IBM

System i

Programming
XML Toolkit

Version 6 Release 1

IBM

System i

Programming
XML Toolkit

Version 6 Release 1

> **Note**
> Before using this information and the product it supports, read the information in "Notices," on page 15.

This edition applies to version 6, release 1, modification 0 of XML Toolkit for System i (product number 5733-XT2) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CISC models.

Contents

XML Toolkit

The IBM® XML Toolkit for System i®, licensed program offering (LPO) 5733-XT2, V1R2M0, is designed to provide key development enablement components, XML parsers, to assist in the use of XML for general application or business-to-business (B2B) solutions. The XSL transformer allows you to reformat or reorganize XML documents to other formats (for example, XML or HTML).

The XML Toolkit for System i is based on cross-platform, open-source code that is designed to be compliant with industry standards. Parser version 5.6.3 corresponds to Apache Software Foundations' Xerces-C Version 2.6 with fixes. The XSL transformer corresponds to Apache Software Foundations' Xalan-C Version 1.10.2. In addition to providing the XML for C++ parser and the XSL transformer for C++, a unique interface called XML Procedural parser is provided to the XML for C++ parser that allows applications written in RPG, C, or COBOL to access an XML parser.

Also included with the XML Toolkit is XML Scripting for System i. XML Scripting for System i provides an XML based build tool.

PDF file for XML Toolkit

You can view and print a PDF file of this information.

To view or download the PDF version of this document, select XML Toolkit (about 303 KB). Note that the PDF version of the XML Toolkit information does not contain the API documentation. The API documentation is an installable option of the XML Toolkit licensed program offering.

Other information

- IBM Redbooks® publications:

  - The Ins and Outs of XML and DB2® for i (about 3 MB)
  - The XML Files: Using XML for Business-to-Business and Business-to-Consumer Applications (about 3 MB)
  - XML in Action: PDML and PCML (about 3.5 MB)
  - Integrating XML with DB2 XML Extender and DB2 Text Extender (about 4.5 MB)

Saving PDF files

To save a PDF on your workstation for viewing or printing:
1. Right-click the PDF link in your browser.
2. Click the option that saves the PDF locally.
3. Navigate to the directory in which you want to save the PDF.
4. Click **Save**.

Downloading Adobe Reader

You need Adobe Reader installed on your system to view or print these PDFs. You can download a free copy from the Adobe Web site (www.adobe.com/products/acrobat/readstep.html) .

XML introduction

Extensible Markup Language (XML) allows you to describe and organize information in ways that are easily understandable by both humans and computers. You can then share that information and its description with others over the Internet, an extranet, network, or in other ways.

XML, like Standard Generalized Markup Language (SGML), is a metalanguage. A metalanguage allows you to define a document markup language and its structure. For example, both XML and Hypertext Markup Language (HTML) are derived from SGML.

You can use XML to create your own markup language that includes a set of rules and tags that describe information suited to your needs, for example, name, title, address, and zip code. You define this markup language in a document type definition (DTD) or XML Schema file that functions as the standard way to describe your information. Using XML to share standardized information means you are no longer required to write programs to focus on proprietary software or convert and translate different data formats.

You and others can use the DTD or XML Schema definition to tag information that you can then use in a variety of ways: printed on an address label, business card, or stationary; displayed in a Web page; or sorted in a list of data with similar attributes. For example, you might want to create an efficient way to share information (such as purchase orders, shipping acknowledgments, order status, and stock status) with your partners and suppliers. You can use XML to share that information by creating and using XML documents that conform to your DTD or XML Schema, in which you specify the standard for the electronic exchange of information.

Although both XML and HTML use tags to describe content, they are also very different:
* HTML describes how to format information for display and is meant for computer-to-human interaction.
* XML describes what the information is and is meant for computer-to-computer interaction.

There are many other sources of information regarding the advantages and uses of XML that are readily available on the Internet and through commercial publications. For a complete and comprehensive understanding of the capabilities of XML, you should consult these sources.

Advantages of XML

Using XML to exchange information offers many benefits.

