DB2 9.5 pureXML Support

A In Depth Look at DB2 9.5 pureXML
February 2009

ORACLE

DB2 9.5 pureXML Support

DB2 9.5 PUREXML SUPPORT

XML is a universal exchange format for both data and documents of all shapes and sizes. XML applications require flexible and comprehensive capabilities to serve a range of uses cases with different characteristics. Therefore, database-native XML support must provide integrated XML data management capabilities including XML datatype, storage, indexing, querying, and schema processing.

"Nonrelational uses of these DBMSs — including, especially, XML document management — that will also help drive growth, particularly once RDBMS vendors demonstrate ways in which XML content can be linked and managed together with relational data and used for integrated reporting", IDC Worldwide RDBMS 2005

A recent IDC report forecasted that native XML support will help drive the growth of enterprise database management systems:

> "Nonrelational uses of these DBMSs — including, especially, XML document management — that will also help drive growth, particularly once RDBMS vendors demonstrate ways in which XML content can be linked and managed together with relational data and used for integrated reporting", IDC Worldwide RDBMS 2005 Vendor Shares

Four years ahead of other RDBMS vendors, Oracle XML DB was introduced in Oracle 9i Release 2 to meet the growing demand of storing, retrieving, and managing the multiplying volumes of XML data alongside relational data. Now at its third release, Oracle XML DB addresses the needs of a broad spectrum of XML use cases by providing multiple native storage options and indexing schemes, scalable XQuery and SQL/XML operations, SQL and XML duality, standards-based implementation, schema evolution, integrated support of Internet protocols, high-performance repository, and full-text search integration on a reliable, available, and scalable Oracle Database platform. As a result, Oracle XML DB has been widely deployed by enterprises to run mission-critical applications.

Oracle XML DB has been widely deployed by enterprises to run mission-critical applications.

The native XML support in DB2 9 pureXML[1] is IBM's attempt to catch up with Oracle XML DB. This paper examines DB2 9 pureXML support in the following five key areas:

- Native Support for the XML data model and other XML standards
- Native XQuery Support
- SQL, XML, and Text Integration
- Feature Completeness and Ease of Programmatic Access
- XML Content Management Capabilities

Incremental enhancements in XML processing capabilities in the recent DB2 9.5 release and an IBM proposed TPoX benchmark are also analyzed..

---

[1] As of Febuary 10, 2009, pureXML has become a core part of DB2 without extra charge.

NATIVE SUPPORT FOR THE XML DATA MODEL AND OTHER XML STANDARDS

The foundation of a native XML support is the comprehensive support of XML data model. A native XML datatype with associated storage, indexing, and schema processing capabilities are essential components of this foundation.

In previous DB2 releases, the XML Extender product provided minimum XML capabilities (e.g., XML shredding) without a native XML datatype. Introduced in the DB2 9.1 release, a native XML datatype is finally introduced, but only a single storage option and one indexing scheme are available for use with the new XML datatype. Further, although XML schema integration is critical to XML data model support in the database, DB2 9 pureXML only provides limited XML schema processing capability. This approach shifts the burden of schema evolution to application developers and DBAs throughout the life cycle of XML data and their dependent applications. The "compatible XML schema evolution" capability introduced in the recent DB2 9.5 release only allows schema changes that remain compatible with the original one.

Although IBM claims its only XML storage option as the "true native" XML storage and touted it as a major differentiator, the truth is that the "true native" label is only relevant in the context of comparing with its predecessor non-native XML extender product.

What We Know

DB2 9.5 pureXML release supports a new XML datatype with a single storage option and only one XML node value indexing scheme (i.e., the XMLPattern index). XML schema processing is limited to user-invokable schema validation of stored XML data against XML schema stored in the XML schema repository. The list below highlights these weaknesses.

Rigid XML storage format

With the XML datatype tied to only one storage format, different XML use cases will amplify the limitation of this approach. For example, this storage format does not support XML document-fidelity.

