IBM® DB2® XML Extender

Release Notes

Version 7, Fixpak 9

IBM® DB2® XML Extender

Release Notes

Version 7, Fixpak 9

Contents

About the Release Notes for IBM DB2 XML Extender Version 7.2 FixPak 9

This document contains information about FixPak 9 for DB2 XML Extender Version 7.2. This supplements the information in *DB2 Universal Database® XML Extender Administration and Programming, Version 7*, released for DB2 Universal Database Version 7.2. The information contained in these release notes includes all topics except installation. For up-to-date information about installing DB2 XML Extender, see the DB2 UDB Version 7.2 Installation Notes in the readme.txt file, in the root directory of the product CD-ROM.

For users of DB2 XML Extender for iSeries, this document also supplements information found in *DB2 Universal Database Extenders for iSeries: XML Extender Administration and Programming, Version 7*.

Chapter 1. What's new in XML Extender with Fixpak 9?

The following fixes and updates are provided in Fixpak 9. Links are provided for changes that have more detailed information or documentation updates.

All updates in Fixpak 9 are now available on DB2 for iSeries: XML Extender in `PTF 5722DE1 V5R1M0 SI08323`.

- "Migrating to Fixpak 9 from prior releases" on page 7
- APAR fixes
  - APAR #JR17509: Fix for dxxExtractTimestamp problem. XML Extender will normalize timestamps automatically conforming to DB2 if needed.
  - APAR #IY38725 and #IY38895: Fixed entity replace truncation problem when generating an XML document. If a CLOB column or a VARCHAR column contains many entities that are to be expanded , the data will not be truncated.
  - APAR #IY38140: extractVARCHAR will return the correct result if the resulting string length is less than or equal to 3000.
  - APAR # IY27403: Fixed data truncation caused by entity replace during side table update.
  - APAR #JR17555: extractUDFs will now return correct data even when the XML document has a CDATA section.
  - APAR #IY35935: Fixed the memory leak caused by DB2 XML Extender's UDFs. DB2 will no longer eventually run out of space while doing bulk inserts.
  - APAR #IY37856: During decomposition of shift-jis document into utf8 database, not all data are inserted into tables due to incorrect string comparison between strings of different code pages.
  - APAR #IY37755: SQL error message will be displayed when DXXDISABLEDB fails.
- Documentation updates
  - Incorrect example for creating a new procedure with an increased CLOB limit
  - XMLFILE data type incorrectly defined
  - Optional table space parameter for enabling databases
  - XML Extender usage restrictions in a DB2 Enterprise-Extended Edition
- Miscellaneous fixes
  - Automatic normalization of timestamps while using extractTimestamp() or extractTimestamps().
  - Handling of entity characters by the extractChar, extractVarchar, extractClob, and Update UDFs

Chapter 2. Installation Requirements

The following items are required to install this fix pack:

- DB2 UDB version 7.2 or higher
- JDK version 1.1.8 or higher

Operating system requirements:

- AIX version 4.2.1 or higher
- Solaris Operating Environment version 2.6 or higher
- Microsoft Windows NT 4.0 SP4 or higher, Microsoft Windows 2000
- Linux (Intel only), Linux kernel 2.2.12 or higher, glibc 2.1.2 or higher, libstdc++ 2.9.0

The Web Services servlet requires JDK 1.2.2+ and JDBC 2.0+.

Chapter 3. Installation notes

If you have older versions of DB2 XML Extender installed, uninstall the previous version before installing DB2 XML Extender Version 7.2.

Before beginning installation, check that DB2 is not running by entering `db2stop force` on the DB2 command line.

After installing DB2 XML Extender on the Solaris Operating Environment, it is necessary to run the following script:

`/opt/IBMdb2xml/V7.1/bin/xmlsetup`

You must have root authority to run this script.

After installing DB2 XML Extender on a Linux environment, it is necessary to run the following script:

`/usr/IBMdb2xml/v7.1/bin/xmlsetup`

You must have root authority to run this script.

Chapter 4. Updates for the *DB2 Universal Database XML Extender Administration and Programming, Version 7*

This section provides updates to information in the *DB2 Universal Database XML Extender Administration and Programming, Version 7,* and follow on release notes for FixPaks 3, 4, 5, 6, and 7, and 8 for UNIX and Windows.

The information in these release notes is divided into several sections::
- "Migrating to Fixpak 9 from prior releases"
- "Documentation updates"
- "Miscellaneous fixes" on page 8

Migrating to Fixpak 9 from prior releases

If you are using XML Extender Version 7.2 FixPak 7 or earlier (on UNIX or Windows operating systems), you must migrate each database enabled for XML Extender before using an existing XML-enabled database with XML Extender Version 7.2 FixPak 9. The migration program migrates you to DB2 XML Extender Version 7.2 Fixpak 9 from prior releases. There is no need to migrate from FixPak 8 to FixPak 9.

