opentext™

OpenText™ Exstream™

Importing External Content

Design and Production Documentation

Release 16.6.0

OpenText™ Exstream
Importing External Content
Rev.: 2019-Apr-30

This documentation has been created for software version 16.6.0.

It is also valid for subsequent software versions as long as no new document version is shipped with the product or is published at https://knowledge.opentext.com.

Open Text Corporation
275 Frank Tompa Drive, Waterloo, Ontario, Canada, N2L 0A1

Tel: +1-519-888-7111
Toll Free Canada/USA: 1-800-499-6544 International: +800-4996-5440
Fax: +1-519-888-0677

Support: https://support.opentext.com
For more information, visit https://www.opentext.com

Contents

Chapter 1: Leveraging External Content in Designs

To help you quickly and efficiently create customer content in Exstream, you can import into your design a wide variety of content, including text, images, PDFs, and more. These resources are called "external content" because they are generated by programs external to Exstream. To utilize external content in a design, you can choose from various leveraging methods which help you to import the content you want and to control how the content is incorporated into the final customer output.

Most organizations have already invested in these various forms of external content. The process of importing external content can help reduce the need to recreate content that already exists. You can also continue to utilize these resources from many of the current content management systems which are already in place in your organization, without the need to move or migrate content to make it more accessible for the design.

For more information about supported locations where external content can be stored, see “Supported Storage Locations for External Content” on page 19.

This chapter discusses the following topics:

1.1 External Content Planning Considerations

When you import external content, Exstream provides you with the flexibility to import the files directly into Designer or to set up design placeholders so that the external files are placed into the customer content during production. Your choice for when to import the content depends on the file types you want to import, your requirements for memory usage and performance during production, and the rules your business has implemented for the content. The following table

provides some questions to consider when planning whether to import content at design time or at run time.

Planning considerations when importing content

Questions to ask	Why you need to know this information
What is the format of the content you want to import?	Not all supported image or text files formats are supported for both design-time and run-time import. In addition, some formats require certain design considerations when using specific output drivers. Review the list of supported formats to see which timing and design options are available for the format you plan to import. For information on supported image formats and specifications, see "File Formats Supported for Importing" on page 11.
Is the content static or does it change periodically? If the content changes, how often does it change? Is the rate of the changes more or less frequent than the rate of change for the over all design of the document?	The rate of change for the content you want to import is one of the most important considerations when deciding on a design approach. Consider the frequency and the amount of time it would take to update a design. Use design-time import for content that has never changed or very rarely changes. Run-time import might be a better choice if you are importing content that changes frequently or without advance notice. Run-time import is also a good choice when you want to import multiple objects at the same time into the same location. For more information about design-time import, see "Importing Content into a Design at Design Time" on page 21. For more information about run-time import, see "Importing Content into a Design at Run Time" on page 42.
Do you want to format the content or modify the content in any way from the original? Is it important that the content not be changed or that it maintain fidelity with the original? For images, is it important to control the placement, size, resolution, or color of the image? For text, is it important to control the formatting of the text?	Not all formatting options are supported for both design-time and run-time import. If you set up the design to import the content at run time, the ways in which you can change the imported content are more limited and the options you have for formatting are more complex to set up than if you import the content at design time. For certain types of content, such as PDF or TIFF, you might want to avoid changing the content when importing at run time. This ability helps maintain the original image fidelity or resolution. For more information about formatting content at design time, see "Formatting Images Imported at Design Time" on page 30 and "Formatting Text Imported at Design Time" on page 38. For more information about formatting content at run time, see "Formatting Content That Will Be Imported at Run Time" on page 59.
Is it important to maintain consistency between the content in the design and the original source file?	Based on your business needs, it might or might not be necessary to keep the content imported into your design in synchronization with the source files that are stored externally to Exstream. For example, if you are importing a logo, many companies have restrictions on what changes can be made visually to their corporate logo. However, if you are using stock photos provided by your company to add visual interest to a design, it might not be as necessary that the image look exactly like the original. Content imported at design time becomes disconnected from the original source. When the source files change, you must update your design if the content must remain in synchronization with the original source files. However, if you set up the design to import the content at run time, then your design does not require an update when the source files change (as long as the file names and paths do not change). For more information about design-time import, see "Importing Content into a Design at Design Time" on page 21. For more information about run-time import, see "Importing Content into a Design at Run Time" on page 42.

Planning considerations when importing content, continued

Questions to ask	Why you need to know this information
Does the content consist of or contain more than one page?	If you are importing content that appears on more than one page per customer, you must import the content at run time. When you import content this way, the content is static and cannot have any formatting applied in the design. For multi-page image files, such as PDF or TIFF, you must use a placeholder variable to import the image into a placeholder document at run time. For text imported at run time that flows beyond one page, you must use a placeholder variable directly on the page, and then set up the text box or table cell for flow. For information on importing content onto more than one page at run time, see "Creating a Placeholder Variable" on page 45.

1.2 Methods Used to Leverage External Content into a Design

With Exstream, you can leverage external content into Exstream designs without investing overhead to redesign the content or investing in new storage locations. You can leverage content into your design that is stored externally from the design database, such as content stored locally on your computer, over a network file server, or on a web server. From Exstream, you can bring external content into your design in one of the following ways:

- **Content import**—Quickly migrate and use content directly from the external resource. These methods are commonly used when importing a full resource (such as an entire image, text, or PDF file).
- **Content reuse**—Convert content so that individual portions from within an external resource can be reused as an Exstream object. These methods are commonly used when you want to use only a portion of a resource, such as a table within an existing PDF, or when you want to extract form data from customers to automatically pre-populate future documents with the customer's data to save time.

1.2.1 Content Import Methods

Content import methods allow you to import content directly from an external resource into a design. These methods allow you to place external content within a design so that it appears in customer output with minimal differences from the original resource. Content import methods are useful if you want to incorporate images and text exactly as they appear from the external resource, or incorporate entire documents (such as full brochures, contracts, or forms) within a design. This guide discusses the following content import methods:

Content import methods

Content import method	When to use this method
Design-time import	When you import content at design time, you place the content directly into the design on a message or page within Designer. You can then change the formatting properties of the content independently of the source file. For example, if you want to include a logo for your company's letterhead, you can import the logo at design time and resize the logo to fit the design. This method is useful in the following situations: • The content rarely or never changes. • It not important that the content stays in synchronization with the original source file. • You want to change the content from the original source file (for example, change the size, resolution, or other formatting). • You are importing an image that is one page or smaller in size.
Run-time import	When you import content at run time, you create a placeholder for the content within the design. At engine run time, the referenced content is placed into the output, with no changes. For example, you can import images at run time if you want to include check images in bank statements or seasonal banners on customer newsletters. This method is useful in the following situations: • The content changes more regularly than you want to update the design. • The content must stay in synchronization with the original source file or must not be changed (for example, standardized or regulated text). • You are importing an image file that includes multiple pages (for example, multiple-page PDF files). • You are importing large amounts of content or multiple-page resources (such as PDF files) into the document and do not want to repackage the application to include the updated content. • You want to reduce the size of your design database and/or your package file. Since run-time content is referenced from your external storage location, the content is not stored in the design database. This division of storage reduces the size of the package file. • You want to leverage content stored in enterprise systems to which Exstream does not natively speak. You must use Dynamic Data Access (DDA) to integrate Exstream into your existing infrastructure. For more information about using DDAs, see *Configuring Connectors* in the Exstream Design and Production documentation.

1.2.2 Content Reuse Methods

Content reuse methods let you convert external resources into a format so that content within the external resource can be reused as functional design objects in Exstream. For example, you can convert the objects within a PDF into Exstream design objects so that you can utilize Exstream design options such as automated table settings or variable use within existing text. You can use any of the following reuse methods, which are discussed elsewhere in the Exstream documentation set:

Content reuse methods

Content reuse method	When to use this method
Convert designs created in other design programs into the Exstream Exchange Format (DXF)	When you want to reuse objects from other design programs, you can convert some designs so that each object becomes an editable and reusable Exstream design object. For example, you could convert a statement PDF so that you could reuse the statement tables as Exstream table objects, complete with automation capabilities. For more information on Exstream conversion tools, see *Importing Designs* in the Exstream Design and Production documentation.
Pre-fill data fields with common data in PDF Extensible Forms Architecture (XFA) forms	When you pre-fill a PDF form, you leverage customer information available in other systems or data sources to fill in the form's fields automatically. This method is useful in the following situations: • You want to send customers PDF forms that already include the customer information you have in your systems, such as names, account numbers, and addresses. • You want to reduce errors in the data and the amount of time required to complete the remaining information on the form. For more information about pre-filling data fields, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.
Mine customer data from PDF XFA forms	When you mine a PDF form, you leverage customer information in a form and use it as input to any Exstream application. This method is useful in the following situations: • You want to use customer-provided information to personalize a document, generate a file, or update a system. • You want to reduce errors in the data and the amount of time required to process data and respond to the receipt of customer forms. For more information about mining data from PDF XFA forms, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

1.3 Modules Required for Leveraging External Content

The Exstream modules related to external content provide a flexible architecture that lets you integrate content from a variety of sources. Exstream can dynamically access and import text, images, logos, and more from content management systems. The external content modules provide many options for PDF integration.

The following Exstream modules are related to the integration of external content:

Modules required to import external content

You must have licensed this module	To do this
Dynamic Content Import module	Import any external image or text files into documents at run time. For more information about including content at run time, see "Importing Content into a Design at Run Time" on page 42. To use the Dynamic Content Import module to import content that is not natively supported by the output format that you have specified, you must install Ghostscript 9.0 or later. Ghostscript is not supported on the z/OS platform. For information on the requirements for Ghostscript, see *Installation and Upgrade Information* in the Exstream Design and Production documentation.
PDF Import as Image module	Include PDF content at run time in designs that will be sent to output drivers that do not natively support the PDF format. For more information about importing PDF content for use in output drivers that do not natively support the PDF format, see "Supporting PDF Content When Output Drivers Do Not Natively Support PDF" on page 120. **Note:** The PDF Import as Image module is not supported on the z/OS platform. To use the PDF Import as Image module, Ghostscript 9.0 or later is required. Ghostscript is not supported on the z/OS platform.
Design PDF module	Import a PDF as images into a design at design time for use as background images and design layers. For example, you can use a correspondence PDF to plan the layout and placement of customer name and address variables in a design. For more information about importing a PDF into a design at design time, see "Importing Content into a Design at Design Time" on page 21. To use the Design PDF module, Ghostscript 9.0 or later is required. Ghostscript is not supported on the z/OS platform.

Ghostscript is a commercially available PostScript and PDF conversion and rendering tool. For information about installing Ghostscript, see *Installation and Upgrade Information* in the Exstream Design and Production documentation.

1.4 File Formats Supported for Importing

The following table provides an overview of the external file formats that can be imported into Exstream designs:

Supported external file formats

File format	File extension	Design time	Run time	Limitations/Requirements
Microsoft Fax	*.awd	Yes	No	—
Windows Bitmap	*.bmp	Yes	No	—
Computer Aided Acquisition and Logistics Support Raster	*.cal	Yes	No	—
Computer Graphics Metafile	*.cgm	Yes	No	—
Windows Cursors	*.cur	Yes	No	—
DICOM Bitmap	*.dic	Yes	No	—
Word document	*.docx	No	Yes	• Exstream supports the Word document (DOCX) format only. You cannot use this option to import documents with the Word 97–2003 Document format (DOC). **Tip:** To update a Word 97–2003 document (DOC) to the DOCX format, open it in Word 2007 or later and resave it. • DOCX import at run time is not supported on z/OS. For more information about importing DOCX files into Exstream, see "DOCX Considerations" on page 78.
Exstream Live Live document	*.dlf	No	Yes	For more information about Live documents and the DLF format, see *Designing for LiveEditor* in the Exstream Design and Production documentation.

Supported external file formats, continued

File format	File extension	Design time	Run time	Limitations/Requirements
Encapsulated PostScript	*.eps	Yes	Yes	• When importing at design time, you must have licensed the Design PDF module. • EPS files must contain `%%BoundingBox`. • If you are importing EPS images at run time, keep in mind that some output types do not support EPS images. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on page 17. • When using outputs types that do not support vector data, the engine rasterizes each EPS image in the output. • For certain output types that support vector image data but not the EPS format, the engine converts the EPS data to supported vector data. For more information about importing EPS files into Exstream, see "EPS Considerations" on page 84.
Raw FAX	*.fax	Yes	No	—
Graphic Interchange Format	*.gif	Yes	Yes	Run-time import of GIF is supported only for HTML import with the following output types: • DOCX • HTML • HTML (email) • Multi-Channel XML Exstream supports LZW compression.
Hypertext Markup Language (HTML)	*.html	No	Yes	When importing at run time, you must have licensed the Dynamic Content Import module. HTML import is supported only on Windows and Linux. For information about supported tags and design considerations for imported HTML files, see "HTML Considerations" on page 85.
AFP IOCA	*.ica	Yes	No	—
Windows Icons	*.ico	Yes	No	—
Interchange File Format	*.iff	Yes	No	—
GEM Paint Image	*.img	Yes	No	—

Supported external file formats, continued

File format	File extension	Design time	Run time	Limitations/Requirements
Intergraph B&W	*.itg	Yes	No	—
Joint Bilevel Image Group	*.jbg	Yes	No	—
Joint Photographic Experts Group 2000	*.jp2	Yes	No	Supports LZW compression
Joint Photographic Experts Group	*.jpg *.iff *.itf	Yes	Yes	• JPG files must contain at least 256 colors. • JPG files cannot be a wrapper for another image type. They must be true JPG files. • Progressive CMYK JPG images are not supported for IJPDS and MIBF outputs. • Run-time import is dependent on the output driver used. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on page 17.
MacPaint	*.mac	Yes	No	—
Microsoft Paint	*.msp	Yes	No	—
Portable Bitmap Utilities	*.pbm	Yes	No	—
Kodak Photo CD	*.pcd	Yes	No	—
Mac Picture	*.pct	Yes	No	—
ZSoft Paint	*.pcx	Yes	No	—

Supported external file formats, continued

File format	File extension	Design time	Run time	Limitations/Requirements
Portable Document Format	*.pdf	Yes	Yes	• Supports single-page or multiple-page images • When importing at design time, you must have licensed the Design PDF module. • Some output drivers do not natively support PDF content; however, you can set up the engine to automatically convert PDF content into an image format that will be supported on these output drivers by licensing the PDF Import as Image module. For additional information about supporting PDF content on output drivers that do not natively support PDF content, see "Supporting PDF Content When Output Drivers Do Not Natively Support PDF" on page 120. • Run-time import is dependent on the output driver used. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on page 17.
Compuserve PNG	*.png	Yes	No	—

Supported external file formats, continued

File format	File extension	Design time	Run time	Limitations/Requirements
Portable Network Graphics	*.png	Yes	Yes	• If you are importing PNG images at design time, keep in mind that only certain output types support transparency in PNG images. For more information about transparency support in PNG images imported at design time, see "Importing Images With Transparency at Design Time" on page 28. • If you are importing PNG images at run time, keep in mind that some output types do not support PNG images. However, transparency is supported in all the output types that support run-time import of PNG images. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on page 17. For more information about transparency support in PNG images imported at run time, see "Importing Images with Transparency at Run Time" on page 57. • If a PNG image imported at run time has a native resolution that is different from the output resolution, the PNG image will appear at a different size in the output to match the output resolution. If you want to ignore the native resolution of the image and process it as if it matches the output resolution, use the PNG_IGNORE_RESOLUTION engine switch. For more information about using the PNG_IGNORE_RESOLUTION engine switch, see *Switch Reference* in the Exstream Design and Production documentation.
Adobe PhotoShop	*.psd	Yes	No	—
AFP PSEG	*.pseg	Yes	Yes	Run-time import is dependent on the output driver used. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on page 17.
SUN Raster	*.ras	Yes	No	—
Rich Text Format	*.rtf	Yes	Yes	For more information about importing RTF files, see "RTF Considerations" on page 126.
Scalable Vector Graphic	*.svg	No	Yes	SVG images are supported only for import with HTML5 output.
Truevision File Format	*.tga	Yes	No	—

Supported external file formats, continued

File format	File extension	Design time	Run time	Limitations/Requirements

Tagged Image File Format	*.tif	Yes	Yes	• Run-time import is dependent on the output driver that you use.
				• Supports single-page or multiple-page images
				• TIFFs using JPG compression are not supported.
				• Most TIFF file formats from 1 to 24 bits per spot in color, grayscale, or black and white, are supported in uncompressed, CCITT Group 4 compression, and Lempel-Ziv-Welch (LZW) compression, except for the following cases:
				  • IJPDS and MIBF output support only true CMYK uncompressed files.
				  • LZW-compressed TIFF files support only color or grayscale static images.
				• When importing TIFF files to IJPDS and MIBF outputs, the engine uses the Floyd-Steinberg method for dithering.
				• When importing TIFF files that contain more than one progressive strip of the image, use an uncompressed version of the file.
				For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on the next page.

Plain text	*.txt	Yes	Yes	—
Windows Metafile	*.wmf	Yes	No	—
WordPerfect Metafile	*.wpg	Yes	No	—
XbitMap	*.xbm	Yes	No	—
XWindowDump	*.xwd	Yes	No	Black and white only
AFP FSXX (FS10, FS11, FS45)	—	FS45 only	Yes	Run-time import is dependent on the output driver used. For more information about output driver restrictions on the types of images you can import at run time, see "Output Driver Considerations for Run-Time Import" on the next page.
ASCII	—	No	Yes	—

1.5 Output Driver Considerations for Run-Time Import

If you import content at run time, some output drivers have specific considerations for which file formats you can use. The following formats have limited support when imported at run time:

- EPS

  > **Note:** Depending on the output that you use, EPS vector data might be used directly in the output, converted to another vector format, or rasterized.
  >
  > For more information about importing EPS files into Exstream, see "EPS Considerations" on page 84.

- FSXX
- PSEG
- JPEG (color)
- PDF
- PNG
- SVG
- TIFF

If you select the **Image each page** option on the **Basic** tab of the output object in Design Manager, then the TIFF output driver is used to create the pages as images create the pages as images, and you can only import formats at run time that are supported in TIFF output.

For more information about creating pages as images, see *Creating Output* in the Exstream Design and Production documentation.

If you want to dynamically import color TIFF files, you must enable AFP color management.

For more information about AFP color management, see *Creating Output* in the Exstream Design and Production documentation.

If you want to import images with transparency, they must be in PNG or SVG format. All of the output types that support run-time import of PNG or SVG images also support any included transparency data. Additional considerations apply when importing PNG or SVG images with transparency.

If you use one of the formats with limited support, review the following table to ensure that the file is supported by the output driver you use:

Support for files imported to specific output drivers at run time

Output driver	EPS	FS45, FS10, FS11, PSEG	JPEG (Color)	PDF	PNG	SVG	TIFF
3211 Line Data	No	No	No	No	No	No	No
AFP	Yes	Yes	Yes	Yes	Yes	No	Yes
DOCX	Yes	No	Yes	Yes	Yes	No	Yes
EDGAR HTML	Yes	No	Yes	No	Yes	No	No
HTML	Yes	No	Yes	Yes	Yes	Yes	No
HTML (email)	Yes	No	Yes	Yes	Yes	No	No
IJPDS	Yes	No	Yes (CMYK non-progressive only)	Yes	No	No	Yes
Metacode	Yes	No	No	Yes	No	No	Yes (B&W only)
MIBF	Yes	No	Yes (CMYK non-progressive only)	Yes	No	No	Yes
PCL	Yes	No	No	Yes	No	No	Yes (B&W only)
PDF	Yes	No	Yes	Yes	Yes	No	Yes
PDF/A	Yes	No	Yes	Yes	Yes	No	Yes
PDF/VT	Yes	No	Yes	Yes	Yes	No	Yes
PostScript	Yes	No	Yes	Yes	Yes	No	Yes
PowerPoint	Yes	No	Yes	Yes	Yes	No	Yes
PPML	Yes	No	Yes	Yes	Yes	No	Yes
RTF	Yes	No	Yes	Yes	Yes	No	No
TIFF	Yes	No	No	Yes	No	No	Yes (B&W only)
TOP	Yes	No	Yes	Yes	Yes	No	Yes

Support for files imported to specific output drivers at run time, continued

Output driver	EPS	FS45, FS10, FS11, PSEG	JPEG (Color)	PDF	PNG	SVG	TIFF
VDX	Yes	No	Yes	Yes	Yes	No	Yes
VIPP	Yes	No	Yes	Yes	Yes	No	Yes
VPS	Yes	No	Yes	Yes	Yes	No	Yes
XML (Composed)	Yes	No	Yes	Yes	Yes	No	Yes
XML (Content)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
XML (Multi-Channel)	Yes	No	Yes	Yes	Yes	No	No
ZPL	Yes	No	No	Yes	No	No	Yes (B&W only)

1.6 Supported Storage Locations for External Content

When leveraging external content, you can use content stored within multiple systems or repositories. This capability of Exstream for accessing and pulling content from a variety of sources allows you to keep files where you would normally store them, and import the content as needed into Exstream designs. You can import content stored on the following locations:

Supported external content storage locations

Location	Description
Local drives	Allows you to use content from the drives and folders on a local computer
Network locations	Allows you to use content from computers or repositories which are networked with a local computer
HTTP repositories	Allows you to use content from an HTTP repository. You can use content directly from HTTP repositories only at run time. **Tip:** If you need HTTP repository files at design time, you can copy the content to your desktop or map the HTTP address as a network location. HTTP repositories are not supported in z/OS.

Supported external content storage locations, continued

Location	Description
Printer resources	Allows you to use content from the print stream stored on specific printers For more information about support available for printer resources, see "Print Resource Considerations" on page 125.

Chapter 2: Importing Content into a Design at Design Time

When you import content at design time, you place the content directly into the design as you are creating customer content in Exstream. You can then change the formatting properties of the content independently of the source file. For example, if you want to include a logo for your company's letterhead, you can import the logo at design time and resize the logo to fit the layout of the design.

Importing content at design time is useful if you are importing content that rarely or never changes, or if it is not important that the content stays in synchronization with the original source file. Importing content directly into a design is also useful if you want to apply formatting (such as changing the size, color, or resolution of an image) to the content that is imported.

To import content at design time, complete the following tasks as needed:

- "Importing an Image at Design Time" below
- "Importing Text at Design Time" on page 37
- "Importing a PDF File at Design Time" on page 39

2.1 Importing an Image at Design Time

When you import images at design time, you can place and arrange images as needed in the design and complete any additional content arrangement to ensure that other content flows correctly around the imported image.

You can add an image to a design at design time using the following methods:

Methods for importing images at design time

To	Complete this task
Place an image directly into the design	"Inserting an image into a design" below
Collect a group of images that will share the same location in a design and then use a selection variable to specify the criteria for determining which image is placed in the customer output	"Adding an Image Selector to a Design" on page 26

2.1.1 Inserting an image into a design

You can place an image directly into the design in Designer from a local file system (including any mapped network drives) or from a connected repository. When you add an image to the

design and save it, the image is stored in the design database in its original format and with its original settings. You can move, resize, and adjust the color of images just as you would any other design object.

For information about working with design objects, see *Designing Customer Communications* in the Exstream Design and Production documentation.

To insert an image into a design:

1. Open a design in Designer.
2. From the **Drawing Objects** toolbar, click  .
3. Depending on whether you have an image repository configured and connected to Exstream, complete one of the following sets of steps on the **Import an Image** dialog box:

To	Do this
Insert an image from your local file system (if you do not have a repository set up)	a. Select the image you want to place in the design. b. Click **Open**.
Insert an image from your local file system (if you have a repository set up)	a. Click  . b. Select the image that you want to place in the design. c. Click **Open**.
Insert an image from the common asset service (CAS) or in another external repository connected through the CAS (such as OpenText Media Management) **Note:** In order to import images from the CAS repository, you must have connection settings for the management gateway and Resource Browser configured in the **System Settings** in Design Manager. For more information about configuring connection settings for the management gateway and the Resource Browser, see *System Administration* in the Exstream Design and Production documentation or contact your system administrator.	a. Click  . b. From the Resource Browser, select the image you want to place in the design. c. Click **Select**. **Tip:** Shared images that are inserted from the CAS repository are subject to change, so Designer notifies you of available changes and allows you to update to the latest version of the image either manually or automatically. For more information about updating images inserted from CAS, see "Updating an image inserted from the Common Asset Service" on the next page.