Poor performance for path-based queries

Similarly, as the only indexing scheme, the XMLPattern index can only address limited use cases. For example, due to its lack of XML path index, retrieving XML fragments from large XML documents will lead to poor query performance. XMLPattern index is also inadequate in use cases where query predicates are not known a priori.

Poorly integrated XML schema processing

Without integrated XML schema-aware processing capabilities, the XMLPattern index requires end users to specify datatype information explicitly. Matching datatypes in user queries with what are specified in the indexes will be a constant chore for both application developers and DBAs.

Limited schema evolution capability

Although it is inevitable for an XML application to evolve its schema throughout its life cycle, DB2 9.5 pureXML only supports a "compatible" XML schema evolution which only allows schema changes that remain compatible with the original one. This approach shifts the burden of "incompatible" schema evolution completely to application developers and DBAs.

Oracle's Proven Leadership and Value

Oracle was a pioneer in native XML datatype support when it was first introduced in Oracle 9i Release 1. With the vision of storage and processing flexibility, Oracle XML DB offered an innovative approach of supporting native XMLType as an SQL-level abstraction to leave room for additional storage and indexing options optimized for evolving use cases.

Since the introduction of XML schema-aware structured and unstructured storage options in Oracle 9i Release 2, Oracle XML DB has enjoyed broad adoption by real world mission critical applications. These applications have been deployed to global enterprises with use cases ranging from data-centric to document-centric. With insights gained from a rapidly growing customer base, Oracle XML DB continues to enrich its capabilities. The new binary XML storage option and the XMLIndex indexing scheme in the recent Oracle Database 11g release have further extended our lead over DB2 9 pureXML.

NATIVE XQUERY SUPPORT

XQuery is a W3C standard for querying XML documents currently in its 1.0 release. JSR 225 and ISO SQL/XML 2006 are respective standardization effort for XQuery support in Java and SQL. A key promise of XQuery is its ability to integrate and query diverse data sources (structured, unstructured, semi-structured, and relational) as XML data sources. In addition, XML transformation and publishing are also popular XQuery use cases.

While DB2 9 pureXML does provide XQuery support for XML type data, it doesn't support static typing features. It is optimized only for XML data stored persistently in XML type columns. Querying a view with XML type columns mapped to relational tables will perform poorly. As a result, DB2 9 pureXML's XQuery support is not suitable for XML publishing use cases.

Finally, DB2 9.5 pureXML also claims support for XQuery Update. Since the XQuery Update Facility is still undergoing the W3C standardization process,

DB2 9.5 only supports an early working draft. With many changes since the early working draft, the current version of XQuery Update Facility departs from DB2 9.5 implementation.

What We Know

DB2 9 pureXML's XQuery implementation doesn't support static typing features. Its XQuery support is only optimized for XML data stored persistently in XML type columns. Querying a view with XML type columns mapped to a relational table performs poorly. In short, its XQuery support is not well integrated with its relational capabilities.

Poor XQuery rewrite capability

XMLQuery() function can used in SQL 'select' clause for XML-centric queries. Since XQuery expression in an XMLQuery() function can have its own 'where' clause, a native XML implementation should be able to take advantage of underlying indexes for optimal query performance. However, DB2 9.5 pureXML fails to support this critical capability.

Poor integration with relational engine

Native XML implementation in a relational database needs to integrate seamlessly together to handle XML, relational, and mixed use cases with optimal performance. DB2 9.5 pureXML cannot interoperate effectively with its relational counterpart.

XML publishing/repurposing of relational data

With SOA becoming prevalent, a common use case for native XML database is to publish and repurpose relational data into different XML structures in response to web service requests. A simple way to satisfy this use case is to first build an XML type view over relational data, and then XQuery can be used to further tailor the XML structure. Therefore, a native XQuery implementation should be able to handle such use case with ease by seamless integration of its relational engine and native XML engine.

