valid JavaScript date */
    var strDate = record.fields["date"];
    var dateParts = strDate.split("-");
    var date = new Date(dateParts[2], (dateParts[1] - 1), dateParts
[0]);
```

### Note

JavaScript counts months from 0 to 11. January is 0. December is 11.

Another way to put a string in a Date is to use the `Date.parse` function; see [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date/parse](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date/parse).

The `date` variable can be used as the value in the `date`, `dateTime` or `time` functions of the `formatter`.

```
var myDate = formatter.date(date, "MM/dd/yyyy");
```

The custom pattern that the script provides, outputs the month and day in two digits each and the year in four digits: 05/21/2016. For more examples of formatting patterns, see "Date and time patterns" below.

### Date and time patterns

Dates and times in a template originating from a `date` field in a record set can be displayed using a custom pattern. In the Script Editor, the pattern can be passed to a `date`, `dateTime` or `Time` function of the `formatter`; see "formatter" on page 456.

The custom pattern may consist of pattern letters (see below), for example: "MM/dd/yyyy". The components can be separated with a space or a symbol, e.g. `.`, `/`, `-`. Text must be put in quotes.

The repetition of pattern letters determines the exact presentation. For example, if the number of pattern letters for a month is less than 3 (M or MM), the month is displayed as a number. If the number of pattern letters is 3 (MMM), it will be displayed as text; if available, a short or

abbreviated form of the month's name will be used. If the number of pattern letters is 4 or more (MMMM), the month's full name is displayed.

**Note**

The pattern letters and patterns on this page are only suitable for **displaying** dates and times in templates, not for extracting dates in the DataMapper module.

**Pattern letters**

Letter	Component	Presentation	Examples
G	Era designator	Text	AD
y	Year	Year	1996; 96
Y	Week year	Year	2009; 09
M	Month in year	Month	July; Jul; 07
w	Week in year	Number	27
W	Week in month	Number	2
D	Day in year	Number	189
d	Day in month	Number	10
F	Day of week in month	Number	2
E	Day name in week	Text	Tuesday; Tue
u	Day number of week (1 = Monday, ..., 7 = Sunday)	Number	1
a	Am/pm marker	Text	PM
H	Hour in day (0-23)	Number	0
k	Hour in day (1-24)	Number	24
K	Hour in am/pm (0-11)	Number	0
h	Hour in am/pm (1-12)	Number	12
m	Minute in hour	Number	30
s	Second in minute	Number	55
S	Millisecond	Number	978
z	Time zone	General time zone	Pacific Standard Time; PST; GMT-08:00
Z	Time zone	RFC 822 time	-0800

		zone	
X	Time zone	ISO 8601 time zone	-08; -0800; -08:00

For more information about patterns, see <http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html>.

## Each

A generic iterator function, to iterate over the elements in the result set.

### each(callback)

Iterates over the elements in a set, such as the enumerable properties of an object, in arbitrary order. For each distinct property, statements can be executed.

### callback

*A function. The callback function is passed the iteration index and the current element. In the scope of the callback function, `this` refers to the current element.*

## Examples

The following two scripts demonstrate a simple iteration over the elements in the `results` (the set of HTML elements that match the selector of the script).

This script sets the background color of each of the elements to red. (This is just to demonstrate how this function works. It is easier to change the style of a set of HTML elements using the `css()` function; see "css()" on page 443.)

```
results.each(function(index) {
    results[index].css('background-color', 'red');
});
```

The following script adds a random integer to each element in the result set.

```
results.each(function(index) {
    var test = Math.floor(Math.random() * 10) + 1;
    this.html(test);
});
```

Selector	Matched element	Matched element after script execution
p	<p></p> <p></p> <p></p>	<p>3</p> <p>1</p> <p>7</p>

This script gets the row index (of the current element in the set) and puts it in a paragraph.

```
results.each(function(index) {
    this.text(index);
})
```

Selector	Matched element	Matched element after script execution
p	<p></p> <p></p> <p></p>	<p>0</p> <p>1</p> <p>2</p>

### Using each () in a translation script

The following script first loads a snippet containing translation strings, depending on the value of a field. Then it inserts translations by iterating over elements in the `results` (the set of HTML elements that match the selector of the script) and setting the HTML of each element with a value from the array of translation strings.

```
var strings = loadjson('snippets/' + record.fields.locale +
'.html');
results.each(function(index) {
    if( strings[this.attr('data-translate')] )
        this.html(strings[this.attr('data-translate')]);
});
```

Note: for documentation on the `data-*` attribute, see [http://www.w3schools.com/tags/att\\_global\\_data.asp](http://www.w3schools.com/tags/att_global_data.asp).

Selector	Matched element	Matched element after script execution
p	<pre>&lt;p data-translate="first"&gt;&lt;/p&gt; &lt;p data-translate="last"&gt;&lt;/p&gt; &lt;p data-translate="email"&gt;&lt;/p&gt;</pre>	<pre>&lt;p&gt;primero&lt;/p&gt; &lt;p&gt;último&lt;/p&gt; &lt;p&gt;dirección de correo electrónico&lt;/p&gt;</pre>

### empty()

Remove the contents (child elements and inner HTML) from one element or a set of elements in the template.

Use `remove()` to remove the elements themselves.

### Example

This script empties all Span elements found in the template.

```
results.empty();
```

Selector	Paragraph before script execution	Paragraph after script execution
span	<pre>&lt;p&gt;Lorem ipsum &lt;span&gt;dolor sit&lt;/span&gt; amet, consectetuer adipiscing elit.&lt;/p&gt;</pre>	<pre>&lt;p&gt;Lorem ipsum &lt;span&gt;&lt;/span&gt; amet, consectetuer adipiscing elit.&lt;/p&gt;</pre>

### filter()

#### filter(callback)

Returns a subset of a set. All elements for which the callback function returns `true` will be included in the result.

#### callback

*A function used as a test for each element in the set. Filter() passes the iteration index and the current element to the callback function. In the scope of the callback function, `this` refers to the current element.*

### Example

The selector of the following script is `li` (list item), so the `results` object contains all list items in the template. The script filters the third and sixth line items from the `results`, taking advantage of the index that is passed to the filter function, and colors them red. It uses the modulus operator (%) to select every item with an index value that, when divided by 3, has a remainder of 2. (The index starts counting at zero.)

```
results.filter(function(index) {  
    return index % 3 === 2;  
}).css("background-color", "red");
```

### **filter(selector)**

Returns a subset of a set. All elements matching the selector will be included in the result.

The difference between `results.filter(selector)` and `query(selector, results)` is that `query()` searches throughout the entire `results` while `filter()` only takes the top-level elements into account.

### **selector**

*A String containing a CSS selector. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for CSS selectors and combinations of CSS selectors.*

### Example

The selector of the following script is `tr` (table row), so the object `results` contains all rows in the template. The script filters all even rows from the `results` and colors them red.

```
results.filter(":nth-child(even)").css("background-color", "red");
```

### **find()**

#### **find(textToFind)**

Performs a deep search for `textToFind` in the children of each element, and returns a new result set with elements that surround the occurrences.

#### **textToFind**

*A String that contains the search text.*

### Example

The following piece of code loads a snippet, then looks for placeholders using find(), and replaces them with a text.

```
var mysnipppet = loadhtml('snippets/snippet.html');
mysnipppet.find('@var1@').text('OL Connect 1');
mysnipppet.find('@var2@').html('<i>OL Connect 2</i>').css('text-decoration', 'underline');
results.replaceWith(mysnipppet);
```

### **For...in**

Can be used to iterate over fields in a data set or rows in detail table. Also see <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in>.

**for(variable in object) { ... }**

Iterates over the enumerable properties of an object, in arbitrary order. For each distinct property, statements can be executed.

### Examples

This script iterates over field names in the current record and adds them to a paragraph.

```
for(var i in record.fields){
    results.after("<p>" + i + "</p>");
}
```

Selector	Matched element	Matched element after script execution
#test	<h1 id="test">Fields</h1>	<h1 id="test">Fields</h1> <p>first</p> <p>last</p> <p>email</p>

This script iterates over fields in the current record, retrieving their values. Then it adds the values to a paragraph.

```
for(var i in record.fields){
    results.after("<p>" + record.fields[i] + "</p>");
}
```

Selector	Matched element	Matched element after script execution
#test	<h1 id="test">Fields</h1>	<h1 id="test">Fields</h1> <p>Peter</p> <p>Parker</p> <p>pparker@localhost.com</p>

## formatter

The `formatter` is a global object that allows you to format values in a script.

The Text Script Wizard also allows you to format variable data; see "Using the Text Script Wizard" on page 214 and "Formatting variable data" on page 217.

### Note

The `TextFormatter` object is now deprecated and will eventually be removed.

## Functions

Function	Description
<ul style="list-style-type: none"> <li>• <code>date()</code></li> <li>• <code>dateLong()</code></li> <li>• <code>dateMedium()</code></li> <li>• <code>dateShort()</code></li> <li>• <code>dateTime()</code></li> <li>• <code>dateTimeLong()</code></li> <li>• <code>dateTimeMedium()</code></li> <li>• <code>dateTimeShort()</code></li> <li>• <code>timeLong()</code></li> </ul>	<p>The <code>date()</code>, <code>dateTime()</code> and <code>time()</code> functions allow you to format a date and/or time in different ways. See "Date, date/time and time functions" on page 445.</p>

Function	Description
<ul style="list-style-type: none"> <li>• timeMedium()</li> <li>• timeShort()</li> </ul>	
<ul style="list-style-type: none"> <li>• lowerCase()</li> <li>• upperCase()</li> <li>• properCase()</li> </ul>	<p>The text formatting functions are used on Strings. <code>lowerCase()</code> transform all characters to lowercase, <code>upperCase()</code> transforms all characters to uppercase and <code>properCase()</code> transforms the first character of each word to uppercase and all other characters to lowercase.</p>

## hasClass()

### hasClass(classname) : Boolean

Returns true if the first element in this result set has the specified class.

#### classname

*String containing one class name.*

#### Example

This script checks if the first of the `results` (the set of elements matching the selector of the script) has the class 'green'. If so, it colors the text of all the elements in the `results` green.

```
if (results.hasClass('green')) {
    results.css('color', 'green');
}
```

## hide()

Hides the elements in a set. This doesn't remove the elements; to make them visible again, use the function "show()" on page 475.

These functions are used by the Conditional Script Wizard, as you can see when you open a Conditional Script and click the **Expand** button; see "Showing content conditionally" on page 219.

## Example

This script hides or shows the elements matched by the selector of the script (which are stored in the `results` object), depending on the value of the data field `Country` in the current record.

```
if (record.fields["Country"] == "CANADA") {
  results.show();
} else {
  results.hide();
}
```

## html()

### html() : String

Returns the inner HTML of the first element in this result set.

### html(value)

Replaces the inner HTML of each element in this result set by the supplied value.

### value

*A String that may contain HTML tags.*

## Examples

The following script loads part of a snippet based on the value of a field, and then inserts the content into the document using `html()`.

```
var promoTxt = loadhtml('snippets/promo-en.html', '#' +
record.fields['YOGA']);
results.html(promoTxt);
```

The following script loads a snippet. Then it looks for a placeholder (`@var2@`) in the text of that snippet and replaces every found placeholder by the text `'<i>OL Connect 1</i>'`. It uses `html()` so the HTML formatting (`<i>` and `</i>`) will indeed be interpreted as HTML. Finally, it places the snippet in the template.

```
var mysnippet = loadhtml('snippets/snippet.html');
mysnippet.find('@var1@').html('<i>OL Connect 1</i>');
results.replaceWith(mysnippet);
```

## loadhtml()

Global function that replaces the content (inner html) of each matched element in the result set, alternatively load the data into a variable. The location should be an URL or a relative file path.

### Note

Loadhtml() is cached per batch run (based on the URL) in print/email.

## loadhtml(location)

Loads all HTML from the specified HTML file.

### location

*String containing a path that can be absolute or relative to the section/context. Use: snippets/<snippet-name> to retrieve the content from a HTML file residing in the Snippets folder on the Resources panel.*

## Examples

This script loads a local HTML snippet (from the Resources panel) directly into the matched elements

```
results.loadhtml("snippets/snippet.html");
```

The following script loads a local HTML snippet (Resources panel) into a variable. The `replaceWith()` command is used to replace the element(s) matched by the script's selector with the contents of the snippet.

```
var mysnipppet = loadhtml('snippets/snippet.html');  
results.replaceWith(mysnipppet);
```

Same result as the previous script, but a different notation:

```
results.replaceWith(loadhtml('snippets/snippet.html'));
```

The following script loads a snippet into a variable and finds/replaces text in the variable before inserting the content into the page. The second `find` command also adds formatting to the replacing text.

```
var mysnipppet = loadhtml('snippets/snippet.html');  
mysnipppet.find('@var1@').text('OL Connect 1');
```

```
mysnippet.find('@var2@').html('<i>OL Connect 2</i>').css('text-decoration', 'underline');
results.replaceWith(mysnippet);
```

This last script loads a snippet into a variable and retrieves an element from the snippet using `query()`.

```
var mysnippet = loadhtml('snippets/text-root-wrapped.html');
var subject = query("#subject", mysnippet).text();
results.append("<p style='font-weight: bold;'>" + subject +
"</p>");
```

### **loadhtml(location, selector)**

Retrieves specific content from the specified HTML file.

#### **location**

*String; the location can be absolute or relative to the section/context. Use: snippets/<snippet-name> to retrieve the content from a HTML file residing in snippets folder of the Resources panel.*

#### **selector**

*String. The supplied selector should conform to CSS selector syntax and allows you to retrieve only the content of matching elements.*

### Examples

This script loads a specific element from the snippet.

```
var mysnippet = loadhtml('snippets/snippet-selectors.html', '#item3');
results.replaceWith(mysnippet);
```

This script loads the children of the selected element.

```
var snippet = loadhtml('snippets/snippet.html', 'foobar').children();
results.replaceWith(snippet);
```

Another example is given in the following how-to: [Using a selector to load part of a snippet.](#)

## loadjson()

Creates a JSON object based on the text retrieved from the supplied location. The function lets you retrieve content from an JSON enabled server using a standard HTTP request. Popular content management systems, like WordPress (requires JSON API plug-in) and Drupal provide a JSON service/API to retrieve content.

### Note

Loadjson() is cached per batch run (based on the URL) in print/email.

This online JSON viewer is handy to debug JSON data: <http://jsonviewer.stack.hu>

## loadjson(location)

Loads json data from a remote location.

### location

*String; the supplied location should be either a URL or a relative file path.*

### Examples

This sample script retrieves JSON data from a snippet.

```
var localJSON = loadjson('snippets/jsonsnippet.html');
if(localJSON.post){
    results.html("<h3>" + localJSON.post.title + "</h3><p>" +
localJSON.post.modified + "</p>");
}
```

This script retrieves a post from a WordPress site.

```
var wpPost = loadjson('http://192.168.101.58/2013/06/leave-the-
third-dimension-behind-and-focus-on-real-printing-
innovation/?json=1');
if(wpPost.post){
    results.html("<h1>" + wpPost.post.title + "</h1>"
+ wpPost.post.content);
}
```

This script retrieves multiple posts from a WordPress site.

