

Stylus Studio XML Deployment Adapter URLs

Stylus Studio XML Deployment Adapters allow you to access and convert non-XML files to XML, and vice versa. Adapters can be invoked programmatically, in an XQuery application, for example, allowing you to treat non-XML data as XML, manipulate it as needed, and, optionally, write it back to its source in its original format.

Stylus Studio provides adapters for over one dozen common file formats (binary, comma- and tab-separated values, EDI, and others).

This document provides background information on using Stylus Studio XML Deployment Adapter URLs in your applications. It covers the following topics:

- [“Available Adapters” on page 2](#)
- [“Adapter URL Syntax” on page 4](#)
- [“Adapter Properties” on page 4](#)
- [“Additional Sources of Information” on page 16](#)

Available Adapters

Stylus Studio currently provides the following adapters, all of which are bidirectional, with the exception of mBox and WinWrite, which operate only as native-to-XML.

Table 1. Adapters

Name	Description	For More Information
Base-64	Converts a binary file into a Base-64 one-element file. This adapter can be useful for converting binaries that are to be included within another file.	“Base-64 Adapter Properties” on page 6
Binary	Converts a binary file into a hex-encoded one-element file. This adapter can be useful for converting binaries that are to be included within another file.	“Binary Adapter Properties” on page 7
CSV	Converts a comma-separated text file to XML, and vice versa	“Comma-Separated and Tab-Separated Values Adapter Properties” on page 8
dBase_II, dBase_III, dBase_III_plus, dBase_IV, dBaseV	Converts dBase database management system files to XML, and vice versa.	“dBase Adapter Properties” on page 9
DIF	Converts Data Interchange Format files to XML, and vice versa. Sometimes used for spreadsheet interchange.	“DIF, SDI, SYLK Adapter Properties” on page 10
DotD	Converts Progress OpenEdge text dump files to XML, and vice versa.	“Properties for Other Adapters” on page 16
EDI	Recognizes and decodes EDIFACT and X12 files to XML, and vice versa.	“EDI Adapter Properties” on page 11
HtmlTidy	Uses JTidy to convert HTML to XHTML, and vice versa.	“HtmlTidy Adapter Properties” on page 13
JavaProps	Converts a Java .properties file to XML, and vice versa.	“Properties for Other Adapters” on page 16

Table 1. Adapters

Name	Description	For More Information
Line	When converting text to XML, wraps each line of the incoming file with an XML tag; when converting XML to text, it writes each tag of depth two's content as a line.	“Whole-line Text Adapter Properties” on page 15
MBox	Parses the .mbox file and wraps the header and body of a text message with an XML tag. The MBox adapter can be used for mbox-to-XML conversion only. Supports MIME attachments – any message attachments are properly decoded and rendered as XML. Specifically: <ul style="list-style-type: none"> ● Text and plain attachments are embedded as text ● Binary attachments are embedded as hex-encoded data ● XML-encoded attachments are emitted as escaped text 	“Properties for Other Adapters” on page 16
RTF	Reads an RTF file, and converts RTF tags to XML, and vice versa.	“Properties for Other Adapters” on page 16
SDI	Super Data Interchange (SDI) is a later version of DIF.	“DIF, SDI, SYLK Adapter Properties” on page 10
SYLK	Converts Microsoft SYmbolic LinK format files, used for spreadsheet interchange, to XML, and vice versa.	“DIF, SDI, SYLK Adapter Properties” on page 10
TAB	Converts a tab-separated text file to XML, and vice versa	“Comma-Separated and Tab-Separated Values Adapter Properties” on page 8

Table 1. Adapters

<i>Name</i>	<i>Description</i>	<i>For More Information</i>
WinIni	Converts a Windows .ini file to XML, and vice versa.	“Properties for Other Adapters” on page 16
WinWrite	Converts Windows WinWrite .wri files to XML.	“Properties for Other Adapters” on page 16

Adapter URL Syntax

While properties differ from one adapter to the next, the syntax used to invoke them is the same:

```
adapter:name[:property_name=value|:property_name=value|...]?file:file URL
```

Example:

```
adapter:Base64:newline=crlf:encoding=utf-8?file//w:\myfiles\base_to_xml.bin
```

In this example:

- The name of the adapter is Base64
- Properties are newline and encoding
- The file being converted is base_to_xml.bin on w:\myfiles

Properties can be in any order. For example, encoding can precede newline.