Advantages of XML include the following:
* XML uses human, not computer, language. XML is readable and understandable, even by novices, and no more difficult to code than HTML.
* XML is completely compatible with Java™ and 100% portable. Any application that can process XML can use your information, regardless of platform.
* XML is extendable. Create your own tags, or use tags created by others, that use the natural language of your domain, that have the attributes you need, and that makes sense to you and your users.

The following example illustrates, in a simplified way, the readability and extensibility of XML:

HTML example	XML example

<pre><HTML>
<H1 ID="MN">State</H1>
<H2 ID="12">City</H2>
<DL>
  <DT>Name</DT>
    <DD>Johnson</DD>
  <DT>Population</DT>
    <DD>5000</DD>
</DL>
<H2 ID="15">City</H2>
<DL>
  <DT>Name</DT>
    <DD>Pineville</DD>
  <DT>Population</DT>
    <DD>60000</DD>
</DL>
<H2 ID="20">City</H2>
<DL>
  <DT>Name</DT>
    <DD>Lake Bell</DD>
  <DT>Population</DT>
    <DD>20</DD>
</DL>
</HTML></pre>
	<pre><?XML VERSION="1.0" STANDALONE="yes" ?>
	<STATE STATEID="MN">
	  <CITY CITYID="12">
	    <NAME>Johnson</name>
	    <POPULATION>5000</POPULATION>
	  </CITY>
	  <CITY CITYID="15">
	    <NAME>Pineville</NAME>
	    <POPULATION>60000</POPULATION>
	  </CITY>
	  <CITY CITYID="20">
	    <NAME>Lake Bell</NAME>
	    <POPULATION>20</POPULATION>
	  </CITY>
	</STATE></pre>

HTML tag names reveal nothing about the meaning of their content. The example above uses an HTML definition list, but the problems inherent in using HTML occur if the data were contained in a table or some other kind of HTML tags. Examples follow:

- Many of the HTML tags are acronyms, so they are not as readable as common language.
- HTML tags represent data (in this example, city names and populations) as items to display, for example, as definitions in a list or cells in a table. This makes it difficult to manipulate the data or to exchange it between applications.

The XML tag names are readable and convey the meaning of the data. The information structure is easily discerned by both humans and computers as each XML tag immediately precedes the associated data. The data structure follows a noticeable and useful pattern, making it easy to manipulate and exchange the data.

Uses of XML

XML has a variety of uses for Web, e-business, and portable applications.

The following are some of the many applications for which XML is useful:

- **Web publishing**: XML allows you to create interactive pages, allows the customer to customize those pages, and makes creating e-commerce applications more intuitive. With XML, you store the data once and then render that content for different viewers or devices based on style sheet processing using an Extensible Style Language (XSL)/XSL Transformation (XSLT) processor.
- **Web searching and automating Web tasks**: XML defines the type of information contained in a document, making it easier to return useful results when searching the Web:

  For example, using HTML to search for books authored by Tom Brown is likely to return instances of the term 'brown' outside of the context of author. Using XML restricts the search to the correct context (for example, the information contained in the <author> tag) and returns only the information that you want. By using XML, Web agents and robots (programs that automate Web searches or other tasks) are more efficient and produce more useful results.
- **General applications**: XML provides a standard method to access information, making it easier for applications and devices of all kinds to use, store, transmit, and display data.

- **e-business applications**: XML implementations make electronic data interchange (EDI) more accessible for information interchange, business-to-business transactions, and business-to-consumer transactions.
- **Metadata applications**: XML makes it easier to express metadata in a portable, reusable format.
- **Pervasive computing**: XML provides portable and structured information types for display on pervasive (wireless) computing devices such as personal digital assistants (PDAs), cellular phones, and others. For example, WML (Wireless Markup Language) and VoiceXML are currently evolving standards for describing visual and speech-driven wireless device interfaces.

XML standards and extensions

Other standards and extensions to XML work together to make your information more portable and useful.

