ALTOVA®

WhitePaper

Ready for IBM® DB2® data server software

Integration of Altova Tools with IBM® DB2® 9 pureXML™ demonstrated via the Online Brokerage Application Scenario

Altova, Inc. · 900 Cummings Center, Suite 314-T · Beverly, MA, 01915-6181, USA · Tel: 978-816-1600 · Fax: 978-816-1606 · Web: www.altova.com

Integration of Altova Tools with IBM® DB2® 9 pureXML™ demonstrated via the Online Brokerage Application Scenario

February 27, 2008

Irina Kogan ikogan@ca.ibm.com
Erin Cavanaugh erin.cavanaugh@altova.com
Liz Andrews liz.andrews@altova.com

Abstract

This paper demonstrates how the deep integration of the graphical Altova tools with DB2 pureXML software can help XML developers, DBAs and end users solve real-life business problems easily and quickly. The examples from the financial industry (the online brokerage scenario) are chosen to explain and to highlight the benefits of this integration.

1 Introduction

The IBM DB2 data server offers a scalable and reliable information infrastructure that enables users to manage a wide variety of structured and unstructured data in a unified engine that provides high scalability, high availability, effortless maintainability, seamless integration of XML and relational services, and more. Now with the deep integration of industry-leading Altova® tools and DB2 9 software, users can query, exchange, transform, and edit their XML artifacts more quickly, easily, and accurately.

This integration offers a rich and powerful toolset for:

- Designing and perfecting XQuery statements faster, with real-time validation for a faster and more robust design time experience
- Previewing, editing, and validating XML data stored in the database with just a few clicks
- Easily managing XML Schemas in the DB2 9 Schema Repository
- Mapping XML data from the database to a CSV file or converting it to an HTML table or other format
- And much more

This important new functionality is demonstrated using the simulated online brokerage scenario from the “Transaction Processing over XML” (TPoX) XML database benchmark [8][10]. The solutions to the business problems presented below show how DBAs and developers working with real-world XML applications can benefit from the integration of Altova tools with IBM DB2 9.

2 Background

This section provides background information on DB2 9 pureXML, on the Altova tools that feature DB2 9 integration, and on the online brokerage scenario used in all examples within this paper to demonstrate the new features.

2.1 DB2 9 pureXML

The DB2 9 release [1][2] provides comprehensive XML support, including pureXML storage, XML indexes, XQuery, SQL/XML, and sophisticated XML Schema handling. "Pure" refers to XML documents that are stored and processed as type-annotated trees. This technique differs from other commercial relational databases [6], which perform translation and/or assembly steps, for example:

- Storing XML documents as large objects (BLOBS or CLOBs)
- Dividing XML documents into a set of relational tables [7]

Database applications involving XML are very diverse and include mechanisms for publishing relational data as XML, content and document management with XML full-text search, consolidation of diverse data sources, managing highly variable data, forms processing, and backend support for Web services and SOA, as well as message-based transaction processing and XML-based OLTP, especially in the financial industry.

2.2 Altova Tools: XMLSpy, MapForce, StyleVision, DatabaseSpy

This section lists the Altova products that include deep integration with IBM DB2 9 pureXML. The details of this integration are discussed in Section 3.

XMLSpy – the industry-leading XML editor and IDE for working with all XML-related technologies, including XML instance documents, XML Schema, XQuery, XSLT, XPath, and more

MapForce – a data mapping and conversion tool that builds and maintains relationships between XML, databases, flat files, EDI, and Web services, making it easy to orchestrate artifacts for inclusion in a service-oriented architecture (SOA), Web service, or custom data integration application

StyleVision – a visual stylesheet creation tool for delivery of XML and relational data through HTML, PDF, and Word/RTF, as well as electronic forms

DatabaseSpy – an end-to-end development tool for database query and design

2.3 TPoX Benchmark: Simulated Online Brokerage Scenario

"Transaction Processing over XML" (TPoX) is an open source, application-oriented and domain-specific benchmark that simulates a financial multi-user workload with XML data conforming to the FIXML standard [11]. It is used to evaluate the performance of XML database systems, focusing on XQuery, SQL/XML, XML storage, XML indexing, XML Schema support, XML updates, logging, concurrency, and other database aspects.

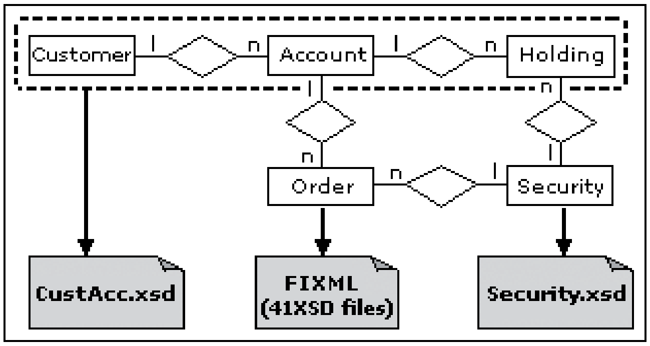
TPoX is publicly available at tpox.sourceforge.net. Here we describe a test scenario; much more information can be found on the benchmark's Web site [8].

The test scenario models an online brokerage. It was created based on a thorough analysis of real-world financial applications. This scenario is purposefully simplified, yet still representative in terms of documents, transactions, and XML Schemas.

The main logical data entities in the scenario are as follows [9]:

- *Customer:* a single customer can have one or more *accounts*
- *Account:* each account contains one or more *holdings*
- *Holding:* the number of shares of a *security*
- *Security:* identifier for a holding (for example, the name of a stock)
- *Order:* each order buys or sells exactly one *security* for exactly one *account*

This is shown on the Entity-Relationship diagram below. The XML schemas are also included in the diagram.



Document handling and size varies by type:

- For each customer there is a CustAcc document defined by the CustAcc.xsd schema that contains all customer information, account information, and holding information for that customer. The CustAcc document size ranges from 4KB to 20KB.
- Orders are represented using FIXML 4.4. FIXML (www.fixprotocol.org) is an industry standard XML Schema for trade-related messages such as buy/sell orders. Order document size ranges from 1KB to 2KB. Order documents generally have many attributes and a high ratio of nodes to data.
- Security documents (fixed at 20833) represent the majority of US-traded stocks and mutual funds, use actual security symbols and names, and are defined by the Security.xsd schema. Their size ranges from 3KB to 10KB.

The Toxgene data generator [12] was used to produce instance documents for all three schemas.