```

var numPosts = 3;
var wpPost = '';
var wpRecentPosts = loadjson('http://192.168.101.58/?json=get_
recent_posts&count=' + numPosts);
if(wpRecentPosts.posts){
    for (var i = 0; i < numPosts ; i++) {
        wpPost += "<p>" + wpRecentPosts.posts[i].title + "</p>";
    }
}
results.after(wpPost)

```

## logger

This is a global `ScriptLogger` object that allows logging messages such as error, warning or informational messages. The messages will appear in the **Messages** pane (see "Problems and messages" on page 332 and "Designer User Interface" on page 254).

## Methods

These are the methods of the logger object.

Method	Parameters	Description
<code>error()</code>	<code>message: string</code>	Logs an error message
<code>info()</code>	<code>message: string</code>	Logs an informational message
<code>warn()</code>	<code>message: string</code>	Logs a warning message

## pageRef()

Returns a **marker** that will be replaced with the element's page number after pagination. This only works for elements in the section that is currently being merged.

## Example

### Creating a table of contents

The following script creates a table of contents for all level 1 headings (<h1> elements) with the class `title` in one section.

```
var toc = '<ul ID="toc">';
query('h1.title').each(function()
{toc += '<li>' + this.text() + ' <span class="li_toc">' +
this.pagerf() + '</span></li>';
});
toc += '</ul>';
results.after(toc);
```

The first line creates a variable for the table of contents, which will be a list (a <ul> element with the ID `toc`). The start tag of the list is added to the variable.

The next line does a query for all level 1 headings (<h1> elements) with the class `title` in the current section. With `each()` the script loops through them. For each of the headings it adds a line item to the list, with the text (`this.text()`) and the page reference of the respective heading.

After the loop, the end tag of the list is added to the variable.

Finally, the script adds the variable - that now contains the table of contents - after the `results`. The `results` object contains the elements that match the selector of the script. So, if the script's selector selects the title of the table of contents, the table of contents will be added after that.

The following style rules, added to the style sheet, will align the chapter titles to the left and the page numbers to the right:

```
#toc li {
text-align:left;
}
#toc span {
float: right;
}
```

Note that these styles use the list's ID, that was defined in the first line of code. For information about style sheets, see "Styling templates with CSS files" on page 171.

## parent()

Returns the parents of the elements in a set. (In HTML, a parent is an element that contains another element.)

To get an ancestor that matches a particular selector, use `closest()` (see "closest()" on page 443).

### Example

Assume that there are three paragraphs in a Box and that one of those paragraphs matches the selector of this script. The paragraph is stored in the `results` object (see "results" on page 472). The script retrieves the Box (which is the parent of the paragraph) using `results.parent()`, and then changes its background color to red.

```
results.parent().css('background-color' , 'red');
```

## prepend()

Insert content at the beginning of each element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). See also: "append()" on page 435.

## prepend(content)

Insert content as the first element to each element in the set of HTML elements that match the selector of the script or of another query in the template (see "query()" on page 467). `Append` creates a new result set.

## content

*HTML string, string or HTML string to insert after the matched elements. In case a plain text string is provided, it is automatically wrapped in a `<span>` element to avoid orphan text nodes to appear in the `<body>` element.*

### Examples

This script inserts a heading as the first element in an element that has the ID `#box`.

```
results.prepend("<h1>Personal information</h1>");
```

Selector	Matched element	Matched element after script execution
#box	<div id="box"> <p>Peter Parker</p> </div>	<div id="box"> <h1>Personal information</h1> <p>Peter Parker</p> </div>

This script inserts a heading as the first element in an element that has the class `name`.

```
results.prepend("<b>Name: </b>");
```

Selector	Matched element	Matched element after script execution
.name	<div> <h1>Personal information</h1> <p class="name">Peter Parker</p> </div>	<div> <h1>Personal information</h1> <p class="name"><b>Name: </b>Peter Parker</p> </div>

This script inserts content in multiple `<div>` elements at the same time.

```
results.prepend("<h1>Personal information</h1>");
```

Selector	Matched element	Matched element after script execution
div	<div id="box"> <p>Peter Parker</p> </div> <div id="box"> <p>Peter Parker</p> </div>	<div id="box"> <h1>Personal information</h1> <p>Peter Parker</p> </div> <div id="box"> <h1>Personal information</h1> <p>Peter Parker</p> </div>

This script prepends a snippet that contains the text "`<h1>Personal information</h1>`".

```
var a = loadhtml('snippets/snippet.html');
results.prepend(a);
```

Selector	Matched element	Matched element after script execution
div	<pre>&lt;div id="box"&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box"&gt;   &lt;h1&gt;Personal information&lt;/h1&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

This script uses the function `query()` to find a box. Then it inserts a heading as the first element in that box.

```
query("#box").prepend("<h1>Personal information</h1>");
```

Matched element	Matched element after script execution
<pre>&lt;div id="box"&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box"&gt;   &lt;h1&gt;Personal information&lt;/h1&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

This script uses the function `query()` to find a box, prepends a heading and sets the text color of the entire box to red.

```
query("#box").prepend("<h1>Personal information</h1>").css(
  "color", "red");
```

Matched element	Matched element after script execution
<pre>&lt;div id="box"&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>	<pre>&lt;div id="box" style="color: red;"&gt;   &lt;h1&gt;Personal information&lt;/h1&gt;   &lt;p&gt;Peter Parker&lt;/p&gt; &lt;/div&gt;</pre>

Note: the way the functions `prepend()` and `css()` are used in this script is called 'chaining'. Chaining is optional; the same could be achieved by storing the result of the query in a variable:

```
var box = query("#box");  
box.prepend("<p>Peter Parker</p>");  
box.css("color", "red");
```

## query()

This function creates a new result set, containing the HTML elements that match the supplied CSS selector. The context (optional) allows you to restrict the search to descendants of one or more context elements.

- query(selector)
- query(selector, context)

The new result set is of the type `QueryResults`. All functions that can be used with the `results` object can also be used with this result set; see "results" on page 472.

## query(selector)

Creates a new result set containing the HTML elements in the template that match the supplied CSS selector.

### selector

*A String containing a CSS selector. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for CSS selectors and combinations of CSS selectors.*

## query(selector, context)

Creates a new result set containing the HTML elements that match the supplied CSS selector. The context (optional) allows you to restrict the search to descendants of one or more context elements.

### selector

*A String containing a CSS selector. See [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) for CSS selectors and combinations of CSS selectors.*

### context

*A result set or an HTML string. If the passed context is not a result set of HTML string it will be coerced to a String and interpreted as HTML.*

## Examples

### Look for an element with a certain ID

This script applies a style rule to the queried elements.

```
query("#test1").css("color", "yellow");
```

Matched element;	Matched element after script execution
<p id="test1">foo</p>	<p id="test1" style="color: yellow;">foo</p>

### Look for an element in a snippet

The following script loads a snippet. Then it looks up an element in a snippet and sets its text. Finally, it replaces the elements matched by the script's selector by the snippet.

```
var snippet = loadhtml('snippets/mysnippet.html');  
query("#foo", snippet).text("bar");  
results.replaceWith(snippet);
```

## record

The `record` object gives access to the record that is currently being merged with the template.

## Properties

Field	Type	Description
<b>fields</b>	Array	The field values that belong to this record. You can access a specific field value using either a numeric index or the field name: <code>record.fields['fieldname']</code> or <code>record.fields.fieldname</code> .
<b>id</b>	Number	The id of this record.
<b>index</b>	Number	The one-based index of this record, or zero if no data is available.
<b>tables</b>	Array	The detail tables that belong to this record. You can access a specific table using either a numeric index or the table name.

## Examples

The following template script evaluates the data field `Country` in the current `record`. If the value is 'CANADA' it will show the `results`, otherwise it will hide them. (The `results` object contains the elements that match the script's selector; see "results" on page 472 and "Writing your own scripts" on page 223.)

```
if (record.fields["Country"] == "CANADA") {
    results.show();
} else {
    results.hide();
}
```

In a Control Script, an entire section could be enabled or disabled based on the same condition:

```
if (record.fields["Country"] == "CANADA") {
    merge.template.contexts.PRINT.sections["Section 1"].enabled =
true;
} else {
    merge.template.contexts.PRINT.sections["Section 1"].enabled =
false;
}
```

For more information about Control Scripts, see "Control Scripts" on page 240.

## **remove()**

Removes each element in a set from the DOM.

This function returns a new result set containing each removed element. These can be changed and inserted in the document. This could be beneficial in terms of performance, as manipulating elements inside the DOM is relatively time consuming.

## Examples

This script removes all Span elements found in the template.

```
results.remove();
```

Selector	Paragraph before script execution	Paragraph after script execution
span	<p>Lorem ipsum <span>dolor sit</span> amet, consectetuer adipiscing elit.</p>	<p>Lorem ipsum amet, consectetuer adipiscing elit.</p>

The selector of the following sample script is `tbody`. Before this script runs, the table body consists of a single placeholder row with three cells. After running the script, it contains thirty rows. To improve performance, most of the DOM manipulation takes place on detached elements.

```
// Detach the placeholder row from the DOM
var row = query("tr", results).remove();

// Modify the cells of this row
var cells = row.children();
cells[0].html("some text").css("background-color", "yellow");
cells[1].html("some text").css("font-weight", "bold");
cells[2].html("some text");

// Create a number of copies
var rows = row.clone();
for (var i = 0; i < 30; i++) {
    rows = rows.add(row.clone());
}

// Attach all copies to the DOM as children of tbody
results.append(rows);
```

### **removeAttr()**

Removes the specified HTML attribute from an element or from each element in a set of elements. To add or change an attribute, use `attr()` (see "attr()" on page 438).

## **removeAttr(attributeName)**

### **attributeName**

*String; the name of the attribute.*

### Examples

This script looks up an email field in a form (which is an <input> with the ID #email1) and removes its `readonly` attribute.

```
query("#email1").removeAttr('readonly');
```

## **removeClass()**

Removes the specified class from each element in this result set. Has no effect if the class is not present.

## **removeClass(classname)**

### **classname**

*String, space separated list of class names.*

### Examples

This script removes the class name "foo" from all elements in the results that have this class.

```
results.addClass("foo");
```

Selector	Matched element	Matched element after script execution
p	<p class="foo">Hello world</p>	<p>Hello world</p>

## **replaceWith()**

Replaces each element in a set of HTML elements.

## **replaceWith(content)**

Replaces each element in a set of HTML elements. Returns the result set.

### **content**

*A query result. This can be an HTML string or a result set.*

## Examples

### Replace elements with a snippet

The following script loads a snippet and then replaces the elements matched by the script's selector with the snippet.

```
var snippet = loadhtml('snippets/mysnippet.html');
results.replaceWith(snippet);
```

### Replace elements with a set of snippets

The following script loads snippets and adds their elements to a new, empty result set (using `query()`). Then it replaces a placeholder in the template with the set of snippets.

```
var chapters = query();
for ( var i = 1; i <= 4; i++) {
chapters = chapters.add(loadhtml('snippets/Chapter' + i +
'.html'));
}
results.replaceWith(chapters);
```

## results

The `results` object (type: `QueryResults`) contains the HTML element or set of HTML elements matched by the selector of the script. The selector of a script can be specified in the Script Editor and is visible in the second column of the Scripts pane, next to the name of the script.

If, for example, a script would have the selector `p.onlyCanada`, the script would apply to all paragraphs that have the class `onlyCanada`. (Classes can be defined in the **Attributes** pane at the right: select the element in the content and type the class(es) in the **Class** field.)

The script could then use the `results` object to hide or show those paragraphs, depending on the value of the data field `Country` in the current record:

```
if (record.fields["Country"] == "CANADA") {
results.show();
} else {
results.hide();
}
```

### Note

This object can't be used in Control Scripts, because they don't have a selector.

## Property

Field	Type	Description
<b>length</b>	Number	Number of elements in this result set. Equivalent to calling size().

## Functions

The functions below can be called by the `results` object or by the result set that is returned by a query, see "query()" on page 467.

Function	Description
"add()" on page 431	Adds elements to a set of HTML elements.
"addClass()" on page 432	Adds the specified class to each element in a set of HTML elements. Has no effect if the class is already present.
"after()" on page 433	Inserts content after each element in a set of HTML elements..
"append()" on page 435	Inserts content at the end of each element in a set of HTML elements.
"attr()" on page 438	Change the given attribute of the element or set of HTML elements with the given value.
"before()" on page 439	Inserts content before an element or before each element in a set of HTML elements.
"css()" on	Gets the value of a style property for the first element in set of HTML

Function	Description
page 443	elements or sets one or more CSS properties for every element in a set of HTML elements.
"children()" on page 441	Returns the immediate children of an HTML element.
"clone()" on page 441	Returns a new result set containing a copy of each element in a set of HTML elements.
"filter()" on page 453	Returns a subset of the current result set.
"find()" on page 454	Performs a search for a text in the children of each element in a set of HTML elements, and returns a new result set with elements that surround the occurrences.
"hasClass()" on page 457	Returns <code>true</code> if the first element in this result set has the specified class.
"hide()" on page 457	Hides the HTML element or set of HTML elements.
"html()" on page 458	Replaces the inner HTML of the element or of each element in a set of HTML elements with the supplied value, or returns the HTML of the first element if no value is supplied.
is(selector)	Returns true if at least one of the elements in a set of HTML elements matches the supplied CSS selector.
"pageRef()" on page 462	Returns a <b>marker</b> that will be replaced with the element's page number after pagination. This only works for elements in the section that is currently being merged.
"parent()" on page 464	Returns the parents of the elements in a set of HTML elements.

Function	Description
"prepend()" on page 464	Inserts content at the beginning of an HTML element or of each element in a set of HTML elements.
"remove()" on page 469	Removes an HTML element or a set of HTML elements from the document.
"removeAttr()" on page 470	Removes the specified attribute from each element in this result set.
"removeClass()" on page 471	Removes the specified class from an element or from each element in a set of HTML elements. Has no effect if the class is not present.
"replaceWith()" on page 471	Replaces an HTML element or a set of HTML elements (with a snippet, for example). Returns the result set.
"show()" below	Shows the HTML element or a set of HTML elements.
size()	Gets the number of elements in this result set. Equivalent to the <code>length</code> property.
"text()" on the facing page	Replaces the text content of an HTML element or of each element in a set of HTML elements with the supplied value, or returns the text content of the first element if no value is supplied.

### show()

Shows the elements in a set. To hide elements (again), use the function "hide()" on page 457.

These functions are used by the Conditional Script Wizard, as you can see when you open a Conditional Script and click the **Expand** button; see "Showing content conditionally" on page 219.

## Example

This script hides or shows the elements matched by the selector of the script (which are stored in the `results` object), depending on the value of the data field `Country` in the current record.

```
if (record.fields["Country"] == "CANADA") {
  results.show();
} else {
  results.hide();
}
```

## text()

### text() : String

Returns the text content of the first element in a result set.

## Example

This script loads a snippet into a variable and retrieves an element from the snippet using `query()` and `text()`.

```
var mysnippet = loadhtml('snippets/text-root-wrapped.html');
var subject = query("#subject", mysnippet).text();
results.append("<p style='font-weight: bold;'>" + subject +
"</p>");
```

## text(value)

Replaces the text content of each element in a result set by the supplied value.

## Example

This script loads a snippet, then looks for placeholders using `find()`, and replaces them using `text(value)`.