Adapter Properties

Adapters for different file types have different properties – the CSV adapter allows you to specify an escape character, but the binary adapter does not, for example. This section describes values for the line separator property, which is shared across most adapters, and it describes the properties for individual adapters.

- [“Line Separator \(newline\) Values” on page 5](#)
- [“Base-64 Adapter Properties” on page 6](#)
- [“Comma-Separated and Tab-Separated Values Adapter Properties” on page 8](#)
- [“dBase Adapter Properties” on page 9](#)
- [“DIF, SDI, SYLK Adapter Properties” on page 10](#)
- [“EDI Adapter Properties” on page 11](#)

- [“Binary Adapter Properties” on page 7](#)
- [“HtmlTidy Adapter Properties” on page 13](#)
- [“Whole-line Text Adapter Properties” on page 15](#)
- [“Properties for Other Adapters” on page 16](#)

Line Separator (newline) Values

Most adapters allow you to specify some type of line separator (referred to in the adapter URL as *newline*). The following table summarizes commonly occurring values. All values are case-insensitive.

Table 2. Common Values for the Line Separator (newline) Property

<i>Value</i>	<i>Description</i>
cr or mac	The Macintosh standard.
lf or unix	The Unix standard.
crlf or dos	The DOS standard.
lfcr	The Windows standard.
ne1	0x85 (commonly found in mainframes).
null	A null byte.
platform	If another value has not been specified, the line separator uses the platform value as returned by the <code>System.getProperty("line.separator")</code> method.

Base-64 Adapter Properties

The following table shows adapter properties for Base-64 encoded binary files, as documented in RFC 1341.

Adapter Name in URL

Base-64

Table 3. Properties for Base-64 Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used only when converting a binary file to a Base-64-encoded XML value. This property does not affect newlines in the input file when coming from the native format; similarly, it does not affect newlines being written to the native format. The default is <code>crlf</code> . See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is <code>utf-8</code> .

Binary Adapter Properties

You can convert binary files that have been encoded as a sequence of digits from base 2 to base 36, and vice versa. Typically, binary data is encoded for use in XML as hexadecimal or Base-64. For hexadecimal output, use base 16.

Base-64, despite the similar nomenclature, is the specific name of an encoding format, and not a reference to a numeric base, which is why it has its own adapter. See [“Base-64 Adapter Properties” on page 6](#) for more information.

Adapter Name in URL

Binary

Table 4. Properties for Binary Base-2 to Base-36 Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used when converting a binary encoded file to XML, and vice versa. The default is crlf. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.
base	The numeric base of the encoded file. The default is 16 (hexadecimal). base 2 is binary; base 8 is octal; and base 10 is decimal.
wrap	Whether you want to wrap lines (wrap=yes) or output all values on a single line (wrap=no).
space	Whether or not byte values should be contiguous (no value) or separated with the value specified for this property. For example, if you set space=, the value 000FFF would be output as 00,0F,FF.

Comma-Separated and Tab-Separated Values Adapter Properties

With a few exceptions, noted in the following table, adapter properties are the same for both the comma-separated values and tab-separated values adapters.

Adapter Names in URL

CSV (comma-separated values)

TAB (tab-separated values)

Table 5. Properties for CSV and TSV Adapters

<i>Property Name</i>	<i>Description</i>
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML.
newline	See “Line Separator (newline) Values” on page 5 for a list of values.
sep	The separator value between each value. This can be 'TAB', any single character (a comma (,) is the default), or the %XX-escaped value (%2c, for example).
first	Generated field names depend on the values in the <code>first</code> and <code>number</code> fields. If <code>first=yes</code> and <code>number=no</code> , field names are read from the first row. Any field names after that are named <code>column.xxx</code> , where <code>xxx</code> is the column number, starting from one and including explicitly named columns in the count. If <code>number=yes</code> , extra columns (those after the first) are named just <code>column</code> .
number	If <code>number=yes</code> (<code>no</code> is the default), each row will also have an attribute, named <code>row</code> , which will contain the row number from the source document, starting from one. Also, each column, even those explicitly named, will have a <code>column</code> attribute numbering the column from one. Any empty columns are omitted from the output, but the numbering of subsequent columns will reflect that a column(s) was skipped.