You need to know about these standards and extensions in order to do the following:
- Use XML with your Java, C++, C, RPG, and COBOL programs.
- Perform complex data searches in XML documents.
- Display XML data on different types of devices.
- Provide your XML documents with hierarchical linking capabilities.
- Produce standard structures for related DTDs.

XML is good for describing information, but it cannot do everything. For example, XML documents do not contain the kind of information that current browsers and many other devices require to display it in a useful way. The same is true for linking to other information, transporting XML data so that it can be used in a meaningful way by the receiving application, and so on.

The XML community has and continues to develop standards and extensions to expand the capabilities of XML:
- DOM and SAX APIs
- Namespaces
- XSL and XSLT
- XLink
- XPath and XPointer
- XML Schema

APIs

Application programming interfaces (APIs) allow applications to work with XML information using a standard set of portable interfaces.

DOM 1.0 and DOM Level 2.0

The Document Object Model (DOM) API enables you to build XML documents as well as parse them. These interfaces enable you to access, manipulate, and create XML documents (and the data within) as programming objects that have methods and events. Your programs can construct or change a DOM tree in memory and then persist that DOM tree to a file or stream. DOM is best suited for instances where you will parse few XML documents but require extensive control over the contents.

Namespaces

Namespaces are pointers that enable you to differentiate between duplicate XML elements or attribute names, a situation that can occur when using XSLT style sheets or more than a single DTD. For example, the <code> element from one DTD might mean something different from a <code> element in another

DTD. To avoid name collisions and ambiguity, give each pointer a unique local name. This makes it simple to distinguish between the different namespaces.

SAX 1.0 and SAX 2.0

The Simple API for XML (SAX) is a read-only, single-pass interface best suited for processing many documents or very large documents. You can use this API to extract information from the XML documents, but you cannot use it to add new data to or to change the content of the XML documents. The SAX API is event-driven, notifying your application when certain events happen as it parses your document. For example, your application might need to know when the parser encounters the start or end of an element node. Note that it is your application that must keep the necessary state information to determine the content and context of these XML events.

XSL and XSLT

Extensible Stylesheet Language (XSL) and XSL Transformation (XSLT) work in combination to enable you to display XML data in a variety of ways, for example, displayed in a browser or on a PDA, or printed in a brochure. XSL and XSLT processing also enable you to transform an XML message or document from one XML markup language to another, which has key applications in e-business.

See "XSL introduction" on page 11 for more information.

XLink

XML Linking Language (XLink) enables you to link your XML document to other resources on the web, including files of just about any format, database searches, and so on. Moreover, you can link to the structure of the resource, not a predetermined place holder, like an HTML <A NAME> anchor tag. Multiple links allow users to traverse the linked information in any order according to restrictions that you specify.

XPath and XPointer

XML Path Language (XPath) and XML Pointer Language (XPointer) enable you to search for and identify data in the hierarchical XML document structure.

XPath defines a syntax for locating data in an XML document. Both XSLT and XPointer use XPath. XPath defines an XML document as a hierarchy of nodes, with the top node being the root. Just like using a regular expression finds one or more patterns in text, using XPath finds patterns in data within the nodes of one or more XML documents.

XML Pointer Language (XPointer) extends XPath to enable locating specific portions of data, called fragments, based on XML attribute values, types, content, or relative position. These fragments can be discrete pieces of data, a range of information between two points, or a continuous series of elements.

XML Schema

XML Schema Language defines the logical structure of an XML document, much like a document type definition (DTD).

The significant difference between DTDs and XML Schemas are that schemas do the following:
- Are written as XML markup language itself, making them extensible, unlike DTDs
- Focus on the problem of cardinality, enabling the enumeration of minimum and maximum allowed elements
- Allow constraints on values
- Allow additional data types and definitions of data types that can be inherited

All of these enhancements give you more control over the allowable content of the XML document or message. For example, you can add a different type of element to an existing schema as long as your addition does not break the original schema. Schemas also have many more available data types than do DTDs, making importing and exporting data somewhat easier.