```
var mysnippet = loadhtml('snippets/snippet.html');
mysnippet.find('@var1@').text('OL Connect 1');
mysnippet.find('@var2@').html('<i>OL Connect 2</i>').css('text-
decoration', 'underline');
results.replaceWith(mysnippet);
```

# Control Script API

The table below lists the objects that are the most important in Control Scripts. Click through to the object to find a description and sample scripts.

See "Control Scripts" on page 240 for information about this kind of scripts, how to insert them and what you can do with them.

Object	Usage
"section" on page 480	Much of the Control Script magic is performed by setting one of the fields of the <code>section</code> object. Via the <code>section</code> object you can omit, select and clone sections; add a background to a Print section; add a header to an email; etc. A section can be retrieved via the context that it belongs to, using <code>merge.template.contexts.ContextType.sections["section name"]</code> . For example: <code>merge.template.contexts.PRINT.sections["Section EN"]</code> .
"context" on the facing page	Object that contains one context and its sections. It is accessed through the <code>template</code> object: <code>merge.template.contexts</code> . To get access to one context, you have to specify the <code>ContextType</code> (see "ContextType" on page 490), for example: <code>var printContext = merge.template.contexts.PRINT;</code> Through the <code>merge</code> object you can find out which context is currently being merged: <code>merge.context</code> .
"template" on page 487	The <code>template</code> object contains all contexts and sections. It is accessed through the <code>merge</code> object: <code>merge.template</code> .
"merge" on page 479	The <code>merge</code> object gives access to the template with all of its contexts and sections .
channel (see "Channel" on page 489)	The channel for which output is generated. This is registered in the <code>merge</code> object: <code>merge.channel</code> . Note that the channel doesn't change when the output consists of different contexts. When generating email, for example, the channel is EMAIL, even when merging the Print context to attach it to the email.
"record" on	The current record in the main data set. To get the value of a field in the

Object	Usage
page 468	record, use <code>record.fields['fieldname']</code> or <code>record.fields.fieldname</code> .

## Other objects that are available to Control Scripts

The list above isn't exhaustive: most of the objects listed in the Designer API (see "Designer Scripts API" on page 429) are also available in Control Scripts. Not all of those objects can be used in Control Scripts, however. This is because Control Scripts differ from template scripts in two ways:

- Control Scripts don't have a selector, like template scripts do. A selector selects parts of the content of a section and stores them in the `results` object, so that they can be modified in the script. As Control Scripts don't have a selector, the `results` object can't be used there. Control Scripts don't touch the content - meaning, the text flow - of the sections.
- Control Scripts run before all other scripts. When a template consists of several contexts, and these contexts are combined in the output - for example, when an Email is generated with the Print context as attachment - all scripts run once for each context, but Control Scripts always go first.

## context

In a Control Script, the `context` object represents one context in the template.

Which contexts are available in the template can be queried using `merge.template.contexts`.

The context being merged can be queried using `merge.context`.

Field	Type	Description
<b>sections</b>	Array	<p>Array of sections (see "section" on page 480) inside a particular context defined in the template.</p> <p><b>Note:</b> When using <code>merge.context.sections</code> keep in mind that for example 'Section X' might only exist in your Print context, so using <code>merge.context.sections['Section X']</code> without enclosing it in the <code>if</code> statement <code>if (merge.context.type ==</code></p>

Field	Type	Description
		<code>ContextType.PRINT) {}</code> will yield an error when the script runs for other contexts. Alternatively, use the <code>template</code> object to access a specific context: <code>merge.template.contexts.PRINT.sections['Section X']</code> .
<b>type</b>	ContextType	The context type: PRINT, EMAIL or WEB (see "ContextType" on page 490).

### Example

This script checks if the output channel is EMAIL and if the context to be merged is the Print context (which happens if the Print context is attached to an email). If this is the case, it includes and excludes certain Print sections from the output.

```
if (channel == Channel.EMAIL) {
    if (merge.context.type == ContextType.PRINT) {
        merge.context.sections['Section 1'].enabled = false;
        merge.context.sections['Section 2'].enabled = false;
        merge.context.sections['Section 3'].enabled = true;
    }
}
```

### merge

In Control Scripts, the root level instance of the object `merge` is the entry point from where you can query and change the way contexts are merged. It gives access to the template with all its contexts and sections.

For sample scripts, follow the links to the respective objects.

For more information about Control Scripts, see "Control Scripts" on page 240 and "Control Script API" on page 477.

Field	Type	Description
<b>channel</b>	"Channel" on page 489	The final output channel: EMAIL, PRINT or WEB. The channel doesn't change when the output consists of different contexts. When generating an email, for example, the channel is EMAIL,

Field	Type	Description
		even when merging the Print context to attach it to the email.
<b>"context"</b> on page 478	Context	The context rendered by this merge run. If for one record, different contexts need to be output (for example, when the Print context is attached to an email) a record is merged multiple times: once per context. Per merge run, <code>merge.context</code> shows with which context the record is merged.
<b>"section"</b> below	Section	In template scripts, this object defines the section that is being merged. <b>Note!</b> In Control Scripts, <code>merge.section</code> is <b>only</b> available when the output channel is WEB. To make sure that it is defined, use the following statement: <code>if (merge.channel == Channel.WEB &amp;&amp; merge.context.type == ContextType.WEB) { ... }</code> . To retrieve any section in a Control Script, use: <code>merge.template.contexts.ContextType.Section ['Section name'];</code> (for example: <code>merge.template.contexts.PRINT.sections ["Section EN"]</code> ).
<b>"template"</b> on page 487	Template	This object contains the template and all of its contexts. It can be used to find out which contexts are available in the template, using <code>merge.template.contexts</code> (see "context" on page 478) and to manipulate the sections in those contexts (see "section" below).

## section

The `section` object can be used to query and modify how the section (and the related context) will be outputted. It is one of the most important objects in Control Scripts (see "Control Scripts" on page 240 and "Control Script API" on page 477).

## Retrieving a section

A section can be retrieved using `merge.template.contexts.ContextType.sections["section name"]`, for example: `merge.template.contexts.PRINT.sections["Section EN"]`.

A section can also be retrieved via `merge.context.sections['section name']`. Remember, however, that when several contexts need to be merged (for example, when the Print context is attached to an email), the script needs to check if the current context is of the type that contains the desired section (for example: `if (merge.context.type == ContextType.PRINT) {}`). When sections in different contexts have the same name, it is safer to use `merge.template.contexts.ContextType.sections["section name"]`.

## Fields

Field	Type	Description
<b>background</b>	String	<b>Print sections only.</b> Used to set a PDF background on a Print section. See "Control Script: Setting a Print section's background" on page 248 and "BackgroundResource" on page 488.
<b>enabled</b>	boolean	Enables or disables this section for output (see "Examples" on page 484). Note that even if a section is disabled, the <code>part</code> and <code>restartPageNumber</code> fields are still effective to define the parts division and page numbering over multiple sections when applicable.  The default enabled state for sections (before any control script runs) is as follows: For <b>Web</b> channel requests, the requested web section is enabled by default. It is possible to redirect to another section by disabling the requested section and enabling another section. For <b>Email</b> channel requests on the <b>Web</b> context, only the default section is enabled by default. It is possible to enable different or multiple sections, to control which sections will be attached to the email. For <b>Email</b> channel requests on the <b>Print</b> context <b>all</b>

Field	Type	Description
		<p>Print sections are enabled by default. It is possible to enable different or multiple sections to control which sections will be attached to the email.</p> <p>For <b>Print</b> channel requests on the <b>Print</b> context <b>all</b> sections are enabled by default.</p>
<b>headers</b>	String	<b>Email sections only.</b> Used to set custom email headers. For examples, see "Adding custom ESP handling instructions" on page 515.
<b>name</b>	String	<p>Used to get or set the name of the section. Note that section names must be unique and that sections cannot have an integer as its name. The name should always include alphanumeric characters.</p> <p>To rename email attachments, use the field <code>part</code>.</p>
<b>ownerPassword</b>	String	<b>Print sections only.</b> Used to set the owner password for a PDF attachment.* Setting only the owner password creates a secured PDF that can be freely viewed, but cannot be manipulated unless the owner password is provided. (Note that the recipient needs Adobe Acrobat to do this, because the Acrobat Reader does not allow users to enter the owner password.) See "Control Script: Securing PDF attachments" on page 252.
<b>part</b>	String	Name for the part. <code>part</code> is used to specify where a new part starts and the title for the part. This is used to split Email attachments. The Email output can, for example, attach 3 PDFs generated from the Print context. The part name will be used as the file name for the attachment. See "Parts: splitting and renaming email attachments" on page 246.
<b>password</b>	String	<b>Print sections only.</b> Used to set the user password and owner password for a PDF attachment to the same

Field	Type	Description
		value. See "Control Script: Securing PDF attachments" on page 252.*
<b>restartPageNumber</b>	boolean	<p><b>Print sections only.</b> Enables or disables a restart of the page numbering. When generating Print output this can be used to let page numbering continue over multiple sections.</p> <p>The default value is <code>false</code>, meaning that each section will start with page 1 (to emulate behavior of previous versions).</p>

\*The password(s) should be set on the first Print section when producing a single attachment, or on the first section of each part when producing multiple attachments. Each of the parts (attachments) may have a different (or no) set of passwords. Passwords set in the Control Script override the password set through the Email PDF password script (see "Email PDF password" on page 101). This allows you to change or remove the password from a specific part. Removal is done by setting the `password` field to `null` or "" (empty string).

## Functions

### Note

For cloned sections, functions are not available.

Function	Description
<b>clone()</b>	Clone this section. See "Dynamically adding sections (cloning)" on page 250.
<b>addAfter()</b>	Add a cloned section after this section.
<b>addBefore()</b>	Add a cloned section before this section.

## Examples

### Conditionally skipping or printing Print sections

This script disables all Print sections and then re-enables one of them, depending on a value in the current record.

```
var printSections = merge.template.contexts.PRINT.sections;
printSections['Section EN'].enabled = false;
printSections['Section FR'].enabled = false;

if(record.fields.Language === 'FR'){
    printSections['Section FR'].enabled = true;
} else {
    printSections['Section EN'].enabled = true;
}
```

### Selecting different sections for Print output and Email PDF attachment

This script selects a different Print section for output, depending on the output channel (Email or Print).

```
var printSections = merge.template.contexts.PRINT.sections;

if(merge.channel === Channel.EMAIL){
    printSections['Section 1'].enabled = false;
    printSections['Section 2'].enabled = true;
}

if(merge.channel === Channel.PRINT){
    printSections['Section 1'].enabled = true;
    printSections['Section 2'].enabled = false;
}
```

### Setting the name of Email PDF attachments

This script renames the file name of an attachment by setting the part name of a section (see "Parts: splitting and renaming email attachments" on page 246).

```
var section = merge.template.contexts.PRINT.sections['Section 1'];
section.part = 'Invoice ' + record.fields['InvoiceNo'];
```

### Controlling multiple Email attachments

The following script attaches the following sections to an email:

- Print section 3 + 4 as attachment with continued page numbers
- Print section 6 as separate attachment (also see "Parts: splitting and renaming email attachments" on page 246)
- Web sections A and B as separate attachment

```

if (channel == Channel.EMAIL) { // only when generating Email
output
if (merge.context.type == ContextType.PRINT) {
    merge.context.sections['Section 1'].enabled = false;
    merge.context.sections['Section 2'].enabled = false;
    merge.context.sections['Section 3'].enabled = true;
    merge.context.sections['Section 3'].part = "PDFAttach1";
    merge.context.sections['Section 4'].enabled = true;
    merge.context.sections['Section 4'].restartPageNumber = false;
    merge.context.sections['Section 5'].enabled = false;
    merge.context.sections['Section 6'].enabled = true;
    merge.context.sections['Section 6'].part = "PDFAttach2";
} else if (merge.context.type == ContextType.WEB) {
    merge.context.sections['default Section'].enabled = false; //
disable whatever is the default section
    merge.context.sections['Section A'].enabled = true;
    merge.context.sections['Section A'].part = "WebPartA";
    merge.context.sections['Section B'].enabled = true;
    merge.context.sections['Section B'].part = "WebPartB";
}
}

```

#### Note

For another example, see this how-to: [Output sections conditionally](#).

#### Note

If the Email PDF Password Script Wizard defines a password, and a template has a Control Script that creates multiple PDF attachments, all the attachments are secured by the same password by default. Using a Control Script, you can set set different passwords for attachments; see "Control Script: Securing PDF attachments" on page 252.

## Positioning the background of a Print section

These scripts both set the background of a Print section to the same PDF, but they position it differently.

### **Using absolute positioning**

```
var activeSection = merge.template.contexts.PRINT.sections['Section 1'];
activeSection.background.source = BackgroundResource.RESOURCE_PDF;
activeSection.background.position = MediaPosition.ABSOLUTE;
activeSection.background.left = "10mm";
activeSection.background.top = "10mm";
activeSection.background.url = "images/somepage.pdf";
```

### **Scaling to Media size**

```
var activeSection = merge.template.contexts.PRINT.sections['Section 1'];
activeSection.background.source = BackgroundResource.RESOURCE_PDF;
activeSection.background.position = MediaPosition.FIT_TO_MEDIA;
activeSection.background.url = "images/somepage.pdf";
```

See also: "BackgroundResource" on page 488, "MediaPosition" on page 491 and "Control Script: Setting a Print section's background" on page 248.

## Cloning Print sections

For background information on cloning Print sections, see: "Dynamically adding sections (cloning)" on page 250.

### **Cloning a section based on the number of records in a detail table**

This script creates as many clones of a section as there are records in a detail table. It assigns the new sections a unique name.

```
var printSections = merge.template.contexts.PRINT.sections;
var numClones = record.tables['detail'].length;
for( var i = 0; i < numClones; i++){
    var clone = printSections["Section 1"].clone();
    clone.name = "my_section_clone_" + i;
    printSections["Section 1"].addAfter(clone);
}
```

## Cloning a section based on data and assign a background PDF

This script clones a section based on data fields. It disables the source section first and then calls the `addPolicy` function. `addPolicy` clones the section, renames it and sets a PDF from the resources as its background. It explicitly enables the clone and then adds it to the Print context.

```
var printSections = merge.template.contexts.PRINT.sections;
merge.template.contexts.PRINT.sections["Policy"].enabled = false;
if(record.fields.policy_a == 1) {
    addPolicy('a');
}
if(record.fields.policy_b == 1) {
    addPolicy('b');
}
function addPolicy(policy){
    var resourceUrl = 'images/policy-' + policy + '.pdf';
    var clone = printSections["Policy"].clone();
    clone.name = "policy_" + policy;
    clone.background.url = resourceUrl;
    clone.enabled = true;
    printSections["Policy"].addAfter(clone);
}
```

### template

The `template` object represents the template with all its contexts and sections. It is used frequently in Control Scripts (see "Control Scripts" on page 240 and "Control Script API" on page 477).

It is retrieved via the merge object: `merge.template` (see "merge" on page 479).

Which contexts are available in the template can be queried using `merge.template.contexts`. To get access to a specific context, you have to specify the `ContextType` (see "ContextType" on page 490).

Field	Type	Description
<b>contexts</b>	Array	Array of contexts (see "context" on page 478) available in the template. The contexts contain the sections (see "section" on page 480).

## Example

The following Control Script retrieves two Print sections. Then, depending on a value in the current record, it enables one section or the other, so that only one of the two sections appears in the output.