4. To set the resolution of the image as it appears in the design, select one of the following options in the **Change resolution** list:

To	Do this
Maintain the original resolution settings of the image being imported	Select **Keep as is**.
Convert the image to a specific resolution	a. Select **Convert to specified**. b. In the **New resolution** box, enter a new resolution for the image.
Convert the image to the default design resolution specified for the design environment	Select **Convert to design resolution**.

Tip: For best results, images should be the same resolution as, or a multiple of, the resolution of the output.

5. To specify the number of bits used to save each pixel and the number of possible colors in the image, select the appropriate option in the **Color** list.
6. On the **Import an Image** dialog box, update the resolution or color settings (optional) and click **OK** twice.

Updating an image inserted from the Common Asset Service

If your design includes shared images that are inserted from the common asset service (CAS) or another external repository connected through the CAS (such as OpenText Media Management), Designer notifies you of available image updates, allowing you to choose whether or not to update an image. By default, when you insert an image from the CAS, the image is not automatically updated when a new version is available. However, you can set images to update to the latest version automatically in Designer.

This section discusses the following topics:

Checking for image updates in your design

When you open or refresh (F5) your design, you will be notified of available updates in the following ways:

- In the design window, you will see the  icon on the image.
- In the Container Viewer or the Outline Viewer, you will see the  icon next to the image.

  Tip: Click  at the top of the viewer to show only images that come from CAS.

- On the Status Bar and in the session log, you will see a notification of the number of image updates that are available.

Note: If you see an error message in the status bar that indicates Designer is unable to check for image updates, you can click the status bar to review the session log for more details. The most likely cause of this error is an issue with your connection to the management gateway.

To receive notification of image updates available in the CAS repository, you must have the connection settings for the management gateway and Resource Browser configured in the **System Settings** in Design Manager. For more information about configuring connection settings for the management gateway, see *System Administration* in the Exstream Design and Production documentation or contact your system administrator.

Checking for image updates at the application level

If you want to check for updates to CAS images at the application level, as opposed to the page, message, or template level, you can do so in Design Manager by running an application report.

For more information about running an application report, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Checking for image updates during packaging

If you want to check for updates to CAS images during packaging, you can do one of the following:

- If you are packaging the application in Design Manager, on the **Build Package File** dialog box, select the **Check CAS for image updates** check box.

For more information about creating a package file in Design Manager, see *Preparing Applications for Production* in the Exstream Design and Production documentation.

- If you are packaging from the command line, or using a control or batch file to run `packager.exe`, use the CHECK_CAS_IMG_UPDATES switch. For more information about using packaging switches, see *Switch Reference* in the Exstream Design and Production documentation.

Updating images manually

To update an image in your design:

1. To begin the update process, do one of the following:
   - Select the image in the design window and click  .
   - Right-click the image in the design window, the Container Viewer, or the Outline viewer, and select **Update image**.
2. On the **Import an Image** dialog box, update the resolution or color settings (optional) and click **OK** twice.
3. In the **Preview Image Update** dialog box, you can see the current version of the image in your design side-by-side with the new version of the image that is available in CAS. Click **Accept** to update the image.

Note: If you have resized the image or made any modifications in the **Image Color Management** dialog box, these changes will be lost when you update the image.

Updating images automatically

You can also have Designer check the CAS repository for image updates when you open your design, and then automatically replace an image with the latest available version. You must individually set each image you want to automatically update. If you are using the same image multiple times in your design, you must update each instance of the image individually.

To set a CAS image to automatically update:

1. In Designer, right-click the image and select **Image Properties**.
2. On the **Image** tab, in the **Automatically update image from repository** list, select **Always update to the latest version**.

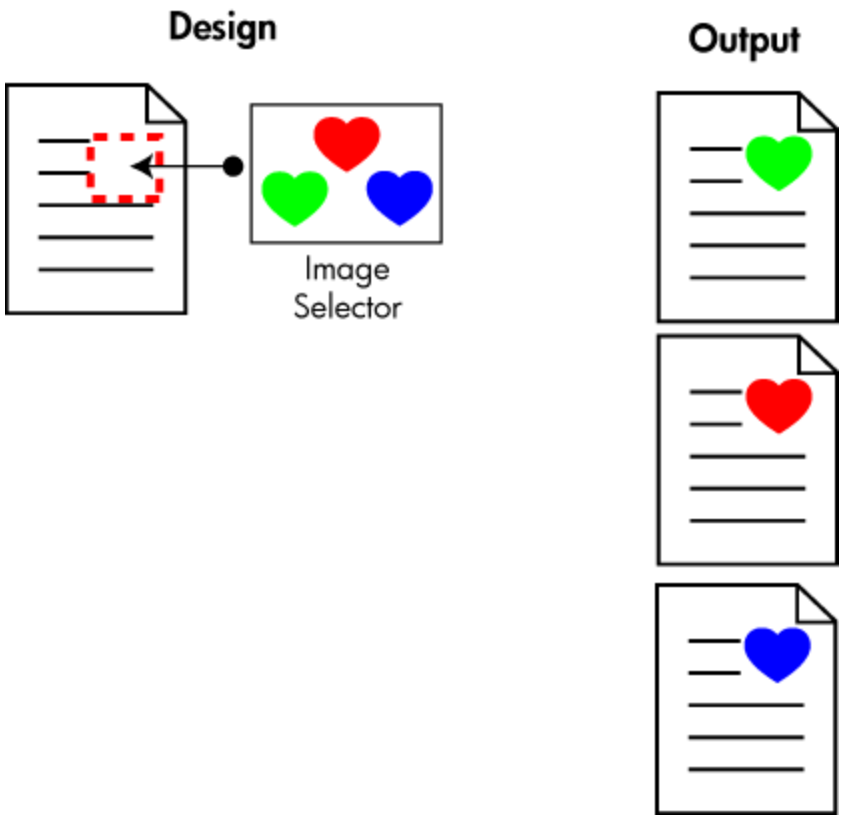
Note: If you have resized the image or made any modifications in the **Image Color Management** dialog box, these changes will be lost when you update the image.

2.1.2 Adding an Image Selector to a Design

An image selector is a feature that lets you define a set of possible images that the engine can import to a specific location in a document. When you use an image selector, a image selector object is associated with a selection variable. You specify selection criteria for each image included in the image selector based on matching values of a selection variable, which is typically a string or integer. At run time, the engine places the correct image for each customer based on matching the variable value. If the variable value does not match any of the selection criteria, then no image is included in the output. You can use the image selector feature instead of using inclusion rules to control the inclusion of separate images that would occupy the same location in the design.

For example, if you want to include an image of the mascot for a local sports team on a company newsletter going out to customers in New York, Chicago, and Los Angeles, you can set up an image selector that includes the three different images for the mascots. The selection criteria assigned to the images would be `NY`, `IL`, and `LA`, respectively, with `NY` set as the default image. In this example, the selection variable is a user-defined variable with values that represent the state abbreviation for the customers. As long as the variable value for the customer is `NY`, `IL`, or `LA`, then one of the mascots will be placed in the newsletter. If the variable value is anything else, such as `OH` or `CA`, then the default image will be placed in the newsletter.

Sample of design and output when using an image selector



Before you complete this task, you must have created a selection variable to control the image selection.

For more information on creating variables, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

To add an image selector to a design:

1. Open a design in Designer
2. Add a single image to the design.

   For information on importing an image, see "Inserting an image into a design" on page 21.
3. Right-click the image and select **Image Properties**.

   The **Image Properties** dialog box opens.
4. Click the **Image** tab.
5. In the **Selection variable** box, click  and select a variable that stores the value of selected images.

> **Tip:** You can use this variable in a formula to drive other content on the page.

6. Below the **Image selections** box, click  to add additional images to the image selector. You can add as many images as you want to the image selector.

> **Tip:** For best results in your output, all images used in the image selector should be the same size and file type.

7. Complete any of the following optional tasks as needed:

To	Do this
Drive other processes based on the selection of a specific image	a. In the **Image selections** box, select an image. b. In the **Selection value** box, enter the variable value that should be used for the engine to select the image. The selection value you enter can then be referenced in formulas or other logic.
Provide a description of an image to help end users in interactive documents select the correct image	a. In the **Image selections** box, select an image. b. In the **Caption** box, enter descriptive information for the selected image. For more information about creating content for end users in Live, see *Designing for LiveEditor* in the Exstream Design and Production documentation.
Select an image to placed in the design when the variable value does not match any of the selection criteria	a. In the **Image selections** box, select an image. b. Select the **Default selection** check box.

8. Click **OK**.

   The **Image Properties** dialog box closes and the default image appears on the design page.

2.1.3 Importing Images With Transparency at Design Time

If you want to import images that include transparency at design time, Exstream supports PNG images that include transparency (including both palette-based and alpha channel). For example, if you have a logo with a transparent background that you want to appear over other objects in a design, you can make sure that the image is in the PNG format and import it into your design as you would any other image. Any transparency in an image appears in Designer, as well as in output types that support transparency in PNG images imported at design time.

The following output types support transparency in PNG images imported at design time:

- AFP
- DOCX
- EDGAR HTML
- HTML
- HTML (email)
- PDF
- PDF/A
- PDF/VT
- PostScript
- PowerPoint
- RTF
- TOP
- VDX
- VIPP
- VPS
- XML (Composed)
- XML (Multi-Channel)

For general information about support for importing PNG and other file formats, see "File Formats Supported for Importing" on page 11.

Keep in mind the following considerations:

- If you are including transparent images for use in PDF, PDF/A, or VDX output, you must place images at the top of the print stream or the transparency will be lost. To do this, open the output object in the Property Panel. On the **Resource Management** tab, in the **Image processing** list, select **Place at top of print stream**.

  For more information about specifying where to place static images in the print stream, see *Creating Output* in the Exstream Design and Production documentation.

- If you select the **Image each page** option on the **Basic** tab of the output object in Design Manager, then the TIFF output driver is used, and PNG images are not supported.

  For more information about creating pages as images, see *Creating Output* in the Exstream Design and Production documentation.

- If you are using the PostScript output driver, you must select **Level 3** as the PostScript version on **Basic** tab of the output object in Design Manager. PostScript Level 2 supports

PNG images but not transparency.

- If you use the Exstream feature in Designer that lets you make a range of colors within an image transparent, then the transparency you apply in Designer overrides any transparency already present within the image. In other words, when you apply transparency within Designer, any transparent areas from the original image appear as a solid color in the output, and the selected range of colors will be transparent instead.

  For more information about applying transparency within Designer, see "Making a Range of Colors Within an Image Transparent" on page 35.

2.1.4 Formatting Images Imported at Design Time

Since design-time content is imported directly into a design as you are creating customer content, you can arrange and format images as needed. Keep in mind that it is a best practice to ensure that any major formatting has been applied to content prior to importing it. Design-time formatting controls for images in Exstream are meant only for fine-tuning to the final appearance of images in the design, since you have more control over the appearance of an image from the program in which it is created.

To format images imported at design time, complete the following tasks as needed:

- "Resizing an Image" below
- "Adjusting the Color of an Image" on the next page
- "Applying Spot Colors to an Image" on page 34
- "Making a Range of Colors Within an Image Transparent" on page 35

You can also use general formatting options available from Exstream, such as moving and rotating objects in the design.

For more information about general formatting options available from Exstream, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Resizing an Image

You can resize an image to make it fit better within your design. In most cases, you can resize, change the scale, or skew (change the aspect ratio) imported images. However, some output drivers have limited support of resizing formatting options. The following table includes those output drivers which have limited support of resizing options:

Output driver resizing support

Output driver	Resize support	Skewing support	Scaling support
AFP	Yes	No	Yes

Output driver resizing support, continued

Output driver	Resize support	Skewing support	Scaling support
IJPDS	Yes	Yes	No
Metacode	Yes	No	Yes
MIBF	Yes	Yes	No
PCL	Yes	No	No
PPML	Yes	Yes	No
TIFF	Yes	No	Yes
ZPL	Yes	No	Yes

All other output drivers support resizing, scaling, and skewing.

To resize design-time imported images:

1. Open the design in Designer.
2. Right-click the image and select **Image Properties**.
3. Complete the following optional tasks as needed:

To	Do this
Resize an image to specific dimensions	a. Click the **Placement** tab. b. In the **Width** and **Height** boxes, enter the measurements of the image.
Scale the image to be a percentage of its original size	a. Click the **Placement** tab. b. In the **Scale width** and **Scale height** boxes, enter a percentage.
Resize the image to the size of the original image file	a. Click the **Placement** tab. b. Select the **Resize to original size** check box.
Resize an image to fit the size of the placeholder frame	a. Click the **Placeholder Frame Properties** tab. b. Select the **Stretch image to fit** check box.

4. Click **OK**.

Adjusting the Color of an Image

When you import an inline image, you can optionally change the color and dithering properties of the image to customize it for your design. For example, you can change a color image to a black-

and-white image, or you can make an image semi-transparent.

Use the **Image Color Management** dialog box to apply color conversion to an image. The **Image Color Management** dialog box adjusts one image at a time. Additionally, if you have defined a selection variable on the image properties, you cannot access the **Image Color Management** dialog box. When you have multiple images, the images are forced to match those of the original pre-selection variable image.

To adjust the color of an image:

1. In Designer, right-click the image and select **Image Color**.
2. In the **Image Color Management** dialog box, make selections based on your design needs. You can select standard color conversion methods, highlight color modes, and transparency settings.

Color Conversion Option	Description
Color Conversion Method	The **Color conversion method** drop-down list lets you select different dithering patterns. Refer to your output device documentation for the best choice. The methods are: • **None** • **Floyd-Steinberg** • **Stucki** • **Burkes** • **Sierra** • **Stevenson Arche** • **Jarvis** • **Ordered** • **Clustered** Keep in mind that if your monitor's resolution is different from the design resolution, set the zoom to **Dot** to preview the image as it will be produced as output. **Important:** If you select **None** in the **Color conversion method** list and are using a black-and-white printer, the object appears in stark contrasts of black and white, and you lose detail.

Color Conversion Option	Description
Highlight Color Mode	If your output devices support highlight color (sometimes called spot color), the **Highlight color mode** list allows you to select the appropriate setting. Options are: • **Black and white**—Converts the image to black and white. • **Highlight color**—Converts the image to use a highlight color. • **Dual tone**—Converts the image to a two-tone image using black and the first highlight color. • **Tri-color**—Converts the image to a tri-color image. This color mode is supported only for AFP output. **Note:** To use highlight color, you must specify highlight color settings on the **Basic** tab of your output. See the output guide associated with your specific output type for more information.
Color to Convert to Black	The **Color to convert to black** color well allows you to select the color in your image that you want to convert to black. **Note:** When you select **Highlight color** in the **Highlight color mode** list, the **Color to convert to black** option changes to **Color to convert to highlight color**.
Color to Convert to Highlight Color 1	The **Color to convert to highlight color 1** color well lets you select the color in your image you want to convert to the first highlight color. This highlight color is determined by the settings on the **Basic** tab in your output type. For more information on output **Basic** tab settings, see the Basic Tab section of the Defining AFP Output chapter in the AFP Output guide.
Color to Convert to Highlight Color 2	The **Color to convert to highlight color 2** color well lets you select the color in your image you want to convert to the second highlight color. This highlight color is determined by the settings on the **Basic** tab in your output type. The second highlight color is only available for certain AFP printers.
Preview	The **Preview** button temporarily applies the **Image Color Management** settings to the image in Designer. **Note:** The highlight color used for previewing highlight color, dual tone, and tri-color images in Designer and in the Outline Viewer is blue. The second highlight color used for previewing tri-color images in Designer and the Outline Viewer is yellow.
Show Original	The **Show Original** button allows you to view the image in its original format.
Replace Original	The **Replace Original** button saves the image based on the current settings. This cannot be undone.
Black-and-White Halftone	The **B/W halftone** check box lets you create photographic screen halftones. For more information on creating photographic screen halftones when working with a black-and-white image, see "Black-and-White Halftone" on the next page.

3. Click **OK**.

Black-and-White Halftone

When working with a black-and-white image, you can create photographic screen halftones. Halftones can improve black-and-white image quality.

To create photographic screen halftones:

1. In Designer, right-click the image and select **Image Color**.
2. In the **Image Color Management** dialog box, select the **B/W halftone** check box.
3. Make selections based on your design needs. The following table describes the black-and-white halftone options that are available:

Black-and-White Halftone Option	Description
Shape	The **Shape** list is active if you select the **B/W halftone** check box. The **Shape** list lets you select the halftone shape to determine the shape of the dots used to create the halftone in your image. Shape options are: • **Default** • **Rectangular** • **Circular** • **Elliptical** • **Random** • **Linear**
Angle	The **Angle** box is active depending on the shape you select from the **Shape** box. The **Angle** box allows you to select an angle to control the angle of the axis for the dots in your image. Valid options range from 0 to 180 degrees.
Grain Size	The **Grain size** box is active depending on the shape you select from the **Shape** box. The **Grain size** box allows you to select a grain size for the dots in your image. Valid options range from 1 to 15 pixels.

4. Click **OK**.

Applying Spot Colors to an Image

When you add an image to the design at design time, you can optionally apply spot colors to the image. When you apply a spot color to an image, you can select specific colors within the image that should print using a specific spot color. For example, if you have a black and white image, you can specify that the spot color replace either all instances of the color white or all instances of the color black within the image. You can also select to apply special spot color finishes (such as gloss or metallic finishes) to the image.

To apply spot colors to an image:

1. In Designer, right click the image and select **Image Color**.
2. In the **Spot Color** area, complete one of the following sets of steps:

To	Do This
Apply a spot color to a specific color in the image	a. In the **Spot color method** list, select **Use defined spot color**. b. Click the **Spot Color to use** color well and select the spot color you want to apply to the image. c. Click the **Color to convert to spot** color well and select the color within the image that you want to replace with the selected spot color.
Apply a spot color finish to the entire image area	a. In the **Spot color method** list, select **Apply spot color finish**. b. Click the **Spot color to use for finish** color well and select the spot color you want to apply as a finish. This finish will be applied to the entire area of the image.
Apply both a spot color and a spot color finish to the image	a. In the **Spot color method** list, select **Defined spot color and spot color finish**. b. Click the **Spot Color to use** color well and select the spot color you want to apply to the image. c. Click the **Color to convert to spot** color well and select the color within the image that you want to replace with the selected spot color. d. Click the **Spot color to use for finish** color well and select the spot color you want to apply as a finish. e. Do one of the following to specify how the spot color finish should be applied to the image: • To apply a spot color finish only to the spot color areas of the image, clear the **Apply finish to entire image** check box (default). • To apply a spot color finish to the entire area of the image, select the **Apply finish to entire image** check box.

3. Click **OK**.
4. From the Menu bar, select **File > Save**.

Making a Range of Colors Within an Image Transparent

If you want to make certain colors within an image transparent, you can apply transparency to certain images within Designer. Applying transparency within Designer is useful if you are overlapping images that do not already include transparency with other content and you want the other content to be visible through part of the image. This option is commonly used to remove background colors (such as black or white) in images that do not already have transparent backgrounds so that only the main image is visible. You can apply transparency only to color images. All pixels of a color that falls within the selected range become fully transparent. In the following illustrations, for example, the color red has been made transparent.

Image without transparency

Image with transparency

Transparency applied within Designer is supported when generating output using only the following output drivers:

- PostScript
- PDF
- PDF/VT
- VDX
- PPML
- VPS

If you are including images for use in PDF or VDX output, the output must be configured to place images at the top of the print stream to support transparency. Transparency will be lost in images placed on-page in PDF or VDX output.

For more information about specifying where to place static images in the print stream, see *Creating Output* in the Exstream Design and Production documentation.

Keep in mind that when you make a range of colors transparent within Designer, this process removes any type of transparency already present within the image. When you apply transparency within Designer, any transparent areas from the original image appear as a solid color in the output, and the selected range of colors will be transparent instead.

For more information about importing images that include transparency, see "Importing Images With Transparency at Design Time" on page 28.

To make a range of colors within an image transparent:

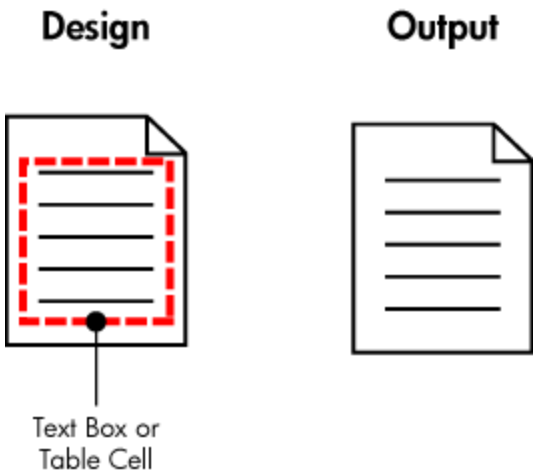
1. Open the design in Designer.
2. Right-click the image and select **Image Color**.
3. In the **Image Color Management** dialog box, select the **Make the following range of colors transparent** check box.
4. Click the color wells to specify the range of colors you want to be transparent.
5. Click **OK**.

   The **Image Color Management** dialog box closes and the transparency is applied to the image.

2.2 Importing Text at Design Time

When you import text at design time, you import the text directly into the design on a message or page within Designer. You can then change or reformat the text based on your design needs. The appearance of the text during design reflects how the text appears in the output. When working with DBCS applications, you must specify the encoding used to translate characters in the imported text.

Sample of design and output when importing text at design time



To import text at design time:

1. Open the design in Designer.
2. Place the cursor anywhere that accepts text (such as a text box or table cell) where you want to import the text file.
3. From the Menu bar, select **Insert > Import > Text File**.

   The **Open** dialog box opens.
4. In the **File name** area, click  and select the text file you want to import.
5. If the file uses a specific encoding, which is common with DBCS files, click  and select the encoding that is applied to the text file.
6. Click **OK**.

   The text is imported into the design, and can be edited or reformatted.

2.2.1 Formatting Text Imported at Design Time

After you add text to your design, you can format that text to achieve the appearance you want. Designer provides many of the formatting features available in traditional word processing programs, such as features you can use to adjust the font style and size. If you use a style sheet to help maintain consistency or brand standards in your organization's designs, the formatting options disabled by the style sheet are not available in Designer.

For information on using style sheets, see *Designing Customer Communications* in the Exstream Design and Production documentation.

You can format text in the following ways:

- Applying basic text formatting
- Adding a bulleted list
- Adding a numbered list
- Automatically resizing text to make it fit
- Changing text spacing

For more information about formatting text, see *Designing Customer Communications* in the Exstream Design and Production documentation.

2.3 Importing a PDF File at Design Time

If you have content stored as a PDF (such as forms or policy information), you can import the pages of a PDF file into a design. Each page of a PDF is imported as an image, so you cannot format or edit the PDF content after you use this method to import a PDF into Exstream. PDF content that is imported at design time is intended to be used as background images and design layers within a design. To allow you to choose the content you want to import, you can also choose to import only specific portions of the PDF. For example, if you only want two pages from a ten-page PDF, you can choose to import a single page or a range of pages rather than importing the entire file.

When you import a PDF file, Design Manager automatically adds the Library objects needed to accommodate the imported content, so you are not required to create a document or pages ahead of time. To accommodate the imported PDF, the following objects are automatically added to the Library:

Objects automatically created by Design Manager when importing a PDF file

This object	Contains this
A Library component (one per PDF page)	Each page of the PDF is converted into an image that is stored as a Library component that can be reused as needed. The Library component name is the same name as the name of the imported PDF file.
A page object (one for each PDF page)	Within the design for each page object is a single page from the PDF (in the form of a Library component). You cannot resize the image in Designer, but you can reposition the image on the design page.
A document (optional)	You can optionally request that Exstream create a new document to contain all the pages created to hold the imported PDF.