3 Integration of Altova Tools with DB2 9 pureXML

To enhance productivity when working with XML, XML Schema, XQuery, and other XML-related technologies supported by DB2 9, Altova has built specialized capabilities that deliver deep integration with the DB2 9 pureXML data server into its XML and data management products: XMLSpy, MapForce, StyleVision and DatabaseSpy. The extended capabilities that have been added to each of these products are discussed in the sections below. The detailed information on how to use these new features is provided in [5].

3.1 XMLSpy

Working with XML Schemas and other XML-based documents in a professional XML editor is substantially more productive than coding XML in a regular text editor. This is especially true for large industry standard XML documents/schemas.

The Altova XMLSpy XML editor and XML IDE include graphical and text-based views customized for working with different XML technologies. Features such as intelligent entry helpers, validation checking and troubleshooting, debuggers, and profilers help users develop XML applications with more speed and accuracy than ever before.

The XQuery language, which is still relatively new, is an important technology for working with XML stored in databases. The powerful XQuery editor, debugger, and profiler in XMLSpy support standard XQuery as well as the IBM DB2 XQuery extensions, which enables users to quickly adapt to working with XQuery to develop XML-based applications.

XMLSpy also includes robust database integration features for querying DB2 databases, generating XML Schemas from database schemas, inferring database schemas from XML Schemas, and more.

XMLSpy's deep integration with DB2 9 enables users to:

- Edit, debug, and profile XQuery statements with support for IBM’s DB2 XQuery extensions
- Visualize the database structure and easily query tables using SQL, SQL/XML, and XQuery statements
- Edit XML data seamlessly with optional validation and persistence services
- Manage XML Schemas stored in DB2 XML Schema Repository with a complete set of services available from a point-and-click interface
- Transform XML data for use in other applications

3.2 MapForce

A common requirement for DB2 9 users is mapping, converting, and integrating XML and relational data with other prevalent formats.

Altova MapForce provides a graphical interface for mapping between today's most common data formats: XML, relational database, flat file, EDI, and Web services.

DB2 9 integration in MapForce enables users to:

- Map XML data directly to and from DB2 9 databases by assigning an XML Schema to the data using an intelligent drag-and-drop interface
- Easily access, preview, and integrate database data
- Employ high-performance filtering of database sources within data mapping projects
- Graphically build Web services that utilize or write to DB2 9 databases

3.3 StyleVision

Presenting business users and customers with access to database data in useful, attractive formats is a challenge that is simplified using Altova StyleVision. StyleVision is a visual stylesheet and database report designer for rendering XML and relational data in HTML, RDF, Word/RTF, PDF, and electronic forms – all based on a single design. Simple drag-and-drop functionality lets users design attractive, effective output to multiple media using a straightforward, graphical interface.

DB2 9 users can use StyleVision to:

- Create XSLT and XSL:FO stylesheets to publish XML and/or relational data using drag-and-drop functionality and advanced design features
- Quickly produce multiple output documents in HTML, Word/RTF, and PDF for publishing and communicating data in the DB2 9 pureXML repository

3.4 DatabaseSpy

Database administrators and other IT professionals often need to explore and maintain multiple proprietary and third-party databases. With a unified interface for visualizing, exploring, and modifying DB2 and disparate databases, Altova DatabaseSpy eases SQL editing and simplifies database design.

DatabaseSpy enables users to:

- View database tables and associated XML Schemas in a hierarchical tree
- Manage XML Schemas registered in the database
- Organize frequently used queries into project files for quick retrieval
- Graphically view and modify database tables and their relationships

4 Business problems solved through the integration of Altova Tools with DB2 9

Following is a description of the common tasks of a database administrator/developer working in the online brokerage:

1. A new project requires that data be migrated from an existing CLOB format table to a newly created table with an XML column (and an XML index defined in it) to store customer information directly as XML.
2. It turns out that some outdated information about a customer is inserted into the table, so this customer's document needs to be selected, reviewed, and then modified and saved back to the database table.
3. The same customer's partial profile needs to be obtained in different formats: XML, CSV, and HTML, for integration in other office systems and publishing on the company's sales intranet.
4. A request comes in to insert a new order document for a transaction in which a customer sold some shares of the security he/she owned. Validation needs to be performed to ensure that the order document conforms to the FIXML schema.
5. The manager of the online brokerage firm sends a request to the DBA to produce a list of securities that have the current open price higher than or equal to the specified one.
6. The online brokerage firm issues a change in the security schema.

These tasks can now be easily performed using the Altova tools connected to DB2 9 pureXML. The solutions to each issue are detailed in the next section. The TPoX benchmark introduced in Section 2.3 is used to demonstrate these transactions.

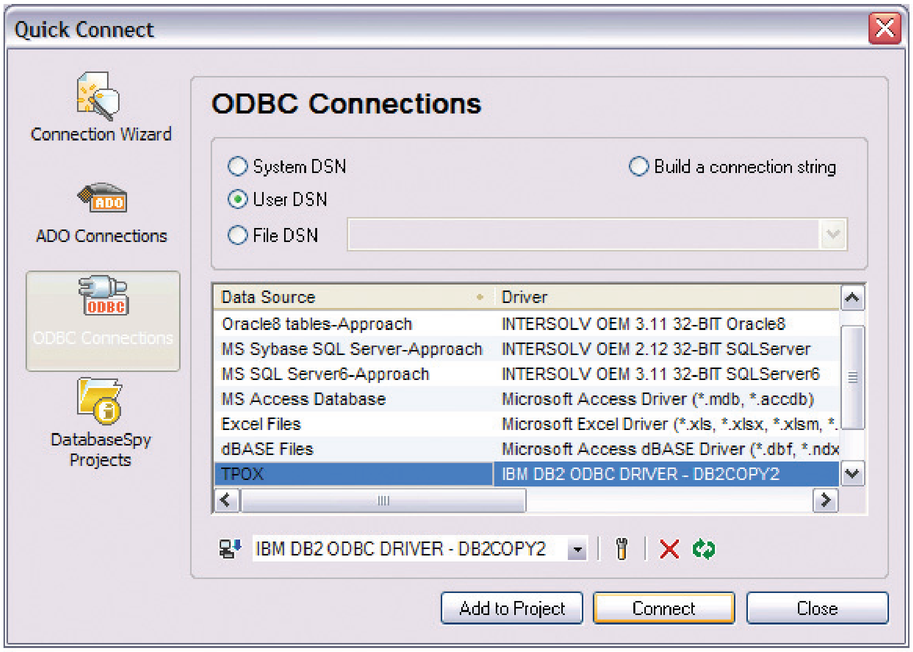
5 Solutions to the Identified Business Problems

The solutions to the problems discussed in the previous section are provided below. The screenshots of Altova tools (v2008) working with DB2 9 help illustrate the tools' capabilities in each situation (Alternatively, DB2 9.5 software could be used in all examples).