```
var printSections = merge.template.contexts.PRINT.sections;
printSections['Section EN'].enabled = false;
printSections['Section FR'].enabled = false;

if(record.fields.Language === 'FR'){
    printSections['Section FR'].enabled = true;
} else {
    printSections['Section EN'].enabled = true;
}
```

## BackgroundResource

BackgroundResource is an enumeration for the types of background resources for a Print section (see "Control Script: Setting a Print section's background" on page 248 and the topic about the `section` object: "section" on page 480).

A Print section can be retrieved in script using

```
merge.template.contexts.ContextType.sections["section name"], for example
merge.template.contexts.PRINT.sections["Section EN"].
```

Field	Description
<b>DATAMAPPER_PDF</b>	<p>A PDF file retrieved via the active Data Mapping Configuration. This can be the PDF file that was used as input file, or another type of input file, converted to PDF.</p> <div data-bbox="462 1522 1453 1795"><p><b>Note</b></p><p><b>From DataMapper input</b> cannot be used in PrintShopMail Connect. The DataMapper is included only in PlanetPress Connect and PreS Connect.</p></div>

Field	Description
<b>NONE</b>	No PDF background.
<b>RESOURCE_PDF</b>	A PDF file stored in the template or on the network. Note that it isn't possible to use a remotely stored PDF file as a section's background.

## Example

The following script sets the background for a section called 'Policy' to `RESOURCE_PDF` and specifies a path for it, using a data value:

```
// Enable the section background and specify that the PDF should be
read
// from a resource file rather than using a PDF DataMapper
background
merge.template.contexts.PRINT.sections['Policy'].background.source
= BackgroundResource.RESOURCE_PDF;

// Specify the path
var resourceUrl = 'images/policy-' + record.fields.policy + '.pdf';
merge.template.contexts.PRINT.sections['Policy'].background.url =
resourceUrl;
```

### Note

To learn how to set a PDF file as a background image on a Print section without a Control Script, see "Using a PDF file as background image" on page 65.

## Channel

`Channel` is an enumeration for the output channels. The active output channel is registered in `merge.channel`.

The channel doesn't change when the output consists of different contexts. When generating email, for example, the channel is `EMAIL`, even when merging the Print context to attach it to the email.

Value	Description
<b>EMAIL</b>	The merge request is for output to Email.
<b>PRINT</b>	The merge request is for output to Print.
<b>WEB</b>	The merge request is for output to Web.
<b>THUMBNAIL</b>	The merge request is for generating a template preview.

### Example

The following Control Script selects different sections for Print output and for Email with the Print context attached to it.

```
var printSections = merge.template.contexts.PRINT.sections;

if(merge.channel === Channel.EMAIL) {
    printSections['Section 1'].enabled = false;
    printSections['Section 2'].enabled = true;
}

if(merge.channel === Channel.PRINT) {
    printSections['Section 1'].enabled = true;
    printSections['Section 2'].enabled = false;
}
```

### ContextType

ContextType is an enumeration for the context types.

The type of the context that is going to be merged next can be retrieved via `merge.context.type`.

The context type needs to be specified when retrieving a section with `merge.template.contexts.ContextType.sections["section name"]`, for example `merge.template.contexts.PRINT.sections["Section EN"]`.

Value	Description
<b>HTML_EMAIL</b>	The context is the Email context.
<b>PRINT</b>	The context is the Print context.
<b>WEB</b>	The context is the Web context.

### Example

This script retrieves two Print sections. Then, depending on a value in the current record, it enables one section or the other, so that only one of the two sections appears in the output.

```
var printSections = merge.template.contexts.PRINT.sections;
printSections['Section EN'].enabled = false;
printSections['Section FR'].enabled = false;

if(record.fields.Language === 'FR'){
    printSections['Section FR'].enabled = true;
} else {
    printSections['Section EN'].enabled = true;
}
```

### MediaPosition

In a Control Script, the `position` is an enumeration for the position of background resources for a Print section. It is retrieved and set via `background.position`.

Field	Description
<b>ABSOLUTE</b>	Places the PDF at a specific location on the page. Set the background's top ( <code>background.top</code> ) and left ( <code>background.left</code> ) measured from the top and left side of the section.
<b>CENTERED</b>	Centers the PDF on the page, vertically and horizontally.
<b>FIT_TO_</b>	Stretches the PDF to fit the page size.

Field	Description
<b>MEDIA</b>	

## Examples

This script applies **absolute positioning** to the background of a Print section.

```
var activeSection = merge.template.contexts.PRINT.sections['Section 1'];
activeSection.background.source = BackgroundResource.RESOURCE_PDF;
activeSection.background.position = MediaPosition.ABSOLUTE;
activeSection.background.left = "10mm";
activeSection.background.top = "10mm";
activeSection.background.url = "images/somepage.pdf";
```

The next script scales the background of a Print section to the size of the **Media**.

```
var activeSection = merge.template.contexts.PRINT.sections['Section 1'];
activeSection.background.source = BackgroundResource.RESOURCE_PDF;
activeSection.background.position = MediaPosition.FIT_TO_MEDIA;
activeSection.background.url = "images/somepage.pdf";
```

# Generating output

When merged with a record set, the templates made in the Designer can generate twotypes of output: Print, and Email.

## Print output

Print templates, also called Print *sections*, are part of the Print context. They are meant to be printed to a printer or printer stream, or to a PDF file (see "Generating Print output" on page 495).

The Print context can also be added to Email output as a PDF attachment; see "Generating Email output" on page 509. When generating output from the Print context, each of the Print sections is added to the output document, one after the other in sequence, for each record.

To dynamically select a section for output, use a Control Script; see "Control Scripts" on page 240.

There is a number of settings in the Print context and Print sections that have an impact on how the Print context is printed; see "Print settings in the Print context and sections" on page 60.

To split the Print output into several files, see "Splitting printing into more than one file" on page 500.

## Email output

The Email context outputs HTML email with embedded formatting to an email client through the use of an email server. The HTML generated by this context is meant to be compatible with as many clients and as many devices as possible.

Although the Email context can contain multiple Email templates, only one of them can be merged with each record. Which one is used, depends on a setting; see "Email output settings in the Email context and sections" on page 510.

Email is sent in a single batch for the whole record set.

To test a template, you can test the scripts (see "Testing scripts" on page 231) and send a test email first (see "Send Test Email" on page 307), before actually sending the email (see "Generating Email output" on page 509).

## Attachments

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

To learn how to attach other files, see "Email attachments" on page 511.

# Optimizing a template

## Scripts

In the process of output generation, the execution of scripts may take up more time than necessary. To optimize a template, it helps to disable scripts that don't have an effect on the context that you're generating output from; see "Managing scripts" on page 228.

Other ways to speed up script execution are described in another topic: "Optimizing scripts" on page 234.

## Images

When a template that contains lots of images is merged with a large record set, the many file requests may slow down the process of output generation. The solution is simple: combine the images into a single image file and display the part that holds the image. This reduces the number of file requests and can improve the output speed significantly.

### **Step 1. Create a file that contains a collection of images.**

Static images may go in any type of image file. Store images that need be added dynamically to the template, in one PDF file, one image per page.

There are several tools to combine image files into a single PDF. **ImageMagick** is one of them. You could use the convert command of the ImageMagick library:

```
convert C:/myimages/*.jpg C:/myimages/image-collection.pdf
```

You could also use **Connect Designer** itself: create a print template with the size of your images and set the page margins to 0. Create a script that loops over your images and adds them to the text flow of the template. Subsequently generate PDF output and use the resulting file as your collection file.

**Step 2. Add the file that contains the collection of images to the template's Resources** (see "Adding images" on page 159).

**Step 3. Display part of the collection file as an image in the template.**

- **Static images** that are part of an image file can be displayed via Cascading Style Sheets (CSS). This technique is much used in web design. In this technique, the file that contains a collection of images is called an **image sprite**. The trick is to create a Box (or Div) for each image and give that box an ID (see "Boxes" on page 150). Then use the ID in a style sheet to select the Box and write a style rule (see "Styling templates with CSS files" on page 171) that sets its background image to the image sprite and positions the image. For an explanation and examples of this style rule, see [http://www.w3schools.com/css/css\\_image\\_sprites.asp](http://www.w3schools.com/css/css_image_sprites.asp).
- **Dynamically added images** are loaded in a script. To retrieve one page from a PDF file in a script, add the page parameter to the file path and set that as the source of the image. Here is an example (assuming that the page number is stored in a variable `pageNumber`):

```
var imageStr = "";
var imagePath = "file:///C:/image-collection.pdf?page=" +
pageNumber;
imageStr += ' Print Presets** allows you to create or modify printing Presets (which contain all the printing options), which can be saved for re-use in later print runs. This can be particularly handy when creating special print runs, that need to be run periodically.

These presets make it possible to do such things as filtering and sorting records, grouping documents and splitting the print jobs into smaller print jobs, as well as the more standard selection of printing options, such as binding, OMR markings and the like.

See "[Job Creation Presets](#)" on page 401 and "Output Creation Settings" on page 410 for more details.

# Connect Printing options that cannot be changed from within the Printer Wizard.

There are a number of settings for the Print context and Print sections that have an impact on how Print sections are printed, which **cannot** be influenced through either a job creation preset or output creation preset.

These settings are:

- **Duplex printing.** Duplex printing has to be enabled for a Print section, in order to print that section on both sides of the paper. See "Enabling double-sided printing" on page 68.
- **Finishing.** The Print context , as well as each of the Print sections, can have its own **Finishing** settings. In printing, Finishing is the way pages are bound together after they are printed. See "Setting the binding style for the Print context" on page 61 and "Setting the binding style for a Print section" on page 67. Also see "Finishing Options" on page 403 for an explanation of the Finishing options.
- **Bleed.** The margins *around* a page are called the Bleed. It can be used on some printers to ensure that no unprinted edges occur in the final trimmed document. See "Page settings: size, margins and bleed" on page 70.

## Print Using Standard Print Output Settings

When using the **File > Print...** option, the Print Configuration dialog appears. This dialog allows you to print the template using **Default** printer settings, or the **Last Used** printer settings or by using previously created Printing Presets.

To learn how to create Printing Presets please see [Job Creation Presets](#) and [Output Creation Presets](#).

- **Configuration Selection** Group:
  - **Output Creation:** Use the drop-down to select existing Output Creation Presets. Use the Gear button to edit the currently selected Preset or to reload the list of Presets from the system. 
  - **Job Creation:** Use the drop-down to select existing Job Creation Presets. Use the Gear button to edit the currently selected Preset or to reload the list of Presets from the system. 

- **Preset Summary:** Displays a summary of the settings for the currently selected Presets.

#### Note

The Default output type of PDF Output is actually a built in system Preset, whilst the Last Used settings can likewise be considered an un-named and un-saved Preset.

- **Records Group:**
  - **All:** Outputs all records in the active dataset.
  - **Selection:** Allows selection of a range of records or a custom selection. You can specify individual records separated by semi-colons (;) or ranges using dashes. For example: 2;4;6-10 would print pages 2, 4, 6, 7, 8, 9 and 10.
  - **Apply filtering and sorting to record selection** checkbox: Check to filter and/or sort records. Selecting this will open both the "Data Filtering Options" on page 404 and "Sorting Options" on page 405 pages.
- **Copies Group :**
  - **Copies:** Enter the number of output copies you want.
  - **Collate:** When printing multiple copies you can check this checkbox to have the record copies printed together. For example in a three record job the records would print out as 1-1-2-2-3-3, rather than 1-2-3-1-2-3.

#### Wizard navigation buttons

- **Advanced** button: Click to open the "Print Using Advanced Printer Wizard " on the next pagewhere you can manually change the printing options.

### Note

Any settings made within the **Advanced Print Wizard** do not permanently update any Preset(s) being used.

- **Print** button: Click to produce print output according to the current settings.
- **Cancel** button: Cancels the Print Wizard, without creating any printout.

## Print Using Advanced Printer Wizard

The **Advanced Printer Wizard** allows you to select from any and all output settings.

The Wizard can be used to generate once-off print runs (either entirely from scratch, or based upon selected pre-existing Presets).

**Note:** These print runs cannot be saved as presets and can only be replicated in the following print run, using the **Last Used** option.

The output settings are determined by selections made throughout the Wizard. For example, if you want to add Inserter Marks to the output, you select the Add Inserter Marks option on the first page of the Wizard, and the Inserter Options page will then appear later in the Wizard.

To open the Advanced Printer Wizard, select the **File > Print...** option and then click the **Advanced** button.

The first page of the Advanced Printer Wizard is the "Print Options" on page 411 page.

## Adding print output models to the Print Wizard

Connect comes with several pre-prepared print output models. These include Advanced Function Presentation (AFP), Intelligent Printer Data Stream (IPDS), Printer Control Language (PCL), Portable Document Format (PDF) and PostScript (including the PostScript variants of PPML, VIPP and VPS).

To keep the Print Wizard interface manageable only a limited range of print output models are available by default. Additional print output models can be added to the list at any time, though.

They can be selected from the range of pre-prepared models that come with Connect. The following topic describes how to do this.

After they have been added, the newly selected print output models will be available in the Print Wizard thereafter.

### How to add print output models from within the Print Wizard

Here is how to add print output options from within the Print Wizard dialog itself.

1. Select **File > Print...** from the menu. The Print dialog will be launched.
2. Click on the **Advanced** button. The Print Wizard will be launched.
3. Click the settings button  at the end of the Model selection.
4. Select **Edit available printers** from options.
5. In the **Preferences** dialog, select the print output models to be added to the Print Wizard, then click OK.

### How to add print output models from within the Designer

Here is how to add print output models from within the main Designer interface itself.

1. Select **Window > Preferences...** from the menu. Preference dialog is launched.
2. Select **Print > Available Printers** from the options.
3. In the **Available Printers** area, select the print output options to be added to the Print Wizard, then click OK.

## Splitting printing into more than one file

By default, when Connect saves the print output spool file to a directory, it creates one spool file that contains all the generated documents. It is, however, possible to output one spool file per document, or to create groups of documents and store those in separate spool files.

Where the output should go, and how documents should be grouped, is set in a *Job Creation Preset*.

To make one document or a group of documents go into a separate file, the print job needs to be 'separated'. Separation is one of the options to set in an *Output Creation Preset*.

See "[Generating Print output](#)" on page 495 for a further explanation about Job Creation Presets and Output Creation Presets.

## Variables available in the Output

In the Output Module, there are some variables available that offer more control over how templates are generated, or the data added to them.

Templates can be used in the following locations:

- The **Job Output Mask** field in the [Print Options](#), when using the **Directory** option.
- The Text, Barcodes, OMR and Image data available in the [Additional Content Options](#) page.

### Available Variables

The following is a list of variables that can be used independently of any job options loaded.

### The Template object

Contains information about the template. The default use of `${template}` expands to a name based on the template name. A four digit sequence number is added at the end of the basename. The file extension is determined by the selected output technology.

`${template}` is basically a short hand for `${template.base}_  
${template.nr,0000}.${template.ext}`

### `${template}`

The `0000` in `${template.nr,0000}` is a format pattern that takes care of formatting the number with at least four digits and leading zero's. See "Formatting date and number values " on page 506, below.

### Example

If the template file is `C:\Data\My-Invoices-EN.OL-template` which gets printed to PDF, then `${template}` expands to `My-Invoices-EN_0001.pdf`

Returns the base name of the template, which is the name of the template file without its path and without the trailing file extension.

### **`${template.base}`**

#### **Example**

If the template file is `C:\Data\My-Invoices-EN.OL-template`, then `${template.base}` expands to `My-Invoices-EN`  
Returns the name of the template file without the path.

#### **Example**

### **`${template.name}`**

If the template file is `C:\Data\My-Invoices-EN.OL-template`, then `${template.name}` expands to `My-Invoices-EN.OL-template`

Note, that `${template.name}` this still includes the extension of the template file (`.OL-template` in the example above).

An automatic sequence number belonging to the current output file. It is automatically incremented for each new output file that gets created. Note, that multiple output files are created, for example, when output separation has been selected for output creation.

### **`${template.nr}`**

It is possible to format the number using a pattern and locale. See "Formatting date and number values " on page 506, below.  
The extension that corresponds to the chosen output technology.