Table 5. Properties for CSV and TSV Adapters

<i>Property Name</i>	<i>Description</i>
escape	This character escapes quotes and separators so that they can be embedded in values. The backslash (\) is the default.
quote	A list of characters the adapter should interpret as quotation characters. Double and single quote marks (" ") are the default values.

dBase Adapter Properties

Adapter properties are the same for all dBase adapters – dBase II, dBase III, dBase III+, dBase IV, and dBase V.

Adapter Names in URL

dBase_II

dBase_III

dBaseIII_plus

dBase_IV

dBase_V

Table 6. Properties for dBase Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used only to convert a dBase file to XML, not vice versa. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.
deleted	Whether or not records marked with a "deleted" attribute are included in the output to XML and preserved in the conversion from XML. Stylus Studio generates the "deleted" attribute on output, and looks for it on input when this property is set to yes.

DIF, SDI, SYLK Adapter Properties

Adapter properties are the same for both Data Interchange Format (DIF), Super Data Interchange Format (SDI), and Symbolic Link Format (SYLK) adapters.

Adapter Names in URL

DIF (Data Interchange Format)

SDI (Super Data Interchange Format)

SYLK (Symbolic Link Format)

Table 7. Properties for DIF, SDI, and SYLK Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used when converting DIF, SDI, or SYLK file to XML, and vice versa. The default is crlf. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.

DotD

See [“Properties for Other Adapters” on page 16](#) for information about the Progress .d Data Dump (DotD) adapter.

EDI Adapter Properties

Adapter properties are the same for both Data Interchange Format (DIF), Super Data Interchange Format (SDI), and Symbolic Link Format (SLK) adapters.

Adapter Name in URL

EDI

Table 8. Properties for EDI Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used when converting EDI to XML, and XML to EDI when the eo1 is set to yes. The default is crlf. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.
val	Validates the XML against the structure in the relevant EDI dictionary. An error is generated if the EDI dialect (EDIFACT, X12, for example) isn’t recognized; an error is also generated if the dialect is recognized, but the message type isn’t. Missing mandatory segments, or segments that not specified for a particular group, will also generate errors. Consider leaving this option on (yes, the default). If this option is off (no), the adapter is unable to synchronize its position within the EDI dictionary, preventing nested loops from being generated with the output. When possible, leave this property set to yes and ensure that the EDI input conforms to the specification.
decode	Adds to each code that is looked up in a table a comment that explains the code’s value. For example, <!--Production Data--> in the following code: <pre><ISA15><!--I14: Interchange Usage Indicator-->P<!-- Production Data--></ISA15></pre> Set decode and field to no to disable all comment generation.

Table 8. Properties for EDI Adapters

<i>Property Name</i>	<i>Description</i>
field	<p>Creates a comment at the start of each element that includes the element's name and number. For example, <!--I14: Interchange Usage Indicator--> in the following code:</p> <pre><ISA15><!--I14: Interchange Usage Indicator-->P<!-- Production Data--></ISA15></pre> <p>Set field and decode to no to disable all comment generation.</p>
len	Checks each value against the upper and lower length limits defined in the EDI specification.
seg	Relaxes the rules that require that segments come in the specified order. However, if this property is off (no), some looping constructs might break, resulting in data being grouped in correctly.
tbl	Generates an error if the value for an element is not in the codelist associated with that element. If this property is off (no), values are not checked for the presence of a codelist.
typ	Ensures that only characters that are appropriate for a given field are included in the value for that field. For example, this property ensures that dates are valid and numbers are well-formed.
opt	<p>If set to yes (no is the default), Stylus Studio assumes that all segments are optional. This property can be useful if your provider declines to provide segments that are considered mandatory according to the EDI specification, but you are aware of what the missing values are.</p> <p>This property is not used if val is set to yes.</p>
eol	Allows you to put each segment on its own line when converting XML to EDI. (Extra linefeeds are ignored when converting EDI to XML.) If this property is set to yes (the default), the value specified in the newline property is used to separate each segment. The normal segment output character is also generated.
prefix	Namespace prefix to be added, with the value of the uri property, to the root element. The prefix alone is added to all elements.

