DB2® Universal Database for OS/390 and z/OS
XML Extender

IBM

Release Notes

Version 7

DB2® Universal Database for OS/390 and z/OS
XML Extender

Release Notes

Version 7

Contents

About the Release Notes for DB2® Universal Database for OS/390 and z/OS, XML Extender Version 7

This document contains information about the DB2 Universal Database for OS/390 and z/OS, XML Extender Version 7, supplementing information in *DB2 Universal Database for OS/390 and z/OS® XML Extender Administration and Programming, Version 7*. The information in these Release Notes includes all topics except installation.

Chapter 1. What's in this document?

This PTF fixes several problems reported by customers using DB2 XML Extender for UNIX or Windows. This PTF updates the current version of DB2 XML Extender for zSeries to include the fixes listed in DB2 XML Extender Version 7.2 FixPak 8 for UNIX or Windows. The documentation updates listed described in this document are based on APAR PQ66251 in PTF UQ74105.

For more information about the DB2 UDB XML Extender Version 7.2 FixPak 8, please see the release notes. They can be found at:

`http://www.ibm.com/software/data/db2/extenders/xmlext/library.html`

This document provides updates to information about topics in the *DB2 Universal Database for OS/390 and z/OS(R) XML Extender Administration and Programming, Version 7*, and the online error messages.

You can find *DB2 Universal Database for OS/390 and z/OS(R) XML Extender Administration and Programming, Version 7* on the DB2 XML Extender Web site:

`www.ibm.com/software/data/db2/extenders/`

Chapter 2. Fixes and updates

The following topics describe fixes provided in this PTF, as well as new and updated information for *DB2 Universal Database for OS/390 and z/OS(R) XML Extender Administration and Programming, Version 7.*

Migrating to the PTF from prior releases

If you have been using DB2 XML Extender Version 7 you must migrate each DB2 subsystem enabled for DB2 XML Extender before using them with this PTF.

Migration from PTF UQ65515 will create XMLDBCLOB user-defined types (UDTs) and user-defined functions (UDFs) for use with Unicode and DBCS databases.

Migration from PTFs prior to PTF UQ65515 will be performed as well. Details of migration from those prior PTFs is documented in the release notes for PTF UQ65515.

To migrate the subsystem: Customize and run the following two jobs in the SDXXJCL data set. Instructions for customizing the jobs are in the JCL prologue for each job.

DXXRBIND
> This job rebinds all the DB2 packages and plans.

DXXMIGRA
> This job drops, creates, and alters functions and stored procedures.

Failing to do the migration step can cause the **DXXADM disable_server** option and the dxxGenXMLCLOB and dxxRetrieveXMLCLOB stored procedures to fail

Miscellaneous Fixes

Null values in SQL Composition

> Key columns having NULL values (either as attribute node or text node) in the retrieved result set will not be emited as a part of the resulting document. Also, child elements of such an element will not be emitted as well, regardless of whether or not the child elements have NOT NULL data. Key columns are specified in the ORDER BY clause

> For example, when generating a document from the following table:

> | ACTUAL | CHILD1 | CHILD2 | CHILD3 | CHILD4 | X |
> |---|---|---|---|---|---|
> | 1 | 2 | 3 | 4 | 5 | 1 |
> | 1 | 2 | 3 | 4 | 6 | 2 |
> | 1 | 2 | 3 | 4 | 7 | 3 |
> | 1 | 2 | 8 | - | - | 4 |
> | 1 | 2 | 9 | 10 | - | 5 |
> | 1 | 2 | 9 | 11 | - | 6 |

> XML Extender might produce the following XML document:

> ```
> <?xml version="1.0"?>
> <!DOCTYPE topic SYSTEM "/mydtd.dtd">
> <topic_actual toc_title="1">
>   <topic_1 toc_title="2">
> ```

```
            <topic_2 toc_title="3">
              <topic_3 toc_title="4">
                <topic_4 toc_title="5">
                  <topic_X>1</topic_X>
                </topic_4>
                <topic_4 toc_title="6">
                  <topic_X>2</topic_X>
                </topic_4>
                <topic_4 toc_title="7">
                  <topic_X>3</topic_X>
                </topic_4>
              </topic_3>
            </topic_2>
            <topic_2 toc_title="8">
            </topic_2>
            <topic_2 toc_title="9">
              <topic_3 toc_title="10">
              </topic_3>
              <topic_3 toc_title="11">
              </topic_3>
            </topic_2>
          </topic_1>
        </topic_actual>
```