### **`${template.ext}`**

For example, for PDF output, `${template.ext}` would be `PDF`, for PostScript output, `${template.ext}` would return `PS`

Note, that `${template.ext}` does not include a leading dot.

### **The File object**

### **`${file}`**

`${file}` is basically a short hand for `${file.base}_`  
`${file.nr,0000}.${file.ext}` where `0000` in `${file.nr,0000}` is a format pattern that takes care of formatting the number with at least four digits including leading zero's. See "Formatting date and number values " on page 506, below.

### Server context:

On the Server, `${file}` expands to a file name based on the job name. A four digit sequence number is added at the end of the basename. The suffix (the extension) is defined by the selected output technology.

### **Example**

If the job name is my-invoices-reprint and is printed to PDF, then `${file}` expands to my-invoices-reprint\_0001.pdf

### Designer context:

In the Designer, `${file}` returns a generated name based on the current template name. A four digit sequence number is added at the end of the basename. The suffix is defined by the selected output technology.

### **Example**

If the template file is my-invoices.OL-template and is printed to PostScript, then `${file}` expands to my-invoices\_0001.ps  
The name of the template without dot extension (designer context) or the name of the job without dot extension (server context)

### **`${file.base}`**

### **Example (Designer context)**

If the template file is C:\Data\my-invoices.OL-template, then `${file.base}` returns my-invoices  
The extension that corresponds to the chosen output technology.

### **`${file.ext}`**

For example, for PDF output, `${file.ext}` would be PDF, for PostScript output, `${file.ext}` would return PS

Note, that `${file.ext}` does not include a leading dot.  
The name of the template (designer context) or the name of the job (server context)

### **`${file.name}`**

### **Example**

If the template file is C:\Data\my-invoices.OL-template, then

`${file.name}` returns my-invoices.OL-template

An automatic sequence number belonging to the current output file. It is automatically incremented for each new output file that gets created.

Note, that multiple output files are created, for example, when output separation has been selected for output creation.

**`${file.nr}`**

It is possible to format the number using a pattern and locale. See "Formatting date and number values " on page 506, below.

This variable is intended for internal use only.

It was introduced for use in Printer Definitions for PostScript printers.

**`${file.pageCount}`**

**Usage of this variable in an Output Preset or in the Print Wizard is discouraged and it should be regarded as deprecated.**

## The Job object

`${job}` expands to a name based on the job name. A four digit sequence number is added at the end of the basename. The file extension is determined by the selected output technology.

`${job}` is basically a short hand for `${job.base}_  
${job.nr,0000}.${job.ext}`

**`${job}`**

The 0000 in `${job.nr,0000}` is a format pattern that takes care of formatting the number with at least four digits including leading zero's. See "Formatting date and number values " on page 506, below.

### Example

If the job name My-Invoices-Reprint.XY2016 gets printed to PDF, then

`${job}` expands to My-Invoices-Reprint\_0001.pdf

Returns the base name of the job without any extension.

**`${job.base}`**

### Example

If the job name is My-Invoices-Reprint.XY2016, then `${job.base}` expands to My-Invoices-Reprint

**`${job.name}`**

Returns the name of the job.

## Example

If the job name is My-Invoices-Reprint.XY2016, then `${job.name}` expands to My-Invoices-Reprint.XY2016

An automatic sequence number belonging to the current output file. It is automatically incremented for each new output file that gets created.

Note, that multiple output files are created, for example, when output separation has been selected for output creation.

`${job.nr}`

It is possible to format the number using a pattern and locale. See "Formatting date and number values " on the facing page, below. The extension that corresponds to the chosen output technology.

`${job.ext}`

For example, for PDF output, `${job.ext}` would be PDF, for PostScript output, `${job.ext}` would return PS

Note, that `${job.ext}` does not include a leading dot. Value of a meta data property of the job.

`${job.metadata.  
propertyname}`

The *propertyname* must have been defined as a *Tag Name* on the *Job Tags* tab of the [Metadata Options](#) page in the *Advanced Print Wizard*.

Note: this is only available if separation is disabled or if *Separation* based on *Job* has been selected on the [Separation](#) page in the *Advanced Print Wizard*.

`${job.count.pages}` The total number of pages in the *Job*.

## Other available properties

These are various properties available to the Output module that are not part of a larger object:

`${system.time}` Displays the current system data and/or time. Can be formatted using the "Formatting date and number values " on the facing page, as seen below.

`${page.nr}` Page number, within *Document*.

`${page.sequence.job}` Page number, within the *Job*.

`${page.sequence.segment}` Page index, within the *Job Segment*.

`${page.sequence.set}` Page index, within the *Document Set*.

`${document.metadata.` Value of a meta data property of the document.

The *propertyname* must have been defined as a *Tag Name* on the *Document Tags* tab of the [Metadata Options](#) page in the *Advanced Print Wizard*.

*propertyname*}

Note: this is only available if *Separation* based on *Document* has been selected on the [Separation](#) page in the *Advanced Print Wizard*.

**`\${document.sequence.job}`** Document index, within the *Job*.

**`\${document.sequence.set}`** Document index, within the *Document Set*.

**`\${document.count.pages}`** Total pages within the *Document*.  
Value of a meta data property of the document set.

The *propertyname* must have been defined as a *Tag Name* on the *Document Set Tags* tab of the [Metadata Options](#) page in the *Advanced Print Wizard*.

**`\${set.metadata.  
propertyname}`**

Note: this is only available if *Separation* based on *Document Set* has been selected on the [Separation](#) page in the *Advanced Print Wizard*.

**`\${set.count.pages}`** Total pages of the *Document Set*.

**`\${set.count.documents}`** Number of Documents within the *Document Set*.  
Value of a meta data property of the job segment.

The *propertyname* must have been defined as a *Tag Name* on the *Job Segment Tags* tab of the [Metadata Options](#) page in the *Advanced Print Wizard*.

**`\${segment.metadata.  
propertyname}`**

Note: this is only available if *Separation* based on *Job Segment* or *Split At Exactly n Sheets* has been selected on the [Separation](#) page in the *Advanced Print Wizard*.

**`\${segment.count.pages}`** Total pages within *Job Segment*.

### Formatting date and number values

Date and number values can be formatted using an optional pattern and/or locale.

| Form                    | Description    | Example                        | Result       |
|-------------------------|----------------|--------------------------------|--------------|
| <b>`\${expression}`</b> | Do not format. | <code>`\${system.time}`</code> | July 4, 2009 |

|                                            |                                                          |                                                  |                      |
|--------------------------------------------|----------------------------------------------------------|--------------------------------------------------|----------------------|
|                                            |                                                          |                                                  | 12:30:55 PM          |
| <code>\${expression,pattern}</code>        | Apply pattern with system locale                         | <code>\${system.time, yyyyMMdd-HH:mm:ss}</code>  | 20090704-12:30:55    |
| <code>\${expression,pattern,locale}</code> | Apply pattern with the specified country locale          | <code>\${system.time, "dd MMMM yyyy", nl}</code> | 4 juli 2009          |
| <code>\${expression,,locale}</code>        | Apply a default format with the specified country locale | <code>\${system.time,,nl}</code>                 | 4 juli 2009 12:30:55 |

It is possible to enclose the values of the pattern and locale in single or double quotes. This is required for including whitespace in a pattern, or when the `${expression}` would otherwise be ambiguous.

At run-time, the output engine determines the type of the value yielded by the expression. If this is a number, a number pattern is expected. For date/time-like types, a date pattern is expected. When no pattern is specified, some default format is applied. For other types, it is not possible to specify a pattern or locale.

## Generating Tags for Image Output

It is possible, even easy, to generate specific tags and indexes for PrintShop Mail Image. This can be used to send email, archive with Search or output to image formats.

### Required Components

The following components are required in order to output to Image:

- A PrintShop Mail Imaging license.
- A Job Preset adding the appropriate metadata fields
- An Output preset generating a PDF/VT file.
- A PrintShop Mail Workflow process outputting to the PrintShop Mail Image task.

### Job Preset Configuration

For email output, the following metadata fields must be added to the [Metadata Options](#) page:

- **ImageSendTo:** Add to have PrintShop Mail Image use the specified field as the E-mail address to which to send the PDF file.
- **ImageSendCc:** Add to have PrintShop Mail Image use the specified field as the E-mail address to which to send the PDF file as a carbon copy (CC).
- **ImageSendBcc:** Add to have PrintShop Mail Image use the specified field as the E-mail address to which to send the PDF file as a blind carbon copy (BCC).
- **ImageSubject:** Add to have PrintShop Mail Image use the specified field as the subject of the email that is sent.
- **ImageBody:** Add to have PrintShop Mail Image use the specified field as the body of the email that is sent.

Note that the PDF file generated by the Print context is sent as an attachment to the email sent using the information above.

### **PrintShop Mail Search Indexing**

For PrintShop Mail Search indexing, you can add your own custom fields. Each field that is not included in the above or in Generating Fax output is added as an index for PrintShop Mail Search. For example you could add CustomerID and this would appear as the CustomerID index in Search. Yes, it's that easy!

### **Output Preset Configuration**

The following settings must be used in the Output Preset:

- In the [Print Options](#), a PDF type should be selected, such as Generic PDF.
- In the [PDF Options](#), the PDF Type should be set to PDF/VT

### **PrintShop Mail Workflow Process**

The following Workflow will produce Image output:

- The four regular Connect tasks to generate print output:
  - **Execute Data Mapping**
  - **Create Print Content**
  - **Create Job** using the above *Job Preset*

- **Create Output** using the above *Output Preset*. The task's **Output Management** must be set to be *Through Workflow*.
- The **PrintShop Mail Image** connector task set to Passthrough (the first "Document" on the list). If sending Email, choose the "Send Email" option of the Output group. Otherwise, choose Archive Output, ensure the output type is PDF, and optionally fill in the PrintShop Mail Search Database tab appropriately.

## Generating Email output

The Email context outputs HTML email with embedded formatting to an email client through the use of an email server. The HTML generated by this context is meant to be compatible with as many clients and as many devices as possible.

Email is sent in a single batch for the whole record set.

To test a template, you can send a test email first.

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

To learn how to attach other files, see "Email attachments" on page 511.

### Before generating Email output

- Decide on the use of an Email Service Provider; see "Using an ESP with PrintShop Mail Connect" on page 513.
- Make sure that a data set is loaded, that any necessary files, such as images and attachments, are in place, and that the correct settings are selected (see below).
- You may want to **rasterize** certain elements, such as <div> elements, business graphics, or headings with a special font type. Rasterizing converts the element to a JPG or PNG image. This is very useful to support as many clients as possible. For example, some email clients may not support SVG, so converting a resource to JPG instead would ensure that most email clients would actually see the output.

To rasterize an element, right-click it and select **Rasterize options**. For a JPG image you can set the quality of the resulting image in a percentage.

## Email output settings in the Email context and sections

The following settings for the Email context and Email sections have an impact on how the actual emails are sent.

- An Email To Script must be available in the template and refer to a valid email address; see "Email header settings" on page 98. If any record does not have a valid email, this record is skipped automatically when generating email output.

### Note

When you send a test email, the Email To Script will not be used; instead, the email will be sent to the address that you specify in the Send Test Email dialog.

- The subject of the email is a property of an email section. See "Subject" on page 99.
- The sender(s), recipient(s) and the subject can be set using Script Wizards; see "Email header settings" on page 98.
- Default SMTP settings can be set in the preferences; see "Email header settings" on page 98.
- If there are multiple Email sections, only one of them can be merged with each record. Make sure that the correct section has been set as the default; see "Setting a default Email template for output" on page 97.  
To dynamically select a section for output, use a Control Script; see "Control Scripts" on page 240.
- PDF attachments can be compressed to make the files smaller; see "Compressing PDF attachments" on page 94.

To generate Email output:

1. Open a template with an Email context.
2. Load a data file or database compatible with this template. See "Loading data" on page 204.
3. On the **File** menu, click **Send Email** or **Send Test Email**. In the dialog that appears you can, among other things, attach the Print context or the Web context to the email. See

"Send Email" on page 306 or "Send Test Email" on page 307 for a description of all the options. Finally, click OK.

### Note

#### About testing emails

When you send a test email, the Email To Script will not be used; instead, the email will be sent to the address that you specify in the Send Test Email dialog. If you have a Litmus account, you can enter your Litmus test address. To make the test address appear by default, you can set the default test address in the Email Preferences: select **Window > Preferences**, click the arrow next to **Email**, click **General** and type the test address next to **Email Test address**.

For a description of how to test your email for different email clients, see this how-to: [Test your emails with Litmus](#). For more information on Litmus, please see <http://litmus.com/>

### Tip

For a detailed description of how to use Mandrill with Connect to send and track emails, see the following how-to: [Using Mandrill](#).

## Email attachments

Output, generated from an Email template, can have the following attachments:

- The contents of the Print context, in the form of a single PDF attachment.
- Other files, an image or a PDF leaflet for example.

Attaching the Print context is one of the options in the Send (Test) Email dialog.

By default, when adding the Print context to an email, all Print sections are output to a single PDF file, named after the email subject, which is then attached to the email. The PDF can be protected with a password (see "Email PDF password" on page 101).

## Note

To split the Print context into multiple attachments, or to attach multiple Web sections as separate attachments, you need to create a Control Script that specifies **parts**; see "Parts: splitting and renaming email attachments" on page 246.

This topic explains how to attach files other than those generated by the Print or Web context. This is also described in a how-to; see [Add custom email attachments](#).

## Attaching external files

To attach files other than those generated by the Print or Web context to Email output:

1. Add the files to the template; see Adding images, or put them in a folder that is available to the machine that outputs the emails.
2. Create a script: on the **Scripts** pane at the bottom left, click **New**. A new script appears in the list. Double-click on it to open it. If you are not familiar with scripts, see "Writing your own scripts" on page 223 for an explanation of how scripts work.
3. Change the name of the script, so that it reflects what the script does.
4. Choose the option **Selector** and in the **Selector** field, type **head**.
5. Write a script that appends a <link> element to the `results` (the selector is `head`, so the `results` contain the <head> of the email).
  - Make sure to set the **rel** attribute to **related**.
  - The **href** attribute determines where the file comes from. For resources inside of the template, use `'images/file.extension'`, or `'fonts/myfont.otf'`, etc. For external resources, you need the full path to the file, such as `'file:///c:/resources/attachments/instructions.pdf'`. Of course, you can also use dynamic calls such as `'file:///c:/clientfiles/' + record.fields.client_id + '/invoices/' + record.fields.invoice_number + '.pdf'`.

## Examples

The following script attaches a PDF file named letter-CU00048376.pdf to each generated email. The PDF file is located in the Images folder on the Resources panel.

```
results.append("<link rel='related' href='images/letter-CU00048376.pdf'>");
```

If that same file would be located on the C: drive, the script should refer to it as follows:

```
href='file:///C:/letter-CU00048376.pdf'.
```

The link doesn't have to be static; you could use data from the record set to build the link, for example:

```
var customerID = record.fields.ID;
results.append('<link rel="related" href="images/letter-' +
customerID + '.pdf">');
```

## Using an ESP with PrintShop Mail Connect

An email service provider (ESP) is a company that offers email marketing or bulk email services.

This topic explains why and how to use an ESP with PrintShop Mail Connect

### Reasons to use an ESP

These are a number of reasons why you would need an ESP:

- ESPs ensure a high deliverability, as most ESPs are whitelisted or approved by ISPs (Internet Service Providers) as legitimate email delivery service. So they help you to avoid having mail detected as spam.
- ESPs provide comprehensive tracking options to measure open rates and they log which links were clicked and by who. Typically this information is available via an online dashboard.
- Most ESPs provide Bounce Management options. They will stop sending messages to addresses that return a hard bounce and retry for soft bounces before removing that address.
- EPSs can handle unsubscribes and prevent accidental sends in the future.