XML Toolkit introduction

To ensure the availability of the latest versions of the XML parsers, XSL transformer, and XML Scripting for System i, the XML Toolkit for System i (5733-XT2) was developed. Each available version of the XML parser for both the C++ and the procedural languages and the XSL transformer consists of two installation options in the 5733-XT2 product.

The first installation option provides the service programs that support that parser version and the XSL transformer. The second installation option provides include files, samples, API documentation, and bind instruction associated with the parser version and transformer. XML Scripting for System i contains a program, service program, samples, and documentation.

Available XML parsers, XSL transformers, and XML Scripting functions

The XML Toolkit currently consists of the XML parser, XSL transformer, and XML Scripting function versions listed on the table below.

Each version of the parser consists of two installation options: service programs (parsers and transformers) and development environments (API documentation, samples, and include files). The product will be periodically refreshed as these new parsers are released. This provides a single point of contact for the latest in i5/OS™ XML-related technologies.

Options 4 and 5 of 5733-XT2, detailed below, may be installed on systems running IBM i V5R3 and later. Options 6 and 7 may only be installed on systems running V6R1 and later. Other than the differences in target releases, the functionality of these two sets of options is the same.

Options 6 and 7 are available for those who want to run XML Toolkit on V6R1 and avoid initial retranslation costs.

Install option in 5733-XT2, V1R2M0	Contents	Minimum target release supported	Apache Software version
1	XML parser version 5.6 and XSL transformer version 1.10 service programs	V5R4M0	2.6 with fixes (XML), 1.10 (XSL)
2	Include files, API documentation, and samples of XML parser version 5.6 and XSL transformer version 1.10	V5R4M0	Not applicable
3	XML Scripting for System i	V5R4M0	XML Scripting for System i includes C++ code that uses Apache Software Foundation's Ant version 1.5.4, which is written in Java as a base.
4	XML parser version 5.6.3 and XSL transformer version 1.10.2 service programs	V5R3M0	2.6 with fixes (XML), 1.10.2 (XSL)

Install option in 5733-XT2, V1R2M0	Contents	Minimum target release supported	Apache Software version
5	Include files, API documentation, and samples of XML parser version 5.6.3 and XSL transformer version 1.10.2	V5R3M0	Not applicable
6	XML parser version 5.6.3 and XML transformer version 1.10.2 service programs	V6R1M0	2.6 with fixes (XML), 1.10.2 (XSL)
7	Include files, API documentation, and samples of XML parser version 5.6.3 and XSL transformer version 1.10.2	V6R1M0	Not applicable

Related information:

➥ XML for i5/OS home page

Installing XML Toolkit

The XML Toolkit allows multiple XML parsers, XSL transformer versions, and XML Scripting to be installed on the same system simultaneously.

The installation procedures are structured to allow you to separately install the parsers and transformer (service programs) or the development environment (API documentation, include files, and samples) or both for each XML parser and XSL transformer version you want.

XML Toolkit V1R2M0 Options	Contents
*BASE	Base code functions
Option 1	XML version 5.6 parser service programs. This option is installed as QXML4C560 and QXML4PR560 in library QXMLLIB. In addition, XSL version 1.10 transformer service program is installed as service program QXSL4C110 in library QXMLLIB.
Option 2	• XML version 5.6 parser API documentation, sample, and include files. The C, RPG, and COBOL development environment is installed in library QXMLDEV560. The C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkit2/xml5_6_0. • XSL version 1.10 transformer API documentation, sample, and include files. The development environment for this is installed in library QXSLDEV110. The C and C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkit2/xsl_1100.
Option 3	XML Scripting for System i program and service program. This option is installed as QXMLSCR10 for both the program and the service program in library QXMLLIB. The supporting files and documentation are installed in the integrated file system directory /QIBM/ProdData/xmltoolkit2/scripting1_0.