To migrate the database:

The migration files are located in the fix pack CD install image. You must complete these steps from the directory where you extracted the fix pack files.

When using UNIX or Windows, complete the following steps.
1. From the DB2 command line, enter:
   ```
   db2 connect to database_name
   db2 bind @dxxMigv.lst
   ```
2. From the DB2 command line, enter:
   ```
   dxxMigv database_name
   ```

Failing to do the migration step can cause problems and unpredicable results such as failure while disabling databases and inability to access new UDFs.

Documentation updates

The following section provides updates to the XML Extender documentation.

Incorrect example for creating a new procedure with an increased CLOB limit
The example shown in step two of ″Increasing the CLOB limit″ in DB2 Universal Database XML Extender Administration and Programming, Version 7.2 is incorrect. Refer to the following correct example:
```
db2   "create procedure DB2XML.dxxShredXML(in    dadBuf     clob(100K),
                                           in    XMLObj     clob(10M),
                                           out   returnCode integer,
                                           out   returnMsg  varchar(1024)
                                           )
             external name 'DB2XML!dxxShredXML'
             language C
```

```
parameter style DB2DARI
not deterministic
fenced
null call"
```

XMLFILE data type incorrectly defined

In chapter two of *DB2 Universal Database XML Extender Administration and Programming, Version 7.2*, the XMLFILE user-defined type is incorrectly defined with VARCHAR(1024) as the source data type. The correct source data type is VARCHAR(512).

Optional table space parameter for enabling databases

Users of DB2 XML Extender for Windows or Unix-based environments can use the optional -t tablespacename parameter to specify a table space while enabling a database. For example:

```
dxxadm enable_db mydb -t tbsp1,tbsp2
```

There is no space between the two table space declarations.

- When enabling a database, use the table space parameter to specify the name of an existing table space to hold db2xml.XML_USAGE and db2xml.DTD_REF tables. A second tablespace can be optionally specified.

  If no tablespace option is provided while enabling the database, DB2 XML Extender will check to see if the DXXDTDRF and DXXXMLUS table spaces exist. The db2xml.dtd_ref table will be created in the DXXDTDRF table space if that table space exists. The db2xml.xml_usage table will be created in the DXXXMLUS table space if it exists as well. If one of the DXXDTDRF or DXXXMLUS table spaces does not exist, then the respective table (db2xml.dtd_ref or db2xml.xml_usage) will be created in the most appropriate table space as determined by DB2.

  If only one table space is provided while enabling the database, both tables will be created in the specified table space. If two table spaces are provided while enabling the database, the db2xml.dtd_ref table will be created in the first table space listed, and the db2xml.xml_usage table will be created in the second table space listed.

XML Extender usage restrictions in a DB2 Enterprise-Extended Edition

Users should consider that on DB2 Enterprise-Extended Edition, all files referenced by a UDF or within an XML document must be accessible from each participating node using the same path and file name via a mounted/shared file system.

On Windows, you can also use the Universal Naming Convention (UNC) of naming files among computers on a local area network, so that a file has the same path name when accessed from any of the computers on the network. For example:

```
\\server\sharedrivename\directoryname\filename.xml
```

The user id specified by the DB2 instance must have access to the UNC file.

Miscellaneous fixes

The following section describes fixes to the XML Extender for Fixpak 9.

Automatic normalization of timestamps while using extractTimestamp() or extractTimestamps()

XML Extender will automatically normalize timestamps extracted from

XML documents to comply with the DB2 timestamp format if needed. Timestamps are normalized to the `yyyy-mm-dd-hh.mm.ss.nnnnnn` format or the `yyyy-mm-dd-hh mm.ss.nnnnnn` format. See Table 1 below for examples of timestamp normalization.

Table 1. Timestamp Normalization

Pre-normalized timestamp	Normalized timestamp
1998-1-11-11.12.13	1998-01-11-11.12.13.000000
2003-2-12-10 23.12	2003-02-12-10 23.12.000000

Handling of entity characters by the extractChar, extractVarchar, extractClob, and Update UDFs

This section describes what is returned by the extractChar, extractVarchar, extractClob, and Update UDFs when entity characters and CDATA sections are part of the XML document. If there are entity characters in the xml document outside of a CDATA section, the extractVarchar and extractChar UDFs return those entity characters unescaped, and the data in the CDATA section will be returned as-is. The extractCLOB and Update UDFs return markup, and all entity characters are returned escaped whether or not they are within CDATA sections.

For example, consider the following XML document:

```
<?xml version="1.0" encoding="UTF-8"?>
<a>
<b>3</b>
<c>out:&amp; <![CDATA[in: &amp;]]></c>
</a>
```

- The expected output for:

  ```
  extractVarchar(,/a/c)
  ```

  would be:

  ```
  out:& in: &amp;
  ```

- The expected output for:

  ```
  extractClob(..., '/a/c')
  ```

  would be:

  ```
  <c>out:&amp; in: &amp;amp; </c>
  ```

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.