### Choosing an ESP

The first thing to do to use an ESP with PrintShop Mail Connect is to choose an ESP and create an account.

Mandrillapp.com, a popular ESP, used to have a free account but now requires a paid MailChimp account. Luckily there are plenty of alternatives that provide free accounts (often capped to a max number of emails per month and sometimes having throttled output).

PrintShop Mail Connect has been tested with: Mandrillap.com, SendGrid (easy user management), MailGun (nearly instant statistics) and MailJet (shows best performance on the free account).

### Adding an SMTP Preset for an ESP

After creating an account, add a SMTP settings preset in PrintShop Mail Connect for the chosen ESP, via the Preferences dialog of the Designer (see "Email SMTP settings" on page 98).

Make sure **Use authentication** is checked, and put in your SMTP Username in the box below.

#### Note

Presets for different ESPs are already available in the list of default presets.

### Sending an email with an ESP

To send an email or test email with the use of an ESP, start generating the email as usual (see "Generating Email output" on page 509). In the Send (Test) Email dialog, pay attention to the following settings:

- In the **Outgoing mail settings** area, select the preset for your ESP in the Presets dropdown.
- In the **Password** box, type the password provided by the ESP.

#### Note

The ESP might also have a test function you can use. Check the options of your ESP.

#### Tip

For a detailed description of how to use Mandrill to send and track emails, see the following how-to: [Using Mandrill](#).

## Adding custom ESP handling instructions

Most ESPs allow you to provide custom handling instructions as part of the email message, via custom headers. Typically these include instructions to enable open rate tracking, click through rate tracking and assign tags/categories to messages. Assigning a tag/category allows you to view statistics per email type in the dashboard of the ESP. Note that each ESP has its own notation and instructions.

In a Connect template, adding these custom headers is handled through a Control Script (see "Control Scripts" on page 240 and "Control Script API" on page 477). The following samples show how to assign a tag or category to a message (e.g. 'invoice', 'confirmation', 'newsletter-jan-2017') for various ESPs.

### SendGrid

Dashboard: <https://app.sendgrid.com/>

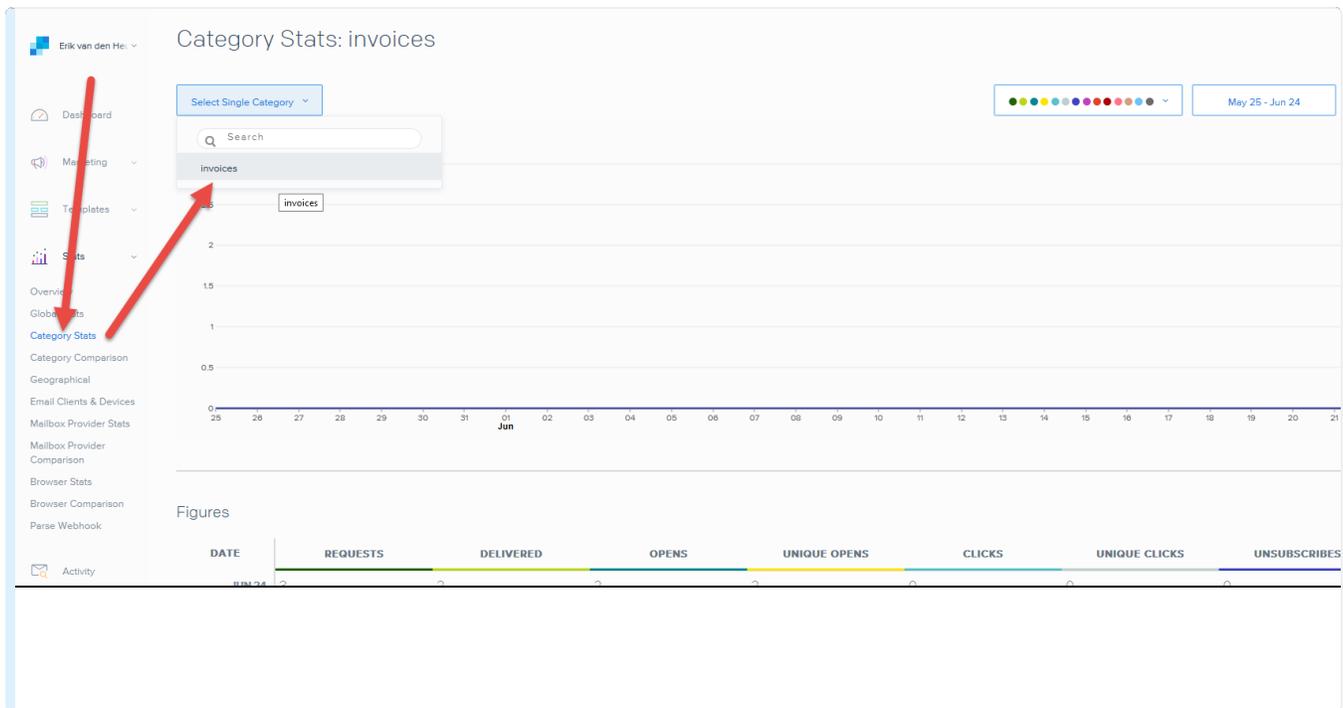
Documentation: [https://sendgrid.com/docs/API\\_Reference/SMTP\\_API/using\\_the\\_smtp\\_api.html](https://sendgrid.com/docs/API_Reference/SMTP_API/using_the_smtp_api.html)

Sample Control Script to assign a category:

```
var headerObj = {
    "category": ["invoices"]
};
merge.context.sections["Content"].headers = {
    "X-SMTPAPI": JSON.stringify(headerObj)
};
```

#### Note

Sendgrid strips out their mail headers. The results need to be verified via their Dashboards (e.g. the Stats section lets you verify the stats for specific categories). Alternatively one can use their Web API to retrieve stats in JSON format. To view the category stats, log in to Sendgrid and choose: Stats > Category Stats > your category name.



## MailGun

Dashboard: <https://mailgun.com/cp/stats>

Documentation: <https://documentation.mailgun.com/api-sending.html#sending>

Sample Control Script to assign a tag:

```
merge.context.sections["Content"].headers = {
    "X-Mailgun-Tag": "invoices"
};
```

### Note

The Mailgun tag allows you to view the stats per tag. Mailgun has a quick refresh and stats are available almost instantly.

|                  |   |   |   |   |   |   |   |   |
|------------------|---|---|---|---|---|---|---|---|
| Bounces          | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Spam reports     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Unsubscribes     | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Incoming         | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Posts via routes | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

\* Note: Counters are UTC-based.

### Categorized By Tag

Delete Selected  Search or Reset

|                                                            | Jun 18 | Jun 19 | Jun 20 | Jun 21 | Jun 22 | Jun 23 | Jun 24 | Total |
|------------------------------------------------------------|--------|--------|--------|--------|--------|--------|--------|-------|
| <input checked="" type="checkbox"/> Invoices-tag Delivered | 0      | 0      | 0      | 0      | 0      | 0      | 10     | 10    |

Navigation: < < > >

Mailgun domain's tag limit is 4000.

### Open & Click Event Settings For The Current Domain

Tracking clicks:  Yes and opens:  Yes

If you enable this feature, Mailgun will scan your messages for links and rewrite them.

Looking for webhooks? We've moved them to a [new Webhooks page](#).

## MailJet

Dashboard: <https://app.mailjet.com/dashboard>

Documentation: [https://app.mailjet.com/docs/emails\\_headers](https://app.mailjet.com/docs/emails_headers)

Sample Control Script to assign a campaign:

```
merge.context.sections["Content"].headers = {
    "X-Mailjet-Campaign": "invoices"
};
```

### Note

Mailjet strips out their own mailheaders like X-Mailjet-Campaign. The results can only be verified via the respective campaign stats page in the Mailjet dashboard. There is no need to pre-create the campaign: adding it to the email header via a Control Script auto-generates the campaign. To view the campaign, login to Mailjet and choose: Campaigns > All.

Campaigns Transactional Automation <sup>BETA</sup> Contacts Stats Help E Erik Van Den Heuvel

My campaigns Compare campaigns My templates

## My campaigns

All ★ Starred ● Draft ● Programmed ● Sent A/X test Archived

Search by campaign title or subject... New campaign

|   | Title                          | Subject                                          | Emails | Opens  | Clicks | Delivery date / Status |          |
|---|--------------------------------|--------------------------------------------------|--------|--------|--------|------------------------|----------|
| ☆ | ● foobar                       | Foo                                              | -      | -      | -      | -                      | Manage ▾ |
| ☆ | ● campaign_o_l_connect_foo     | Foo                                              | -      | -      | -      | -                      | Manage ▾ |
| ☆ | ● "o_l_connect_invoices"       | <span>SMTP</span> [Test] Email subject INV101... | 35     | 5.71%  | 0%     | 23/06/2016 19:59       | Manage ▾ |
| ☆ | ● "campaign_o_l_connect_fo..." | <span>SMTP</span> [Test] Email subject INV884... | 6      | 33.33% | 0%     | 23/06/2016 19:53       | Manage ▾ |
| ☆ | ● "campaign_o_l_connect_te..." | <span>SMTP</span> [Test] Take action             | 3      | 33.33% | 0%     | 23/06/2016 19:48       | Manage ▾ |

# Print Manager

The **Print Manager** must be used when generating output for IPDS printers, but can also be used in conjunction with PCL and PostScript outputs.

IPDS printing requires a *print stream* and PrintShop Mail Connect only produce *files*, thus this intermediary **Print Manager** is necessary to communicate to IPDS printers. It feeds the IPDS output to the printer, as a print stream, rather than as a file.

Neither PCL nor PostScript printers require a *print stream*, so such output files can be sent directly to a printer, from within Connect. You could use the **Print Manager** application, however, to extra print management functionality, such as organizing job priorities and job queues.

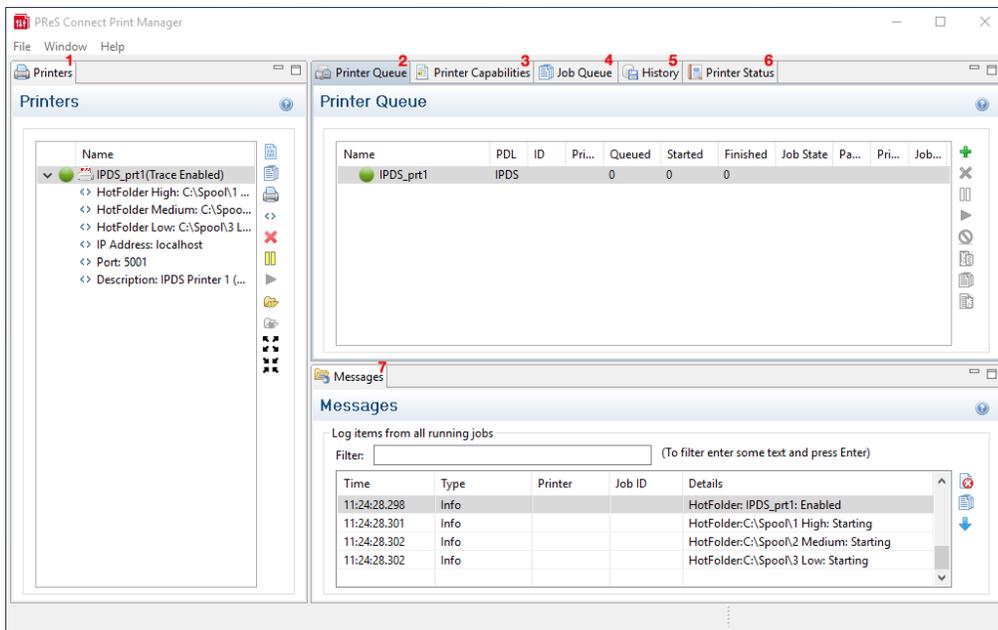
## Print Manager Introduction Video

## Print Manager usage

For a full description of how to use **Print Manager**, see "Print Manager Interface" below

## Print Manager Interface

The Connect Print Manager Interface is broken up into multiple tabs. These can be re-organised and moved as desired, but appear by default as in the following image.



1. "Printers pane" on the next page
2. "Printer Queue pane" on page 524
3. "Printer Capabilities pane" on page 525
4. "Job Queue pane" on page 526
5. "History pane" on page 527
6. "Printer Status pane" on page 528
7. "Messages pane" on page 529

## Menu options

- **Windows > Preferences ...:** The PrintShop Mail Connect Print Manager general settings can be changed via the **Windows > Preferences ...** menu option. Information on the preferences available can be found on this page: "Print Manager Preferences" on page 529.
- **Windows > Reset Perspective:** The PrintShop Mail Connect Print Manager perspective can be reset via the **Windows > Reset Perspective** menu option. This returns all the Print Manager panes (which can be moved around as desired) to their original default positions.

# Printers pane

The Printers pane displays a list of printers that have been added to the Print Manager.

The following information (all editable using the "Printer Settings Dialog" on the facing page) is displayed for each printer:

- **Printer Name:** Displays the name of the printer.  
The background color and the icon to the left of the printer indicates the current status of the queue. The two options are:
  - **Green:** The queue is active and any job sent to it is immediately streamed to the printer.
  - **Yellow:** The queue is paused and jobs will wait before being streamed to the printer.
- **Hot Folders:** Displays the paths of any hot folders.
- **IP Address:** Displays either the IP Address or machine name of the IPDS printer.
- **Port (IPDS Only):** Displays the port the printer communicates on.
- **LPR (PCL and PostScript Only):** Displays the printing file type.
- **Description:** Displays printer description.

To the right of the printer list, the following controls are available, as well as in right-mouse click context menus:

-  **Add job to selected printer:** Click to load an file directly and add it to the printer queue.
-  **Add job with Page Range:** Click to load an file directly and add it to the printer queue while specifying the page range to print.
-  **Add Printer:** Click to add a new printer using the "Printer Settings Dialog" on the facing page.
-  **Edit Printer:** Click to edit the currently selected printer using the "Printer Settings Dialog" on the facing page.
-  **Delete:** Click to delete the currently selected printer.
-  **Pause:** Click to pause the currently selected printer's queue, which prevents any new jobs from being printed.

- **▶ Resume:** Click to resume the currently selected printer's queue. Any job within the queue will start being printed.
- **📁 Pause HotFolder:** Click to pause the hot folder for the currently selected printer. This does not prevent jobs in the printer queue from generating output - only new jobs will stop being captured in the Hot Folder. This button is only activated when the Hot Folder is currently active.
- **📁 Resume HotFolder:** Click to resume capturing new jobs in the Hot Folder for the currently selected printer. This button is only activated when the Hot Folder is paused.
- **🔍 Expand:** Click to expand the current printer details within the Printer pane, if not already expanded.
- **🔍 Collapse:** Click to hide the current printer details within the Printer pane, if they were currently expanded.

## Printer Settings Dialog

This dialog allows you to add or modify custom settings for printers. The settings available will differ between printer types. IPDS output has more options than the others.

Each printer setting subsection appears in it's own area, which can be folded up or down by clicking on the blue title bar atop each subsection.

The available entries include the following options:

- **Printer Details:**
  - **Enable Printer:** Check to enable the printer settings.
    - **Printer Type:** Select the type of printer to add, from the drop down list.
    - **Name:** Enter a name that describes the printer in the list.
    - **IP Address:** Enter the IP Address or machine name of the printer.
    - **Port (IPDS Only):** Enter the port on which the printer communicates.
    - **LPR (PCL and PostScript Only):** Enter the printer queue.