Querying relational views over XML type columns

In a reverse direction from the use case above, once XML documents are persistently stored in a native XML storage, additional processing steps (e.g., business intelligence) will follow on the relational side of the database. A simple, effective, and flexible approach to handle this use case is to build relational views over persistently stored XML documents. Again, a seamless integration of XML and relational engines should produce optimal query plan when such relational views are queried.

Oracle's Proven Leadership and Value

Oracle was the first major database vendor to introduce database-native XQuery support. With deep integration and innovative XQuery rewrite technology, Oracle XQuery offers optimized query execution against multiple data sources (e.g., persistently stored XML data, relational data, XML DB repository), and excels at XML transformation and publishing use cases.

SQL, XML, AND TEXT INTEGRATION

Integrated full-text query is a common use case in XML applications. To elaborate further,

- Full-text support for XML documents is another area exposing the poor integration of DB2 9.5 pureXML with other DB2 components. Full-text search of XML documents is only possible with a separate DB2 Net Search Extender product. This separate extender in only accessible through an SQL interface but not the XQuery interface.
- With its inferior XQuery implementation, XML applications will have a hard time to repurpose XML data.

What We Know

Poor integration with full-text search

Full-text search of XML documents is poorly integrated in DB2 9.5 pureXML

Oracle's Proven Leadership and Value

Oracle XML DB provides deep integration with other Oracle Database components, namely,

- Oracle XQuery supports full-text search on text nodes via an integrated ora:contains() function.

FEATURE COMPLETENESS AND EASE OF PROGRAMMATIC ACCESS

The new generation of XML applications accesses data using standard Internet protocols (e.g., HTTP, FTP) as well as programming language-specific APIs (e.g., JDBC). DB2 9.5 pureXML release supports language-specific APIs but not Internet protocols. The language-specific APIs for XML type is also limited to a serialized format instead of a high performance parsed XML format.

What We Know

Lack of Support for Internet Protocols

DB2 9.5 pureXML only provides language-specific APIs but not Internet protocols. The language-specific APIs for XML type is limited to a simple serialized format.

Poor Integration with the Database

DB2 9.5 pureXML and Database Partitioning are mutually exclusive in a database instance. This is another proof of DB2 9.5 pureXML's poor integration of its native XML feature with other essential components of the database.

Oracle's Proven Leadership and Value

Oracle XML DB supports high performance Internet protocols (e.g., HTTP/HTTPS, SOAP, WebDAV, FTP) and APIs. The API support for XMLType uses a high performance parsed XML format.

XML CONTENT MANAGEMENT CAPABILITIES

Document-centric XML applications have different requirements from data-centric applications. The primary difference is that XML documents are often viewed as resources (files and directories) in a content repository as opposed to rows in relational tables. A content repository should support access control, foldering, versioning, and rich APIs.

What We Know

DB2 9.5 pureXML does not provide any content repository capabilities.

Oracle Proven Leadership and Value

To target document-centric use cases, Oracle XML DB has provided repository support since its first release in Oracle9i Release 2. With access control, versioning, and deep integration with SQL and XQuery APIs, Oracle XML DB repository has been widely used by XML content management applications.

DB2 9.5 PUREXML ENHANCEMENTS

Support for XML in non-Unicode database

There was a serious flaw in the first DB2 9 pureXML release which only supported databases using the UTF-8 codepage. The DB2 9.5 release patched this flaw to support additional codepages.

Analysis: In contrast, Oracle XML DB has always supported all database character sets since its initial release.

Sub-document update

In DB2 9 pureXML, any updates of an XML document can only be done by replacing the entire XML documents. This is particularly inefficient for updating a single node value of large XML documents. DB2 V9.5 now allows a user to update parts of an XML document by using an XQuery transform expression, which is based on an early draft of the XQuery Update facility.