If the PDF file you import contains bookmarks and you are generating PDF, PDF/A, PDF/VT, or VDX output, you can retain the existing bookmarks so that they appear in the final output. Customers can then use the bookmarks to navigate the PDF, just as they can use Exstream-created bookmarks. The structure of bookmarks in imported PDFs is maintained exactly. That

is, even if a bookmark points to an object to which you cannot add a bookmark marker in Exstream, the bookmark will work as designed in the final output.

For information about adding bookmarks to PDF output, see *Designing Customer Communications* in the Exstream Design and Production documentation.

If the PDF file you import contains accessibility tagging and you are generating accessible PDF or PDF/A output, the accessibility information in the imported file is retained so that it is present in the final output. For more information about retaining PDF accessibility tags or creating accessible PDF output in Exstream, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Note: If you import a multi-page accessible PDF, the read order of the PDF output might change if the imported PDF has accessibility tags that span pages.

To import PDF content into a design at design time, you must have licensed the Design PDF module.

For more information on the modules related to external content, see *Importing External Content* in the Exstream Design and Production documentation.

To import a PDF file at design time:

1. In Design Manager, in the Library, right-click either the **Documents** or **Pages** heading and select **Import PDF**.

   The **Import PDF** dialog box opens.

2. Browse to and select the PDF you want to import.
3. To control the way in which PDF pages are added to the Library, do one of the following:

To	Do this
Import the PDF pages only	a. Clear the **Create Document** check box. b. In the **Base Page Name** box, enter the name to apply to each imported page. For example, if you enter `Page`, the pages are named `Page 1` and `Page 2`.
Import the PDF pages and automatically add them to a new document	a. Select the **Create Document** check box. b. In the **Document Name** box, enter the name to apply to each imported page and the new document. For example, if you enter `New PDF`, the document is named `New PDF` and the pages are named `New PDF 1` and `New PDF 2`.

4. To define the type of page used to import each page from the PDF, select one of the following options from the **Page Type** drop-down list:

To	Do this
Add each PDF page to a specified paper type	a. Select **Use specified Paper Type**. b. In the **Paper Type** box, select a paper type. Make sure you select a paper type that is the same size as the pages of the PDF file. If the sizes differ, the PDF content could be cut off when viewed in Designer or when printed.
Add each PDF page to a specified template	a. Select **Use specified Page Template**. b. In the **Page Template** box, select a page template. Make sure you select a page template that is the same size as the pages of the PDF file. If the sizes differ, the PDF content could be cut off when viewed in Designer or when printed.

5. To identify which pages to import from the selected PDF, do one of the following:

To	Do this
Import all of the page in the PDF file	In the **Page Range** area, select the **All** radio button.
Import specific pages from the PDF file	In the **Page Range** area, select the **Page** radio button and in the adjacent box, enter the pages you want to import. To indicate a range, separate numbers with a hyphen (for example, `1-5`). To indicate separate pages or identify multiple ranges, separate numbers with a comma (for example, `3-7, 12-18, 30`).

6. Click **Open**.

   The pages and, if specified, the document are added to the Library. The **Log** dialog box opens.

7. Click **OK** to close the Log dialog box or click **Save to File** to save the log.

Chapter 3: Importing Content into a Design at Run Time

Run-time import allows you to design documents to include frequently changing content (such as check images, marketing brochures, or legal statements), and ensure that the most up-to-date content is included in the customer output without sacrificing page design or layout. For run-time import, rather than place content directly into your design, you build placeholder elements into the design to reference the external content. Then, when the engine runs, the referenced content is placed into the customer output.

For example, suppose that the marketing department provides an updated brochure on an irregular basis. With design-time import, you would place the image of the brochure directly into the design. However, you would need to update that image every time marketing releases a new brochure, and you could miss an update. With run-time import, you can set up a placeholder to reference the location where the marketing department stores the brochure on your internal file structure. Then, at run time, the engine will automatically pull the brochure from the repository into the customer output, ensuring that the most up-to-date copy of the content is always provided to customers.

If you import an accessible PDF at run time, the accessibility tagging will be preserved if you create accessible PDF output. The accessibility standards for the imported PDF and the PDF output object should match, or the imported PDF might lose some accessibility features in your output. For more information about specifying the accessibility standard for accessible PDF output in Exstream, see *Designing Customer Communications* in the Exstream Design and Production documentation.

> **Note:** If you import a multi-page accessible PDF, the read order of the PDF output might change if the imported PDF has accessibility tags that span pages.

Importing content at run time offers benefits that you cannot achieve with design-time import. Importing content into a design at run time can be useful in the following situations:

- The content changes more regularly than you want to update the design.
- The content must stay in synchronization with the original source file or must not be changed (for example, standardized or regulated text).
- You are importing an image file that includes multiple pages (for example, multiple-page PDF files).
- You are importing large amounts of content or multiple-page resources (such as PDF files) into the document and do not want to repackage the application to include the updated content.
- You want to reduce the size of your design database and/or your package file. Since run-time content is referenced from your external storage location, when you import content at run time, the content is not stored in the design database. This division of storage reduces the

size of both the design database and the package file.

- You want to leverage content stored in enterprise systems to which Exstream does not natively speak. You must use Dynamic Data Access (DDA) to integrate Exstream into your existing infrastructure.

For more information about using DDAs, see *Configuring Connectors* in the Exstream Design and Production documentation.

For more information on the modules required to leverage external content dynamically at run time, see "Modules Required for Leveraging External Content" on page 10.

This chapter discusses the following topics:

3.1 About Placeholder Objects

To import content into a design at run time, you must set up placeholders in your design at the location where the imported content will be placed. Since run-time content is not available until run time, these placeholder objects act as templates which allow you to pre-define the sizing, placement, and page layout necessary to incorporate run-time content in the design. For example, if you are importing an image into a newsletter at run time, a placeholder image object lets you define where the image will be placed and how big the image will be on the page. You can then design the remaining objects surrounding the image on the page in the desired page layout.

The following table lists the placeholder objects available in Exstream and explains when you should use them:

Placeholder objects in Exstream

Placeholder object	Description
Placeholder variable	Required for all objects. A placeholder variable is a variable that identifies the file you want to import at run time. All types of content imported at run time require a placeholder variable. For text, this variable can be placed directly inline with content, just like any other variable. However, other types of content require a container object as well (such as an empty image object or a placeholder document).

Placeholder objects in Exstream, continued

Placeholder object	Description
Empty image object	Required only with images that are a single page or smaller in size. An empty image object is a design object that allows you to pre-define the size, placement, and formatting of the image that will be imported into the design at run time. For more information about formatting run-time images, see "Formatting Images That Are Imported at Run Time" on page 59.
Placeholder document	Required with multiple-page images. A placeholder document is used to contain all the pages where multiple-page run-time content will be placed. Using a placeholder document, you can import multiple-page DLF, TIFF, or PDF files into a document at run time. This functionality lets you import all the pages in the file, a range of pages, or only certain pages specified by a variable. A placeholder document can also be used to import a single-page image that you want to appear in the output exactly the same as it does in the source file, such as check images. To import PDF files at run time, you must use a placeholder document.
Placeholder frame	Optional with multiple-page images. Used only on pages within a placeholder document. A placeholder frame is a special type of frame that allows you to define the size, placement, and formatting of the multiple-page images that will be imported into the design at run time. • To control the placement, scale, and rotation of imported content, use a placeholder frame on the pages within the placeholder document. • To place the image in the upper left corner of the page (0,0) and not reposition the content, do not use a placeholder frame. This option is useful if the image is the same size as the page. You can only apply run-time formatting to an image if you use a placeholder frame. For more information about formatting run-time images, see "Formatting Images That Are Imported at Run Time" on page 59.

3.2 Adding Placeholder Objects to a Design

Depending on the type of content you are adding to the design at run time, you must complete one of the following sets of tasks to define where run-time content is placed by the engine:

Defining where run-time content is placed by the engine

To define where this content is placed	Do this
Text	1. Create a placeholder variable. 2. Add the placeholder variable to an object that can contain text in the design (such as a text box or table cell). For more information about adding variables to a design, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Defining where run-time content is placed by the engine, continued

To define where this content is placed	Do this
Print resources	1. Create a placeholder variable. 2. Create an empty image object.
Images (single-page or smaller)	1. Create a placeholder variable. 2. Create an empty image object.
Images (multiple pages)	1. Create a placeholder variable. 2. Create a placeholder document. 3. Create a placeholder frame (optional).

To add placeholder objects to a design, complete the following tasks as needed:

- "Creating a Placeholder Variable" below
- "Creating an Empty Image Object" on page 52
- "Creating a Placeholder Document" on page 53
- "Creating a Placeholder Frame" on page 56

3.2.1 Creating a Placeholder Variable

No matter which type of content you are importing at run time, your design must include a special type of variable called a placeholder variable. A placeholder variable is a user-created string variable that contains instructions for the engine on how to locate the external file you want to import at run time. This connection between the variable and the location of the external content you want to import allows the Exstream engine to identify the correct file and controls how the file is incorporated into the final customer output.

Some placeholder variables are used to pass external content directly into the print stream of the output device. However, in some instances (for example, when you import PDF content at run time to an output device that does not natively support PDF format), content might change before it is placed in the print stream. These changes can occur even if you use a placeholder variable to import the content at run time.

For more information about what files are supported natively and non-natively for specific output drivers, see "File Formats Supported for Importing" on page 11.

To create a placeholder variable:

1. In the Library, right-click the **Data Dictionary** heading and select **New Variable**.
2. In the **Name** box, enter a name. In the **Description** box, enter a description (optional).

3. In the **Type** list, select **Placeholder**.
4. In the **Placeholder Type** list, select the option that corresponds to the type of file you want to import at run time.

To import this	Do this	Notes
Text files	Select one of the following: • For plain text files, select **Plain text**. **Note:** Plain text that you import at run time must use a specific encoding, as determined by the following factors: • In a DBCS application on any platform other than z/OS, the text must use ISO-8859-1 encoding. • In a DBCS application on a z/OS platform, the text must use IBM-1047 encoding. • In an SBCS application, the text must use the encoding that matches the locale of the platform on which the package was originally created. For example, if the Windows system locale was set to "English (United States)" when the package was created, then plain text that you import at run time must use Windows-1252 encoding, regardless of the platform on which you are running the engine. If the text uses a different encoding from what is expected, certain characters might appear incorrectly. • For text files that use a specific encoding, select **Specified encoding**. **Note:** If you are generating DLF output, text content with a specified encoding is not supported. • For RTF files, select **RTF text**. • For text from a Live application, select **Live**. • For files in the Exstream Exchange Format (DXF), select **DXF**. • For Word document (*.docx) files, select **Word document (*.docx)**. • For HTML files, select **HTML**.	The engine automatically substitutes variable values for any variables in the text file. Variable names in the text file must be enclosed in brackets to be recognized as variables (for example, `Dear <CustomerName>`). To prevent variable substitution in a plain text file, you can use the DONT_RESOLVE_VARIABLES_IN_PLAIN_TEXT_IMPORT switch. For more information about using the DONT_RESOLVE_VARIABLES_IN_PLAIN_TEXT_IMPORT switch, see *Switch Reference* in the Exstream Design and Production documentation. For imported HTML files, you can use angle brackets, but you must use the HTML entities `&lt;` and `&gt;` (for example, `Dear &lt;CustomerName&gt;`). You can also use double braces (for example, `Dear {{CustomerName}}`). For more information about using variables with imported HTML content, see "HTML Considerations" on page 85.

To import this	Do this	Notes

Single-page images or PDFs	Select one of the following:
	• For AFP page segment image files, select **APF PSEG**.
	• For FS45, FS10, and FS11 IOCA images, select **AFP FSXX**.
	• For EPS files, select **EPS**.
	• For JPEG files, select **JPEG**.
	• For PDF files, select **PDF**.
	• For PNG files, select **PNG**.
	• For SVG files, select **SVG**.
	• For TIFF files, select one of the following:
	  • For Group 4 Tiff Standard or 1-bit uncompressed black and white TIFF images, select **TIFF (B&W G4 or uncomp)**.
	  • For full-color TIFF files, select **TIFF (Color)**.
		• If you are generating DLF output, **AFP FSXX** and **AFP PSEG** are not supported. Also, inline PDF placeholder variables are not supported for DLF output.
		• If you want to import images that include transparency, the images must be in PNG or SVG format.
		For more information about importing images with transparency at run time, see "Importing Images with Transparency at Run Time" on page 57.
		• SVG images are supported only for import with HTML5 output.

Printer resources	Select one of the following: • For images from a printer resource (such as PostScript TIFF images or Metacode IMGs), select **Image resource**. • For overlays from a printer resource (such as Metacode forms (FRMs), AFP overlays, PostScript forms, and page segments), select **Overlay resource**.	If you are generating DLF output, printer resources are not supported.

Multiple-page resources	Select one of the following:
	• For multiple-page files in the Exstream Exchange Format (DXF), select **DXF**.
	• For a multiple-page Live file, select **Live**.
	• For a multiple-page PDF file, select **PDF**.
	• For TIFF files, select one of the following:
	  • For Group 4 Tiff Standard or 1-bit uncompressed black and white TIFF images, select **TIFF (B&W G4 or uncomp)**.
	  • For full-color TIFF files, select **TIFF (Color)**.
	• For Word document (*.docx) files, select **Word document (*.docx)**.	

5. If the variable must reference more than one external file, select the **Array** check box. For example, you could use an array to import different images into repeating table rows. For multiple-page file imports, the placeholder variable must always be an array. Every page in the file becomes one element in the array, and the array is loaded with the pages of each file after the engine reads the data before composition.

For information about controlling how the engine reads the array to select the correct external file, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

6. In the **Design sample** box, enter some sample text to show in Designer (optional).
7. Click **Finish**.

   The variable opens in the Property Panel.
8. Click the **Placeholder** tab.
9. If you are importing a file that is stored in the common asset service (CAS) or another external repository connected through the CAS:

   a. Select the **CAS resource** check box and set the value of the variable to the CAS resource ID for the file.

      The CAS resource ID is a Base64-encoded string and acts as an identifier that is used to locate the file in the CAS repository. For more information about the format for entering a CAS resource ID, see "Setting the Initial Value for a Placeholder Variable" on page 50.

   b. In the control file for your application or from the command prompt, use the MGWUSER and MGWPASSWORD engine switches to specify the user name and password for the OTDS user that you want to use to sign in to the CAS repository at run time.

   c. Optionally, use the engine switches listed in the following table to override the OTDS and management gateway connection settings specified on the **Integration** tab in **System Settings**.

To	Use this switch
Specify the URL for the OTDS server	OTDSURL This switch overrides the URL that is specified in the **OTDS base URL** box on the **Integration** tab in **System Settings**.
Specify the OTDS resource ID for OTDS authentication	OTDSRESOURCEID This switch overrides the OTDS resource ID that is specified in the **Resource ID** box on the **Integration** tab in **System Settings**.
Specify the URL for the management gateway server	MGWURL This switch overrides the URL that is specified in the **Management Gateway URL** box on the **Integration** tab in **System Settings**.
Specify the management gateway tenant that corresponds to the CAS repository	MGWTENANT This switch overrides the tenant that is specified in the **Tenant name** box on the **Integration** tab in **System Settings**.

To	Use this switch
Specify the application domain that corresponds to the CAS repository	MGWAPPDOMAIN This switch overrides the application domain that is specified in the **Application domain** box on the **Integration** tab in **System Settings**.

For information about using these switches, see *Switch Reference* in the Exstream Design and Production documentation.

> **Tip:** You can also use Adaptive Media Delivery (AMD) to access images directly from OTMM instead of connecting through the CAS. To access images directly from OTMM, set the value of the placeholder variable to the AMD URL instead of configuring the placeholder variable to access a CAS resource.
>
> For more information about setting the value of the placeholder variable, see "Setting the Initial Value for a Placeholder Variable" on the next page.
>
> For more information about using AMD with OTMM, see the OpenText Media Management documentation.

10. If you are importing an HTML file to use in HTML output, select an option to specify how the engine includes the HTML content in the output.

To	Do this
Place the HTML content into the output without converting it to the Exstream format	Select the **For HTML output, do not convert** check box. This is the default.
Convert the HTML content to the Exstream format before placing the HTML content into the output	Clear the **For HTML output, do not convert** check box.

For more information about importing HTML files, see "HTML Considerations" on page 85.

11. Select an option to specify the file(s) to import for the customers in the run.

To	Do this
Import a different file for each customer in the run	Select the **The file for each customer is unique** check box.
Import the same file for each customer in the run	a. Clear the **The file for each customer is unique** check box. b. In the **Maximum number of files to hold in memory** box, enter the number of files to place in memory, so they are not read for each customer. The maximum number allowed is `9999`.

12. If you are importing a text file that uses an encoding (DBCS applications only), click  , and select the encoding that is applied to the file.
13. If you are importing multiple-page images, make a selection in the **Pages to import** list to

specify which pages to import at run time:

To	Do this
Import all the pages in the file	Select **All**.
Specify a range of pages to import	a. Select **Range**. b. In the adjacent box, enter the starting and ending pages. To indicate a range, separate the numbers with a hyphen (for example, `1-5`). To indicate separate pages or identify multiple ranges, separate the numbers with a comma (for example, `3-7, 12-18, 30`).
Use a variable to identify which pages to import	a. Select **Variable**. b. In the adjacent box, click  and select a string or integer variable to control the imported pages.

14. Click  .

3.2.2 Setting the Initial Value for a Placeholder Variable

After you set up the placeholder variable, you must set the initial value of the variable to reference the location of the external file. You can use any of the available methods to specify the initial variable value (for example, a user specified value, a mapped reference file, a formula, or using the IMPORTDIRECTORY switch in a control file). You can map a placeholder variable to a customer driver, initialization, post-sort initialization, or reference data file. The array is loaded with the pages of each file after data is read but before composition.

For more information on variables, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

For information on the IMPORTDIRECTORY switch, see *Switch Reference* in the Exstream Design and Production documentation.

The variable value is the location of the external file to import at run time, such as the following:

External resource name formatting requirements

Resource	Sample entry format
The file name	`test.dxf`
The file name and path	`x:\images\test.dxf`

External resource name formatting requirements, continued

Resource	Sample entry format
The file name and HTTP path	`http://images.com/test.dxf` **Tip:** All HTTP resources must begin with the `http://` or `https://` prefix to be accurately located by the engine. **Note:** HTTP resources are not supported in z/OS.

The name of the printer resource	`S1TEST`
	
	`test.img`

The key value for a connector to access	`key=https://servername/images/test.dxf`

The CAS resource ID for a file stored in the common asset service (CAS) repository	`Y3hyOi8_`
	`aWQ9YzE0M2Y5MGMtODQwMy0yZTQ0LWFlNTUtZDU0MzM4NzMxYTQ1O3Y9M`
	`Tt0PWQ2ZWE1OTUzLTBhMWUtNzQ1MC0wMmUwLTQyZDMzNGY2ZDk0ZQ==`
	
	Tip: The CAS resource ID is a Base64-encoded string and acts as an identifier that is used to locate the file in the CAS repository. The CAS resource ID can be obtained from the Resource Browser in Designer or from Workshop.
	
	To view the CAS resource ID for an image from the CAS in Designer, complete the following steps:
	
	1. From the **Drawing Objects** toolbar, click  .
	2. From the **Import an Image** dialog box, click  .
	3. From the Resource Browser, select the image for which you want to view the CAS resource ID.
	4. Click  above the image preview to view the properties of the image, which include the CAS resource ID.

The Adaptive Media Delivery (AMD) URL for an image from OpenText Media Management (OTMM)	`http://myotmmserver/adaptivemedia/rendition?id=`
	`7e5aae68d8f83196c16899cfef948e5f881d2834&version=3`
	
	Tip: The `id` parameter in the URL specifies the OTMM ID for the image, which can be obtained from the OTMM user interface. The `version` parameter identifies a specific version of the image and can be omitted to retrieve the latest version.
	
	For more information about using OTMM, see the OpenText Media Management documentation.
	
	Note: HTTP resources are not supported in z/OS.

3.2.3 Creating an Empty Image Object

An empty image object is used for importing single-page images or PDF files at run time. When you include an empty image object as a placeholder in a design, you can see how the imported content affects the remaining content on the page, such as shifting text and other objects around the placed image. The empty image object allows the design to resemble the expected customer output to help you plan and lay out the remaining content in your design. Keep in mind, however, that unlike images imported at design time, an empty image appears blank on the design page. The content of an empty image object will not be visible until the content is generated as output.

If you are importing an image into a document which will be produced as Live output (a DLF file), you must use an empty image object to import the image at run time. When designing for DLF output, you can also design interactive image import features for LiveEditor end users.

For information about importing images into designs for DLF output, see *Designing for LiveEditor* in the Exstream Design and Production documentation.

Before you can add an empty image into a design, you must set up a placeholder variable that references the file you want to import.

For more information about creating a placeholder variable, see "Creating a Placeholder Variable" on page 45.

To create an empty image object to pre-define the placement of an image:

1. Open the design in Designer.
2. From the Drawing objects toolbar, click  .

   The **Create an empty image** dialog box opens.
3. In the **Placeholder variable** box, click  and select the placeholder variable that references the external location of the file(s) you want to import.
4. From the **Content change** drop-down list, select **No change**.
5. Click **OK**.

   The empty image object is placed in the design, where you can move or resize the empty image as needed.
6. Right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.
7. Click the **Image** tab.
8. Complete the following optional steps, as needed:

To	Do this
Override the automatic names assigned by Designer (such as `XSFT01`, `S1EXnnnn`, and so on)	In the **Image Name** box, enter a name for the image. The name you enter appears on the design page inside the empty image. Some print streams have restrictions on file names. What can be read in one output driver might fail in another. Unless you limit the application to one output driver, or you are knowledgeable about file naming requirements for all the output drivers included in the applications, you should leave the image name box empty. If you are creating Multi-Channel XML output, however, and you specify an image path, you must specify an image name. Some outputs do not support DBCS characters in image resource names. For example, Metacode output accepts only ASCII characters in the **Image Name** box. If you use AFP output, you must use SBCS characters in the **Image Name** box. If you use DBCS characters in the **Image Name** box with AFP as a selected output type, Exstream issues an error message and replaces the image name with a unique 8–byte name. For more information about specifying an image path for XML (Multi-Channel) output, see *Creating Output* in the Exstream Design and Production documentation.
Maintain the size of the original image referenced by the placeholder variable	From the **Image placement and size** drop-down list, select **Maintain size from file**. Regardless of the size of the empty image object in the design, the image will be generated as output using the original dimensions of the external file.
Rotate the image during import	From the **Image Rotation** drop-down list, select one of the following options: • **None** • **90** • **180** • **270**
Resize the image during import	Specify how you want to resize the image during import. For example, you might want to scale or skew the image, or to maintain the original image aspect ratio. For more information about resizing images that are imported at run time, see "Resizing Images That Are Imported at Run Time" on page 60.

9. Click **OK**.

   The **Image Properties** dialog box closes.

3.2.4 Creating a Placeholder Document

Placeholder documents are used when you import multiple pages of content at run time. Like other documents in Exstream, placeholder documents are used to contain and organize pages. The pages of the imported content are applied to the design pages you store within the

placeholder document. A placeholder document must contain at least one page to receive the imported content.

> **Tip:** Instead of creating multiple pages, you can create one default page and specify that the engine uses this page as a template to create as many pages ("filler pages") as needed for the imported content.

For information on setting page properties, see *Designing Customer Communications* in the Exstream Design and Production documentation.

When you create pages for use with placeholder documents, you must consider where on the page you want the engine to place the imported content. Depending on whether you want to resize, scale, or rotate the imported content, you can add the following types of pages to a placeholder document:

Pages to use in a placeholder document

Type of page	When to use
Blank pages	Use blank pages if you do not want to resize or reposition the imported content. At run time, the imported content is placed in the upper left corner of the design page (0,0). You cannot reposition this content on the page. This method can be useful if you are importing pages that are the same size as the design page, and do not require any formatting changes.
Design pages containing placeholder frames	Use design pages that contain placeholder frames if you want to control the placement, scale, or rotation of the imported content. At run time, the engine inserts the source image file directly into the print stream without adding it to the application, and places the external image referenced by the placeholder variable into the design based on the location of the placeholder frame. For more information about creating a placeholder frame, see "Creating a Placeholder Frame" on page 56.