XML Toolkit V1R2M0 Options	Contents
Option 4	XML version 5.6.3 parser service programs. This option is installed as QXML4C563 and QXML4PR563 in library QXMLLIBV4. In addition, XSL version 1.10 transformer service program is installed as service program QXSL4C112 in library QXMLLIBV4.
Option 5	• XML version 5.6.3 parser API documentation, sample, and include files. The C, RPG, and COBOL development environment is installed in library QXML563V4. The C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkitv4/xml5_6_3. • XSL version 1.10.2 transformer API documentation, sample, and include files. The development environment for this is installed in library QXSL112V4. The C and C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkitv4/xsl_1102.
Option 6	XML version 5.6.3 parser service programs. This option is installed as QXML4C563 and QXML4PR563 in library QXMLLIBV6. In addition, XSL version 1.10 transformer service program is installed as service program QXSL4C112 in library QXMLLIBV6.
Option 7	• XML version 5.6.3 parser API documentation, sample, and include files. The C, RPG, and COBOL development environment is installed in library QXML563V6. The C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkitv6/xml5_6_3. • XSL version 1.10.2 transformer API documentation, sample, and include files. The development environment for this is installed in library QXSL112V6. The C and C++ development environment is installed in the integrated file system directory /QIBM/ProdData/xmltoolkitv6/xsl_1102.

All samples and documentation are distinguished by the XML parser, XSL transformer, and XML Scripting version. For XML, this is library QXMLDEVnnn and integrated file system directory /QIBM/ProdData/xmltoolkit2/XMLn_n_n, where nnn and n_n_n refer to the XML version.

For XSL, this is library QXSLDEVnnn and integrated file system directory /QIBM/ProdData/ xmltoolkit2/XSLn_n_n, where nnn and n_n_n refer to the XSL version.

For XML Scripting, this is integrated file system directory /QIBM/ProdData/xmltoolkit2/scriptingn_n, where nnn and n_n refer to the XML Scripting version.

To install XML Toolkit, use the Restore Licensed Program (RSTLICPGM) command. Install the *BASE option and any other required options. For example:

```
RSTLICPGM LICPGM(5733XT2) DEV(xxxxxx) OPTION(*BASE)
```
      Installs the *BASE code required for all other options.

RSTLICPGM LICPGM(5733XT2) DEV(xxxxxx) OPTION(1)
       Installs the XML version 5.6 parser service programs and the XSL version 1.10 transformer service program.

RSTLICPGM LICPGM(5733XT2) DEV(xxxxxx) OPTION(2)
       Installs the XML version 5.6 parser development environments and the XSL version 1.10 transformer development environments.

To delete a portion of the XML Toolkit, run the Delete Licensed Program (DLTLICPGM) command. For example:

DLTLICPGM LICPGM(5733XT2) OPTION(2)
       Deletes Option 2 (development environment for XML version 5.6 parsers).

API documentation

The documentation for each XML parser version is available in browser format when you install the appropriate licensed program development option for the XML parser version of your choice.

To view the documentation, map a network drive to your target system and, using your browser, open the file location specified in the table below.

The documentation includes API descriptions, information about how to create and run the sample programs, some programming information, and frequently asked questions (FAQ).

Table 1. Documentation locations

Option	Version	Documentation location in the integrated file system
2	XML4C version 5.6	/QIBM/ProdData/xmltoolkit2/xml5_6_0/doc/xml4c/index.html
2	XML4PR version 5.6	/QIBM/ProdData/xmltoolkit2/xml5_6_0/doc/xml4pr/index.html
2	XSLT4C version 1.10	/QIBM/ProdData/xmltoolkit2/xsl_1100/doc/xsl4c/index.html
3	XML Scripting for System i	/QIBM/ProdData/xmltoolkit2/scripting1_0/docs/index.html
5	XML4C version 5.6.3	/QIBM/ProdData/xmltoolkitv4/xml5_6_3/doc/xml4c/index.html
5	XML4PR version 5.6.3	/QIBM/ProdData/xmltoolkitv4/xml5_6_3/doc/xml4pr/index.html
5	XSLT4C version 1.10.2	/QIBM/ProdData/xmltoolkitv4/xsl_1102/doc/xsl4c/index.html
7	XML4C version 5.6.3	/QIBM/ProdData/xmltoolkitv6/xml5_6_3/doc/xml4c/index.html
7	XML4PR version 5.6.3	/QIBM/ProdData/xmltoolkitv6/xml5_6_3/doc/xml4pr/index.html
7	XSLT4C version 1.10.2	/QIBM/ProdData/xmltoolkitv6/xsl_1102/doc/xsl4c/index.html

For information about mapping network drives, see Accessing file shares from a Windows client.