5.1 Migrating the customer data (CLOB to XML)

Problem Description:
A new project requires that data be migrated from an existing CLOB (Character Large Object) format table to a newly created table with an XML column (and an XML index defined in it) to store customer information directly as XML.

Solution:
First, the user needs to connect to the database using the graphical wizard in Altova DatabaseSpy (a similar wizard is available from the other tools as well):



Quick Connect

Connection Wizard
ADO Connections
ODBC Connections
DatabaseSpy Projects

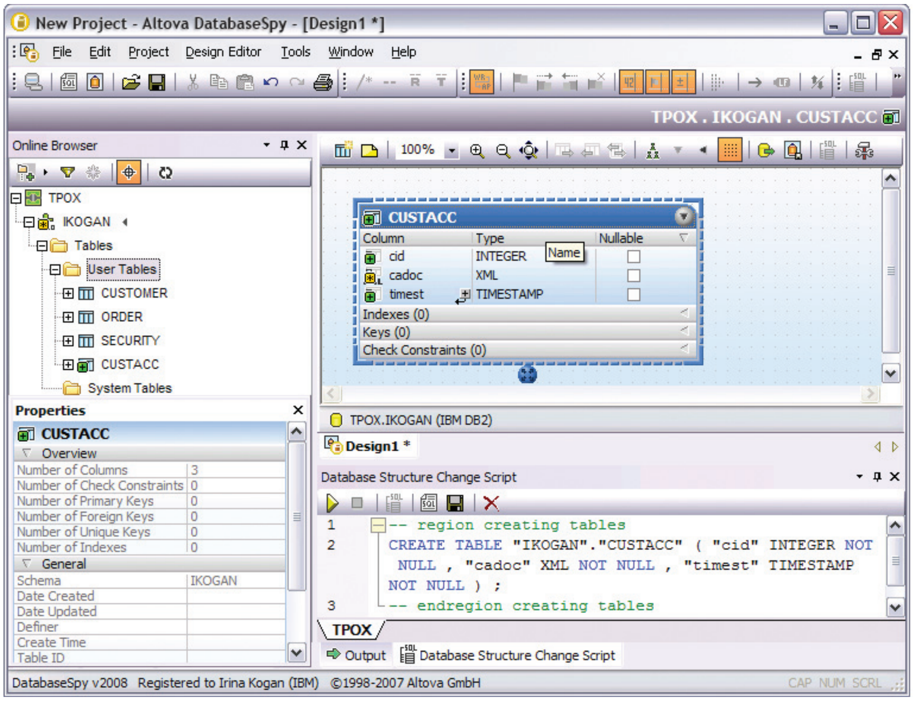
ODBC Connections

○ System DSN
● User DSN
○ File DSN
○ Build a connection string

Data Source	Driver
Oracle8 tables-Approach	INTERSOLV OEM 3.11 32-BIT Oracle8
MS Sybase SQL Server-Approach	INTERSOLV OEM 2.12 32-BIT SQLServer
MS SQL Server6-Approach	INTERSOLV OEM 3.11 32-BIT SQLServer6
MS Access Database	Microsoft Access Driver (*.mdb, *.accdb)
Excel Files	Microsoft Excel Driver (*.xls, *.xlsx, *.xlsm, *.
dBASE Files	Microsoft Access dBASE Driver (*.dbf, *.ndx
TPOX	IBM DB2 ODBC DRIVER - DB2COPY2

IBM DB2 ODBC DRIVER - DB2COPY2

Next, a CUSTACC table with an XML, an INTEGER ID, and a TIMESTAMP column should be created. One of the ways to do this is to use the Design Editor in DatabaseSpy that has been extended to provide support for XML columns.


New Project - Altova DatabaseSpy - [Design1 *]
File Edit Project Design Editor Tools Window Help
TPOX . IKOGAN . CUSTACC
Online Browser
TPOX
IKOGAN
Tables
User Tables
CUSTOMER
ORDER
SECURITY
CUSTACC
System Tables
100%
Name
TPOX.IKOGAN (IBM DB2)
Design1 *
Database Structure Change Script
TPOX
Output
Database Structure Change Script
DatabaseSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH
CAP NUM SCRL

CUSTACC

Column	Type	Nullable
cid	INTEGER	☐
cadoc	XML	☐
timest	TIMESTAMP	☐

Indexes (0)
Keys (0)
Check Constraints (0)

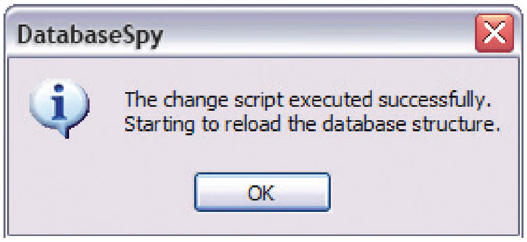
Properties

CUSTACC

Overview	
Number of Columns	3
Number of Check Constraints	0
Number of Primary Keys	0
Number of Foreign Keys	0
Number of Unique Keys	0
Number of Indexes	0
General	
Schema	IKOGAN
Date Created	
Date Updated	
Definer	
Create Time	
Table ID	

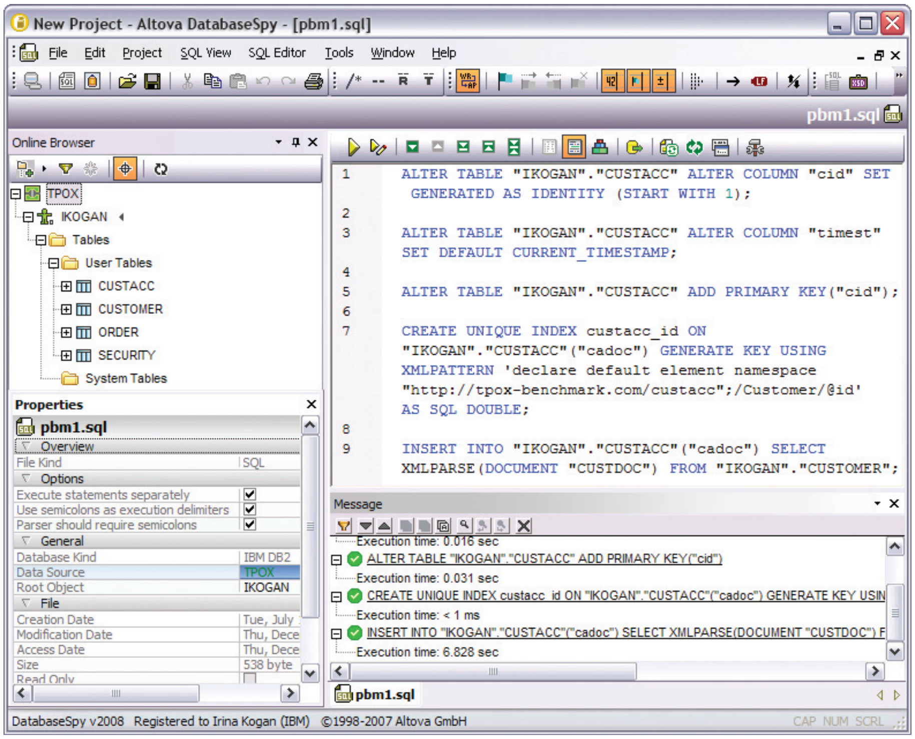
```
-- region creating tables
CREATE TABLE "IKOGAN"."CUSTACC" ( "cid" INTEGER NOT
 NULL , "cadoc" XML NOT NULL , "timest" TIMESTAMP
NOT NULL ) ;
-- endregion creating tables
```