Updates to UDF's

The following user-defined functions have a new parameter that specifies the encoding of the source or output file. The value of this parameter is any code page name that is recognized by ICU.

```
db2xml.XMLVarcharFromFile(filename varchar(512), src_encoding  varchar(100))
returns XMLVarchar
```

```
db2xml.XMLCLOBFromFile(filename varchar(512), src_encoding varchar(100))
returns XMLCLOB AS LOCATOR
```

```
db2xml.XMLFileFromVarchar(doc varchar(3000), targetfilename varchar(512),
targetencoding varchar(100))
returns XMLFile
```

```
db2xml.XMLFileFromCLOB(doc CLOB(2G) as LOCATOR, targetfilename varchar(512),
 targetencoding varchar(100))
returns XMLFile
```

```
db2xml.Content(doc XMLVarchar, targetfilename varchar(512),
 targetencoding varchar(100))
returns varchar(512)
```

```
db2xml.Content(doc XMLCLOB as LOCATOR, targetfilename varchar(512),
 targetencoding varchar(100))
returns varchar(512)
```

Example 1: To import the contents of a file /home/collins/xml/entail.xml into a varchar buffer and to specify that the source file is encoded in iso-8859-1:

```
db2xml.XMLVarcharFromFile('/home/collins/xml/entail.xml', 'iso-8859-1')
```

The file is imported into a varchar and converted from iso-8859-1 to the database code page.

Example 2: To export a varchar buffer into a file /home/raskolnikov/xml/confession.xml and to specify that the output file should be encoded in ibm-808:

```
db2xml.Content('<?xml version="1.0" encoding="utf-8"?>
<sequence><thought>I did it!</thought></sequence>',
'/home/raskolnikov/xml/confession.xml', 'ibm-808')
```

The contents of the buffer are exported to a file and converted from the database code page to `ibm-808`, and the encoding declaration of the XML file is updated appropriately.

DB2 XML Extender limit changes
The limit for the number of tables allowed in a RDB_node mapping DAD is increased to 30, and the number of columns allowed per table has been increased to 500. In addition, the number of times each table/column can be specified in the `join` predicates of the condition statement is now unlimited.

Change in XML_USAGE primary key
The table_schema column has been added to the primary key of the DB2XML.XML_USAGE table.

XMLVarchar UDT size limit removed
You can now define the size of the XMLVARCHAR user-defined type prior to enabling the server. Previously, the XML Extender failed when the server was enabled and a previously defined XMLVARCHAR existed.

To change the size of the XMLVARCHAR UDT of an already enabled server:

1. Back up all data in the XML Extender-enabled server.
2. Disable all XML collection tables and XML column side tables.
3. Disable the server with the `dxxadm disable_server` command. See *DB2 Universal Database for OS/390 and z/OS(R) XML Extender Administration and Programming, Version 7* for the syntax of the `dxxadm disable_server` command.
4. Create the XMLVARCHAR user-defined type.
5. Enable the server with the `dxxadm enable_server` command. See *DB2 Universal Database for OS/390 and z/OS(R) XML Extender Administration and Programming, Version 7* for the syntax of the `dxxadm enable_server` command.
6. Re-enable all XML collection tables and XML column side tables.
7. Recreate and reload the tables.

Chapter 3. Message updates

The following messages have been updated or added to the XML Extender.

DXXQ046W (message number changed from 'DXXQ004W')   The DTD ID was not found in the DAD.

Explanation: In the DAD, VALIDATION is set to YES, but the DTDID element is not specified. No validation check is performed.

User Response: No action is required. If validation is needed, specify the DTDID element in the DAD file.

DXXQ047E   Parser error on line *linenumber* **column** *colnumber*: *msg*

Explanation: The parser could not parse the document because of the reported error.

User Response: Correct the error in the document, consulting the XML specifications if necessary.

DXXQ052E   An error occured while accessing the DB2XML.XML_USAGE table.

Explanation: Either the database has not been enabled or the table DB2XML.XML_USAGE has been forcefully dropped.

User Response: Ensure that the database has been enabled and the table DB2XML.XML_USAGE is accessible.

DXXQ053E   An SQL statement failed : *msg*

Explanation: An SQL statement generated during XML Extender processing failed to execute.

User Response: Examine the trace for more details. If you cannot correct the error condition, contact your Software Service Provider. When reporting the error, be sure to include all the messages, the trace file and how to reproduce the error.

DXXQ054E   Invalid input parameter: *param*

Explanation: The specified input parameter to a stored procedure or UDF is invalid.

User Response: Check the signature of the relevant stored procedure or UDF, and ensure the actual input parameter is correct.

DXXQ055E   ICU error: *uerror*

Explanation: ICU error encountered during conversion operation.

User Response: Report the error to your software service provider. Include trace file, error message, and instructions to reproduce the error.

Trademarks

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

```
AIX
DB2
DB2 Universal Database
IBM
MQSeries
OS/400
OS/390
```

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

Java, JDK, and JDBC are registered trademarks of Sun Microsystems, Inc..

UNIX is a registered trademark of X/Open Company Limited.

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

IBM ®

Printed in U.S.A.