Related concepts:

Accessing file shares from a Windows client

"Samples"
Samples for each XML parser, XSL transformer, and XML Scripting version provided with the XML Toolkit are available by installing the appropriate development option for each.

Samples

Samples for each XML parser, XSL transformer, and XML Scripting version provided with the XML Toolkit are available by installing the appropriate development option for each.

For XML, C++ samples are provided through integrated file system files. RPG, C, and COBOL samples are provided through source physical files.

For XSL, the C and C++ samples are provided through integrated file system files.

For XML Scripting, the samples are provided through integrated file system files.

The version for these is indicated in either the integrated file system path location (n_n_n or n_n) or in the name of the tools development library (QXMLDEVnnn for XML or QXSLDEVnnn for XSL). Information describing the function of each sample and how to create, bind, and run the samples for a given supported language are provided. See the "API documentation" on page 9.

A source file, QCLSRC, is provided in the tools development library and contains members that create the samples provided for each language version.

Table 2. Samples locations

Option	XML parser version	Samples location
2	XML4C version 5.6	Integrated file system location: /QIBM/ProdData/ xmltoolkit2/xml5_6_0/samples. The include files are located in the /QIBM/ProdData/xmltoolkit2/xml5_6_0/ include/xercesc directory.
2	XML4PR version 5.6	Library: QXMLDEV560. This library contains the QRPGLESRC, QCBLLESRC, QCSRC, and H source files. The include files are contained in the corresponding file member QXML4PR560.
2	XSLT4C version 1.10	Integrated file system location: /QIBM/ProdData/ xmltoolkit2/xsl_1100/samples. The include files are located in the /QIBM/ProdData/xmltoolkit2/xsl_1100/include/ xalanc directory.
3	XML Scripting for System i	Integrated file system location: /QIBM/ProdData/ xmltoolkit2/scripting1_0/samples
5	XML4C version 5.6.3	Integrated file system location: /QIBM/ProdData/ xmltoolkitv4/xml5_6_3/samples. The include files are located in the /QIBM/ProdData/xmltoolkitv4/xml5_6_3/ include/xercesc directory.
5	XML4PR version 5.6.3	Library: QXML563V4. This library contains the QRPGLESRC, QCBLLESRC, QCSRC, and H source files. The include files are contained in the corresponding file member QXML4PR563.
5	XSLT4C version 1.10.2	Integrated file system location: /QIBM/ProdData/ xmltoolkitv4/xsl_1102/samples. The include files are located in the /QIBM/ProdData/xmltoolkitv4/xsl_1102/ include/xalanc directory.

Table 2. Samples locations (continued)

Option	XML parser version	Samples location
7	XML4C version 5.6.3	Integrated file system location: /QIBM/ProdData/ xmltoolkitv6/xml5_6_3/samples. The include files are located in the /QIBM/ProdData/xmltoolkitv6/xml5_6_3/ include/xercesc directory.
7	XML4PR version 5.6.3	Library: QXML563V6. This library contains the QRPGLESRC, QCBLLESRC, QCSRC, and H source files. The include files are contained in the corresponding file member QXML4PR563.
7	XSLT4C version 1.10.2	Integrated file system location: /QIBM/ProdData/ xmltoolkitv6/xsl_1102/samples. The include files are located in the /QIBM/ProdData/xmltoolkitv6/xsl_1102/ include/xalanc directory.

Related concepts:

"API documentation" on page 9

The documentation for each XML parser version is available in browser format when you install the appropriate licensed program development option for the XML parser version of your choice.

XSL introduction

Extensible Stylesheet Language (XSL) allows you to format and reorganize existing XML documents into another format (for example, XML or HTML).