This produces a script that can be run against DB2 9 (by clicking the yellow arrow). The execution produces the following confirmation message, and the database structure shown in the Online Browser on the left now lists the CUSTACC table.



The change script executed successfully.
Starting to reload the database structure.

The rest of the commands can be saved in an SQL script (named pbm1.sql for this example) that can be executed from the SQL Editor, which enables the user to run all semicolon-separated commands in sequence.



```
ALTER TABLE "IKOGAN"."CUSTACC" ALTER COLUMN "cid" SET
 GENERATED AS IDENTITY (START WITH 1);

ALTER TABLE "IKOGAN"."CUSTACC" ALTER COLUMN "timest"
SET DEFAULT CURRENT_TIMESTAMP;

ALTER TABLE "IKOGAN"."CUSTACC" ADD PRIMARY KEY("cid");

CREATE UNIQUE INDEX custacc_id ON
"IKOGAN"."CUSTACC"("cadoc") GENERATE KEY USING
XMLPATTERN 'declare default element namespace
"http://tpox-benchmark.com/custacc";/Customer/@id'
AS SQL DOUBLE;

INSERT INTO "IKOGAN"."CUSTACC"("cadoc") SELECT
XMLPARSE(DOCUMENT "CUSTDOC") FROM "IKOGAN"."CUSTOMER";
```

Overview	
File Kind	SQL
Options	
Execute statements separately	☑
Use semicolons as execution delimiters	☑
Parser should require semicolons	☑
General	
Database Kind	IBM DB2
Data Source	TPOX
Root Object	IKOGAN
File	
Creation Date	Tue, July
Modification Date	Thu, Dece
Access Date	Thu, Dece
Size	538 byte
Read Only	□

```
  Execution time: 0.016 sec
⊟ ✓ ALTER TABLE "IKOGAN"."CUSTACC" ADD PRIMARY KEY("cid")
  Execution time: 0.031 sec
⊟ ✓ CREATE UNIQUE INDEX custacc_id ON "IKOGAN"."CUSTACC"("cadoc") GENERATE KEY USIN
  Execution time: < 1 ms
⊟ ✓ INSERT INTO "IKOGAN"."CUSTACC"("cadoc") SELECT XMLPARSE(DOCUMENT "CUSTDOC") F
  Execution time: 6.828 sec
```

In the screenshot above, all commands have been executed successfully. The commands also include creation of the unique XML index that prevents the customer IDs from having duplicate values (@id attribute).

All data is now migrated to the newly created CUSTACC table.

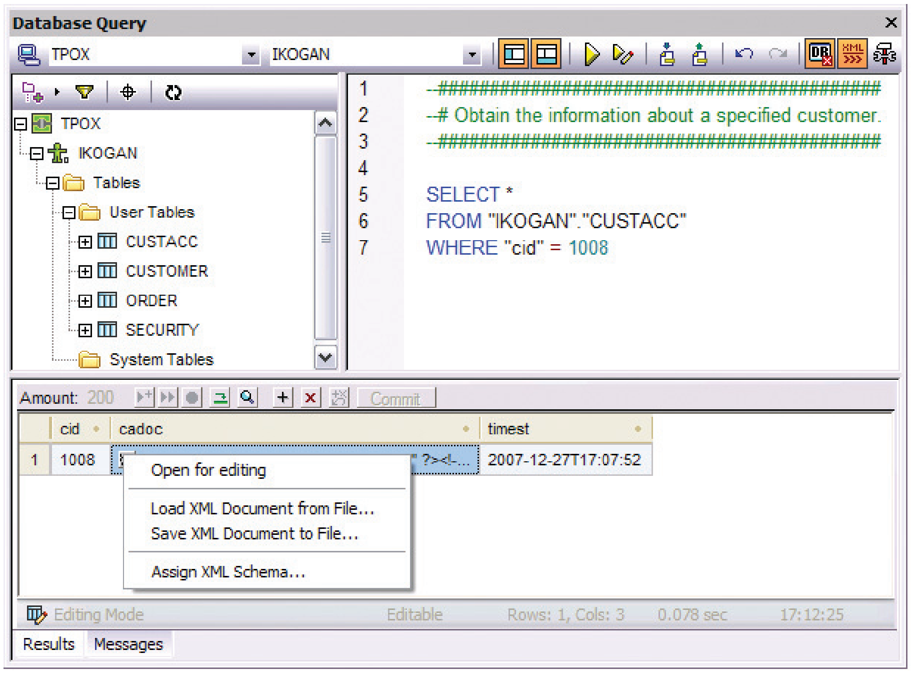
5.2 Selecting (SQL) and updating a customer document

Problem Description:

It turns out that some out-dated information about a customer is inserted into the table. As a result, this customer's document needs to be selected, reviewed, and then modified and saved back to the database table.

Solution:

A simple SELECT statement run from the Database Query window can be used to obtain the row for a customer with the specified customer ID. Note that the Database Query window is available in Altova XMLSpy, DatabaseSpy, and MapForce. XMLSpy is used in this particular exercise.

