



www.databasejournal.com/features/mssql/article.php/3880091

[Back to Article](#)

An Overview of the SQL Server xml Data Type
May 7, 2010

XML is, it seems, everywhere. SQL Server has ever-improving functionality that helps us peek into, shred, store, manipulate and otherwise utilize XML. This article covers XML variables, XML columns, typed vs. untyped XML, and the xml data type methods.



Introduction

XML is, it seems, everywhere. Businesses use it to coordinate, collaborate, and exchange data with other businesses. SQL Server has ever-improving functionality that helps us peek into, shred, store, manipulate and otherwise utilize XML. This article covers XML variables, XML columns, typed vs. untyped XML, and the xml data type methods.

xml Data Type

The much anticipated (at least by me) xml data type was introduced into SQL Server 2005. The new data type allows variables and columns to be created to specifically hold xml data, whether it be an entire XML document or just some XML content.



XML Variables

Prior to SQL Server 2005, your options for storing XML data were limited to text or varchar columns or variables. Let's look at a couple now and then scenarios.

Scenario:

You need to create a stored procedure to process incoming XML from a partner.

Then (SQL 2000):

You would have created a procedure with a text or varchar parameter and loaded the value of that parameter into memory with `sp_xml_preparedocument`. You would then use `SELECT` statements with `OPENXML` to get data out utilizing `XPath 1.0` expressions. Finally, you execute `sp_xml_removedocument` to get the XML out of memory.



MARKETPLACE

[Windows 7 will benefit your business](#)

Learn how the new features in Windows 7 will improve your IT and Ops efficiency.
www.microsoft.com

[IT HelpDesk & Customer Support Software](#)

Internal IT HelpDesk Software with Asset Mgmt. Customer Support Software with Account & Contact Mgmt

the XML data type.

Now (SQL 2005, 2008)

You now simply create a procedure with an xml parameter. Use new xml data type query() and nodes() methods to get the data out. (We'll dive into these shortly.)

[DOWNLOAD YOUR FREE TRIAL TODAY! www](#)

Revenue for IT Resellers

Provide Online Data Backup Services Perfect For SMB Clients. Try It Today.

[DataDepositBox.com/Resellers](#)

Scale-out your web app data, lower costs

New data mgt tool offloads RDBMS to boost scalability/save money. Free download.

[www.northscale.com](#)

Scenario:

You need to store off XML data you've received from a third party so you can join it with your system's data in a report.

Then (SQL 2000)

Prior to the xml data type, large XML could be stored in a text column, and you could read it into a temp table that had a text column in it. However, you could not then use OPENXML with that text column of the temporary table to examine the XML.

Now (SQL 2005, 2008)

Today if you want to store the large XML, you would store it in an XML column and simply read it back into an xml data type variable for processing. You could then proceed to examine and manipulate it utilizing the xml data type methods discussed below.

XML Columns

SQL Server tables can now contain xml data type columns. These columns can contain XML content and XML documents. You must specify this at create time. The default is to create the column to hold XML content. The difference being that XML content can have multiple top-level elements and the XML document has to be an XML 1.0 document. In both cases, the XML will be checked to determine if it is well-formed.

Typed vs Untyped xml data types

There are two types of XML data, untyped and typed. For untyped XML data, no schema is defined and no validation is required. Conversely, schemas are defined and validation information is a concern for typed XML data. Additionally, the schema must be stored in SQL Server prior to adding the typed XML column to a table. (See [MSDN](#) for information on creating schema collections.) Below is first an example of creating an untyped XML column and an example of creating a typed XML column follows.

```
CREATE TABLE myTable(
IDCol BIGINT IDENTITY,
MyXMLCol XML NULL)
CREATE TABLE myTable(
IDCol BIGINT IDENTITY,
MyXMLCol XML NULL (myDBSchema.mySchemaCollection))
```

xml data type methods

The xml data type comes with four methods that assist with querying and manipulating the XML data stored in the column or variable. Those methods are query(), value(), nodes(), and modify().

query()

The query() method of the xml data type uses the XQuery Language to search inside xml data types. You can find out more about XQuery support in SQL Server 2008 at [MSDN](#).

The following example shows creating some sample XML in an XML variable and then querying into that data. The first query gets the first exercise element. The second query gets the data for the Reps attribute.

```
DECLARE @myDoc xml
SET @myDoc = '
< Location="Chest">
Reps = "15">Exercises in order.
Bench Press
Flies
'

-- query the first exercise in the xml
SELECT @myDoc.query('/Workout/BodyPart/Exercise[1]')
--query the attribute Reps
SELECT @myDoc.query('data(/Workout/BodyPart[1]/@Reps)')
```

First Result:

Bench Press

Second Result:

15

Exist()

If you needed to determine if the XML data contains a certain node, you could use the exist() method of the xml data type. In the example below, the exist() method is used in

the where clause to allow only those nodes containing an Exercise element to be queried.

```
DECLARE @myDoc xml
SET @myDoc = '
< Location="Chest">
Reps = "15">Exercises in order.
Bench Press
Flies
< Location="Triceps">
Reps = "12">Exercises in order.
'
-- query the first exercise in the xml
SELECT @myDoc.query('/Workout/BodyPart/*')
WHERE @myDoc.exist('/Workout/BodyPart/Exercise') = 1
```