Analysis: Oracle XML DB has implemented piece-wise updates for XML documents stored with structured storage option since its introduction in Oracle 9i Release 2. In Oracle Database 11g, binary XML storage option using SecureFiles also supports highly efficient piece-wise updates. Since XQuery Update standardization effort is still underway, instead of premature adoption of a frequently changing early draft specification as DB2 9.5 did, partial updates of XML documents stored in Oracle XML DB can be accomplished with XPath-based SQL functions.

Base-table row storage/compression

DB2 9.5 pureXML added limited support for compression of XML documents if they are less than 32KB in size.

Analysis: The 32KB size limit for possible compression in DB2 9.5 pureXML makes it impractical for most uses cases where an XML document can easily go over the small size limit. In contrast, the binary XML storage format introduced in Oracle Database 11g can take advantage of the SecureFile compression for XML documents of any size to achieve high compression ratio.

Compatible XML schema evolution

Without any schema evolution capability in DB2 9 pureXML, a schema cannot be changed at all once its registered. In DB2 9.5 pureXML, a compatible XML schema evolution feature is introduced.

Analysis: Schema evolution is a critical capability for applications that are subject to may change throughout their life cycle. Oracle XML DVB introduced copyEvolve support for all possible schema evolution scenarios in Oracle Database 10g Release 1. In the recent Oracle Database 11g release, in-place schema evolution is also introduced for schema changes that don't invalidate dependent XML documents. In-place schema evolution in Oracle XML DB is similar to the approach used in DB2 9.5.

Load support

Loading large numbers of XML documents is a common for XML use cases. Missing in DB2 9 pureXML was the capability to in the Load utility for bulk loading of XML documents. In DB2 9.5 pureXML, the Load utility now provides such supports.

Analysis: Oracle XML DB supports a number of ways to load XML documents since its introduction in Oracle 9i R2. Among them, direct path load in SQL*Loader offers the highest performance for bulk loading a large number of XML documents. With its unique support of Internet protocols, customers have also been using FTP or WebDAV to load XML documents into Oracle XML DB.

XML index enhancements

DB2 9.5 pureXML introduces minor enhancement to its XML indexing with allow data type validation.

Analysis. Critical to the scalability of an XML database are efficient indexing schemes along with a query processor and a cost-based optimizer that can take advantage of existing indexes. DB2 pureXML only supports one XML indexing scheme for its lone XML storage format. This XML indexing scheme is similar to a function-based index in an Oracle database. A function-based XML index serves limited use cases where query predicates are known a priori. In comparison, Oracle XML DB provides multiple indexing schemes for its use-case optimized storage options.

New publishing functions to map relational data to XML data

DB2 9.5 adds the XMLGROUP scalar function to return a single top-level element to represent a table, or the result of a query, and the XMLROW scalar function returns a sequence of row elements to represent a table, or the result of a query.

Analysis: Both functions are simply unnecessary wrappers around standard SQL/XML functions. Oracle has been supporting the complete SQL/XML publishing standard since 9.2 and has the best performing, standards based, solution for generating XML content from relational data. Unlike IBM, Oracle is not trying to introduce proprietary extensions to the standard that lock unsuspecting customers into non-standard implementations. The only time Oracle has introduced non-standard extensions or operators is to meet customer requirements that CANNOT be satisfied within the existing SQL/XML framework. Both of these operators were discussed and rejected by the SQL standards committee. Basically XMLROW is nothing more than select XMLElement("row",xmlForest(A, B, C, ...)) from TABLE and XMLGROUP is nothing more than select XMLElement("Table", xmlagg(XMLElement("row",xmlForest(A, B, C, ...)))) from TABLE.

SQL function transforms XML documents using a XML style sheet

The XSLTRANSFORM function can be used to convert XML documents into HTML, plain text or between different XML schemas. XSLTRANSFORM uses style sheets to convert XML into other data formats. Users can convert part or all of an XML document and select, or rearrange, the data using the XPath query language and XSLT (Extensible Style Sheet Language Transformation).