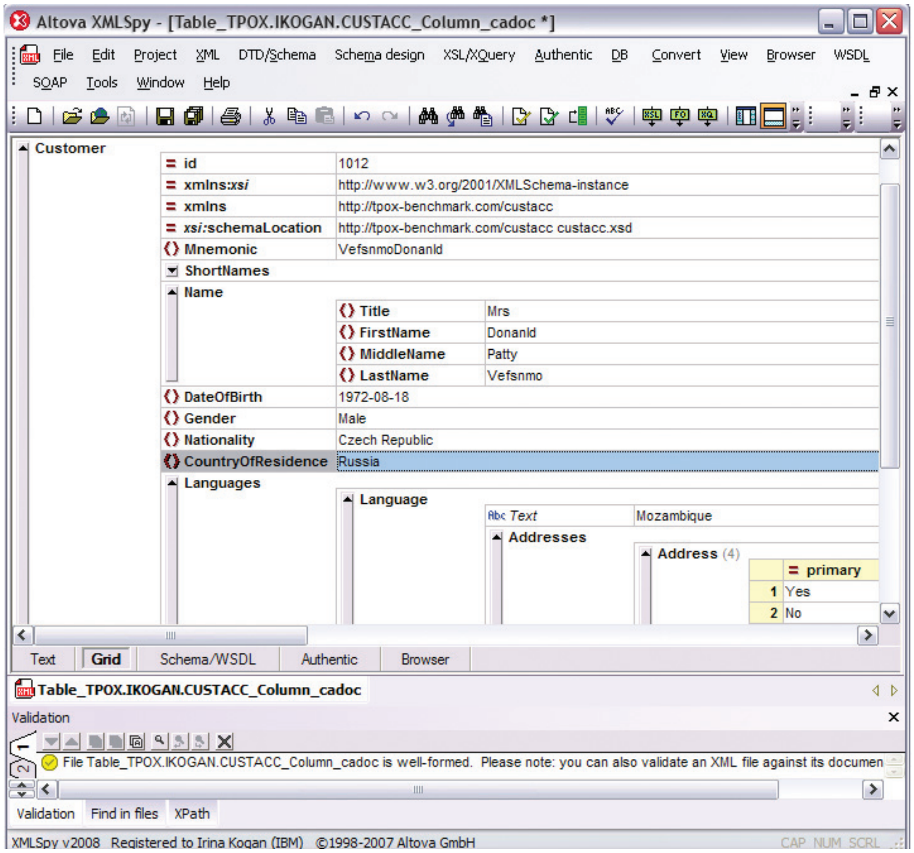


```
--###############################################
--# Obtain the information about a specified customer.
--###############################################

SELECT *
FROM "IKOGAN"."CUSTACC"
WHERE "cid" = 1008
```

	cid	cadoc	timest
1	1008	" ?><!-...	2007-12-27T17:07:52

The retrieved XML document can then be edited in XMLSpy, which also enables the user to verify that the document is still well-formed after editing:



Customer

= id	1012
= xmlns:*xsi*	http://www.w3.org/2001/XMLSchema-instance
= xmlns	http://tpox-benchmark.com/custacc
= *xsi*:schemaLocation	http://tpox-benchmark.com/custacc custacc.xsd
() Mnemonic	VefsnmoDonanld
ShortNames	
Name	
() Title	Mrs
() FirstName	Donanld
() MiddleName	Patty
() LastName	Vefsnmo
() DateOfBirth	1972-08-18
() Gender	Male
() Nationality	Czech Republic
() CountryOfResidence	Russia
Languages	
Language	
Text	Mozambique
Addresses	
Address (4)	

	= primary
1	Yes
2	No

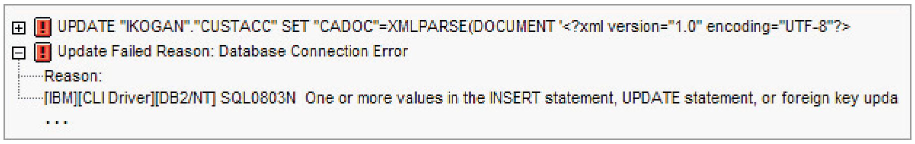
File Table_TPOX.IKOGAN.CUSTACC_Column_cadoc is well-formed. Please note: you can also validate an XML file against its documen

XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH

If the update on DB2 9 requires XML Schema validation, the user can select the 'Assign XML Schema…' menu item to choose a schema from the DB2 XML schema repository, and validation will run during the database update.

Clicking the 'Commit' button in the Database Query window saves the document back to DB2 9. Note that the XML document can also be easily saved to the local file system.

If during editing a user accidently changes the @id attribute of the customer to a pre-existing customer ID, then he/she would not be able to commit the changes and would receive the following error message:



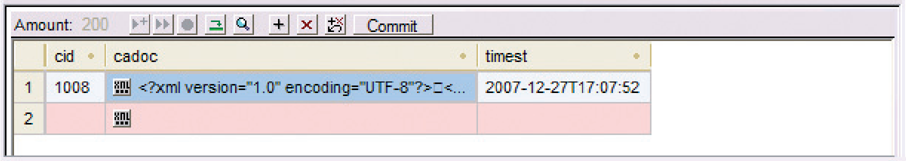
```
UPDATE "IKOGAN"."CUSTACC" SET "CADOC"=XMLPARSE(DOCUMENT '<?xml version="1.0" encoding="UTF-8"?>
Update Failed Reason: Database Connection Error
    Reason:
    [IBM][CLI Driver][DB2/NT] SQL0803N  One or more values in the INSERT statement, UPDATE statement, or foreign key upda
    ...
```

This is the whole explanation of the SQL0803N error code:

```
SQL0803N  One or more values in the INSERT statement, UPDATE statement,
          or foreign key update caused by a DELETE statement are not valid
          because the primary key, unique constraint or unique index
          identified by "<index-id>" constrains table "<table-name>" from
          having duplicate values for the index key.
```

This is due to the fact that the unique XML index custacc_id created in section 5.1 would be violated if such an update succeeded. The error messages produced by DB2 are retrieved by XMLSpy and displayed in the Message pane of the Database Query window.

The Database Query window also enables the user to insert new rows or delete existing rows from a database table using the "+" and "x" buttons:



	cid	cadoc	timest
1	1008	<?xml version="1.0" encoding="UTF-8"?>□<...	2007-12-27T17:07:52
2			

5.3 Obtaining a partial customer profile

Problem Description:

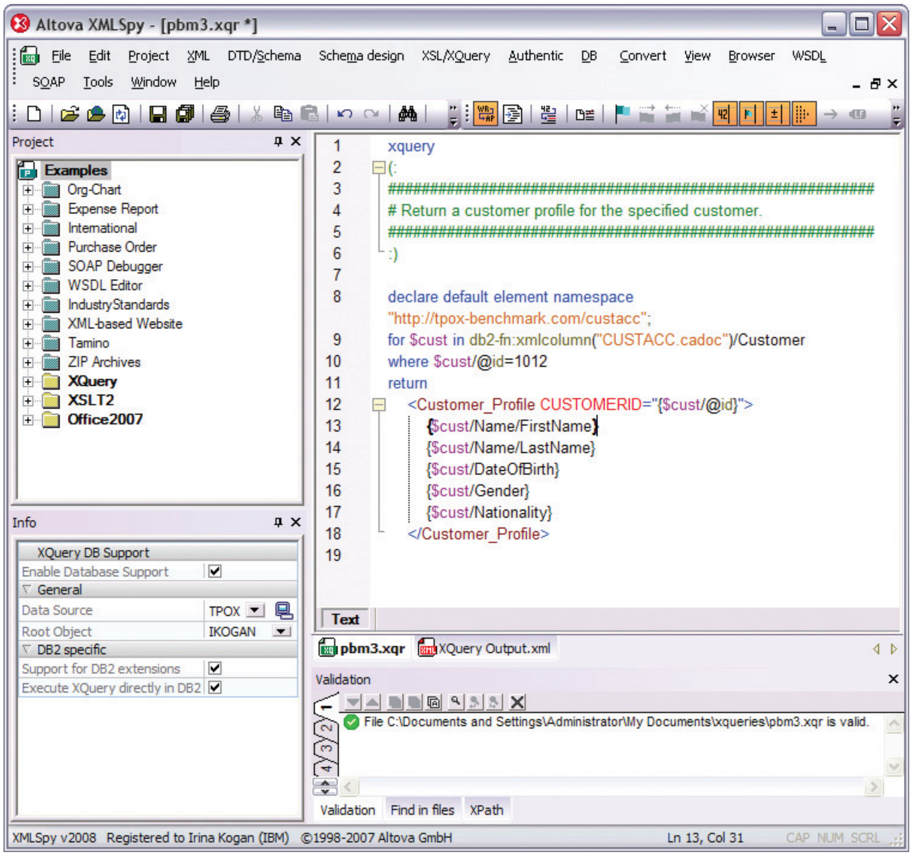
The same customer's (see section 5.2) partial profile needs to be obtained in three different formats: XML, CSV, and an HTML table. This will enable the user to export it to other office systems and publish the information on the sales intranet.

Solution:

This problem is solved using three of the Altova tools. XQuery executed from XMLSpy produces an XML output. MapForce converts the XML document from the database to a CSV (comma separated values) file. StyleVision transforms the document into an HTML table. These three ways of obtaining a customer profile are demonstrated next.

5.3.1 XML output (XQuery in XMLSpy)

The XQuery statement run against DB2 to produce XML output is displayed in the XMLSpy XQuery editor below:



```
xquery
(:
##############################################################
# Return a customer profile for the specified customer.
##############################################################
:)

declare default element namespace
"http://tpox-benchmark.com/custacc";
for $cust in db2-fn:xmlcolumn("CUSTACC.cadoc")/Customer
where $cust/@id=1012
return
   <Customer_Profile CUSTOMERID="{$cust/@id}">
      {$cust/Name/FirstName}
      {$cust/Name/LastName}
      {$cust/DateOfBirth}
      {$cust/Gender}
      {$cust/Nationality}
   </Customer_Profile>
```

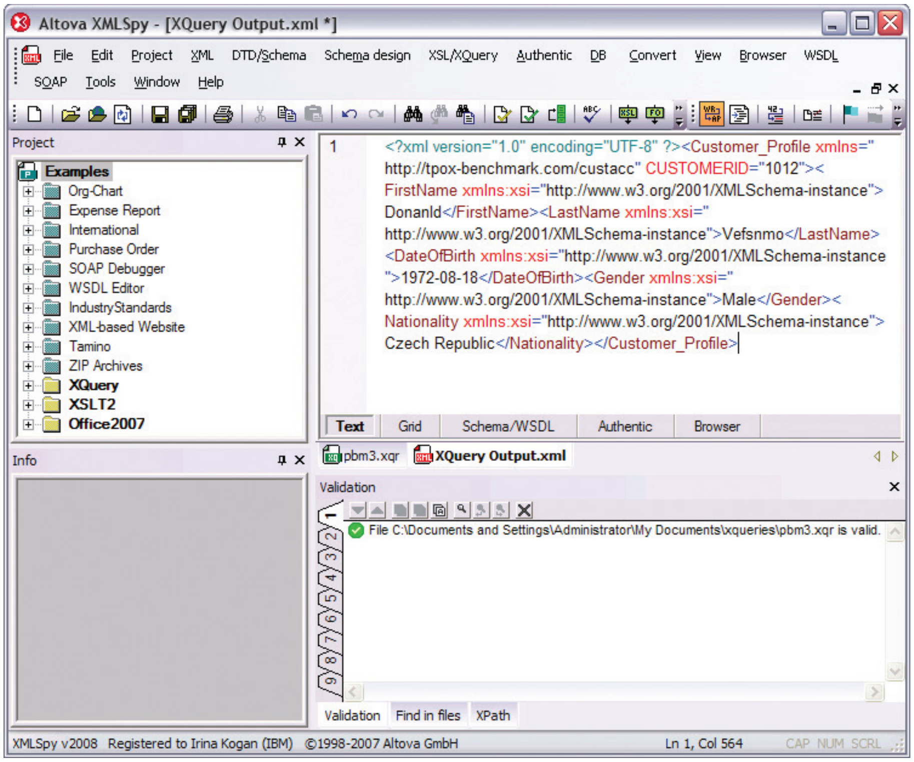
XQuery DB Support
Enable Database Support ☑
General
Data Source TPOX
Root Object IKOGAN
DB2 specific
Support for DB2 extensions ☑
Execute XQuery directly in DB2 ☑

File C:\Documents and Settings\Administrator\My Documents\xqueries\pbm3.xqr is valid.

XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH Ln 13, Col 31 CAP NUM SCRL

As shown above, ‘Support for DB2 extensions’ and ‘Execute XQuery directly in DB2’ need to be selected in the XMLSpy Info window. Then the XQuery can be validated by XMLSpy against DB2 9, with functions using DB2 XQuery extensions, such as `db2-fn:xmlcolumn`, being understood. Had the user encountered any problems with the XQuery document, XMLSpy also includes an XQuery debugger and profiler for troubleshooting and optimizing queries.