You can use XSL to reformat an XML file to specify the how the data is to be presented on the web via XSLT stylesheets. You can also write XSLT stylesheets to reorganize an XML document to remove or add elements or to change it into another XML document. XSLT stylesheets are written in XML to achieve this.

This process has two basic components:

- Use XSL stylesheets to define a set of patterns and templates you want to use to replace XML elements. A pattern identifies the XML element, and the corresponding template is used by an XSL processor to actually replace the XML element. For example, you can transform data elements in an XML document to display appropriately in a browser or mailing label.
- Use XSLT documents to transform the hierarchy of XML data into a different kind of tree, reordering elements as you prefer. For example, you can add a table of contents or an index to a set of data that does not have one. You can also use XSLT to transform the grammar of XML documents. For example, you can transform the grammar for a set of incoming XML request documents to a different XML grammar required by the receiving application.

These technologies do more than format the display of an XML document. They change it so that it becomes a different kind of document. In conjunction with other XML tools and extensions, such as parsers and XML Linking Language (XLink), you can produce new documents formats such as specific word processing formats, portable document format (PDF), HTML, and more.

Uses of XSL

XSL can be used for many XML transformation and reformatting tasks.

The transformation and reformatting tasks that you can do with XSL include the following:
- XML to HTML rendering
- XML vocabulary conversion
- XML to plain text conversion

- Simple XML queries

XML Scripting for System i introduction

XML Scripting for System i uses an XML formatted file to accomplish a set of tasks on the system. These tasks are grouped in "targets" which allow a different set of tasks to be run in different invocations of the XML Scripting tool. The targets can have dependencies on each other.

The XML Scripting for System i tool is inspired by Apache Ant, but it is not compatible with that product and has a different focus. Currently, the tool is available only in the English language and has no extension API.

Getting support

Review the statement of support for the XML Toolkit.

Support services for the XML Toolkit are provided under the typical terms and conditions for IBM software products. Support services include program services, voice support, and consulting services. Use the online information provided at the System i5® Home Page under the topic Support for more information, or contact your local IBM representative. These terms and conditions include the following:

- You may, at IBM direction, be required to obtain a more current level of the XML Toolkit and move to a newer version of the XML parser, XSL transformer, or XML Scripting to receive Continued Program Services (CPS) support.
- Resolving defects with the XML Toolkit parsers are supported under program services or voice support. Resolving application programming or debugging problems is supported under consulting services.
- XML Toolkit application programming interface (API) calls are supported under consulting services unless:
    - It is clearly an XML Toolkit API defect as demonstrated by recreation in a relatively simple program.
    - It is a question that asks for documentation clarifications.
    - It is a question about the location of samples or documentation.
- All programming assistance is supported under consulting services. This includes the program samples that are provided in the XML Toolkit licensed program.
- If you believe that there is a potential defect in the XML Toolkit APIs, a simple program that demonstrates the error is required.

Related concepts:

System i5 Home Page

Related information for XML Toolkit

The XML community continually improves existing tools, adds new tools, and helps evolve new standards and extensions that increase the usefulness and flexibility of XML.

Use these IBM Redbooks publications and Web sites to keep up to date on developments in the XML community and to see more information about XML, including introductory information, tutorials, advanced and reference materials, evolving XML standards, and XML tools that you can download.

IBM Redbooks publications

- The Ins and Outs of XML and DB2 for i (about 3 MB)
- The XML Files: Using XML for Business-to-Business and Business-to-Consumer Applications (about 3 MB)

- XML in Action: PDML and PCML (about 3.5 MB)

- Integrating XML with DB2 XML Extender and DB2 Text Extender (about 4.5 MB)

Web sites

- W3C - XML
- www.xml.org
- www.xml.com

Other information

- developerWorks® XML Zone
- XML on System i

Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

This XML Toolkit publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of XML Toolkit for System i.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

DB2
developerWorks
i5/OS
IBM
iSeries
Redbooks
System i
System i5

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Terms and conditions

Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER

EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF
MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

IBM ®

Printed in USA