Analysis: XSL transformation has been part of Oracle XML DB since 9iR2. Unlike IBM who is just introducing this functionality, Oracle has had production customers using database based XSLT for over 6 years now.

New check constraints and triggers on XML

New constraint definition clause options can be used to specify and ensure the validity of XML column data before it is processed. The new clauses of the CREATE TABLE and ALTER TABLE statements that can be specified for XML columns are IS VALIDATED and IS NOT VALIDATED. Before triggers can now be used to automatically validate XML documents against registered XML schemas.

Analysis: Oracle XML DB has supported check constraints and triggers on XML since the first release of Oracle XML DB in 9iR2.

THE TPOX BENCHMARK

Although numerous benchmarks for measuring XML processing capabilities have been published in the past, IBM introduced the TPoX benchmark as an data-centric "application-level" benchmark based on a financial information exchange standard – FIXML. IBM ran the benchmark against DB2 9 pureXML to tout its XML processing capabilities.

Analysis: As we have explained earlier this paper, XML has diverse use cases that can be categorized as either data-centric or content-centric. Data-centric applications further span a broad spectrum that can be characterized by their typical workloads in a production environment throughout their life cycles. As a narrowly scoped benchmark specification proposed by a single vendor, TPoX hasn't been able to raise any interests among major XML database vendors in the industry. Further, the TPoX benchmark attempted to follow the approach of TPC database benchmarks, even though they have been known to deviate substantially from production system workloads[2]. In short, it is futile for IBM to conduct a costly benchmark effort of a narrowly scoped specification that doesn't reflect production system workloads.

[2] Characteristics of production database workloads and the TPC benchmarks

Table 1 : Oracle XML DB Vs DB2 9.5 pureXML

Feature	Oracle XML DB	DB2 9.5
XML Processing in the Database Server		
XQuery Support	Yes	Yes
Streaming XML schema validation	Yes	No
Streaming XPath processing	Yes	No
High performance XML/SQL duality	Yes	No
Full-text search	Yes	Limited
XML Query optimization	Yes	Limited
Database-native Web Services	Yes	No
Included as standard feature of the Database	Yes	No
Sold as separately charged option	No	Yes
XML Storage and Indexing		
XML data type	Yes	Yes
Separation of Logical Model from Physical Representation	Yes	No
Multiple storage options to match use cases	Yes	No
A storage format for maintaining Document Fidelity	Yes	No
XML Schema based optimizations	Yes	No
Support for Schema Extension, Versioning and Evolution	Yes	Limited
Multiple indexing schemes	Yes	No
XML data and index compression	Yes	Limited
API's, Tools & Utilities		
Complete support by data management utilities and tools	Yes	Limited
Development APIs and tools support	Yes	Yes
File System Abstraction		
File / Folder organization of content	Yes	No
Internet protocols (FTP, HTTP, WebDAV)	Yes	None
Versioning	Yes	No
Access Control Lists	Yes	No
XQuery and SQL support for file metaphor	Yes	No
Compound document support (XLink and XInlcude)	Yes	No
Event notification	Yes	No

CONCLUSION

After examining DB2 9.5 pureXML support in five areas critical to XML applications, major holes have been uncovered in all five areas. In a sharp contrast, Oracle XML DB excels in all five areas due to the following strengths:

- First to market with wide spread adoption around the world satisfying a full range of XML application uses cases
- A flexible architecture to adapt quickly with customers' evolving requirements
- A superior implementation with deep integration to offer simplified development and high performance on a reliable, available, and scalable Oracle database platform

Oracle XML DB continues to be years ahead of DB2 9.5 pureXML in native XML support. The recent Oracle Database 11g release has further bolstered the leadership position of Oracle XML DB in native XML support among RDBMS vendors.

In summary, Oracle XML DB continues to be years ahead of DB2 9.5 pureXML in native XML support. The recent Oracle Database 11g release has further bolstered the leadership position of Oracle XML DB in native XML support among RDBMS vendors.

DB2 9.5 pureXML Support
February 2009

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com