When the XQuery is executed against DB2 9, the XML file shown below is produced:



```
<?xml version="1.0" encoding="UTF-8" ?><Customer_Profile xmlns="
http://tpox-benchmark.com/custacc" CUSTOMERID="1012"><
FirstName xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
Donanld</FirstName><LastName xmlns:xsi="
http://www.w3.org/2001/XMLSchema-instance">Vefsnmo</LastName>
<DateOfBirth xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance
">1972-08-18</DateOfBirth><Gender xmlns:xsi="
http://www.w3.org/2001/XMLSchema-instance">Male</Gender><
Nationality xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
Czech Republic</Nationality></Customer_Profile>
```

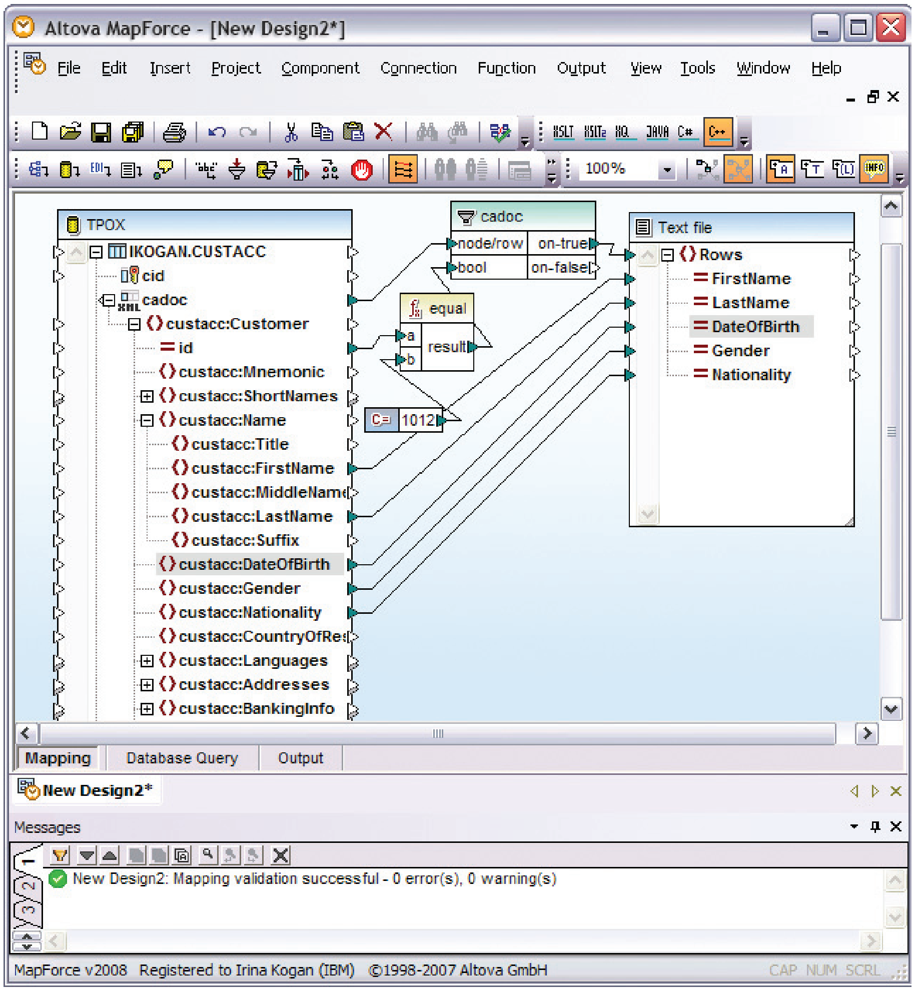
File C:\Documents and Settings\Administrator\My Documents\xqueries\pbm3.xqr is valid.

XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH Ln 1, Col 564

5.3.2 CSV output (MapForce)

CSV output can be easily saved in a text file or opened in other applications, such as Microsoft® Excel. Transforming the XML data to a CSV format is accomplished easily using MapForce. The user creates a visual mapping between the data structures, and MapForce automatically makes the conversion. The first object used for the mapping is the CUSTACC table from the DB2 SAMPLE database; the second is a CSV file. First, to accurately map the XML data, the user needs to specify the schema for the cadoc XML column of the CUSTACC table, which is accomplished by right-clicking the column and selecting “Assign XML Schema to Field…”. Then, the user can drag and drop connecting lines to map the XML elements in the database to the corresponding fields in the CSV target.

The resulting mapping is shown below:

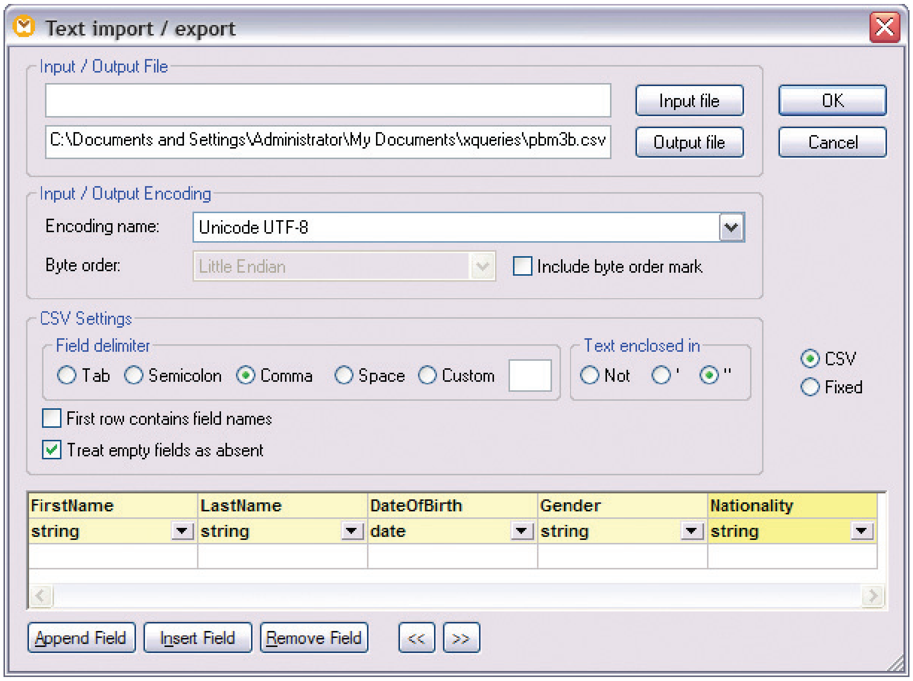


New Design2: Mapping validation successful - 0 error(s), 0 warning(s)

MapForce v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH CAP NUM SCRL

The mapping also uses graphical data processing functions to filter the data to map only the customer documents that have @id equal to 1012 in the database; i.e., the single customer document that we are interested in.

Right-clicking the CSV file component in the MapForce mapping enables us to specify its properties, as shown below.