### Tip

This will almost always be RAW.

- **Hot Folder** subsection: Defines folders from which print output files are automatically captured and sent to the appropriate printer queue.
  - **Enable Hot Folders:** Check to enable hot folders. Then enter the folders from which the Print Manager will capture print files automatically and send them to the appropriate Priority queue on the printer. There are **High**, **Medium** and **Low** priority queues available and each can be set here.
- **Extended Options (IPDS only)** subsection: Defines logging of printer activities.
  - **Ack Pages:** The number of pages to send to the printer for printing before sending an acknowledgement request. Acknowledgement requests slow the printing process down, as the printer needs to process the request and return the information. The default Ack Pages number is 100 pages which means the Print Manager will send an acknowledgement request to the printer in 100 page increments until the end of the job. Given the overhead in processing such requests, the smaller the Page number, the slower the printing becomes.
  - **Create Trace File:** Select this to log all communications to a log file.
- **End of Job Preferences (IPDS only)** subsection: Select from these options to determine job end processing.
  - **Edge Marks:** Select this to add an Edge Mark to the final page of a job. Used to indicate job separation in Continuous Form printing.
  - **Stack Receive Pages:** Add blank pages to the end of the job in order to push the completed job to the stacker.
- **Error Handling Preferences (IPDS only)** subsection: Select these to **STOP** printing upon encountering the following errors:
  - **Undefined Character Check:** If an undefined character is encountered in the job. For example, the Euro symbol might be present in the job, but missing from the printer.
  - **Report Page Position Check:** When printed beyond the edge of the page boundaries. For example, an image that extends beyond the edge of the page.
  - **Handle Errors:** Stops on all other optional errors.

- **Description** subsection: An optional printer description can be entered here, if desired.

## Printer Queue pane

The Printer Queue pane displays jobs, in similar fashion to the [Job Queue pane](#). However in this case, it displays only the jobs that are associated with a specific printer.

- **Printer Queue** table
  - **Name**: Displays the name of the printer where the job is located.
  - **PDL**: Displays the type of printer language.
  - **ID**: The generated ID for the job. The first 2 letters of the ID indicate the origin of that job:
    - **HF**: Hot Folder capture.
    - **DR**: Dragged and dropped directly into the Print Manager.
    - **BR**: Added via a Browse button.
    - **RS**: Resubmitted to the printer.
  - **Priority**: Displays the priority in which the job is being printed: Low, Medium, High.
  - **Queued**: Displays the time that the job entered the printer queue, or the number of jobs queue on the printer.
  - **Started**: Displays the time that the job started being printed, or the number of started jobs on the printer.
  - **Finished**: Displays the time that the job has finished printing, or the number of finished jobs on the printer.
  - **Job State**: Displays the current job status such as Queued, Finished, Started, etc.
  - **Page Range**: Displays the range of pages to be printed.
  - **Printed Pages**: Displays the current number of printed pages in the current job, or the total number of printed pages on the printer.
  - **Job Pages**: Displays the total number of pages in the job.

To the right of the table and also via right-mouse click context menu within the table, the following options are available:

- **Add Job** button / context menu option: Click to add a new job to the selected printer in the list.
- **Remove Job** button / context menu option: Click to delete the currently selected job.
- **Pause Job** button / context menu option: Click to temporarily pause the current job on the printer queue.
- **Resume Job** button / context menu option: Click to start a job that has been paused.
- **Cancel Job** button / context menu option: Click to cancel a job.
- **Resubmit Job** button / context menu option: Click to restart a job from the beginning.
- **Resubmit Range of Pages** button / context menu option: Click to resubmit a selection of pages.
- **Re-assign to Job Queue** button / context menu option: Takes the job out of the Printer Queue and places it back in the generic Job Queue.
- **Pause Printer** button / context menu option: Click to pause an active printer.
- **Resume Printer** button / context menu option: Click to start a printer which has been paused.
- **Remove Finished Jobs** button / context menu option: Click to remove finished jobs from the Printer Queue display, making the visual monitoring of active and scheduled jobs easier.

## Printer Capabilities pane

The Printer Capabilities pane displays the technical capabilities of any printer that has been added to the "Printers pane" on page 521.

- **Filter:** Type in a filter for a specific capability name and press Enter to filter the results below.
- **Name:** Displays the list of printers and their capabilities. By default, capabilities are not displayed for performance reason. To display the capabilities, right-click on the printer and select "**Check Printer Capabilities**" or click on the button on the right.

### Note

Requesting a printer's capabilities can take some time, and not all printers will support it.

- **Capabilities:** Each printer capability is listed under the Printer, in separate divisions. Capabilities are read directly from the printer.

## Job Queue pane

The Job Queue pane is a centralized location where all the jobs that are captured, added or sent to the Print Manager are listed, whether they have been printed or not.

- **Filter:** Type in a search word and hit Enter to filter the job queue. This searches the job name. Clear and press Enter to remove the filtering.
- **Job Queue table:**
  - **Name:** The name of the job filename, as captured or added to the Print Manager.
  - **ID:** The generated ID for the job. The first 2 letters of the ID indicate the origin of that job:
    - **HF:** Hot Folder capture.
    - **DR:** Dragged and dropped directly into the Print Manager.
    - **BR:** Added via a Browse button.
    - **RS:** Resubmitted to the printer.
  - **PDL:** Displays the job printer language.
  - **Page Count:** Displays the total number of pages in the job.
  - **Resolution:** Displays the resolution, in DPI, of the job.
  - **Page Size:** Displays the dimensions of the job.

To the right of the table and also via right-mouse click context menu within the table, the following options are available:

- **Copy selected items to clipboard** button / **Copy** context menu option: This copies job information (job name and folder path) to the clipboard.

- **Properties** button / context menu option: This launches a job Details Window that displays detailed information about the makeup of the print job. Once the Details Window is open it remains atop the Print Manager thereafter, to allow simple comparisons between jobs. To close the Details Window, put the focus upon the Details Window by clicking anywhere within it, and then press the Escape button ("Esc").
- **Add Job** button / context menu option: Click this to add a print Job to the Job Queue.
- **Remove Job** button / context menu option: Click this to remove a print Job from the Job Queue.

## History pane

The History pane displays a history of all jobs that have been printed through the Print Manager.

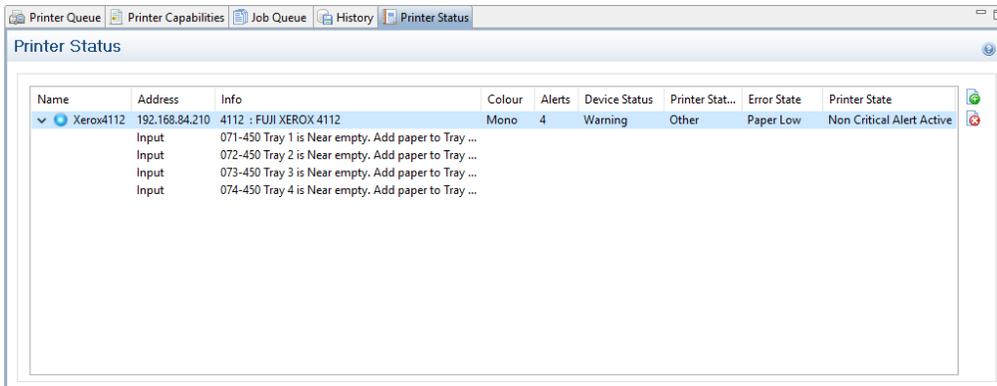
- **Filter:** Type in a search word and hit Enter to filter the job history. Clear and press Enter to remove the filtering.
- **History Log:**
  - **Date:** Displays the date and time when the job was started.
  - **Printer:** Displays the name of the printer the job was printed to.
  - **ID:** Displays the internal ID of the job.
  - **Job:** Displays the name of the job based on its filename.
  - **Path:** Displays the path from where the job was captured or added.
  - **Pages:** Displays the pages printed and the total number of pages in the job.
  - **Status:** Displays the final status of the job, such as Cancelled or Printed.
  - **Priority:** Displays in which queue priority the job was printed on the printer.

To the right of the table and also via right-mouse click context menu within the table, the following options are available:

- **Copy selected items to clipboard** button / **Copy** context menu option: Click to copy all messages to the clipboard to paste them in an external software.
- **Always select last log entry** button / **Tail** context menu option: Click to toggle auto-scrolling so that the last message is always displayed when it appears.
- **Remove Job** button / context menu option: Click to delete job from history.

# Printer Status pane

This pane can be used to monitor the state of PostScript and PCL printers. It contains general printer information, such as the printer's B&W/color settings as well as the quantity of paper in trays, any outstanding alerts or errors and whether the printer is currently in use, or awaiting a job. An example of this can be seen in the following image:



To **Start monitoring a printer** select the start icon to the right of the status display: 

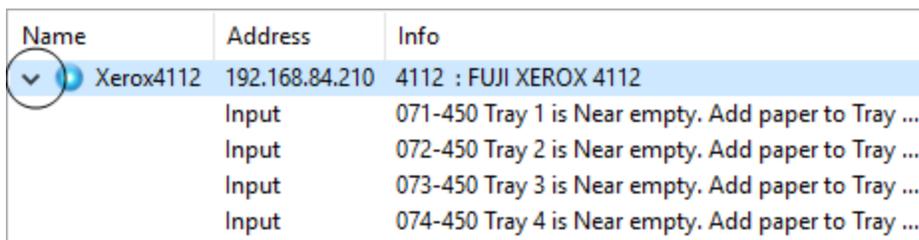
To **Stop monitoring a printer** select the stop icon to the right of the status display: 

Once a printer is being monitored, the details can be expanded and the individual printer trays displayed. These can be done as follows:

- **To expand the printer details**, press the right arrow besides the printer Name, as seen in the following image.



- **To reduce the printer details**, press the down arrow besides the printer Name, as seen in the following image.



# Messages pane

The Messages pane displays any notice, warning or error message that is generated by the Print Manager. This may include indications that jobs have finished or started, that errors have occurred, etc.

- **Filter:** Type in a search word and hit Enter to filter the messages. Clear and press Enter to remove the filtering.
- **Messages table:**
  - **Time:** Displays the timestamp for the message.
  - **Type:** Displays the type of message.
  - **Printer:** Displays the name of the relevant printer, if applicable.
  - **Job ID:** Displays the relevant Job ID, if applicable.
  - **Details:** Displays the message itself.

To the right of the Messages table, the following controls are available:

- **Clear Log Items:** Click to clear the current log. This does not delete log files.
- **Copy selected items to clipboard:** Click to copy all messages to the clipboard to paste them in an external software.
- **Always select last log entry:** Click to toggle auto-scrolling so that the last message is always displayed when it appears.

# Print Manager Preferences

The Print Manager **Windows > Preferences ...** menu option allows the adjustment of Print Manager preferences.

There are three groups of preferences which can be altered, or set back to default settings at any time. These groups are:

1. "Language Choice" on the facing page
2. "General Settings" on the facing page
3. "SNMP Connection Settings" on page 531

## Language Choice

Select the display language.

As of PrintShop Mail Connect version 1.5 the languages supported are:

- German (DE)
- English (EN)
- Spanish (ES)
- French (FR)
- Italian (IT)
- Japanese (JA)
- Chinese Simplified (ZH)
- Chinese Traditional (ZH\_HK, ZH\_MO or ZH\_TW)

## General Settings

- **Hot Folder:** Defines a global hot folder that captures jobs and places them in the [Job Queue pane](#). This does not automatically print the jobs, however.
  - **Enable Hot Folder:** Check to enable the global hot folder.
  - **Monitored Folder:** Enter a path for the hot folder, or click Browse to browse for it.
- **Session Retries:** The number of times Print Manager will try to reconnect to the printer before it stops trying and produces a timeout error in the messages.
- **Printer Timeout (sec):** The startup time of the printer before the timeout is reported in the messages.
- **Printer Ack Timeout (sec):** This is the time that Print Manager waits for a response from the printer after sending a response request.
- **Retry Delay:** This is the time in second that the Print Manager waits before retrying a job after the timeout.
- **Log Level:** Use the drop-down to select which log level to save. Higher log levels such as "Debug" give more helpful messages and details in the log but can impact processing time and resources.

- **Printer Preferences:**

- **Hide disabled printers:** Check box to have any printers that have been set to disabled in the "Printer Settings Dialog" on page 522 not appear in the "Printers pane" on page 521.

## **SNMP Connection Settings**

Options for printers that support Simple Network Management Protocol (SNMP) connection.

These include:

- **Connection attempts:** Defines the number of times to try a connection before abandoning the connection attempt,
- **Polling interval (secs):** Defines the polling interval in seconds. This determines how often the Print Manager communicates with the printer. This setting applies to all communications with the printer. It impacts upon the **Connect attempts** (determining the interval between attempts) as well as how often the printer will be updated in the "Printer Status pane" on page 528.

# Release Notes

## Overview

This document provides an overview of the new features and enhancements in PrintShop Mail Connect 1.5.

### Installing PrintShop Mail Connect 1.5

- PrintShop Mail Connect is released as a 64 Bit version only.
- Full details on installing and licensing PrintShop Mail Connect can be found in the online help in the installer.
- Note that both PrintShop Mail come with 30 day trial licenses by default.

## Connect 1.5 Designer Enhancements and Fixes

### General Designer improvements

- A **color selection eyedropper** has been added, to allow the selection of a color from elsewhere on screen. (SHARED-33561/33646/36293)
- **Improved responsiveness** within the Designer, particularly when dealing with large and complex documents. (SHARED-44309)
- A **configurable Auto Save functionality** has been added for templates. (SHARED-40942/42085)
- Improvements made to **image file selection** functionality. (SHARED-42231/42451/42503/42556//43778)

### Simplified creation of templates based on existing PDFs

- Option added to allow the creation of a **new print document based on an existing PDF**. (SHARED-19220)

- Improved support for **adding PDF files as Section backgrounds**. Files can now be referenced from disk or imported into Template. (SHARED-42496)
- Added support for drag and dropping **Data Fields** directly onto the page as absolutely positioned textboxes. (SHARED-43311)

### Print Layout improvements

- **Page Number** formatting options (start/stop page numbering for sections, set numbering notation) improved in Print Section Properties dialog. (SHARED-39048)
- Added **repeating background images** support for print documents. (SHARED-43201)
- Option added to allow the insertion of **absolute positioned tables** on a master page. (SHARED-21967)

### Email enhancements

- **User-definable SMTP settings**. New defaults are added for Sendgrid and Mailgun (in addition to Mandrill). (SHARED-43897)
- The standard **New email** wizard has been replaced with the new **Basic Email template** wizard. The new wizard has improved HTML structure. (SHARED-43338)
- Sending a **test email** no longer requires data. (SHARED-41889)
- **Tighter compression** for PDF attachments that are based on a print section. (SHARED-38575)
- **Colour picker** support added to the Email template wizards. (SHARED-33561)
- Added support for **PNG barcode** images in email messages. (SHARED-43787)

### Barcode enhancements

- **Improved Barcode creation** with improved dialogs, better data validation and better error messaging. (SHARED-39295/42879)
- Font controls added to the **Barcode Properties** dialogs. (SHARED-22722/43659)
- Barcode improvements made in **Preview** mode. Support added for resizing and dragging of absolute positioned barcode objects, as well as resizing of inline barcode objects. (SHARED-43641)
- Barcodes can now have **transparent backgrounds**. (SHARED-43659)