Pages that are included in a placeholder document cannot be set up for flow.

> **Note:** You cannot use tables with named data sections in placeholder documents.

If you import a PDF file that contains bookmarks and you are generating PDF, PDF/A, PDF/VT, or VDX output, you can retain the existing bookmarks so that they appear in the final output and customers can use them to navigate. The structure of bookmarks in imported PDFs is maintained exactly. That is, even if a bookmark points to an object to which you cannot add a bookmark marker in Exstream, the bookmark will work as designed in the final output. The bookmarks from the imported pages appear under the placeholder document bookmark.

For information about adding bookmarks to PDF output, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Before you can create a placeholder document, you must set up the placeholder variable that references the file you want to import.

For more information about creating a placeholder variable, see "Creating a Placeholder Variable" on page 45.

To create a placeholder document to place runtime content within a separate document:

1. In the Library, right-click the **Documents** heading and select **New Document**.
2. In the **New Document** dialog box, enter a name in the **Name** box. You can also enter a description in the **Description** box (optional).
3. Click **Finish**.

   The document opens in the Property Panel.
4. Click the **Basic** tab.
5. In the **Document type** list, click **Placeholder (use pre-composed content)**.
6. In the **Placeholder** box, click  and select the placeholder variable to associate with the placeholder document.
7. Add pages to the placeholder document:

   a. In the **Contents** area, click  and select **Add Page**.

   b. In the **Select Page** dialog box, select the pages that you want to add to the placeholder document. Be sure that you include the following pages:

      - The design page, either with or without a placeholder frame
      - The design page that you want to use as the last page of the document (optional)

   c. Click **OK**.

   d. Repeat step a through step c until all of the pages that are needed to import the content correctly are listed in the **Contents** area.
8. Save the placeholder document and open it in the Edit Panel.
9. In the Edit Panel, find the design page with the placeholder frame and double-click its far right column.
10. In the **Document Page Properties** dialog box, in the **Position of page in document** list, select one of the following options for positioning the page:

To	Do this
Place the page as it appears in the Edit Panel and Library	Select **As Ordered**.
Create one page for each element in the placeholder variable array	Select **Filler Page (as required)**. The engine uses the layout of the design page as a template in creating the additional pages.

To	Do this
Create one page for each element in the placeholder array, but only up to a maximum specified number of pages	Select **Fixed Number Filler** and in the adjacent box, enter the maximum number of pages that can be generated. Regardless of how many elements in the placeholder variable are mapped, the engine stops importing after the specified number of pages.
Use the selected page as the final page	Select **Last Page (replaces last filler page)**.

11. Click **OK**.

3.2.5 Creating a Placeholder Frame

A placeholder frame is a design object that lets you control the placement, scale, and rotation of external content imported into a placeholder document. Placeholder frames are not connected to an external resource. When a page containing a placeholder frame is added to a placeholder document, the placeholder document controls what external content is placed in the frame.

When using a placeholder frame to place content within a page, keep the following in mind:

- Only one placeholder frame can be placed on a page.
- The contents imported into the frame cannot split and flow to another page.
- The frame expands vertically on the page, as needed, to accommodate the imported content.
- The frame cannot be embedded in a text box or table.

Tip: Instead of creating multiple pages with placeholder frames, you can create one default page and specify that the engine uses this page as a template to create as many pages ("filler pages") as needed for the imported content.

For information on setting page properties, see *Designing Customer Communications* in the Exstream Design and Production documentation.

To create a placeholder frame:

1. Open the page in Designer.
2. On the Drawing Objects toolbar, click  .

   The **New Frame** dialog box opens.
3. Select the **Placeholder** radio button.
4. Click **OK**.

   The **Insert Frame** dialog box opens.

5. In the **Placeholder Frame Properties** area, configure the following options as needed:

To	Do this
Place borders around the outside edges of the frame that holds the image	a. Click the edges of the box where you want to place borders. b. To format the borders, click the box in the **Borders** area. The **Border Properties** dialog box opens. c. In the **Offset** box, enter a number to offset the border so that it does not overlap the text to be imported into the frame. d. In the **Line properties** area, specify the line type, color, and thickness for the frame border. e. Click **OK**. The **Border Properties** dialog box closes.
Scale the images to fit the frame size exactly	Select the **Stretch images to fit** check box.
Change the orientation of the image	From the **Image rotation** drop-down list, select a rotation option.

6. Click **OK**.

   A purple frame appears on the page with a **Document placeholder only** label.

7. Move the frame to the place where you want content to start on the page. The critical placement is the frame's upper left edge. This sets the coordinates that the engine uses to start placing the imported content. By default, the engine starts placing imported content at coordinates 0,0 on a placeholder document page.

8. Resize the frame so that it is wide enough to accommodate the content being imported. The frame will expand vertically.

3.3 Importing Images with Transparency at Run Time

If you want to import images that include transparency at run time, Exstream supports PNG and SVG images that include transparency (including both palette-based and alpha channel). For example, if you have product images with transparent backgrounds that you want to appear over other objects in a design, you can make sure the images are in the PNG or SVG format and import them into your design as you would any other image.

The following output types support transparency in PNG images imported at run time:

- AFP
- DOCX

- EDGAR HTML
- HTML
- HTML (email)
- PDF
- PDF/A
- PDF/VT
- PostScript
- PowerPoint
- RTF
- TOP
- VDX
- VIPP
- VPS
- XML (Composed)
- XML (Multi-Channel)

HTML5 is the only output type that supports transparency in SVG images imported at run time.

For general information about support for importing PNG, SVG, and other file formats, see "File Formats Supported for Importing" on page 11.

Keep in mind the following considerations:

- If you are including transparent images for use in PDF, PDF/A, or VDX output, you must place images at the top of the print stream or the transparency will be lost. To do this, open the output object in the Property Panel. On the **Resource Management** tab, in the **Image processing** list, select **Place at top of print stream**.

  For more information about how to place images at the top of the print stream, see *Creating Output* in the Exstream Design and Production documentation.

- If you select the **Image each page** option on the **Basic** tab of the output object in Design Manager, then the TIFF output driver is used, and PNG images are not supported.

  For more information about creating pages as images, see *Creating Output* in the Exstream Design and Production documentation.

- If you are using the PostScript output driver, you must select **Level 3** as the PostScript version on **Basic** tab of the output object in Design Manager. PostScript Level 2 supports PNG images but not transparency.

3.4 Formatting Content That Will Be Imported at Run Time

Since run-time content is imported by the engine at run time, you must define any additional formatting that you want the engine to apply to the imported image from the placeholder object in the design. Keep in mind that it is a best practice to ensure that any major formatting has been applied to content prior to importing it. Run-time formatting controls in Exstream are meant only for last minute fine-tuning to the final appearance of content, since you have more control over the appearance of content from the program in which it is created.

Before you apply formatting controls to content imported at run-time, it is also important to be knowledgeable about any existing formatting applied to the resource. For example, you should know the original orientation of images before you apply rotation to an empty image placeholder. This information can be especially important since you will not be able to view the resource from the design; the imported content is visible only when the content is generated as output.

To format content that will be imported at run time, complete the following tasks as needed:

- "Formatting Images That Are Imported at Run Time" below
- "Formatting Text That Is Imported at Run Time" on page 71
- "Importing External Content That Contains Variables" on page 73
- "Setting Up a Placeholder Document to Allow Imported DOCX or DXF Files to Flow" on page 75

3.4.1 Formatting Images That Are Imported at Run Time

Depending on the design feature that you use to import images, you have different formatting options available to you. The following table provides information about the formatting options that are available for each feature:

Formatting options for images imported at run time

Formatting option	Empty image object	Placeholder frame	Placeholder object
Border	Yes	No	Yes
Resizing (scaling/skewing/aligning)	Yes	Yes	Yes
Rotating	Yes	Yes	Yes

To format images that are imported at run time, complete the following tasks as needed:

- "Adding a Border to Images That Are Imported at Run Time" below
- "Resizing Images That Are Imported at Run Time" below
- "Rotating Images That Are Imported at Run Time" on page 69

Adding a Border to Images That Are Imported at Run Time

If you import an image using an empty image object, you can add a border around the image to help emphasize it in the design.

This task assumes that you have already created and placed an empty image object in the design.

For more information about creating an empty image object, see "Creating an Empty Image Object" on page 52.

To add a border to an image:

1. In Designer, open the design that contains the empty image object.
2. Right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.
3. Click the **Lines and Fill** tab.
4. In the **Frame** area, make selections to define the style, color, and thickness of the border.
5. Click **OK**.

   The **Image Properties** dialog box closes. The border is not visible from the design; however, the border will appear around the imported image after the content is generated as output.

Resizing Images That Are Imported at Run Time

Many application designs include multiple image sizes for a frequently used image, such as a logo. Other images, such as a photo of an insurance agent, can change for each customer. In most cases, an application includes imported images that have a wide variety of image sizes and image aspect ratios.

To minimize the number of images that must be included in a design database, you can resize images at run time. In addition, you can ensure image consistency by maintaining the original aspect ratio when resizing images. If the original shape, size, or aspect ratio of an imported image does not match the shape, size, or aspect ratio of its placeholder object, some white space might appear between the resized image and its placeholder object. You can select an alignment position, such as top left, center, or bottom right, to control the anchor position for the resized image within its placeholder object.

For example, suppose that you are designing a welcome letter for a new insurance customer. The top of the page includes the company logo, a photo of the customer's agent, and an address block. You want the top edges of the logo, photo, and address block to be aligned. The database of agent photos includes images with various sizes and image aspect ratios. If an image must be resized to fit within the image placeholder object, you can maintain the original image aspect ratio so that the image is not distorted. If you want to resize the image, while at the same time maintain the original image aspect ratio, you might want to select the top center alignment position so that the top edge of the agent photo is aligned with the top edges of the company logo and address block.

In most cases, you can resize, skew (change the aspect ratio), or scale imported images. However, some output drivers have limited resizing options.

The following table lists the output drivers that have limited resizing options:

Output driver resizing support

Output Driver	Resizing support	Skewing support	Scaling support	Aligning support
AFP	Yes	No	Yes	Yes
IJPDS	Yes	Yes	No	Yes
Metacode	Yes	No	Yes	No
MIBF	Yes	Yes	No	Yes
PCL	Yes	No	Yes	Yes
PPML	Yes	Yes	No	Yes
TIFF	Yes	No	Yes	Yes
ZPL	Yes	No	Yes	Yes

All other output drivers support resizing, scaling, skewing, and aligning.

To resize images that are imported at run time, complete the following tasks as needed:

- “Maintaining Original Image Size Within an Empty Image Object” on the next page
- “Maintaining Original Image Size Within a Placeholder Object” on page 63
- “Maintaining Image Aspect Ratio and Resizing an Image to Specific Dimensions Within an Empty Image Object” on page 64
- “Maintaining Image Aspect Ratio and Resizing an Image to Specific Dimensions Within an Empty Image Object” on page 64
- “Resizing an Image to the Size of an Empty Image Object” on page 66
- “Resizing an Image to the Size of a Placeholder Object” on page 67

Maintaining Original Image Size Within an Empty Image Object

You can maintain the original image size within an empty image object. For example, suppose that you work for a print service provider and you are creating a new baby announcement that will include images of handprints and footprints. Customers provide scanned images at a specified size and resolution and you must make sure that the image sizes are maintained in the final output. You place the empty image objects on a page and set the properties of the empty image object to maintain the original image size.

To maintain the original image size within an empty image object:

1. Open the design in Designer.
2. To access the empty image object properties, right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.
3. Click the **Image** tab.
4. From the **Image placement and size** drop-down list, select **Maintain size from file**.
5. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box on the **Image** tab.
6. To specify the alignment position of imported images with the empty image object:
   a. Clear the **Snap bounding box to image** check box on the **Image** tab.

      The **Align From** grid becomes active.
   b. In the **Align From** grid, select the alignment position for the resized image.
7. If you want to use a variable to control the alignment position of imported images within the empty image object, complete the following tasks:
   a. Clear the **Snap bounding box to image** check box on the **Image** tab.

      The **Align from variable** box becomes active.
   b. In the **Align from variable** box, click  and select the variable that controls the

alignment position for imported images.

The following table includes the values that correspond to alignment options:

Alignment option	Integer value	Alignment option	Integer value	Alignment option	Integer value
Top left	0	Top center	1	Top right	2
Center left	3	Center	4	Center right	5
Bottom left	6	Bottom center	7	Bottom right	8

8. Click **OK**.

   The **Image Properties** dialog box closes.

9. From the Menu bar, select **File > Save**.

Maintaining Original Image Size Within a Placeholder Object

Not only can you import images into empty image objects, but you can also import images into other objects, such as a text box, table cell, or graphic message, and maintain the original image size. For example, suppose that you are designing a letter that will include a signature image and you want the image to appear at its original size in the output. You can insert an inline placeholder variable within a text box and set the properties for the placeholder variable to maintain the original image size.

To maintain the original image size within a placeholder object:

1. Open the design in Designer.
2. To access the placeholder variable properties, double-click the placeholder variable, right-click and select **Variable properties**.

   The **Placeholder Variable Use Properties** dialog box opens.

3. From the **Image placement and size** drop-down list, select **Maintain size from file**.
4. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box.
5. To specify the alignment position of imported images within the placeholder object:
   a. Clear the **Snap bounding box to image** check box.

      The **Align From** grid becomes active.

   b. In the **Align From** grid, select the alignment position for the resized image.

6. If you want to use a variable to control the alignment position of imported images within the placeholder object, complete the following tasks:

a. Clear the **Snap bounding box to image** check box.

   The **Align from variable** box becomes active.

b. In the **Align from variable** box, click  and select the variable that controls the alignment position for imported images.

   The following table includes the values that correspond to alignment options:

Alignment option	Integer value	Alignment option	Integer value	Alignment option	Integer value
Top left	0	Top center	1	Top right	2
Center left	3	Center	4	Center right	5
Bottom left	6	Bottom center	7	Bottom right	8

7. Click **OK**.

   The **Placement Variable Use Properties** dialog box closes.

8. From the Menu bar, select **File > Save**.

Maintaining Image Aspect Ratio and Resizing an Image to Specific Dimensions Within an Empty Image Object

You can maintain the original image aspect ratio when you resize an image to fit within the specified dimensions of an empty image object. For example, suppose that you are creating a postcard for an auto dealer that includes a coupon for service discounts. You want to include an image of the customer's vehicle model on the postcard, but you must make sure that the image fits within specified dimensions. To prevent the vehicle images from being distorted when they are resized, you can set the properties for the empty image object to maintain the original image aspect ratio.

To maintain the original image aspect ratio and resize an image to specific dimensions within an empty image object:

1. Open the design in Designer.
2. To access the empty image object properties, right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.

3. Click the **Image** tab.
4. From the **Image placement and size** drop-down list, select **Fit image proportionally**.
5. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box on the **Image** tab.

6. To specify the alignment position of imported images with the empty image object:
   a. Clear the **Snap bounding box to image** check box on the **Image** tab.

      The **Align From** grid becomes active.

   b. In the **Align From** grid, select the alignment position for the resized image.

7. If you want to use a variable to control the alignment position of imported images within the empty image object, complete the following tasks:
   a. Clear the **Snap bounding box to image** check box on the **Image** tab.

      The **Align from variable** box becomes active.

   b. In the **Align from variable** box, click  and select the variable that controls the alignment position for imported images.

      The following table includes the values that correspond to alignment options:

Alignment option	Integer value	Alignment option	Integer value	Alignment option	Integer value
Top left	0	Top center	1	Top right	2
Center left	3	Center	4	Center right	5
Bottom left	6	Bottom center	7	Bottom right	8

8. Click **OK**.

   The **Image Properties** dialog box closes.

9. From the Menu bar, select **File > Save**.

Maintaining Image Aspect Ratio and Resizing an Image to Specific Dimensions Within a Placeholder Object

In addition to empty image objects, you can also maintain the original image aspect ratio when resizing an image to fit within the specified dimensions of a placeholder object. For example, suppose that you are designing a checking account statement that includes check images within table cells. You insert a placeholder variable into the table cells and then set the properties of the placeholder variable to maintain the original aspect ratio of the check images.

To maintain the original image aspect ratio and resize an image to specific dimensions within a placeholder object:

1. Open the design in Designer.
2. To access the placeholder variable properties, double-click the placeholder variable, right-click and select **Variable properties**.

The **Placeholder Variable Use Properties** dialog box opens.

3. From the **Image placement and size** drop-down list, select **Fit image proportionally**.
4. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box.
5. To specify the alignment position of imported images within the placeholder object:
   a. Clear the **Snap bounding box to image** check box.

      The **Align From** grid becomes active.

   b. In the **Align From** grid, select the alignment position for the resized image.
6. If you want to use a variable to control the alignment position of imported images within the placeholder object, complete the following tasks:
   a. Clear the **Snap bounding box to image** check box.

      The **Align from variable** box becomes active.

   b. In the **Align from variable** box, click  and select the variable that controls the alignment position for imported images.

      The following table includes the values that correspond to alignment options:

Alignment option	Integer value	Alignment option	Integer value	Alignment option	Integer value
Top left	0	Top center	1	Top right	2
Center left	3	Center	4	Center right	5
Bottom left	6	Bottom center	7	Bottom right	8

7. Click **OK**.

   The **Placement Variable Use Properties** dialog box closes.
8. From the Menu bar, select **File > Save**.

Resizing an Image to the Size of an Empty Image Object

You can resize an image to fit the size of an empty image object when the size of the empty image object must be maintained and you do not want any white space to appear within the bounding box of the empty image object. For example, suppose that you are creating a document that includes an image with a seasonal theme, such as a snowflake or beach ball. If any distortion that might result from stretching the image to fit within the empty image object would be acceptable, you can set the properties of the empty image object to resize images to the size of the empty image object.

To resize an image to the size of an empty image object:

1. Open the design in Designer.
2. To access the empty image object properties, right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.
3. Click the **Image** tab.
4. From the **Image placement and size** drop-down list, select **Fit image to object**.
5. Click **OK**.

   The **Image Properties** dialog box closes.
6. From the Menu bar, select **File > Save**.

Resizing an Image to the Size of a Placeholder Object

In addition to empty image objects, you can resize images to fit within the specified dimensions of a placeholder object. For example, suppose that you are designing a travel brochure for customers that have requested information about a specific destination theme, such as a beach or mountain resort. You might include images of beach or mountain landscapes in the brochure. If any distortion that might result from stretching the images to fit within the placeholder object would be acceptable, you can set the properties of the placeholder variable to resize images to the size of the placeholder object.

To resize an image to the size of a placeholder object that includes a placeholder variable:

1. Open the design in Designer.
2. To access the placeholder variable properties, double-click the placeholder variable, right-click and select **Variable properties**.

   The **Placeholder Variable Use Properties** dialog box opens.
3. From the **Image placement and size** drop-down list, select **Scale to specific size**.
4. Click **OK**.

   The **Placement Variable Use Properties** dialog box closes.
5. From the Menu bar, select **File > Save**.

Using a Variable to Resize an Image Within an Empty Image Object

You can use a variable to specify how to resize images with an empty image object. The variable can specify if you want to maintain the original image size and aspect ratio, resize the image to the size of the empty image object, or resize the image to specific dimensions and maintain the original image aspect ratio within the empty image object. For example, suppose that you are designing a real estate directory that includes multiple empty image objects of

various sizes and aspect ratios within each page, and the directory will be produced as an 8-1/2 x 11-inch booklet. If you want to test to find out if all of the content on each page would be retained if you changed the output size to a 7 x 9-inch booklet, you can use a variable to specify the resizing method. You would change the variable value for resizing instead of changing the resizing settings for each empty image object.

To use a variable to resize an image within an empty image object:

1. Open the design in Designer.
2. To access the empty image object properties, right-click the empty image object and select **Image Properties**.

   The **Image Properties** dialog box opens.
3. Click the **Image** tab.
4. From the **Image placement and size** drop-down list, select **Control by variable**.

   The **Image placement and size variable** box becomes active.
5. In the **Image placement and size variable** box, click  and select the variable that specifies how you want to resize the imported image.

   The following table includes the values that correspond to resizing options:

Resizing option	Integer value
Maintain size from file	0
Fit image to object	1
Fit image proportionally	2

   For information about creating variables, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.
6. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box.
7. Click **OK**.

   The **Image Properties** dialog box closes.
8. From the Menu bar, select **File > Save**.

Using a Variable to Resize an Image Within a Placeholder Object

In addition to empty image objects, you can also use a variable to specify how to resize images within a placeholder object. The variable can specify if you want to maintain the original image size and aspect ratio, resize the image to the size of the placeholder object, or resize the image to specific dimensions and maintain the original image aspect ratio within the placeholder object. For example, suppose that you are creating a directory that includes multiple placeholder

objects of various sizes and aspect ratios, and the directory will be delivered by the postal service. If you want to test to determine the resulting page count and associated postage cost based on how you choose to resize imported images, you can use a variable to specify the resizing method. You would change the variable value for resizing instead of changing the resizing settings for each placeholder object.

To use a variable to resize an image within a placeholder object:

1. Open the design in Designer.
2. To access the placeholder variable properties, double-click the placeholder variable, right-click and select **Variable properties**.

   The **Placeholder Variable Use Properties** dialog box opens.
3. From the **Placeholder Variable Use Properties** drop-down list, select **Control by variable**.

   The **Image placement and size variable** box becomes active.
4. In the **Image placement and size variable** box, click  and select the variable that specifies how you want to resize the imported image.

   The following table includes the values that correspond to resizing options:

Resizing option	Integer value
Maintain size from file	`0`
Fit image to object	`1`
Fit image proportionally	`2`

   For information about creating variables, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.
5. If you want to change the size of the bounding box to match the size of the image, select the **Snap bounding box to image** check box.
6. Click **OK**.

   The **Placement Variable Use Properties** dialog box closes.
7. From the Menu bar, select **File > Save**.

Rotating Images That Are Imported at Run Time

You can rotate an image up to 270 degrees to make it fit better within a design. The method used for rotating an image depends on the design feature you used to import the image.

To rotate an image:

1. Open the design in Designer.
2. Depending on the placeholder object that you use to import the image, do one of the following:

To	Do this
Rotate an image that was imported using a placeholder frame	a. Right-click the frame and select **Frame Properties**. The **Frame Properties** dialog box opens. b. Click the **Placeholder Frame Properties** tab. c. From the **Image Orientation** drop-down list, select the rotation that you want to apply to the imported image. d. Click **OK**. The **Frame Properties** dialog box closes.
Rotate an image that was imported using an empty image object	a. Right-click the empty image object and select **Image Properties**. The **Image Properties** dialog box opens. b. Click the **Image** tab. c. From the **Image Rotation** drop-down list, select the rotation that you want to apply to the imported image. d. Click **OK**. The **Image Properties** dialog box closes.
Rotate an image that was imported using a placeholder variable within an object	a. Double-click the placeholder variable, right-click and select **Variable properties**. The **Placeholder Variable Use Properties** dialog box opens. b. From the **Rotation** drop-down list, select the rotation that you want to apply to the imported image. c. Click **OK**. The **Placeholder Variable Use Properties** dialog box closes.

3.4.2 Formatting Text That Is Imported at Run Time

The following table provides information on the formatting options that are available for formatting text that is imported at run time:

Formatting methods for text imported at run time

Formatting method	Description
Encoding	An encoding defines how specific DBCS characters should be interpreted. For DBCS text, you can identify the encoding that is applied to the text to ensure that characters are generated accurately in customer output. For information on encoding, see *System Administration* in the Exstream Design and Production documentation.
Font resource management	Font resource management lets you identify the additional fonts, font styles, and DBCS font characters that are used in the text that is imported at run time, so that the appropriate font resources are included in the print stream at run time.
Paragraph properties (for RTF content only)	For RTF content, you can define how the content is incorporated with the surrounding text. By applying special paragraph settings, you can allow RTF content to use the formatting applied to the surrounding text, or allow the content to be imported using the original formatting.
Tag sets	Tag sets are used to set up and define formatting for tagged text. Tag sets can be useful if the original resource contains text that is already tagged with the formatting options you want to apply in Exstream. For more information about setting up a tag set, see *System Administration* in the Exstream Design and Production documentation.