Table 8. Properties for EDI Adapters

<i>Property Name</i>	<i>Description</i>
uri	Namespace URI to be added, with the prefix property, to the root element. If the prefix is set, but the URI is not, the prefix is ignored.
loop	Used to change the group element name. By default, segment loops are wrapped in <GROUP_#> elements, where # denotes the segment group number. The loop element name can be any valid element name, with the following exceptions for these characters: # – the segment group number is used instead. \$ – the name of the first segment of the group is inserted into the element name instead. @ – the portion of the loop element name before the @ symbol is used as the element name. If nothing occurs before the @ symbol, GROUP is used, and the portion after, if any, is used as the attribute name. The attribute value is the group number or the first segment name within the group, depending on whether a # or \$ is within the attribute name.

HtmlTidy Adapter Properties

You can use the HtmlTidy adapter to convert HTML to XHTML.

Adapter Name in URL

HTMLTidy

Table 9. Properties for the HtmlTidy Adapter

<i>Property Name</i>	<i>Description</i>
newline	Used when converting HTML to XHTML. The default is crlf. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.

Table 9. Properties for the HtmlTidy Adapter

Property Name	Description
warnings	Whether or not you want the adapter to fail when it encounters potential problems with the HTML-to-XHTML mapping. warnings=no is the default.
errors	Whether or not you want the adapter to fail when it encounters problems with the HTML-to-XHTML mapping. If errors=no, the adapter continues with the conversion, making a best guess. errors=yes is the default. Note that even if errors=yes, it is still possible for the adapter to fail for catastrophic problems and fail to generate valid XHTML because it cannot find a reasonable mapping for the input file it was provided.

JavaProps

See [“Properties for Other Adapters” on page 16](#) for information about the Java .properties (JavaProps) adapter.

Mbox

See [“Properties for Other Adapters” on page 16](#) for information about the Mbox adapter.

RTF

See [“Properties for Other Adapters” on page 16](#) for information about the RTF adapter.

SDI

See [“DIF, SDI, SYLK Adapter Properties” on page 10](#) for information about the Super Data Interchange (SDI) adapter.

SYLK

See [“DIF, SDI, SYLK Adapter Properties” on page 10](#) for information about the Symbolic Link format adapter.

Whole-line Text Adapter Properties

The following table describes the properties for the Whole-line Text (Line) adapter.

Adapter Name in URL

Line

Table 10. Properties for the Whole-line Text Adapter

<i>Property Name</i>	<i>Description</i>
newline	Used when converting a whole-line text file to XML, and vice versa. The default is crlf. See “Line Separator (newline) Values” on page 5 for a list of values.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.
root	Value used for the root element name. Default is root.
line	Value used for the line element name. Default is line.

WinIni

See [“Properties for Other Adapters” on page 16](#) for information about the Windows .ini (WinIni) adapter.

WinWrite

See [“Properties for Other Adapters” on page 16](#) for information about the WinWrite adapter.

Properties for Other Adapters

Each of the following adapters has the same properties. Names as they appear in adapter URLs are shown following the adapter's proper name.

- E-mail Mbox (MBox)
- Java .properties File (JavaProps)
- Progress .d Data Dump (DotD)
- Rich Text Format (RTF)
- Windows .ini File (WinIni)
- Windows Write (WinWrite)

Table 11. Properties for Other Adapters

<i>Property Name</i>	<i>Description</i>
newline	Used when converting a file to XML, and vice versa. The default is crlf. See “ Line Separator (newline) Values ” on page 5 for a list of values. Note: The Smith Corona Personal Word Processor adapter uses this value when converting to XML only.
encoding	The encoding for the input file when it is not XML; or the encoding for the output file when it is not XML. The default is utf-8.

Additional Sources of Information

You can learn more about using adapters in XQuery applications in the Stylus Studio online documentation.

- http://www.stylusstudio.com/docs/v2006/d_flatfileconversion41.html provides an overview of the Stylus Studio Java API.
- http://www.stylusstudio.com/docs/v2006/d_flatfileconversion45.html describes demo.bat, an example application shows three uses of invoking adapter URLs to convert files to and from XML.