## Scripting improvements

- New **closest()** command added to the Scripting API, to locate closest matching element above it in the Document Object Model (DOM) tree. (SHARED-41789)
- **Script editing improved.** Line numbering now available within the editor, support for code completion and syntax highlighting added, as well as support for various ECMA6 commands. (SHARED-42768/43696)
- Support added for **cloning Sections in a Control Script** to allow a document to have a dynamic number of Sections. (SHARED-43683)
- Improved **Scripts tool tip** warning and error messaging. (SHARED-42550/43758)  
Improvements include:
  - Better tailored error messages and warnings.
  - Icons added representing script type as well as showing the issue severity.
  - Duplicated problems now filtered out.
  - Several other minor improvements.
- Improved support for raw HTML within Designer scripting API commands. (SHARED-43075)

# Connect 1.5 Output Enhancements and Fixes

## General

- Improved **content creation** processing speed for templates featuring PDF backgrounds. (SHARED-44350)

## Email Output

- **Basic Email Action** wizard now made the default for new Email templates. (SHARED-43338)
- Support added for **user defined** SMTP/Email Service Provider (ESP) settings. (SHARED-43897)

## Print Output

- New option added, allowing printing to **Windows Printer Driver**. (SHARED-35536)
- Improvements made to **external sort** option in Job presets. Support added for using input/output file placeholders. (SHARED-40944)
- New **HCF** file added that supports “top down wrap around sequence marks”. (SHARED-42326)
- Use **PostScript Media** name values in the PostScript DSC comments, to improve subsequent searches. (SHARED-42826)
- Option added to allow **storing of job resources** on PostScript printer’s own storage medium. (SHARED-43467)
- **OMR marks improved**, with support added for Match Numbers (Match Code, MC). (SHARED-43589)
- A **Proof preview function** has been added to the Output Wizard, to display onscreen how the current print job would appear when printed. (SHARED-43885)
- **Imposition improvement**. Can now set specific starting position via new Offset option. (SHARED-44022)
- Minor glitches in **Booklet** and **Imposition** output addressed. (SHARED-44340/44430)

- IPDS output: Added **support for levels and metadata**. Future versions of the Print Manager will be able to search and restart based upon these values. (SHARED-44173)

# Connect 1.5 General Enhancements and Fixes

## Installer improvements

- Improvements made to installation robustness. The installer now copes better when encountered **permissions issues** during installation. (SHARED-43732/43737)
- The **Update Client** has been updated to 1.1.9 and has been included in both the Connect 1.5 and Workflow 8.5 installations. (SHARED-47065)

## Print Manager improvements

- **Removing all finished jobs** for a specific printer with a single step. (SHARED-42984)
- **Retry counters** added to interface. (SHARED-44772)
- Minor dialog **updates and improvements** throughout the Print Manager. (SHARED-30040/41975/42189/42804/43890/441113/44214)

# Known Issues

## Installation Paths with Multi-Byte Characters

When installing the Traditional Chinese or Japanese versions of Connect, if the user specifies an alternative installation path containing multi-byte/wide-char characters it can break some of the links to the Connect-related shortcuts in the Start Menu and cause an error to appear at the end of the installer. The workaround for the moment is to use the default installation path. The problem will be addressed in a later release.

## Switching Languages

Changing the language using the **Window>Preferences>Language Setting** menu option does not currently change all of the strings in the application to the selected language. This is a known issue and will be fixed in a later release.

In the meantime we offer the following workaround for anyone who needs to change the language:

1. Go to the .ini files for the Designer and Server Config:
  - C:\Program Files\Objectif Lune\OL Connect\Connect Designer\Designer.ini
  - C:\Program Files\Objectif Lune\OL Connect\Connect Server Configuration\ServerConfig.ini
2. Change the language parameter to the required one under Duser.language=en | es | de | fr | it | ja | pt | tw | zh

Only one of the above language tags should be selected. Once saved, Connect will appear in the selected language at next start-up.

## GoDaddy Certificates

When installing Connect offline, dialogs allow installing the GoDaddy certificates. Most users should use the default settings and click **Next**. In some cases, however, this may not work correctly. For this reason those users should activate **Place all certificates in the following store** and then select the **Trusted Root Certification Authorities** as the target certificate store.

## Available Printer Models

Note that only the single Printer Model (Generic PDF) will appear on the **Advanced** page of the **Print Wizard** by default.

To add additional printer models click on the settings  button next to the Model selection entry box.

## Colour Model in Stylesheets

The colour model of colours defined in a stylesheet can sometimes change after editing the stylesheet. This is a known issue and will be addressed in a subsequent release.

## Online Help Links Point to Introductory Page

Context sensitivity for the online help is not yet enabled in Connect. All links and F1 calls point to the introductory page, where you can Search on keywords to bring up Help pages relating to the topic.

Context sensitivity will be introduced in a subsequent release of Connect.

## Image Preview in Designer

If in the Windows Internet settings (**Connection Settings > LAN configuration**) a proxy is enabled, but "Bypass proxy settings for local addresses" is not checked, the image preview service, conversion service and live preview tab in the Designer will not work and exhibit the following issues:

- Images will be shows as 0 size boxes (no red 'X' is displayed)
- Live preview does not progress, and when re-activated reports "browsers is busy"

To fix the issue you must check the "Bypass proxy settings for local addresses" option.

## VIPP Output

Some templates set up with landscape orientation are being produced as portrait in VIPP. It can also sometimes be the case that text and images can be slightly displaced. These are known issues and will be addressed in a later release of Connect.

# Copyright Information

Copyright © 1994-2017 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, PReS and PrintShop Mail are registered trademarks of Objectif Lune Inc.

# Legal Notices and Acknowledgments

PrintShop Mail Connect, Copyright © 2017, Objectif Lune Inc. All rights reserved.

The license agreements for the associated open source third party components can be downloaded [here](#).

This application uses the following third party components:

- **Adobe PDF Library** which is either a registered trademark or trademark of Adobe Systems Incorporated in the United States and/or other countries.
- **Adobe XMP Core** Copyright © 1999 - 2010, Adobe Systems Incorporated. All rights reserved.
- **Eclipse Persistence Services Project (EclipseLink)**, Copyright © 2007, Eclipse Foundation, Inc. and its licensors. All rights reserved. This is distributed under the terms of the Eclipse Public License Version 1.0 and Eclipse Distribution License Version 1.0.
- **Fugue Icons** by [Yusuke Kamiyamane](#) which are distributed under the terms of the [Creative Commons Attribution 3.0 License](#).
- **Gecko** which is distributed under the terms of the Mozilla Public License Version 2.0. Information on obtaining Gecko can be found on the following page: [https://wiki.mozilla.org/Gecko:Getting\\_Started](https://wiki.mozilla.org/Gecko:Getting_Started)
- **Glassfish Java Mail** which is licensed under the terms of the Common Development and Distribution License (CDDL) Version 1.0. Information on how to download the Glassfish source can be obtained from here: <https://wikis.oracle.com/display/GlassFish/Java+EE+7+Maven+Coordinates>
- **Hamcrest Matchers** Copyright © 2000-2006, www.hamcrest.org. All rights reserved.
- **HyperSQL**, Copyright © 2001-2010, The HSQL Development Group. All rights reserved.
- **ICU4J 4.4.2** Copyright © 1995-2013 International Business Machines Corporation and others. All rights reserved.
- **J2V8** which is distributed under the terms of the Eclipse Public License Version 1.0. The source code for J2V8 can be obtained from the following location: <https://github.com/eclipsesource/j2v8>

- **Jacob Java Com Bridge** which is licensed under the terms of the GNU Lesser General Public License Version 2. The source code for this can be obtained from the following location: <http://sourceforge.net/projects/jacob-project/files/jacob-project/>
- **JavaCraft JSch** Copyright © 2002 - 2012 Atsuhiko Yamanaka, JCraft Inc. All rights reserved.
- **JavaSysMon** Copyright © 2009 ThoughtWorks, Inc. All rights reserved.
- **JavaX Mail** which is distributed under the terms of the Common Development and Distribution License (CDDL) Version 1.1. The source code for this can be obtained from the following location: <https://java.net/projects/javamail/downloads/directory/source>
- **Jersey** which is distributed under the terms of the Common Development and Distribution License (CDDL) Version 1.1. Information on how to obtain the source code can be found at the following location: <http://repo1.maven.org/maven2/org/glassfish/jersey/jersey-bom>
- **jersey-json-1.13** which is licensed under the terms of the Common Development and Distribution License (CDDL) Version 1.1. Information on how to obtain the source code can be found at the following location: <http://mvnrepository.com/artifact/com.sun.jersey/jersey-json/1.13-b01>
- **Jersey Multipart** which is distributed under the terms of the Common Development and Distribution License (CDDL) Version 1.1. Information on how to obtain the source code can be found at the following location: <http://repo1.maven.org/maven2/org/glassfish/jersey/jersey-bom>
- **JGoodies Forms, JGoodies Binding and JGoodies Looks**, Copyright © 2002-2013 JGoodies Software GmbH. All rights reserved.
- **JNA Version 3.5.1** which is distributed under the terms of the GNU Lesser General Public License Version 2.1. The source code for this can be obtained from the following location: <https://github.com/twall/jna/releases>
- **Junit** which is distributed under the terms of the Eclipse Public License Version 1.0. The source code for Junit can be obtained from the following location: <https://github.com/junit-team/junit/tree/master/src>
- **Mimepull** which is distributed under the terms of the Common Development and Distribution License (CDDL) Version 1.1. The source code for this can be obtained from the following location: <https://maven.java.net/content/repositories/releases/org/jvnet/mimepull/mimepull/>
- **Objectweb ASM**, Copyright © 2000-2011 INRIA, France Telecom. All rights reserved.

- **Relique CSV Driver** which is licensed under the terms of the Lesser General Public License Version 2.0. This can be obtained from the following location:  
<http://sourceforge.net/p/csvjdbc/code/ci/master/tree/>
- **Rhino 1.7R4 and 1.7.7.1** which are licensed under the terms of the Mozilla License Version 2.0. The source code for these can be obtained from the following location:  
[https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Rhino/Download\\_Rhino](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/Rhino/Download_Rhino)
- **Saxon** which is distributed under the terms of the Mozilla Public License Version 2.0. The source code for this can be obtained from the following location:  
<http://sourceforge.net/projects/saxon/files/Saxon-HE/9.6/>
- **Servlet API** developed by Sun as part of the Glassfish project and licensed under the terms of the Common Development and Distribution License (CDDL) Version 1.0. Information on how to download the Glassfish source (as part of Java EE platform) can be obtained from here:  
<https://wikis.oracle.com/display/GlassFish/Java+EE+7+Maven+Coordinates>
- **Spring Framework** which is distributed under the terms of the Apache Software License Version 2.0. This product includes subcomponents with separate copyright notices and license terms.
- **Springsource JavaX Mail** which is distributed under the terms of the Common Development and Distribution License (CDDL) Version 1.0. The source code for this can be obtained from the following location:  
<http://ebr.springsource.com/repository/app/bundle/version/detail?name=com.springsource.javax.mail&version=1.4.5&searchType=bundlesByName&searchQuery=mail>
- **Springsource SLF4J 1.6.1**, Copyright © 2004-2008 QOS.ch. All rights reserved.
- **Web Services Description Language for Java** which is distributed under the terms of the Common Public License v 1.0. The source code for this can be obtained from the following location: <http://wsdl4j.cvs.sourceforge.net/viewvc/wsdl4j/>
- **XULRunner** which is distributed under the terms of the Mozilla Public License Version 2.0. The source code for this can be obtained from the following location:  
<http://ftp.mozilla.org/pub/mozilla.org/xulrunner/releases/latest/source/>
- **zziplib** which is licensed under the terms of the Mozilla License Version 1.1. The source code for this can be obtained from the following location:  
<http://sourceforge.net/projects/zziplib/files/zziplib13/>
- **7-Zip SFX** which is licensed under the terms of the GNU Lesser General Public License Version 2.1. The source code for this can be obtained from the following location:  
[http://www.7zsfx.info/files/7zsd\\_src\\_160\\_2712.7z](http://www.7zsfx.info/files/7zsd_src_160_2712.7z)

Portions of certain libraries included in this application which are distributed under the terms of the Mozilla Public License have been modified. To obtain copies of the modified libraries please contact your local Objective Lune Support team.

This application also uses the following components which are distributed under the terms of the **Apache Software License Version 2.0**:

- Apache Ant
- Apache Axis
- Apache CFX
- Apache Commons Beanutils
- Apache Commons CLI
- Apache Commons Codec
- Apache Commons Collections
- Apache Commons Configuration
- Apache Commons DBCP
- Apache Commons Digester
- Apache Commons Discovery
- Apache Commons FileUpload
- Apache Commons Imaging
- Apache Commons IO
- Apache Commons Lang
- Apache Commons Logging
- Apache Commons Net
- Apache Commons Pool
- Apache Commons Validator
- Apache Commons VFS
- Apache Derby
- Apache Felix and dependencies
- Apache Geronimo
- Apache Jakarta HttpClient
- Apache Log4j
- Apache Neethi

- Apache OpenCMIS
- Apache POI
- Apache ServiceMix
- Apache Tomcat
- Apache WSS4J
- Apache Xalan
- Apache Xerces2 Java Parser
- Apache XMLGraphics
- Apache XML-RPC
- Barcode4j
- Google Collections
- Google GSON
- Jetty
- LMAX Disruptor
- OPS4J Pax Web
- org.json.simple
- Spring Dynamic Modules
- StAX
- XMLBeans

### **Eclipse Technology:**

This Software includes unmodified Eclipse redistributables, which are available at [www.eclipse.org](http://www.eclipse.org). The Eclipse redistributables are distributed under the terms of the Eclipse Public License - v 1.0 that can be found at <https://www.eclipse.org/legal/epl-v10.html>.

### **VSS Java FreeMarker:**

This product includes software developed by the Visigoth Software Society (<http://www.visigoths.org/>).

This includes the following subcomponents that are licensed by the Apache Software Foundation under the Apache License, Version 2.0:

- freemarker/ext/jsp/web-app\_2\_2.dtd
- freemarker/ext/jsp/web-app\_2\_3.dtd
- freemarker/ext/jsp/web-app\_2\_4.xsd
- freemarker/ext/jsp/web-app\_2\_5.xsd
- freemarker/ext/jsp/web-jsptaglibrary\_1\_1.dtd
- freemarker/ext/jsp/web-jsptaglibrary\_1\_2.dtd
- freemarker/ext/jsp/web-jsptaglibrary\_2\_0.xsd
- freemarker/ext/jsp/web-jsptaglibrary\_2\_1.xsd

### **Java SE framework and platform:**

This application uses the Java SE framework and platform which is distributed under the terms of the Oracle Binary Code License Agreement for the Java SE Platform Products and Java FX. Copyright 2013, Oracle America, Inc. All rights reserved.

Use is subject to license terms. ORACLE and JAVA trademarks and all ORACLE- and JAVA-related trademarks, service marks, logos and other brand designations are trademarks or registered trademarks of Oracle in the U.S. and other countries.

*Use of the Commercial Features for any commercial or production purpose requires a separate license from Oracle. "Commercial Features" means those features identified Table 1-1 (Commercial Features In Java SE Product Editions) of the Java SE documentation accessible at <http://www.oracle.com/technetwork/java/javase/documentation/index.html>.*

### **Further Components:**

- This product includes software developed by the **JDOM Project** (<http://www.jdom.org/>).
- Portions of this software are copyright © 2010 **The FreeType Project** ([www.freetype.org](http://www.freetype.org/)). All rights reserved.
- This product includes software developed by **JSON.org** (<http://www.json.org/java/index.html>).

[Click to download the EULA as PDF](#)