To format text that is imported at run time, complete the following tasks as needed:

- "Using Font Resource Management to Ensure that Fonts are Available at Run Time" below
- "Formatting Paragraph Properties for RTF Files That Are Imported at Run Time" on the next page

Using Font Resource Management to Ensure that Fonts are Available at Run Time

Font resource management is a feature that helps you ensure that the fonts used in the imported external content are available for the print stream. Imported content can often contain fonts that are not used within the application design that you manage in Exstream. Because the engine is importing the content at run time, the engine does not know ahead of time which fonts or font format options to insert into the print stream. You can identify these font resources from the **Font Resources** tab of the application so that they are available to the engine at run time.

If you do not use font resource management to identify the fonts that are used in the external content, make sure that any fonts and font styles that are used by the imported content appear

on another object in the application, such as a page or a message. If the engine encounters a font in the imported text that does not exist in the application, you receive an error.

Note: You can also include TrueType fonts using the FONTDIRECTORIES engine switch. This switch searches specified directory paths for fonts that are used in content that is imported dynamically into your design at run time. For more information about the FONTDIRECTORIES engine switch, see *Switch Reference* in the Exstream Design and Production documentation.

When your application produces DBCS output, keep in mind that you must specify the DBCS font characters that are included in the package file, so that they are available to the engine at run time.

For information about specifying the DBCS font characters that are included in the package file, see *Creating Output* in the Exstream Design and Production documentation.

To use the font resource management feature to identify the fonts and font styles that are available at run time:

1. From Design Manager, drag an application to the Property Panel.
2. Click the **Font Resources** tab.
3. In the **Additional fonts to include in the package file** area, click  .
4. In the **Select Font** dialog box, select the font, font size, and font format options as needed. To specify a range of font sizes, select the **Add multiple sizes** check box.
5. Click **OK**.

   The **Select Font** dialog box closes and the font (including variations of point sizes and formats) is added to the **Additional fonts to include in the package file** area.
6. Optionally, if you want to include superscript or subscript characters for the font, complete the following steps:
   a. Click  .
   b. In the **Font Properties** dialog box, select the **Include Superscript/Subscript** check box.
   c. Click **OK**.

Formatting Paragraph Properties for RTF Files That Are Imported at Run Time

If you import an RTF text file, you can control how the engine handles paragraph breaks within the imported text. This option is useful if you are importing a text file in the middle of other text contained within a text box or table cell.

To control paragraph breaks for imported RTF text:

1. Open the design in Designer.
2. Right-click the placeholder variable and select **Variable Properties**.

   The **Placeholder Variable Use Properties** dialog box opens.
3. To control how the engine handles formatting paragraph breaks within the imported text file, select one of the following options:

To	Do this
Insert the content as a separate paragraph, with a blank line before and after the imported text	Select the **Import as separate paragraph** radio button.
Insert the content inline with the existing text	Select the **Inline text** radio button. The engine uses the paragraph and font settings specified until it reaches a hard return in the imported content. The engine then uses the paragraph properties of the external content source file for the remainder of the content. When the engine reaches the end of the import, the engine uses the original paragraph properties of the text box. For example, if your text box contains double-spaced paragraphs, followed by the placeholder variable, followed by more double-spaced paragraphs, the imported text will appear single-spaced in the output, but the text around it will appear double-spaced in the output. **Note:** If you inserted the placeholder variable inline with text in the design and you want to apply the paragraph properties of the RTF file to the text created in Designer, you must use the USE_RTF_PARA_PROPS switch in your control file. For information on the USE_RTF_PARA_PROPS switch, see *Switch Reference* in the Exstream Design and Production documentation.

4. Click **OK**.

   The **Placeholder Variable Use Properties** dialog box closes.

3.4.3 Importing External Content That Contains Variables

To keep external documents consistent with the content that is created in Exstream, you can embed Exstream variables in the text for some external files. Similar to variables in content that is designed in Exstream, the engine replaces variables in external content with the appropriate customer data when the document is imported at run time. This process lets external documents use the same customer data as documents that are designed in Exstream and ensures that all of the content that is delivered from Exstream maintains the same level of consistency in customer content, regardless of where the documents were originally created.

For example, suppose that your marketing department writes correspondence in Microsoft Word and these communications are intended to be imported into your Exstream design at run time. You can provide the marketing department with a list of customer variables (such as the customer's name, address, and account information), which they can then incorporate into the text of the letter. Then, when the correspondence is imported into Exstream at run time, the content will be updated to include the appropriate customer data.

This section discusses the following topics:

If you are importing a DOCX or DXF file that contains variables, you must also set up the placeholder document to allow the imported content to flow. Then, the placeholder document prevents content from overflowing the end of the page when the variable content is added during the engine run.

For more information about setting up a placeholder document to allow imported DOCX or DXF files to flow, see "Setting Up a Placeholder Document to Allow Imported DOCX or DXF Files to Flow" on the next page.

Adding Variables to External Content

Exstream variables are supported in the following types of external files:

- DOCX
- DXF
- HTML
- RTF
- TXT

If you want to include variables in external PDF files, you must first convert the PDF file into the DXF format, and then import the DXF file.

For more information about converting PDF files into the DXF format, see *Importing Designs* in the Exstream Design and Production documentation.

To add variables to external content, when you create the external document, enter the name of the variable you want to use within the text and enclose the variable name with angle brackets (<>). For example:

```
Congratulations, <CustomerName>! You qualify for a free <ServiceType>.
```

Because HTML tags are enclosed in angle brackets, you must use the HTML entities `<` and `>` to enclose the variable name. For example:

```
<p>Congratulations, &lt;CustomerName&gt;! You qualify for a free
&lt;ServiceType&gt;.</p>
```

You can also use double braces to enclose variables. For example:

```
<p>Congratulations, {{CustomerName}}! You qualify for a free
{{ServiceType}}.</p>
```

For more information about using variables in imported HTML files, see "HTML Considerations" on page 85.

Keep in mind the following considerations when including variables in external content:

- Any variable used in the external file must also be used somewhere in the application or package file in order for the variable to be recognized by Exstream.
- Variable names are case-sensitive and must match the variable name spelling exactly as it appears in the Exstream Library.
- If the engine identifies an entry as a variable, but is unable to match the variable name to a variable used in the application or package file, the entry in the external content will be ignored and left unchanged in the final output.
- Since variables can add more content to a design, be sure to either set up a placeholder document to allow the imported files to flow or provide enough white space in the original file to accommodate variable content. If variables add more content than can fit within the area provided and the document is not set up to flow, the final imported content can be truncated.

3.4.4 Setting Up a Placeholder Document to Allow Imported DOCX or DXF Files to Flow

When a DOCX or DXF file is imported at run time, content can overflow the end of the pages in the placeholder document if the document contains variables. By default, imported DOCX and DXF files are limited to importing content based on the original page layout and design.

Variables can affect the page layout of the imported DOCX or DXF file because when variable content is added during the engine run, the variables can sometimes add more text than can fit within the space that is provided in the original design. For example, suppose that you have a policy document that fills three pages from top to bottom before the variables are populated. The variables that are used in the document contain updated clauses from the legal department. When these clauses are added to the document, they could add another page of text.

If the placeholder document is set up to allow the imported DOCX or DXF file to flow, the engine can add additional pages to accommodate overflow that is caused by the addition of variable content to ensure that no content is lost when the file is imported.

To set up a placeholder document to allow an imported DOCX or DXF file to flow, you must complete the following tasks:

Do this	Description
Include the ALLOW_PLACEHOLDER_CONTENT_TO_FLOW switch in the control file.	When you prepare for production, you must include the ALLOW_PLACEHOLDER_CONTENT_TO_FLOW switch in the control file. The ALLOW_PLACEHOLDER_CONTENT_TO_FLOW switch allows the engine to repaginate the imported DOCX or DXF file at run time as variable content is added and as the content is placed within the pages of the placeholder document. To indicate that you want the engine to repaginate the imported DOCX or DXF file, you must enter `YES` as an argument for this switch. For example: `-ALLOW_PLACEHOLDER_CONTENT_TO_FLOW=YES`
For single page DXF files, you must set the placeholder page that is included in the placeholder document to flow to a specific flow page.	Single-page DXF files do not include the flow page references that are needed to accommodate overflow. In order to allow the content of a single-page DXF file to flow, you must set the placeholder page that is in the placeholder document to flow to a specific flow page. Exstream will then use the flow settings that are applied to the placeholder document to control the flow of the single-page DXF file, and any other flow settings that were previously applied to the single-page DXF file are ignored. To set a placeholder page to flow to a specific flow page: 1. In Design Manager, open the placeholder page in the Property Panel. 2. Click the **Flow** tab. 3. From the **Destination of overflow from this page** drop-down list, select **Flow to specified page**. 4. From the **Page** drop-down list, select the flow page that you want to use to accommodate overflow from the imported DXF file. For more information about setting up flow pages to accommodate flowing content, see *Designing Customer Communications* in the Exstream Design and Production documentation.
For multiple-page DXF files, copy the flow pages that are used in the original file and place a copy of those flow pages into the placeholder document.	Multiple-page DXF files can reference one or multiple flow pages; However, while the flow page references are maintained in these files, the DXF file does not contain a copy of the flow pages themselves. If you are importing a multiple-page DXF file, you must copy the flow pages that are used in the original DXF design and place a copy of those flow pages into the placeholder document. Including a copy of the flow pages in the placeholder document allows Exstream to continue to use the flow settings that were previously applied to the multiple-page DXF file when the file is imported.
For DXF files, validate the flow and relativity settings in the original DXF file.	As the engine paginates the imported content at run time, grow and relativity settings that were applied to the original DXF file content can affect the appearance of the DXF file in the final customer output. Objects within a DXF file behave as they were originally designed, even as content is repaginated to accommodate variable text. For example, suppose that a DXF file contains a text box, and beneath the text box is a table. If the **Move relative to the object** option for the table is set to **Above**, the table will be pushed down the page as the text box grows to accommodate the added variable text. If the **Move relative to the object** option for the table is set to **None (does not move)**, the text box can grow and overlap the table in the design. To ensure that the DXF file flows as expected after the engine adds variable content to the final customer output, validate the flow and relativity settings in the original DXF file. For more information about flow and relativity in DXF file designs, see *Designing Customer Communications* in the Exstream Design and Production documentation.

Note: When you import files at run time, Exstream can repaginate only DOCX or DXF files to prevent overflow that is caused by variable content or different page sizes. If you want to allow documents in other formats (such as PDF, InDesign, or Quark) to flow when they are imported at run time, you can use the Exstream conversion tools to convert external document formats to the DXF format before importing the document at run time.

For a full list of formats that can be converted to the DXF format and for more information about using the Exstream conversion tools, see *Importing Designs* in the Exstream Design and Production documentation.

Chapter 4: Format-Specific Considerations for Importing Content

Depending on the content you are importing, some formats have specific considerations that you should keep in mind either before importing content or while setting up a design to receive the content (either for design-time or run-time import).

This section discusses the following topics:

4.1 DOCX Considerations

Microsoft Word (DOCX) files are Office Open XML files that can be opened and edited in Microsoft Word 2007 and later. To help you get the most out of your DOCX files, you can import DOCX content inline with your design at run time. However, there are differences in the design features available in Microsoft Word and Exstream that affect the results you receive when importing DOCX content. For example, Exstream does not provide the same image formatting options available in Microsoft Word. For best results, you should review the design considerations in this section to help ensure that existing documents import as expected. You should also consider whether your document contains settings that are not supported in Exstream because there is no equivalent setting.

This section discusses the following topics:

4.1.1 Design Considerations for Importing DOCX Content

Microsoft Word and Exstream can have similar settings that are managed differently in each program. For example, both programs allow tables, but Exstream allows for more advanced table automation than is possible in Microsoft Word. Because of these program differences, you must keep in mind the following considerations that apply to any DOCX file that you want to import into Exstream:

Feature	Design considerations
Engine Switches	In Exstream, the engine requires the `docx2dxf.xsl` file in order to parse the DOCX file you are importing. By default, the engine expects to find the XSL file in the same directory as the engine. If the XSL file is in another directory, or if the XSL file has a different name, you must specify the location of the XSL file, its name, or both, using the DOCX2DXF_LOCATION engine switch. For more information about the DOCX2DXF_LOCATION engine switch, see *Switch Reference* in the Exstream Design and Production documentation.
Footnotes and endnotes	• You can include only one footnote per paragraph. • Endnotes are treated like footnotes in Exstream. Because of this difference, Exstream places endnotes at the end of each page, as they appear, instead of at the end of the document. • Because of the difference between how Microsoft Word and Exstream manage footnotes, the formatting applied to footnotes might appear different when they are imported into Exstream.
Headers and Footers	• Page numbers are supported only if they are placed inside the header or footer of the DOCX file. • If you are importing DOCX documents that contain multiple pages, Exstream supports only plain text headers and footers. Any other header or footer formatting can produce unexpected results. For more information about setting up an imported DOCX file to flow, see "Setting Up a Placeholder Document to Allow Imported DOCX or DXF Files to Flow" on page 75. • You must provide enough whitespace in the original DOCX file to accommodate headers or footers that contain variables. If you do not provide space for the variable text, headers might overlap the body of the page and footers might be truncated to fit the page.
Images	• Exstream can import only JPEG images from the DOCX content. If you include images in any other format, they will be ignored when the content is imported into Exstream. • Images must be placed inline with the text or behind the text. If you use any other setting, the image will be treated as if it is placed behind the text. • Rotated images are not supported. If you import a DOCX file that contains rotated images, the rotated images will be ignored.
Multiple documents that are imported using a single placeholder variable	In Exstream, if you use the same DOCX placeholder variable multiple times on the same design page, make sure that you are not importing complex content (such as documents with multiple pages or complex layouts) because this might cause unexpected results. For more information about creating and applying placeholder variables, see "Creating a Placeholder Variable" on page 45.

Feature	Design considerations
Page numbering	Page numbers are supported only if they are placed inside the header or footer of the DOCX file.
Pagination and indentation	Pagination and indentation of DOCX file content can appear different from the original design when it is generated as output from Exstream.
Section breaks	Section breaks are supported only in DOCX files that are imported using a placeholder document. If you are importing DOCX documents that are intended to flow, section breaks must be set to break to the next page. For more information about setting up an imported DOCX file to flow, see "Setting Up a Placeholder Document to Allow Imported DOCX or DXF Files to Flow" on page 75.
Tables	• Row height is always adjusted by the engine. If your design contains tables, Exstream might adjust the tables size when the table is imported into Exstream. • If a placeholder variable that you are using to import a DOCX file is located within a table in your Exstream design, the imported DOCX content must not include tables. A DOCX file that contains tables will fail to import when using a placeholder variable within a table.
Tables of contents	• Tables of contents must use the default "dot" character as a tab leader character. • Tables of contents should not contain hidden page numbers. When the DOCX file is imported, all of the page numbers will be visible. • Tables of contents support only standard formatting. If additional formatting is applied to a table of contents, then the extra formatting will be ignored when the DOCX file is imported. • Tables of contents do not support captioned items, paragraph outlining, and table of contents entry fields.
Text and Text Formatting	• Bookmarked text entries are not supported in Exstream. These bookmarks are ignored when the DOCX file is imported into Exstream. • Bullet characters in a DOCX file are automatically converted to the bullet characters that are supported in Exstream. • Bullets must be followed by content in order for them to appear in Exstream. For example, if a bulleted item contains only empty spaces, the bullet will be removed when the content is imported into Exstream. • Kerning can be applied to an entire text box, but not to individual words or lines of text. • Rotated text is not supported. If you import a DOCX file that contains rotated text, the rotated text will be ignored. • Single line strike-through formatting on text is supported, but any other strike-through formatting (such as double-line strike-throughis) is not supported in Exstream.
Text boxes	• Embedded text boxes must not be set to split. Exstream does not support splitting embedded text boxes. If the DOCX file you import contains embedded text boxes that are set to split, you will receive a circular flow error. • Text boxes must be placed inline with the text. • Rotated text boxes are not supported. If you import a DOCX file that contains rotated text boxes, the rotated text boxes will be ignored.

For information about how to access and apply specific design features to a DOCX file, see the Microsoft Word help.

4.1.2 DOCX Features That Are Not Supported in Exstream

Some Microsoft Word settings and design elements do not have an equivalent option in Exstream and therefore cannot be replicated in Exstream output. For example, Microsoft Word documents might contain image formats that are not supported in Exstream. The following table lists the features—organized by the object to which they apply—that are not supported in Exstream:

For this object in the DOCX file	The following design objects or features are not supported
Page setup	• Page columns
Text	• Applying a background color to text (including highlight colors) • Applying left indents that are pushed into the page margin • Applying line numbers • Applying special capitalization, including all caps, drop caps, and small caps) • Applying the underline style of "words only"
Paragraphs	• Applying a background color to individual paragraphs of text (including highlight colors) • Applying borders, other than single lines, to paragraphs of text • Applying shape templates to paragraphs of text
Images	• Embossing images • Imprinting images • Outlining images • Applying a shadow to images • Microsoft Visio drawings
Tables	• Allowing rows to span (vertical spanning)

4.2 DXF Considerations

The DXF format is an Exstream file type, based on the XSL Formatting Objects (XSL-FO) standard, that allows you to create and modify Exstream-editable content in external programs.

You can import a DXF file at design time or during production.

For more information about the structure of a DXF file, see the *DXF Reference* in the Exstream Design and Production documentation.

If you maintain or create applications or designs outside of Exstream, you can use conversion tools to convert those designs into DXF files. When you import a DXF file into a design, the content in the DXF file is imported as design objects which are editable just like other design objects created directly in Designer. For example, suppose an organization in your enterprise designs mailings in an external program. The organization can store the mailing designs as PDFs and you can use the PDF Converter to convert those files and import them into the Design Manager Library. You can then interact with the designs in Design Manager and Designer just as you do with other types of objects.

For more information about the converters available from Exstream, see *Importing Designs* in the Exstream Design and Production documentation.

As with other file types, DXF files can also be imported at run time. When you import a DXF file, you can import a portion of a design (for example, a paragraph) and have it appear in the output as if it were designed in Designer. For example, suppose that your marketing organization creates and maintains product descriptions. You can save the content as XML and then create an XSLT that transforms the content into a DXF file. Then, you can import the product descriptions as formatted text at run time so that in the output they retain their formatting and appear inline with text that was created in Designer.

For more information about importing content at run time, see "Importing Content into a Design at Run Time" on page 42.

Before importing a DXF file, review the following considerations:

- You can use inline placeholder variables to import paragraphs (`dlg:paragraph`), sections (`dlg:section`), or formatted text (`dlg:dxf-text`) at run time. The objects and any embedded objects are imported into the application at run time.
- You can use placeholder documents to import pages (`dlg:page`) or documents (`dlg:document`) at run time, but you cannot import applications or application-level objects, such as messages and campaigns, at run time.
- You cannot use a placeholder document to import paragraphs (`dlg:paragraph`), sections (`dlg:section`), formatted text (`dlg:dxf-text`), or library components (`dlg:library-component`) at run time.
- You cannot use an inline placeholder variable to import pages (`dlg:page`), documents (`dlg:document`), or library components (`dlg:library-component`) at run time.
- If you want to import a library component (`dlg:library-component`) at run time, the library component must be embedded in another supported object.
- Library component references (`dlg:library-component-ref`) are not supported because the engine does not read the file in order to find the referenced library component.
- Rotated objects are not supported.

- Live Output supports importing DXF content at run time; however, the content is read-only and you cannot edit it.
- If you are importing DXF content that contains images at run time, then the images must be in JPEG or B&W TIFF (uncompressed or CCITT Group 4 compressed) format. Use the `import-type` attribute of the `dlg:image` element to specify the format of an included image.
- The engine uses the Exstream Object and Content DTD (`ExstreamObjectAndContent.dtd`) to parse the DXF being imported at run time. By default, the engine expects the DTD to be in the same directory as the engine. If the DTD is in another directory, or if the DTD has a different name, you must specify the location of the DTD, the name of the DTD, or both, using the EXSTREAM_DTD_LOCATION engine switch.

  For more information about the EXSTREAM_DTD_LOCATION engine switch, see *Switch Reference* in the Exstream Design and Production documentation.
- You can create a sample DXF file that allows you to see the XML elements for existing objects that are similar to objects you want to import.

  For more information about creating a sample DXF file, see "Creating a Sample DXF File to Help You Understand How an Existing Design is Constructed" below.
- Importing DXF content at run time is not supported on the z/OS platform.

For more information about specific DXF elements, see the *DXF Reference* in the Exstream Design and Production documentation.

4.2.1 Creating a Sample DXF File to Help You Understand How an Existing Design is Constructed

In working with DXF, it can be helpful to understand the XML structure for existing Exstream design objects. You can export a DXF file from the Library to see the structure for existing objects that are similar to objects you want to import. Because DXF is XML, you can open the file in any XML or text editor.

When exporting DXF, keep in mind the following considerations:

- Although most design objects are supported in DXF, certain objects with limited support might not be exported fully.
- Some objects that are not part of the physical design, such as design groups, are not supported in DXF and are therefore not exported.
- If you plan to import the sample DXF back into the design environment, you should first make sure that the DXF contains all the expected objects and properties before overwriting existing objects.
- If attributes in the exported DXF reference objects in the Library, the DXF can be imported only into the same database in order for those for references to stay intact. Importing DXF that contains such references into a different database at design time might create new

objects where referenced objects are missing, and the objects imported from the DXF might need modification to function as expected. DXF that contains such references cannot be imported into a different database at run time.

To create a DXF file from an existing design:

1. In Design Manager, in the Library, right-click a Library object that you want to export to DXF.
2. Click **Export to DXF**.

4.3 EPS Considerations

If you want to use vector images in a design, Exstream preserves and stores the original vector data from imported EPS images, which are commonly used for high-resolution printing. Any PANTONE Color information included in the image, which may be needed during printing, is also stored.

When importing EPS images, keep in mind the following considerations:

- When importing EPS images at design time, you must have licensed the Design PDF module.
- EPS files must contain `%%BoundingBox`.
- Output of each of the following types includes the original EPS vector data from imported EPS images:
  - AFP
  - PostScript
  - PPML
  - TOP
  - VIPP
  - VPS
- The engine converts EPS vector data to a supported vector format for output of the following types:
  - PDF
  - PDF/A
  - PDF/VT
  - VDX

> **Note:** To convert EPS vector data for use in these outputs, you must install Ghostscript.
>
> Ghostscript is a commercially available PostScript and PDF conversion and rendering tool. For information about installing Ghostscript, see *Installation and Upgrade Information* in the Exstream Design and Production documentation.

- When you import an EPS image at design time, Designer rasterizes the image for display, which allows Designer both to display the image at a higher resolution than the thumbnail included in the image file and to display EPS images that do not contain thumbnails. Areas of the image that do not contain vector objects appear transparent in Designer.
- When you use outputs types that do not support vector data, the engine rasterizes each EPS image in the output.
- When the engine rasterizes an EPS image for a particular output, areas of the image that do not contain vector objects appear transparent in output types that support image transparency.