Result:

Bench Press

Flies

Value()

The method value() uses XQuery to examine the XML and return back a scalar value of the specified SQL type. The SQL type cannot be image, text, ntext, xml, a CLR type, or sql_variant . This example uses the value() method to return the value inside the first exercise node. The result is returned as a VARCHAR(20).

```
DECLARE @myDoc xml
SET @myDoc = '
< Location="Chest">
Reps = "15">Exercises in order.
Bench Press
Flies
< Location="Triceps">
Reps = "12">Exercises in order.
'
-- query the first exercise in the xml
SELECT @myDoc.value('/Workout/BodyPart/Exercise')[1], 'VARCHAR(20)')
```

Result:

Bench Press

nodes()

nodes() is used to shred the XML nodes of the XML data into relational columns. This is similar to what OPENXML was used for in SQL Server 2000. Using the same sample XML, this query uses nodes() to first create a rowset from the XQuery provided. In this case, the rowset was for the first exercise node and the select merely queried for the entire rowset of the exercise element.

```
DECLARE @myDoc xml
SET @myDoc = '
< Location="Chest">
Reps = "15">Exercises in order.
Bench Press
Flies
< Location="Triceps">
Reps = "12">Exercises in order.
'
-- query the first exercise in the xml
SELECT T.exercise.query('.')
FROM @myDoc.nodes('/Workout/BodyPart/Exercise')[1] as T(exercise)
```

Result:

Bench Press

modify()

To change or remove a portion of the content in an XML variable or column, the modify() method can be used. XML Data Manipulation Language (DML) is used to specify the action within the modify() method. With XML DML you can insert nodes, change nodes, and delete nodes. More information on XML DML can be found at [MSDN](#). Here is an example of modifying the sample XML to include an additional exercise prior to the one specified in the XQuery statement.

```
DECLARE @myDoc xml
SET @myDoc = '
Exercises in order.
Flies
'
-- update text in the first exercise
SET @myDoc.modify('
insert Decline Press
before (/Workout/BodyPart/Exercise)[1]')
SELECT @myDoc
```

Result:

Exercises in order.

Conclusion

Over the past decade, Microsoft has been evolving the capabilities of SQL Server with regards to XML processing. With this latest version, the syntax used to parse through and manipulate XML has been greatly simplified. SQL Server now has much more powerful and sophisticated XML tools at your finger tips.

For More Information

[Working with the XML Data Type of SQL Server](#)

[MSDN](#) XML (Transact-SQL)

[MSDN](#) xml Data Type Methods

[MSDN](#) XML DML

» [See All Articles by Columnist Deanna Dicken](#)

Business Resources

[Free Online Database](#)

Remote Users Enter and Manage their Data. You Report & Share Results. Start Now !

www.InfoDome.com

FREE WHITEPAPER

New Security Solutions Using Intel vPro Technology

Intel focused on three high-priority, high-value categories and successfully performed lab tests to validate each use case.

[CLICK HERE](#)

internet.com®

The Network for Technology Professionals

Search: [Find](#)

[About Internet.com](#)

Copyright 2010 QuinStreet Inc. All Rights Reserved.

[Legal Notices](#), [Licensing](#), [Permissions](#), [Privacy Policy](#).

[Advertise](#) | [New sletters](#) | [E-mail Offers](#)

Solutions

Whitepapers and eBooks

Article: Adobe Bridges Designer-Developer Gap with Flash Catalyst

Microsoft PDF: What's New in SQL Server 2008 R2

Article: Adobe Broadens the Appeal of RIA Development with Flash Builder

4

PDF: Fact-Based Comparison of Hosted Services: Google vs. Microsoft

Article: Java Developers Finding a Home at Adobe Flex

Microsoft Personalized Whitepapers and Resources for You

Article: Cutting IT Costs with SQL Server 2008

Microsoft PDF: 12 Ways to Reduce Costs with SQL Server 2008

Article: Adobe Helps PHP Developers Create Rich Internet Applications

MORE WHITEPAPERS, EBOOKS, AND ARTICLES

Webcasts

Create Applications Without Infrastructure Limits with Windows Azure:
Access Announcements and Events

MORE WEBCASTS, PODCASTS, AND VIDEOS

Downloads and eKits

Microsoft's Cloud Services
Iron Speed Designer Application Generator

MORE DOWNLOADS, EKITS, AND FREE TRIALS

Tutorials and Demos

Microsoft Article: Learn Windows Azure Platform for Enterprises

Learn Unified Communications

Learn SQL Server 2008

Learn Windows Server 2008 R2

Internet.com Hot List: Get the Inside Scoop on IT and Developer Products

Learn Forefront

Learn System Center

All About Botnets

Learn SharePoint

MORE TUTORIALS, DEMOS AND STEP-BY-STEP GUIDES