4.4 HTML Considerations

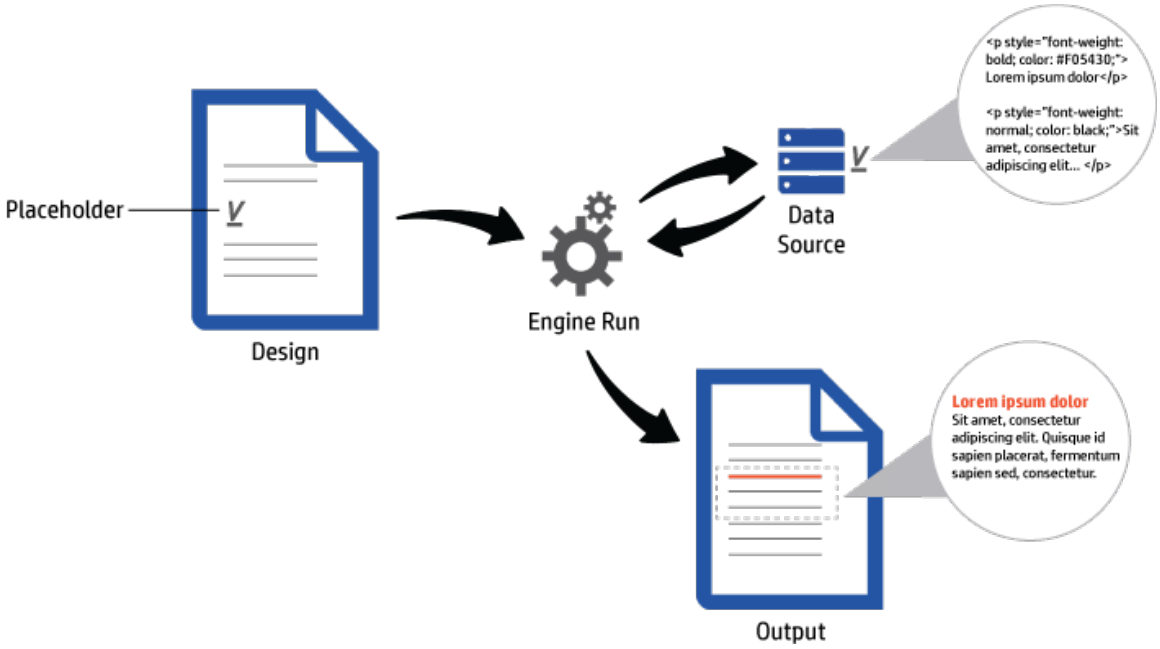
On Windows and Linux, Exstream supports importing HTML content at run time, for use with all output types except Live (*.dlf). This feature is intended to be used with snippets of HTML content rather than complex web applications. For example, suppose that your marketing organization produces promotional content in HTML for use on the company website. You can import this HTML content at run time so that the promotions appear inline along with content that was created in Designer.

For information about importing content at run time, see "Importing Content into a Design at Run Time" on page 42.

As with other types of dynamically imported content, you must use placeholder variables to identify the HTML content to import at run time. Exstream supports HTML placeholder variables as text only. You cannot use an HTML placeholder variable with a placeholder document or an empty image object.

For more information about placeholder variables, see "Creating a Placeholder Variable" on page 45.

Example of how placeholder variables work with HTML import

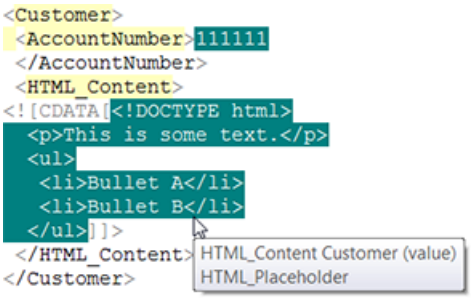


When importing HTML content, you can set up your placeholder variable so that it provides instructions to the engine using one of two different methods. The following table provides an overview of the steps for each method and examples:

Methods for importing HTML content

To use this method	Do this	Examples
Define a path to the HTML content	1. In Design Manager, create an HTML placeholder variable and specify which HTML content to import, using one of the available methods to define the variable value: • Specify a static user value for the file path of the HTML content. • Use a formula to determine which HTML content to use. • Map the placeholder variable to a data file with file paths for the HTML. 2. In Designer, insert the placeholder variable inline in a design object that can contain text (for example: paragraph object, table cell, text box, or text message).	For example, it might look like this if you map the placeholder variable to a file path in an XML data file: <Customer> <AccountNumber>111111 </AccountNumber> <HTML_Content>\\filepath\message.html </HTML_Content> </Customer> HTML_Content Customer (value) HTML_Placeholder

Methods for importing HTML content, continued

To use this method	Do this	Examples
Include the HTML content in a data file	1. In Design Manager, create an HTML placeholder variable with **File only** selected from the **Source** drop-down list in the variable properties. 2. Create a data file that references the HTML data source on the **Test Data Source** and **Production Data Source** tabs. 3. Map the placeholder variable to the data area that contains the HTML content. 4. On the **Data Area Properties** dialog box, select **Placeholder content** from the **Format** drop-down list. 5. From the **Encoding** drop-down list, select an option: • **None**—Select if the content was not encoded. • **Base-64**—Select only if the content was encoded prior to the engine run. 6. In Designer, insert the placeholder variable inline in a design object that can contain text (for example: paragraph object, table cell, text box, or text message).	For example, it might look like this if you map the placeholder variable to HTML content in an XML data file:  **Note:** If you're using XML input, you must wrap the HTML content in a CDATA tag.

When the engine encounters imported HTML, it builds a set of default styles before parsing the HTML. These styles are based on the placeholder variable formatting in Designer. The engine parses each import only one time.

For best results, create an HTML placeholder variable for each instance you want to import. If you insert the same HTML placeholder variable on the design page multiple times, this set of default styles is created for the first instance of the HTML, and then used for all subsequent instances of the placeholder variable.

For example, suppose the first instance of the HTML placeholder variable is in a paragraph formatted as 8-point font and a later instance of the variable is in a paragraph formatted as 15-point font. In the output, all instances of the variable will be formatted as 8-point font, regardless of the formatting applied to the placeholder variable in Designer.

There are two exceptions to this behavior. If the imported HTML includes inline styles, those are honored, since inline styles override formatting applied in Designer. Additionally, if you're producing unconverted HTML or HTML (email) output, the imported HTML will be formatted as expected. For information about unconverted HTML output, see "Converted vs. Unconverted HTML Output" on the next page.

To be sure that imported HTML appears in the output as expected, your imported HTML should be well-formed and consistent with the guidelines outlined in the following topics:

4.4.1 Converted vs. Unconverted HTML Output

If your application produces HTML output, you have the option to convert the imported HTML to Exstream format, or to place it—unconverted—into HTML or HTML (email) output.

You can specify whether the imported HTML is converted in the placeholder variable properties, on the **Placeholder** tab. When **HTML** is selected from the **Placeholder** drop-down list, the **For HTML output, do not convert** check box becomes active. The check box is selected by default, but this setting affects only HTML output. The engine ignores it for all other output types. If you clear the **For HTML output, do not convert** check box, then the imported HTML content is converted.

In most cases, you will leave this check box selected, so that the imported HTML is not converted in the HTML output. This way, the imported HTML is displayed in the browser according to the styles applied inline within the HTML. If you do not convert the imported HTML, the engine includes a comment in the HTML output to identify the imported, unconverted HTML as pre-composed content.

If you decide to convert the imported HTML to Exstream format, then the resulting output will be limited to only those formats that are supported within Exstream. As a result, you might see some visual differences between the imported HTML and HTML output. For example, you can add borders around text in HTML, but not in Designer. You might also see differences in the text alignment due to the way that the engine calculates tab, margin, and padding sizes. More information on expected differences between converted and unconverted HTML in HTML output for various design elements are outlined in this section.

If you produce accessible HTML output, you should also consider which design objects Exstream supports for accessibility, and how the engine creates the HTML structures in HTML and HTML (email) output.

For more information about producing accessible HTML output, see the *Designing Customer Communications* in the Exstream Design and Production documentation guide.

4.4.2 Embedding Variables in Imported HTML

Embedding variables in imported HTML content lets you leverage the same data you already use in the application, so that you can add to documents in Designer. For example, suppose you want to import HTML content that addresses your customer by name. Within the imported HTML, you would include the variable associated with this data, and then the customer name would be included in the output.

You can embed any type of variable except the following:

- Formatted text
- Placeholder
- Tagged text

As with other file formats that support embedded variables, you can identify variables within imported HTML content with angle brackets. However, since HTML uses angle brackets to identify tags, you must use the HTML entities `<` and `>` or `&#60;` and `&#62;`. You can also use double braces. An embedded variable for the customer name could look like `<CustomerName>` or `{{CustomerName}}`.

Variable names are case-sensitive, so any variables included in the imported HTML must exactly match the variable name in the Library. If the variable names in the imported HTML do not match any variables included in the application or package file, then the content will remain unchanged in the output. For example, suppose the variable name used in the application is `'Customer_Name'`, but the imported HTML contains `Dear {{CustomerName}}` and the variable `'CustomerName'` does not exist in the application. Instead of the output including something like `Dear Susan`, it will contain `Dear {{CustomerName}}`.

> **Note:** For converted output, Exstream does not support using embedded variables to define any content other than text. This means that you can not use variables to define file paths or URLs, such as for hyperlinks or image references, or for generating HTML, styles, attributes, or attribute values. Because the engine parses each import one time only and replaces the variables after parsing, these variable values will not be available at the time the engine needs them to set style properties or attributes in the HTML.

If you embed an array variable within the imported HTML content, and the placeholder variable is in an automated table row, then the embedded array variable will resolve to the automatic array element that matches the current table row index. For example, in row 1, the embedded variable resolves to the first array element, row 2 to the second array element, and so on. If you place an array within any other object that contains text, the value in the output will be the first element in the array.

For more information about adding variables to content imported at run time, see "Importing External Content That Contains Variables" on page 73.

4.4.3 Font Management for Imported HTML

For text to appear correctly in the output, the package file must contain all font variants expressed in the imported HTML content, including the font families, sizes, and styles. For example, if your imported HTML contains `<p style="font-family: Arial; font-size: 12pt;">`, then you must include Arial 12-point in your package file.

For font styles, such as bold and italics, you must add bold and italic variants in your package file for each font family and size declared in the HTML. For example, if your imported HTML contains `<p style="font-family: Arial; font-size: 12pt; font-weight: bold; font-style: italic;">`, then you must include Arial 12-point bold italic in your package file. You must also include the font family and size variant for any text that is enclosed within valid elements for bold and italic styles. For example, suppose that you have the following HTML:

```
<div style="font-family: Times New Roman; font-size: 10pt;">
 <p>Some text with <b>bold</b>!</p>
</div>
```

The package file must contain both Times New Roman 10-point regular and Times New Roman 10-point bold variants.

> **Note:** If you convert your imported HTML, any font-family information in your imported HTML content is lost when you convert.

When considering whether you have all of the correct font sizes included in the package file, determine whether your design uses calculated font sizes, such as with superscript and subscript, or font sizes based on pixels, em values, or percentages. For example, if your imported HTML contains `<p style="font-family: Arial; font-size: 150%;">`, and the parent structure defines the font size for `<p>` as Arial 12-point, then you must also include Arial 18-point.

If your imported HTML contains text that uses non-Latin DBCS character sets (such as Arabic, Cambodian (Khmer), Farsi, or Hebrew), then you must make sure that the package file includes the appropriate character set for that language. You must also enable complex text layout. For more information about including character sets and enabling complex text layout, see *System Administration* in the Exstream Design and Production documentation.

If you do not have the correct fonts in the package file, the engine will substitute another font and issue a message.

You can make sure that all of the fonts are included in the package file in the following ways:

- Specify the font variants to include in the package file (recommended)
- Include all of the font variants on a design page or non-printing fonts page

For more information about including fonts in the package file, see “Using Font Resource Management to Ensure that Fonts are Available at Run Time” on page 71.

4.4.4 Images in Imported HTML

You can include references to images in your imported HTML content for outputs that support run-time import. The image source must be defined using a URL or file path rather than a variable. For example, suppose that you use a variable to specify which image to include for a particular locale. If you include that variable in the imported HTML content, the image will not appear in the output. That is, including `<img src="{{ImgPath}}">` results in a missing image in the output.

Exstream supports importing PNG and JPG images for all output types that support them at run time. You can also import GIF images if you're producing any of the following output types:

- DOCX
- HTML
- HTML (email)
- Multi-Channel XML

Keep in mind that if you import animated GIFs, they will be animated only in HTML output. For more information about importing images at run time, see "File Formats Supported for Importing" on page 11

Transparent images in PNG format are also supported for imported HTML content. For information about considerations for importing transparent PNG files at run time, see "Importing Images with Transparency at Run Time" on page 57.

Review the following table for more information about how to apply styling to images in your imported HTML:

Image styling	How it works
Alternate Text	You can specify alternate text using the `alt` attribute in your `img` tag for the following output types: • HTML • HTML (email) • Multi-Channel XML • PDF (with accessibility tagging turned on) If you're producing accessible output, you can specify a reading language for the alternate text by including the `lang` attribute. For more information about specifying the reading language, see "Specifying the Accessibility Reading Language in Imported HTML" on page 96.

Image styling	How it works

Borders	You can add a border to images using border style properties to define the color, style, and width. The engine applies the first border style that it encounters. It searches in the following order: top, right, bottom, left. For example, if you have the top border width defined as 2 pixels and right border defined as 4 pixels, the engine will apply a 2-pixel border to all sides of the image.
	
	Border color values can be expressed using hexadecimal, RGBA values, or HTML color names. If you do not specify a border color, the engine will apply the color defined in the parent element. If there is not one specified, then the engine will apply a default color.
	
	You can use the following border styles: `none`, `dashed`, `dotted`, or `solid`. If you do not specify a border style, the engine will not place a border around the image. If you specify an unsupported border style, the engine applies a solid border.
	
	Border widths can be expressed in points, pixels, or em values. If you do not specify a border width, the engine will assign a default width.
	
	When declaring border styles, you can use the individual style properties for color, style, and width; or, you can use the shortcut border property to define these properties. For example, both of the the following are valid:
	
	`<img src="path/image.png" style="border-color:#504A4B; border-style: solid; border-width=3px;">`
	
	`<img src="path/image.png" style="border:#504A4B solid 3px;">`

Image styling	How it works
Float and Clear	When you "float" an image, the engine places the image to the left or right of the current block and the content wraps around the image. You can use any of the following `float` values: `left`, `right`, `inherit`, or `none`. Subsequent content floats around the image until the engine encounters a `clear` declaration in the HTML. Exstream supports the following `clear` values: `left`, `right`, `both`, or `none`. Keep in mind that wrapping is constrained by the size of the containing object. **Example of an image floated left of the text**  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent consequat pellentesque odio. Nullam molestie lectus non finibus rhoncus. Nam a hendrerit arcu. Nam eu fermentum dui. Nunc vitae sapien a nisl sollicitudin commodo a sit amet nibh. Pellentesque pharetra magna at libero dignissim, nec congue magna tempus. Quisque pellentesque faucibus eros, sed vulputate ligula hendrerit non. **Example of an image floated right of the text** Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent consequat pellentesque odio. Nullam molestie lectus non finibus rhoncus. Nam a hendrerit arcu. Nam eu fermentum dui. Nunc vitae sapien a nisl sollicitudin commodo a sit amet nibh. Pellentesque pharetra magna at libero dignissim, nec congue magna tempus. Quisque pellentesque faucibus eros, sed vulputate ligula hendrerit non.  **Example of an image floated left with a clear declaration before the text**  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent consequat pellentesque odio. Nullam molestie lectus non finibus rhoncus. Nam a hendrerit arcu. Nam eu fermentum dui. Nunc vitae sapien a nisl sollicitudin commodo a sit amet nibh. Pellentesque pharetra magna at libero dignissim, nec congue magna tempus. Quisque pellentesque faucibus eros, sed vulputate ligula hendrerit non. If you do not include `float` in your image styling, then the image is placed inline within the current block, and it affects the line height. **Example of an image without float settings defined**

Image styling	How it works
	 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent consequat pellentesque odio. Nullam molestie lectus non finibus rhoncus. Nam a hendrerit arcu. Nam eu fermentum dui. Nunc vitae sapien a nisl sollicitudin commodo a sit amet nibh. Pellentesque pharetra magna at libero dignissim, nec congue magna tempus. Quisque pellentesque faucibus eros, sed vulputate ligula hendrerit non. Float is supported in the following output types: • AFP • HTML (using container designs) • IJPDS • MIBF • PCL • PDF • PDF/A-2a • PDF/VT • PostScript • PPML • TIFF • VIPP • XML (multi-channel) **Tip:** If you are producing HTML output from a standard design, floated images might not appear as expected in the output. For best results, use a container design. For more information on container design, see *Designing Customer Communications* in the Exstream Design and Production documentation.

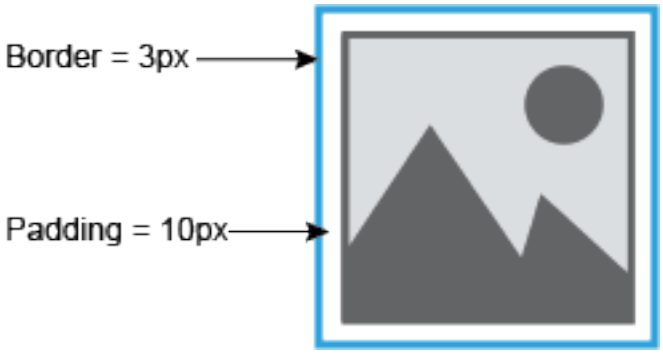
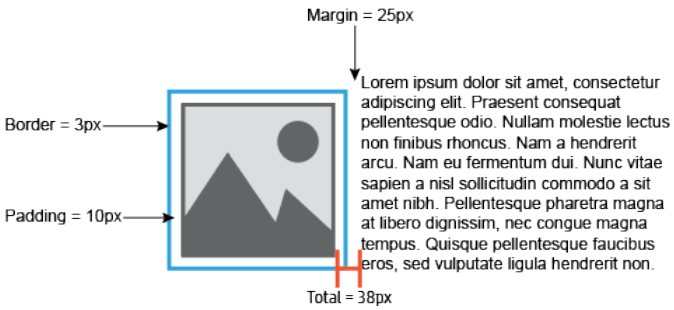
Image styling	How it works
Hyperlinks	Hyperlinks on images are supported in the following output types: • DOCX • HTML • HTML (email) • Multi-Channel XML • PDF • PDF/VT • PowerPoint **Note:** You must use a file path or URL when declaring the `href` value. Using embedded variables to define this value is not supported.
Margins and Padding	The engine honors margins and padding as white space that text cannot encroach upon. Margin and padding widths can be expressed in points, pixels, em values, or percentages. If the image has both padding and a border applied, then there will be white space between the image and the border. **Example of an image with a border and padding**  Margins, padding, and borders are cumulative, which means that the white space between the image and neighboring objects or text will be calculated based on the total width. **Example of an image with a border, padding, and margins** 

Image styling	How it works
Width and Height	Image width and height settings can be expressed in pixels only. If you do not declare a width or height, then the engine imports the image at actual size. If you attempt to use percentages to define the width or height of an image, the engine reads the percentage as pixels. For example, if your imported HTML includes `<img src-"image.jpg" width="50%">`, then your image will be sized to 50 pixels in the output. If you declare a width but not a height, then the engine calculates the height based on the ratio of the declared width to the width of the original image. The inverse is true if you declare a height but not a width. If an image is taller than the object that contains it, and the object is not set to grow to accommodate the image, then the image will not appear in the output.

4.4.5 Specifying the Accessibility Reading Language in Imported HTML

If you're importing HTML content, and you want to specifically define the reading language that will be used by accessibility tools for the imported content, then you must include the `lang` attribute in the imported HTML. If you do not include a `lang` attribute on the top-level tag, then Exstream applies the language setting of the text paragraph that contains the placeholder variable as the default language.

If you add a `lang` attribute on a `<div>`, `<table>`, or `<p>` tag, Exstream applies that language to all child elements contained within that tag, unless the child element includes a `lang` attribute.

For example, suppose that you have the following imported HTML content:

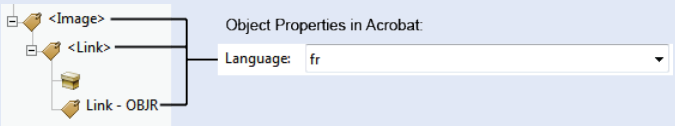
```
<div lang="es">
        <h2>¿Habla español?</h2>
        <p>Llame a uno de nuestros representantes de habla hispana.</p>
        <img src="phone.jpg" alt="Image of a phone" lang="en-US">
</div>
```

In this example, the corresponding Designer element is an image embedded in a text box. The specified reading language for the text box is Spanish, and the reading language for the image's alternate text is English. Because the child element (in this case, the image), includes a `lang` attribute value, the alternate text of the embedded object would be read using English pronunciation rules.

For a list of accessibility languages supported in Exstream, see “Supported Accessibility Language Codes” on page 99.

You can include the `lang` attribute for the following elements:

HTML tag category	Tags that support the `lang` attribute during HTML import
Basic HTML structure	• `<html>` • `<body>` • `<h1>` through `<h6>` • `<div>` • `<p>` **Note:** With unconverted HTML imports, the engine removes the `<html>` and `<body>` tag from the resulting HTML output, and any language declarations made in those tags would be lost. For converted HTML, however, these tags are retained in the HTML output, as would the language declarations. For more information about converted and unconverted HTML content, see the *Importing External Content* in the Exstream Design and Production documentation guide.
Hyperlinks	• `<a>`

<table>
<tr><th>HTML tag category</th><th>Tags that support the <code>lang</code> attribute during HTML import</th></tr>
<tr><td>Images</td><td>
<code><img></code>
Tip: If an image is a hyperlink, and the <code>lang</code> attribute has different values on the <code><a></code> and <code><img></code> elements, the behavior for imported HTML content differs from that of images placed in Designer. For images included in the imported HTML content, the <code>lang</code> attribute value on the <code><img></code> tag is honored in the output. For images placed in Designer, the <code>lang</code> attribute value on the <code><a></code> tag is honored in the output.
For example, suppose that an image has French specified as the reading language for the alternate text and the hyperlink has English specified as the reading language. In typical HTML, it might appear like the following:
<pre>
        <img src="logo.jpg" alt="Notre logo d'entreprise" lang="fr">
</pre>
In accessible HTML and PDF output, French is specified as the reading language for both the alternate text and the hyperlink.
For example, the imported HTML content would result in the following converted HTML output:
<pre>
        <img src="logo.jpg" alt="Notre logo d'entreprise" lang="fr">
</pre>
The same imported HTML content would result in the following language properties in accessible PDF output:


Whereas the Designer content would result in the following HTML output:
<pre>
        <img src="logo.jpg" alt="Notre logo d'entreprise" lang="en">
</pre>
In the HTML output for either case, incorrect rules of pronunciation will be applied to either the image alternate text or the hyperlink. To avoid this issue, make sure that the <code>lang</code> attributes for the <code><a></code> and <code><img></code> elements contain the same value. If you are producing HTML output, you can also you can choose to produce unconverted HTML output to maintain the language declaration for the image and alternate text. PDF output honors the language settings that you apply in Designer.
</td></tr>
<tr><td>Lists</td><td>

<code><dd></code>
<code><dl></code>
<code><dt></code>
<code>
</code>
<code>
</code>
<code>
</code>

</td></tr>
</table>

HTML tag category	Tags that support the `lang` attribute during HTML import
Special text formatting	• `<blockquote>`
Tables	• `<table>` • `<th>` • `<tr>` • `<td>` • `<thead>` • `<tbody>` • `<tfoot>`

4.4.6 Supported Accessibility Language Codes

To specify an accessibility language, you can select the language that you want from the **Accessibility language** list on the **Accessibility** tab of the object properties. You can also use an ISO 639–1 or ISO 639–2 language code by creating a scalar string variable that contains a custom ISO language code as its value. For more information about creating a variable, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

Important: Keep in mind that accessibility languages are the languages that are available to be read by accessibility tools. The list of accessibility languages does not represent the languages supported in Exstream output.

For more information about setting the reading language for accessible HTML output or for accessible PDFs, see *Designing Customer Communications* in the Exstream Design and Production documentation.

ISO 639–1 and ISO 639–2 language codes supported in Exstream (case-sensitive)

Language name	ISO 639–1	ISO 639–2
Amharic	am	
Arabic	ar	
Armenian	hy	
Bengali	bn	
Catalan	ca	
Cebuano		ceb

ISO 639–1 and ISO 639–2 language codes supported in Exstream (case-sensitive), continued

Language name	ISO 639–1	ISO 639–2
Chinese (PRC)	zh-CN	
Chinese (Taiwan)	zh-TW	
Chinese (Hong Kong SAR)	zh-HK	
Chinese (Singapore)	zh-SG	
Czech	cs	
Danish	da	
Dutch	nl	
English (American)	en-US	
English (Australian)	en-AU	
English (British)	en-GB	
Farsi (Persian)	fa	
Finnish	fi	
French	fr	
French (Canadian)	fr-CA	
French (Creole)		cpf
German	de	
Gujarati	gu	
Hawaiian		haw
Hindi	hi	
Hmong		hmn
Hungarian	hu	
Igbo	ig	
Ilokano		ilo
Italian	it	

ISO 639–1 and ISO 639–2 language codes supported in Exstream (case-sensitive), continued

Language name	ISO 639–1	ISO 639–2
Japanese	ja	
Khmer	km	
Korean	ko	
Kru		kro
Lao	lo	
Marshallese	mh	
Navajo	nv	
Nepali	ne	
Norwegian	no	
Norwegian (Bokmål)	nb	
Norwegian (Nynorsk)	nn	
Oromo	om	
Pohnpeian		pon
Polish	pl	
Portuguese	pt	
Portuguese (Brazilian)	pt-BR	
Punjabi	pa	
Romanian	ro	
Russian	ru	
Samoan	sm	
Spanish	es	
Swedish	sv	
Tagalong	tl	
Thai	th	

ISO 639–1 and ISO 639–2 language codes supported in Exstream (case-sensitive), continued

Language name	ISO 639–1	ISO 639–2
Tongan	to	
Turkish	tr	
Ukrainian	uk	
Urdu	ur	
Vietnamese	vi	
Yoruba	yo	

4.4.7 Tables in Imported HTML

Exstream supports importing tables that are constructed using the HTML `<table>` structure.

Important: Exstream does not support embedding an image or a table within a table in the imported HTML content.

With tables, you can specify background colors, colspans, and widths. When you declare styles for multiple parts of a table, the styles are honored in the following way:

- `<table>`—Any style declared in a `<table>` tag applies to the whole table, but can be overridden by styles defined in the `<tr>`, `<td>`, or `<th>` tags.
- `<tr>`—Any style declared in the `<tr>` tag applies to the whole row, and overrides styles defined in the `<table>` tag.
- `<td>`—Any style declared in the `<td>` tag defines the defaults for the cell, and overrides styles defined in the `<table>` or `<tr>` tags.
- `<th>`—Any style declared in the `<th>` tag defines the defaults for the header cell, and overrides styles defined in the `<table>` or `<tr>` tags. Text in a `<th>` tag is centered and bold by default, but this can be overridden with other styles.

Review the following table for more information about how to apply styling to tables in your imported HTML:

Table styling	How it works
Alignment	You can align text horizontally and vertically on the `<td>` and `<th>` tags.

Table styling	How it works
Background color	It's a best practice to declare background colors on the `<td>` and `<th>` tags, but you can also declare them on the `<table>` and `<tr>` tags. Background color values can be expressed using hexadecimal, RGBA values, or HTML color names.

Borders	It's a best practice to declare borders on the `<td>` and `<th>` tags, but you can also declare them on the `<table>` and `<tr>` tags.
	
	Border color values can be expressed using hexadecimal, RGBA values, or HTML color names. If you do not specify a border color, the engine will apply a default color.
	
	You can use the following border styles: `none`, `dashed`, `dotted`, or `solid`. If you do not specify a border style, the engine will not place a border around the table or table cell. If you specify an unsupported border style, the engine applies a solid border.
	
	Border widths can be expressed in points, pixels, or em values. If you do not specify a border width, the engine will assign a default width.
	
	When declaring border styles, you can use the individual style properties for color, style, and width; or, you can use the shortcut border property to define these properties. For example, both of the following are valid:
	
	`<td style="border-color:#504A4B; border-style: solid; border-width=3px;">`
	
	`<td style="border:#504A4B solid 3px;">`

Column Spans	You can set a table cell to span multiple columns by adding the `colspan` attribute to the `<td>` tag. For example, if you declare `<td colspan="3">` in your HTML, then that particular cell will span three columns.
Float and Clear	When you "float" a table, the engine places the table to the left or right of the current block and the text wraps around the table. You can use any of the following `float` values: `left`, `right`, `inherit`, or `none`. Subsequent content floats around the table until the engine encounters a `clear` declaration in the HTML. Exstream supports the following `clear` values: `left`, `right`, `both`, or `none`. Keep in mind that wrapping is constrained by the size of the containing object. If you do not include `float` in your table styling, then the table is placed inline within the current block, and it affects the line height. Float is not supported in the following output types: • 3211 Line Data • Edgar HTML • HTML (email) • Live • PowerPoint • RTF • XML (composed) • XML (content) • ZPL

Table styling	How it works

Layout tables	When designing for accessible HTML output, if you have tables that should not be read in table format (for example, tables used to create the design layout), you can design them as layout tables. To be considered a layout table, the table structure must not contain `<thead>` or `<th>` tags, or any `<td>` tags with `caption` or `scope` attributes. For example:
	`<table>`
	`<tr>`
	`<td>Text</td>`
	`</tr>`
	`</table>`
	If, however, you are designing for accessible PDF output, you must specify that a table is a layout table in Designer. You can do this in the **Table Properties** dialog box, on the **Accessibility** tab. From the **Read as** drop-down list, select **Division**.

Width and Height	The width of a table (`<table>`) or table cell (`<td>` or `<th>`) can be expressed in pixels, em values, or percentages. You can not declare a height on a `<table>` tag, but you can declare a height on a `<tr>` , `<th>`, or `<td>` tag.

If your table has differing numbers of cells in a given row or column, the engine selects the row or column with the most cells to use as the table structure, and then fills in the gaps with empty cells. For example, suppose that your imported HTML includes the following table structure:

```
<table style="background-color:#fff; font-family:Arial; color:#000;
text-align:left;">
   <tr style="border:#000 solid 2px; background-color:#2443FF;
   color=#fff; text-align:center; font-weight:bold;">
       <td>Text</td>
       <td>Text</td>
   </tr>
   <tr style="border:#000 solid 2px;">
       <td>Text</td>
       <td>Text</td>
       <td style="background-color:53E8A2; text-align:right;"><i>Text</i></td>
   </tr>
</table>
```

The first table row contains 2 cells and the second table row contains 3, so the engine will create a table with 2 rows and 3 columns. In the first row, the cell in the third column will be empty. In the second row, the cell in the third column will contain the text included in the imported HTML. For example:

Example of an HTML table with varying numbers of cells in a row

Text	Text	
Text	Text	*Text*

Notice also how the style properties declared in cells and rows override the styles declared in the `<table>` tag.

4.4.8 Tag and Style Support for Imported HTML

You can use HTML tags and style properties to declare inline styles for the imported HTML, including support for multiple attributes within tags. For example, the following HTML tag would instruct the engine to produce Times New Roman 10-point blue text for that paragraph:

```
<p style="font-family: Times New Roman; font-size: 10pt; color: blue;">
```

Declared inline styles override any formatting that you apply in Designer, as well as default formatting for HTML tags (for example, headings). If you do not include inline styles in the imported HTML content, then the engine applies the formatting applied to the placeholder variable in the design.

Important: Exstream does not honor the `display: inline-block` style property for elements in imported HTML.

The engine also applies any styles defined within an HTML `<head>` structure, except in unconverted HTML output. For example, suppose that your imported HTML contains the following style information in the `<head>` structure:

```
<head>
 <style>
  h1 {
   font-family: Arial;
   font-size: 14pt;
  }
 </style>
</head>
```

In this example, setting the font size to 14 point overrides the 2 em font size that is the default for `<h1>`. For converted HTML output or any other output type, the engine applies this style for `<h1>` content (following HTML rules of precedence). For unconverted HTML output, however, the engine discards the `<head>` content.

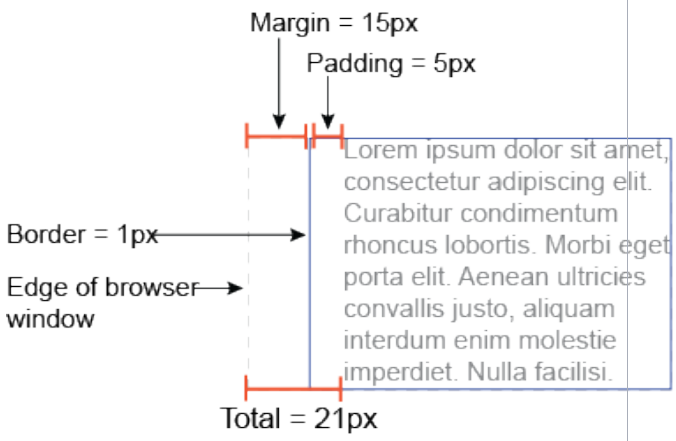
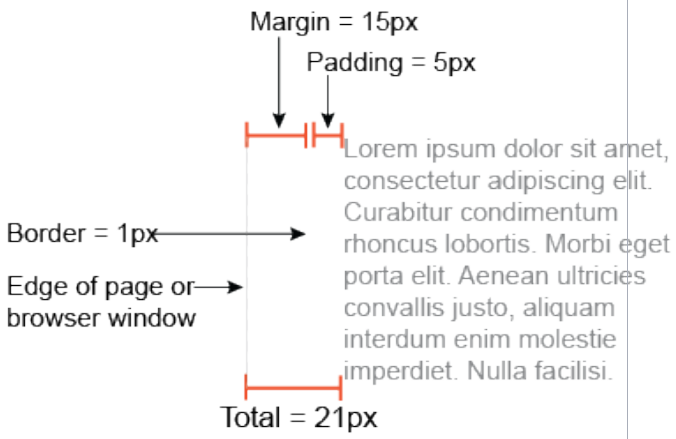
Note: If the `<head>` content contains a link to an external style sheet, the engine does not apply any styles defined in the external style sheets, regardless of whether the HTML is converted.

For more information about converted and unconverted HTML output, see "Converted vs. Unconverted HTML Output" on page 88.

Exstream supports most valid HTML style tags and properties when declaring formats for imported HTML content. Be aware that some text formats require special considerations during the application development process. Review the following table for more information.

Formats	Supported tags	Supported style properties and attributes	Design considerations
Aligned Text	N/A	`text-align: [value]`	Exstream supports `left`, `center`, and `right` alignment values when you use the inline style property `text-align`. The engine ignores text alignment settings applied in Designer.
Block Quote	`<blockquote>`	N/A	The engine indents the text one level in the output. The size of the indent is defined in Designer, in the **Tab size** box of the text paragraph properties. However, when creating unconverted HTML output, the engine ignores these settings. For information about creating unconverted HTML output, see "Converted vs. Unconverted HTML Output" on page 88.
Bold	`<b>` `<strong>`	`font-weight: bold`	Be sure to include the bold variant of the font in the package file or the bold text will not appear correctly in the output. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Formats	Supported tags	Supported style properties and attributes	Design considerations

Borders	N/A	`border-width: [value]`
		`border-top: [value]`
		`border-right: [value]`
		`border-bottom: [value]`
		`border-left: [value]`
		`border-style: [value]`
		`border-color: [value]`
			The border width can be expressed in points, pixels, or em values.
			Border color values can be expressed using hexadecimal, RGB values, RGBA values, or HTML color names.
			If the imported HTML is not converted, borders on text will appear in HTML output as expected.
			Example of unconverted HTML to HTML output
			 
			However, if the imported HTML is converted, any borders around text will not be included in the output. Instead, it will be added to the margin and padding values to control the spacing of the content. If the border width and style are not fully-defined, then the engine will discard the border width from this calculation.
			For example, if the imported HTML includes `<p style="border-width: 1px; border-style: solid; border-color: blue;">`, the engine will add 1 pixel to the margin and padding values. If any of the border style properties are missing from the imported HTML, then the engine will not add 1 pixel to the margin and padding values.
			Example of converted HTML to any output
			 

Formats	Supported tags	Supported style properties and attributes	Design considerations

Bulleted List	`<ul>`
	`<li>`
		N/A	Exstream supports nesting of bulleted lists. The engine adjusts the text indent for each level automatically using the default tab spacing.
			The default bullet types and colors are defined in the **System Settings**, on the **Text and Fonts** tab. You can override the default bullet types and colors for a design in the page template properties, on the **Design Defaults** tab.
			The engine ignores some HTML style properties for bulleted lists included in the imported HTML. For example, suppose the default settings in your design define first-level bullets as round and black. If you include a first-level bulleted list with an alternate style property, such as `<ul style="list-style-type: square; color: blue;>`, in the imported HTML, these bullets will appear as round and black in the output instead of square and blue.
			Be sure to include the bullet font and size for the bullet character in the package file. Otherwise, the bullet will not appear correctly in the output.
			For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.
			Note that when you produce PDF output, bulleted lists with multiple paragraphs are not supported. Suppose that you have the following list structure:
			`<ul>`
			`<li>`
			`<p>List item 1</p>`
			`<p>List item 2</p>`
			`</li>`
			`</ul>`
			The PDF output will contain only the first paragraph as part of the list. The list is then closed and the second paragraph is placed outside of the list.

Formats	Supported tags	Supported style properties and attributes	Design considerations

Description List	`<dl>`
	`<dt>`
	`<dd>`
		N/A	Exstream supports nesting of description lists. The engine adjusts the text indent for each level automatically using the default tab spacing.
			In accessible HTML and PDF output, `<dt>` tags are interpreted as list items, and the list items enclosed in `<dd>` tags are interpreted as items in a nested list. Suppose that you have the following list structure:
			`<dl>`
			`<dt>Term</dt>`
			`<dd>List item 1</dd>`
			`<dd>List item 2</dd>`
			`<dd>List item 3</dd>`
			`</dl>`
			In the HTML and PDF output, the list will be interpreted as having a list with one item and a nested list with three items.
			Note that when you produce PDF output, description lists with multiple paragraphs are not supported. Suppose that you have the following list structure:
			`<dl>`
			`<dt>List Heading</dt>`
			`<dd>`
			`<p>List item 1</p>`
			`<p>List item 2</p>`
			`</dd>`
			`</dl>`
			The PDF output will contain only the first paragraph as part of the list. The list is then closed and the second paragraph is placed outside of the list.

Font Color	N/A	`color: [value]`	Font color values can be expressed using hexadecimal, RGB values, RGBA values, or HTML color names. For example, if you want to use navy blue text for first-level headings, either of the following are valid:
			`<h1 style="color: #000080">`
			`<h1 style="color: Navy">`

Font Family	N/A	`font-family: [value]`	For each font family used in the design, be sure to include all necessary font variants and sizes in the package file. If you convert your imported HTML, any font-family information in your imported HTML content is lost when you convert. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Formats	Supported tags	Supported style properties and attributes	Design considerations
Font Size	N/A	`font-size: [value]`	The font size can be expressed in points, pixels, em values, or percentages. • **Points**—The font size in the output is the font size declared in the HTML. For example, text within `<p style="font-size: 14pt";>` would appear as 14-point text in the output. • **Pixels**—The engine converts the pixel size to points (1 pixel = 0.75 points) and then rounds up or down to the nearest point size. For example, 13 pixels converts to 9.75 points, so `<p style="font-size: 13px";>`,would appear as 10-point text in the output. • **Em Values**—The font size present in the output is calculated based on the font size declared in the parent element, for which the value is equal to 1 em. If no parent structure is present, then the font size is calculated based on the font size of the placeholder variable text in Designer. For example, if the parent element is `<div style="font-size: 10pt";>`, then text within `<p style="font-size: 1.25em";>` would appear as 12.5-point text in the output. Likewise, if the parent element is `<div style="font-size: 12pt";>`, then text within `<p style="font-size: 2em";>` would appear as 24-point text in the output. • **Percentages**—The font size present in the output is calculated based on the font size declared in the parent element. If the HTML does not declare a font size for the parent element, then the font size is calculated based on the font size of the placeholder variable text in Designer. For example, if the parent element is `<div style="font-size: 9pt";>`, then text within `<p style="font-size: 200%";>` would appear as 18-point text in the output. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Formats	Supported tags	Supported style properties and attributes	Design considerations
Headings	<h1> <h2> <h3> <h4> <h5> <h6>	N/A	For all heading levels, the engine adds a line break before the heading text and calculates the font size based on the font size declared in the parent element. Text within all heading levels is bold by default. Be sure to include the heading font size and variant in the package file. Consider the following HTML: <h1>Text</h1> More text In this example, the text within the tag uses the styles defined in the tag, so Arial 10-point must be included in the package file. The default size for <h1> text is twice the size of the font declared in the parent element, and bold, so you must also include Arial 20-point bold in the package file. You can also add any supported inline style attributes within the heading tag in the imported HTML content. If you include inline styles, the heading text is formatted based on the heading defaults plus the style attribute values. Suppose that, in the previous example, the HTML also declared an inline font variant for <h1> text, as in the following: <h1 style="font-style=italic;">Text</h1> More text The default <h1>style of Arial 20-point bold would still be applied to the heading text, and italics would also be added. This means that you must include Arial 20-point bold italic in the package file. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90. If you have nested headings in the imported HTML content, but they are not sequenced properly (for example, if you have an <h1> followed by an <h5> without any levels in between), this design carries through to the output. However, it can cause some output types with structural requirements, such as PDF, to fail a validation check.

Formats	Supported tags	Supported style properties and attributes	Design considerations

Hyperlinks	`<a>`	`href=[url]`
		`title=[text string]`
		`target=[value]`
			Hyperlinks are supported for the following output types:
			• HTML
			• HTML (email)
			• Multi-Channel XML
			• PDF
			• PowerPoint
			The `href` value must be defined using a URL rather than a variable.
			Note: When creating PDF output, the engine ignores the `title` and `target` properties. When creating PowerPoint output, the only valid value for the `target` property is `_self`.

Images	`<img>`	`src=[path]`
		`alt=[text string]`
		`width=[value]`
		`height=[value]`
		`border=[value]`
		`border-color: [value]`
		`border-style: [value]`
		`border-width: [value]`
		`float: [value]`
		`clear: [value]`
			Exstream supports including images in PNG and JPG formats. The image source must be defined using a file path rather than a variable.
			Image height and width settings can be expressed in pixels.
			Image border widths can be expressed in points, pixels, or em values. You can use the following border styles: `none`, `dashed`, `dotted`, or `solid`.
			Image border color values can be expressed using hexadecimal, RGB values, RGBA values, or HTML color names.
			The following `float` values are supported: `left`, `right`, `inherit`, or `none`.
			The following `clear` values are supported: `left`, `right`, `both`, or `none`.
			For more information about using images in imported HTML content, see "Images in Imported HTML" on page 91.

Italic	`<i>` `<em>`	`font-style: italic`	Be sure to include the italic variant of the font in the package file or the italicized text will not appear correctly in the output. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Formats	Supported tags	Supported style properties and attributes	Design considerations

Margin Settings	N/A	`margin: [value]`
		`margin-top: [value]`
		`margin-right: [value]`
		`margin-bottom: [value]`
		`margin-left: [value]`
			Margin widths can be expressed in points, pixels, em values, or percentages. If the HTML expresses the margin width as a percentage, be sure that you define the size of the parent element. You can express the values individually or using the CSS shorthand.
			For example, for 10-pixel margins on all sides,
			`<p style="margin: 10px;">` and
			`<p style="margin: 10px 10px 10px 10px;">` are both valid.
			If the imported HTML is converted, the text will be aligned based on the margin width values, plus any values expressed for borders and padding.

Numbered List	`<ol>`
	`<li>`
		N/A	Exstream supports nesting of numbered lists. The engine adjusts the text indent for each level automatically using the default tab spacing.
			To define the color of a number, you must apply settings in Design Manager. The engine does not honor any color defined in the imported HTML. The default number colors are defined in the **System Settings**, on the **Text and Fonts** tab. You can override the default number colors for a design on the page template properties, on the **Design Defaults** tab.
			The default numbering style applied to all levels of numbered lists is 1, 2, 3, and so on. If you want to define a different numbering style, you must include the `list-style-type` property in the imported HTML. The numbering style must correspond to one of the supported number styles in Exstream. For example, if you want to use lowercase alpha numbering (a, b, c, and so on), then your imported HTML must define that list style, as in the following:
			`<ol style="list-style-type: lower-alpha;">`
			If you specify a font to use for the number characters in the imported HTML, be sure to include the number font and size in the package file. Otherwise, the number character will not appear correctly in the output.
			For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.
			Note that when you produce PDF output, numbered lists with multiple paragraphs are not supported. Suppose that you have the following list structure:
			`<ol>`
			`<li>`
			`<p>List item 1</p>`
			`<p>List item 2</p>`
			`</li>`
			`</ol>`
			The PDF output will contain only the first paragraph as part of the list. The list is then closed and the second paragraph is placed outside of the list.

Formats	Supported tags	Supported style properties and attributes	Design considerations

Padding Settings	N/A	`padding: [value]`
		`padding: [value]`
		`padding-top: [value]`
		`padding-right: [value]`
		`padding-bottom: [value]`
		`padding-left: [value]`
			Padding widths can be expressed in points, pixels, em values, or percentages. If the HTML expresses the border width as a percentage, be sure that you define the size of the parent element. You can express the values individually or using the CSS shorthand.
			For example, for 5-pixel padding on all sides, `<p style="padding: 5px;">` and `<p style="padding: 5px 5px 5px 5px;">` are both valid.
			If the imported HTML is converted, the text will be aligned based on the padding width values, plus any values expressed for borders and margins. The engine ignores values for top and bottom padding if they are placed within a `<span>` element.

Short Quote	`<q>`	N/A	The engine adds double quotes around the text in the output.
Special Characters	N/A	N/A	The engine resolves character entities included in the imported HTML content in the output. **Important:** Be aware, however, that not all double-byte special characters are available for single-byte applications. If you are converting the imported HTML to the Exstream format, the engine replaces the double-byte characters that do not have single-byte representations with the substitution character `0x1A`. If you are not converting the imported HTML, the engine replaces them with spaces in the output.

Strikethrough	`<del>`
	`<strike>`
	`<s>`
		`text-decoration: line-through`	N/A

Subscript	`<sub>`	`vertical-align: sub`	If you use `<sub>`, the engine shifts the text below the baseline and calculates the font size at 65% of the font size declared in the parent element. If the HTML does not declare a font size in the parent element, then the engine calculates the subscript font size based on the font size of the placeholder variable text in Designer. If you use `vertical-align: sub`, the engine shifts the text below the baseline, but does not change the font size. Be sure to include the subscript font size in the package file. For example, if the baseline font size is 10-point Times New Roman font, then you must include 6.5-point Times New Roman for the subscript font. For more information about using font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Formats	Supported tags	Supported style properties and attributes	Design considerations
Superscript	`<sup>`	`vertical-align: sup`	If you use `<sup>`, the engine shifts the text above the baseline and calculates the font size at 65% of the font size declared for the parent element. If the HTML does not specify a font size for the parent element, the engine calculates the superscript font size based on the font size of the placeholder variable text in Designer. If you use `vertical-align: sup`, the engine shifts the text above the baseline, but does not change the font size. Be sure to include the superscript font size in the package file. For example, if the baseline font size is 10-point Times New Roman font, then you must include 6.5-point Times New Roman for the superscript font. For more information about font considerations for imported HTML content, see "Font Management for Imported HTML" on page 90.

Tables	`<table>`
	`<th>`
	`<tr>`
	`<td>`
	`<thead>`
	`<tbody>`
	`<tfoot>`
	`<colgroup>`
		`background-color: [value]`
		`border: [value]`
		`margin: [value]`
		`padding: [value]`
		`width: [value]`
		`colspan="number"`
		`float: [value]`
		`clear: [value]`
		`text-align: [value]`
		`vertical-align: [value]`
			Table border widths can be expressed in points, pixels, or em values. You can use the following border styles: `none`, `dashed`, `dotted`, or `solid`.
			
			Table background and border color values can be expressed using hexadecimal, RGB values, or HTML color names.
			
			The width of a table or table cell can be expressed in pixels, em values, or percentages.
			• **Pixels**—The size of the table or table cell is the exact pixel value.
			• **Em Values**—The engine calculates the table or table cell width based on the default font size. For example, if the default font size is 10-point, then `<td style="width: 2em">` results in 20-point, which the engine converts to 0.28 inches.
			• **Percentages**—The engine calculates the table or cell width based on the percentage of the width of the text paragraph that contains the placeholder variable.
			
			You can add the `colspan` attribute to `<td>` and `<th>` tags. For example, if you declare `<td colspan="3">` in your HTML, then that particular cell will span three columns.
			
			The following `float` values are supported: `left`, `right`, `inherit`, or `none`.
			
			The following `clear` values are supported: `left`, `right`, `both`, or `none`.
			
			The following `vertical-align` values are supported: `top`, `middle`, or `bottom`
			
			All font styles are supported within tables.
			
			For more information about including tables in imported HTML content, see "Tables in Imported HTML" on page 102.

Underline	`<u>`	`text-decoration: underline`	N/A

4.5 PDF Considerations

The PDF format is widely used because you can create a PDF file from almost any content creation program. To help you get the most use out of your external PDF files, you can include PDF content in a design at design time and at run time.

This section discusses the following topics:

4.5.1 How Can I Use PDFs in My Exstream Application?

You can import existing PDF document designs at design time so that they can be used to create new document designs, and you can import PDF documents at run time in order to include them in your output.

This section provides an overview of the ways that you can leverage existing PDF designs and content, and lists the modules that are required for each type of use.

This section discusses the following topics:

Importing PDFs at Design Time

If you want to use an existing PDF document design in order to create an Exstream document design, you can import the PDF as an image and use it as a template for positioning design objects, or you can use the PDF Converter to convert the PDF into Exstream design components.

The following table lists the modules required to use PDF content in various ways in your design at design time.

To	You must have licensed this module	Overview of steps
Use an existing PDF design to position an address block on a page For example, suppose that you are creating an application to produce output for a standardized bill statement that is printed on cut-sheet forms. You want to use an existing PDF design as a template in order to position variables and other design objects.	**Design PDF module** The Design PDF module lets you import a PDF file as an image into an Exstream design.	To import a PDF design at design time: 1. In Design Manager, import the PDF image into a design database. 2. In Designer, add the PDF image to a design layer, and then position variables and design objects. For more information about importing PDF designs at design time, see "Importing a PDF File at Design Time" on page 39.
Convert a PDF design into editable Exstream design objects in the DXF file format For example, suppose that you are converting the design of a complex, multi-page document that was created using a legacy application. You want to be able to modify and reuse the design in Exstream applications.	**PDF Converter module** The PDF Converter module lets you convert a PDF document design into editable documents, pages, and objects. These components can then be used just like any other design components that are created in Exstream.	To convert a PDF design at design time: 1. Use the PDF Converter to convert the PDF document to the DXF file format. 2. In Design Manager, import the DXF content into an Exstream design database. 3. In Designer, add variables and rules, and modify the imported design as required. For more information about importing PDF designs at design time, see "Importing a PDF File at Design Time" on page 39.

For more information about considerations to keep in mind when importing PDF files at design time, see "PDF Design-Time Considerations" on page 119.

Importing PDFs at Run Time

Not only can you use Exstream Design and Production to produce PDFs, but you can also import PDFs at run time in order to include them in a variety of outputs.

The following table lists the modules required to use PDF content in various ways in your design at run time.

To	You must have licensed this module	Overview of steps
Dynamically import a frequently changing set of PDFs at run time For example, suppose that you are creating an Exstream application to produce account statements, and the statements include terms and conditions in both English and Spanish. This content is maintained in PDFs by the legal department, and changes often. You want to be able to pull in the most up-to-date legal content without having to repackage your application and perform regression testing.	**Dynamic Content Import module** The Dynamic Content Import module lets you dynamically import PDFs and other content at run time. **PDF Import as Image module**[1] The PDF Import as Image module is required in addition to the Dynamic Content Import module if you optionally want to: • Import PDFs and then produce an output format that does not natively support PDF or EPS files. • Import PDFs and then use the **Image each page** option to produce PDF, PDF/A, PDF/VT, PostScript, PPML, TOP, VDX, VIPP, or VPS output. • Use a Live document in a fulfillment application, and the Live document includes a PDF that was dynamically imported when the Live document was initially produced.	To dynamically import PDFs at run time: 1. In Design Manager, add a placeholder document to your design database, and then add a page to the placeholder document. 2. In Designer, add a placeholder variable to the page. For more information about dynamically importing content into a design at run time, see "Importing Content into a Design at Run Time" on page 42.
Pre-populate a PDF XFA form, so that the form is partly filled out For example, suppose that case workers use a PDF form to process citizen requests for health services. The form is pre-populated with information in order to reduce errors and the amount of time that it takes to complete. The case worker adds or edits information and then submits the form electronically to back-end systems.	**PDF Form Pre-Fill module**[1] The PDF Form Pre-Fill module lets you reuse PDF XFA forms and populate them with data at run time. **Dynamic Content Import module** The Dynamic Content Import module lets you dynamically import PDFs and other content at run time. **PDF Output module** The PDF Output module lets you produce editable PDF output.	To use a PDF XFA form and pre-fill data fields: 1. In Design Manager, map variables to PDF XFA tag names in the PDF XFA form. 2. Package the application and run the engine to populate the data fields and produce editable PDF output. For more information about mapping variables to a data file, see *Using Data to Drive an Application* in the Exstream Design and Production documentation.

To	You must have licensed this module	Overview of steps
Extract data from a PDF XFA form and use the data in an Exstream application For example, suppose that you want to import data from a PDF form that loan agents use to process automobile loans. A loan agent collects information from a customer and then submits the form electronically to back-end systems. The data is then used to populate customer databases and produce customer communications.	**PDF Form Miner module**[1] The PDF Form Miner module lets you use a PDF XFA form as a data source in an Exstream application. **Dynamic Content Import module** The Dynamic Content Import module lets you optionally import multiple PDF-XFA forms using a placeholder variable. **Dynamic Data Access (DDA) module** The Dynamic Data Access (DDA) module lets you optionally use a DDA routine to extract data from a PDF XFA form and then pass the data to an Exstream application.	To use a PDF XFA form as a data source: 1. In Design Manager, create an XML data file for the PDF XFA data source, create a PDF XFA placeholder variable type, and map variables to the PDF XFA data source. 2. Package the application and run the engine to populate the data fields and produce output. For more information about creating a data file, creating a variable, and mapping variables to a data file, see *Using Data to Drive an Application* in the Exstream Design and Production documentation. For information about configuring a DDA routine, see *Configuring Connectors* in the Exstream Design and Production documentation.

[1] The PDF Import as Image, PDF Form Pre-Fill, and PDF Form Miner modules are not supported on z/OS.

For more information about considerations to keep in mind when importing PDF files at run time, see "Supporting PDF Content When Output Drivers Do Not Natively Support PDF" on the next page.

4.5.2 PDF Design-Time Considerations

When you import a PDF at design time, each page of an imported PDF is automatically converted into an image and placed on a separate page. Keep in mind the following considerations when importing PDF files into a design at design time:

- Ghostscript 9.0 or later is required to import and view PDF files. Ghostscript is a commercially available PostScript and PDF conversion and rendering tool that rasterizes a PDF page to an image so it can be placed on a page.

  For information about installing Ghostscript, see *Installation and Upgrade Information* in the Exstream Design and Production documentation.

- Although you need the PDF Import as Image module to import PDF files at run time, you do not need this module to import PDF files at design time.

- You cannot resize or edit PDF content that is imported at design time.

> **Tip:** If you want to edit PDF content from Exstream, you can use the PDF Converter to convert the content within the PDF into editable Exstream design objects.

For more information about converting PDF content into editable Exstream design objects, see *Importing Designs* in the Exstream Design and Production documentation.

- If an imported PDF contains accessibility tagging, the accessibility information will be retained if you create PDF or PDF/A output for which accessibility tagging has been enabled in Exstream.

  Note: PDFs that are converted into editable design objects using the PDF Converter do not retain accessibility tagging. If you are creating PDF or PDF/A output, you can reapply accessibility tags in your Exstream design. For more information about creating accessible PDF output, see *Designing Customer Communications* in the Exstream Design and Production documentation.

- The resolution settings on the imported PDF file, the objects on the design page in Designer, and the output object properties in Design Manager must all be the same; otherwise you might see quality issues in your output, since these resolution settings can contradict each other if they're not compatible.
- A preview of the imported PDF page is available only if the PDF contains an internal thumbnail. If the thumbnail is not available, the PDF content appears as a gray box.
- You cannot import a PostScript file using the design-time PDF import options.

4.5.3 Supporting PDF Content When Output Drivers Do Not Natively Support PDF

Some output drivers do not natively support PDF content. You can still leverage PDF content when generating output using output drivers with this limitation; however, you must set up Exstream with the correct modules and programs to allow the Exstream engine to complete any conversion or modifications needed to include PDF content into the print stream for these output drivers.

When importing PDF files dynamically to an output driver that natively supports PDF (PDF, PDF/A, PDF/VT, or VDX), the engine passes the content directly into the PDF print stream. However, for output drivers that do not natively support PDF, the engine converts the content to a format that is compatible with the output driver.

Keep in mind the following when importing PDF images:

- When you import PDF files at run time to PDF, PostScript, PPML, VDX, VIPP, or VPS output(s), Exstream supports importing from PDF files that are version 1.7 and earlier.
- When importing PDF images to PDF output, do not select the **Encrypt contents of PDF output** check box in the PDF output object properties.

  For information on setting up output objects, see *Creating Output* in the Exstream Design and Production documentation.

- When you import large PDF files, the engine requires large amounts of memory to store them during processing. Depending on the file size of the imported PDF files, the engine can run out of memory.

  For information on managing memory, see *Preparing Applications for Production* in the Exstream Design and Production documentation.

- If you select the **Image each page** option on the **Basic** tab of the output object in Design Manager, regardless of the output driver, you must license the PDF Import as Image module and install Ghostscript, which is a commercially available PostScript and PDF conversion and rendering tool.

  Note: Ghostscript is not supported on z/OS environments.

- When you have an output object with **Image each page** selected on the **Basic** tab, or when you have a printer that does not support EPS dynamic import, Exstream converts the imported PDF contents to a black-and-white image.

- When you run an application that imports PDF content at run time on an AFP device that supports native PDF content, you can enable AFP color management architecture by selecting the **Use AFP color management architecture** check box on the **Resource Management** tab of the AFP output object properties.

  For more information about AFP color management architecture, see *Creating Output* in the Exstream Design and Production documentation.

- If the AFP device that you use to produce output does not support native PDF, or using AFP color management architecture produces incorrect output, you must include the PDF_PASSTHROUGH_AS_PDF engine switch in your control file to enable the engine to import native PDF content at run time.

  For more information about the PDF_PASSTHROUGH_AS_PDF engine switch, see *Switch Reference* in the Exstream Design and Production documentation.

- When the engine converts images to EPS during import, fonts references are changed to the closest available Base-14 font.

- The engine rasterizes PDF images on output drivers that do not support native PDF import.

Keep in mind the following when producing AFP output:

- AFP output of native PDF content is supported on some AFP devices. Consult your product manufacturer for more information.

- For higher quality PDF content, when you run an application that imports PDF content at run time on an AFP device that supports native PDF content, you can enable the AFP color management architecture option on the Resource Management tab of the AFP output object properties.

  For more information about the AFP color management architecture, see *Creating Output* in the Exstream Design and Production documentation.

- When you run an application that imports dynamic PDF content and run-time image imports, and you do not want the dynamic PDF content processed using the default settings, use the PDF_PASSTHROUGH_AS_PDF engine switch to include dynamic PDF content in the output in object containers.

  For more information about the PDF_PASSTHROUGH_AS_PDF engine switch, see *Switch Reference* in the Exstream Design and Production documentation.

The following table outlines the formats to which PDF content is converted when it is generated as output using an output driver that does not natively support the PDF format:

Conversion formats for dynamically imported PDF files

Output driver	Converted formats	Requirements

AFP	If the AFP printer does not support embedded EPS, files are imported in one of the following formats:
	• If the AFP printer supports embedded EPS, files are imported as EPS. On Windows and UNIX platforms, IBM Infoprint Manager converts the EPS to a native IPDS bitmap using Adobe libraries before printing. This conversion can slow processing time because the EPS must be rasterized.
	• If the AFP printer does not support embedded EPS, files are imported in one of the following formats:
	  • For full-color images, JPEG (RGB color) is used.
	  • For AFP black-and-white images, B&W TIFF is used.
	Note: The JPEG format color space is defined by your AFP output device.
	• For AFP black-and-white images, the B&W TIFF format is used.
	If the AFP printer supports embedded EPS, files are imported as EPS. On Windows and UNIX platforms, IBM Infoprint Manager converts the EPS to a native IPDS bitmap using Adobe libraries before printing. This conversion can slow processing time because the EPS must be rasterized.
		If the AFP printer does not support AFP data with embedded EPS, you must meet the following requirements:
		• You must have licensed the PDF Import as Image module.
		• You must install Ghostscript.
		If the AFP printer supports AFP data with embedded Encapsulated PostScript (EPS) or if you are importing PDF files to AFP output on z/OS, you must use the PDF_PASSTHROUGH_AS_EPS switch in the control file.
		Tip: The PDF_PASSTHROUGH_AS_EPS switch can also be used if you want higher quality images when delivering PDF content to an AFP printer that does support embedded EPS.
		For information on using the PDF_PASSTHROUGH_AS_EPS switch, see *Switch Reference* in the Exstream Design and Production documentation.
		If you want to dynamically import color TIFF files, you must enable AFP color management.
		For more information about AFP color management, see *Creating Output* in the Exstream Design and Production documentation.

Conversion formats for dynamically imported PDF files, continued

Output driver	Converted formats	Requirements
HTML	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
IJPDS	Files are imported in one of the following formats: • For full-color images, the TIFF (CMYK color) format is used. • For IJPDS black-and-white images, the B&W TIFF format is used.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
Metacode	Files are imported in the B&W TIFF format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
MIBF	Files are imported in one of the following formats: • For full-color images, the TIFF (CMYK color) format is used. • For MIBF black-and-white images, the B&W TIFF format is used.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
PCL	Files are imported in the B&W TIFF format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
PDF	PDF content is natively supported in PDF output. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
PDF/A	PDF content is natively supported in PDF/A output. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
PDF/VT	PDF content is natively supported in PDF/VT output. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.

Conversion formats for dynamically imported PDF files, continued

Output driver	Converted formats	Requirements
PostScript	PDF content is natively supported in PostScript output in the EPS format. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
PowerPoint	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
PPML	PDF content is natively supported in PPML output in the EPS format. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
RTF	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
TIFF	Files are imported in the B&W TIFF format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
VDX	PDF content is natively supported in VDX output in the EPS format. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
VIPP	PDF content is natively supported in VIPP output in the EPS format. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
VPS	PDF content is natively supported in VPS output in the EPS format. If you choose to import each page as an image, files are imported in the B&W TIFF format.	There are no additional requirements since this output driver supports PDF or EPS files.
XML (Composed)	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.

Conversion formats for dynamically imported PDF files, continued

Output driver	Converted formats	Requirements
XML (Content)	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
XML (Multi-Channel)	Files are imported in the JPEG (RGB color) format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.
ZPL	Files are imported in the B&W TIFF format.	• You must have licensed the PDF Import as Image module. • You must install Ghostscript.

Ghostscript is a commercially available PostScript and PDF conversion and rendering tool. For information about installing Ghostscript, see *Installation and Upgrade Information* in the Exstream Design and Production documentation.

For more information about the PDF Import as Image module, see *Importing External Content* in the Exstream Design and Production documentation.

4.6 Print Resource Considerations

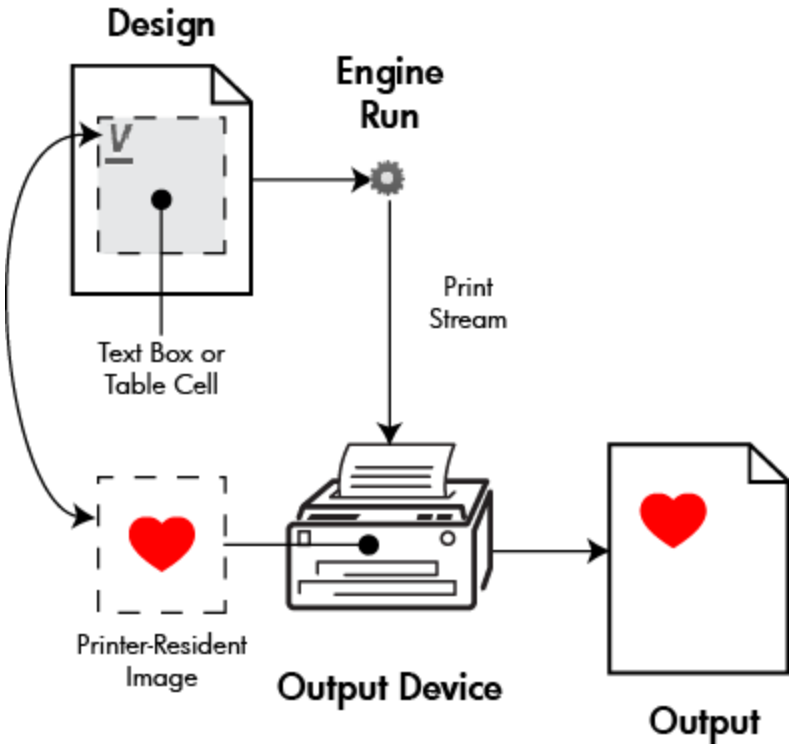
Some output devices support the storage of resources in memory on the device, which can increase processing efficiency for legacy formats or high-quality color images.

Before passing through print resources at run time, review the following considerations:

- Exstream supports the ability to include a reference to forms overlays and printer-resident images from the print stream at run time for the following output drivers:
  - AFP
  - Metacode
  - PCL
  - PostScript
  - PPML
  - TOP
  - VIPP
  - VPS
- When you store images on the output device, they do not have to be rasterized during production.

- Print resource images cannot be resized or rotated because the dimensions cannot be obtained prior to run time.
- To see printer-resident resources when you test the application, you must set up your viewer to view external resources when you test the application.
- When you reference printer resources, you set up the design with a placeholder variable that specifies the location of the resource file so that it can be identified and used during printing. At run time, the engine places the image name (not the image itself) in the print stream and the printer resolves the image when it prints. The following graphic illustrates this process.

Example of referencing printer resources



4.7 RTF Considerations

To take advantage of advanced text formatting and editing options available with word processing programs, you can create text in the word processing program of your choice. You can then generate an RTF file from that program and import the content into an Exstream design. For example, suppose your legal department provides several notice for general use by different offices in the company. You can save those notices in an RTF format and import the notices into an Exstream design.

Depending on the word processing program used to create the RTF file, the final appearance of the imported content in the customer output can vary. Keep in mind the following considerations when you generate the RTF file you want to import:

- Microsoft Word saves background data that is not needed for RTF presentation, which can adversely affect engine performance. Importing RTF content saved in a simple text editor, such as WordPad, can significantly boost engine performance compared to the same content saved in Microsoft Word.
- RTF import supports the importing of only single-column text. If you want to import images as well, you must use separate files and placeholder variables. Not all visual elements that are possible in an RTF file are supported in RTF import. Most unsupported features can be designed on a page using Designer features. For example, during design you can set up the text box into which you are importing the RTF content to contain multiple columns and have a background color.
- Word processing programs offer various formatting options as you create content. When text is imported into Exstream, the design environment must complete some modifications to align the original text formatting with the text formatting options available in Exstream. For example, custom bulleting options available in Microsoft Word might not be available in Exstream. The following table provides an overview of the Exstream support for various formatting options and text features when importing RTF files:

Specifications for supported formatting and text features during RTF import

Text feature	Specifications
Bulleted lists	• The design environment substitutes a filled bullet (xB7) for bullet characters. • In DBCS applications, if the bullet in the RTF file uses Unicode, it is replaced with a question mark (?).
Color	No additional specifications
Document margins	If you use a placeholder frame to hold the imported content, the design environment references only the left margin set in the original RTF file. If you use a text box or table cell to hold the imported content, all document-level margins set in the original document are ignored.
Fonts	To be imported correctly, all fonts and font formats (size, color, underline, bold, and so on) used in the RTF content must be included elsewhere in the application. You can use a font resource management. For more information about using font resource management, see "Using Font Resource Management to Ensure that Fonts are Available at Run Time" on page 71.
Hard returns	No additional specifications
Horizontal alignment	Left, center, right, and justified text is supported.
Line spacing	All text is imported as single-spaced. Options that set text lines to be spaced at 1.5, double, exactly, at least, or multiple are not supported.
Numbered lists	To be imported correctly, lists must be numbered manually. Lists with numbers auto-generated by Microsoft Word are imported without numbers and tabs.

Specifications for supported formatting and text features during RTF import, continued

Text feature	Specifications
Page headers	The design environment imports page headers as static text without the special spacing set by page margin spacing specified in the RTF file.
Paragraph margins	The design environment supports left margin settings of specific paragraphs, including indents and hanging first lines (as with bulleted paragraphs). Every paragraph's right margin stretches to the right of the text box, table cell, or placeholder frame.
Paragraph spacing	The space before and after a paragraph is supported.
Soft returns	No additional specifications
Strikethrough	No additional specifications
Subscript	No additional specifications
Superscript	No additional specifications
Tab stops	Default and customized tabs used in paragraphs are supported; however, they must be left or decimal tab types. Centered and right tab stops are imported as a left tab stop.
Tables	Columns in the table can vary in width, but each column must have a consistent width throughout the entire table. Row heights defined in the RTF file are not supported; to increase the height of a row, place extra paragraphs in the first cell in the row during design. The rest of the row automatically matches the height of the first cell. The following table features are supported: • Shading and background color in individual cells • Alignment settings of left, center, and right in individual cells • Merged cells on the same row • Merged cells spanning multiple rows split by a row • Single-line solid borders in tables with different colors and thicknesses. The design environment applies only a single type of border around a single cell, unlike Microsoft Word, which can apply a different type of border to each edge of the cell.
Variables	Variables can be used in the text file, but must be enclosed in angle brackets in order for the engine to recognize them as variables (for example, `<Dear CustomerName>`).

For more information on controlling paragraph properties, see "Formatting Paragraph Properties for RTF Files That Are Imported at Run Time" on page 72.

4.7.1 Unsupported Features for RTF Import

The following features are not supported for RTF import:

- Background color
- Borders (around paragraphs)
- Charts
- Comments
- Drawn lines
- Embedded vector images
- Extended dashes (em or en dashes)
- Floating text boxes
- Footers
- Footnotes
- Highlighting
- Hyperlinks (that are functional)
- Kerning (scaling)
- Multiple columns
- Raster images
- Revision tracking
- Small capital letters
- WordArt

4.8 SVG Considerations

If you want to use vector images in your HTML output, you can use placeholder variables to import SVG images at run time. Exstream preserves and stores the original vector data from imported SVG images.

Note: If you import an SVG image at design time, Designer rasterizes the image for both display and output. Any transparency, animation, embedded hyperlinks, or accessibility elements in the image are lost. Once an SVG image is rasterized, it can be included in any output type.

When importing SVG images at run time using placeholder variables, keep in mind the following considerations:

- SVG image import is supported only with HTML5 output.
- Any transparency, animation, text, embedded hyperlinks, or accessibility elements in your SVG image are retained during the import.
- If you want to use the **Maintain size from file** option in the placeholder variable properties, be sure the width and height attributes are set in the `<svg>` tag of the SVG image. Otherwise, Exstream interprets the width and height to be "0" and the image will not appear in the output.

  For more information about setting the image size, see "Maintaining Original Image Size Within a Placeholder Object" on page 63.