Text import / export

Input / Output File

C:\Documents and Settings\Administrator\My Documents\xqueries\pbm3b.csv

Input / Output Encoding

Encoding name: Unicode UTF-8

Byte order: Little Endian □ Include byte order mark

CSV Settings

Field delimiter: ○ Tab ○ Semicolon ● Comma ○ Space ○ Custom

Text enclosed in: ○ Not ○ ' ● "

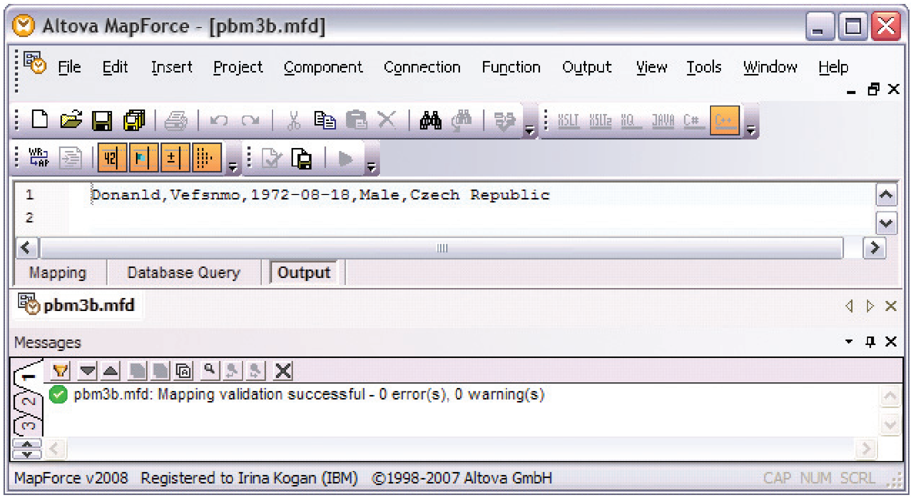
● CSV ○ Fixed

□ First row contains field names

☑ Treat empty fields as absent

FirstName	LastName	DateOfBirth	Gender	Nationality
string	string	date	string	string

Once the mapping is defined, we can view the results immediately by clicking the Output tab. The output contains a list of the requested values separated by a comma.



Altova MapForce - [pbm3b.mfd]

```
Donanld,Vefsnmo,1972-08-18,Male,Czech Republic
```

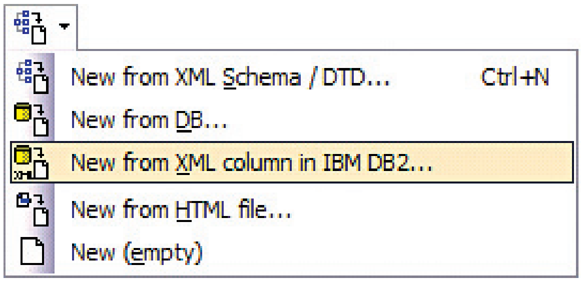
pbm3b.mfd: Mapping validation successful - 0 error(s), 0 warning(s)

MapForce v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH

5.3.3 HTML table output (StyleVision)

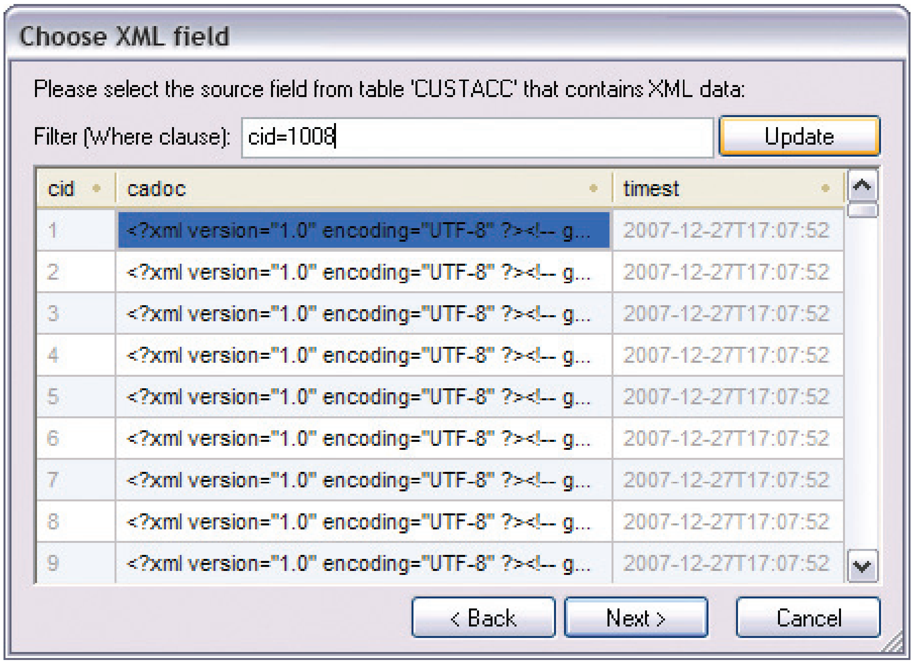
The next step is to publish the customer profile in an HTML table. For this we use Altova StyleVision.

First, we need to insert an XML Schema or instance document from DB2 9 as an object to be used in the transformation.


New from XML Schema / DTD... Ctrl+N
New from DB...
New from XML column in IBM DB2...
New from HTML file...
New (empty)

Then the 'Select XML table' menu pops up, from which we choose the CUSTACC table. (The CUSTOMER table does not appear in the menu as it does not have any XML columns.)

Next, we need to specify which document we would like to work on. The customer we are interested in has the customer ID (from the integer column cid) equal to 1008, which we enter in the filter field (an 'Update' button needs to be clicked to actually limit the selection to this single document).

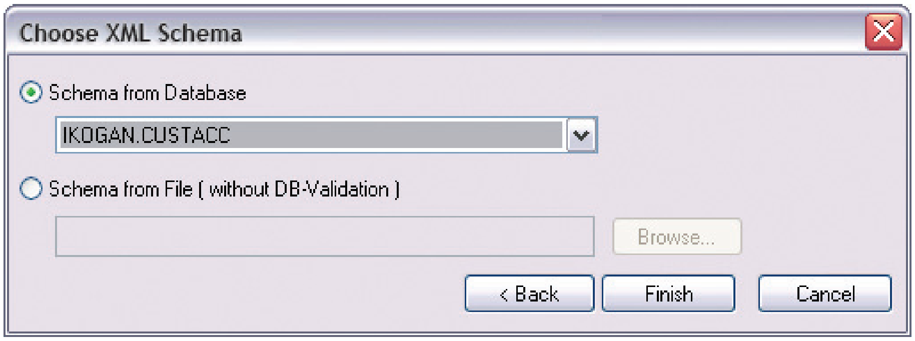


Please select the source field from table 'CUSTACC' that contains XML data:

Filter (Where clause): cid=1008

cid	cadoc	timest
1	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
2	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
3	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
4	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
5	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
6	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
7	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
8	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52
9	<?xml version="1.0" encoding="UTF-8" ?><!-- g...	2007-12-27T17:07:52

Next, a menu pops up that enables the user to choose an XML Schema, either from the database or from our file system.



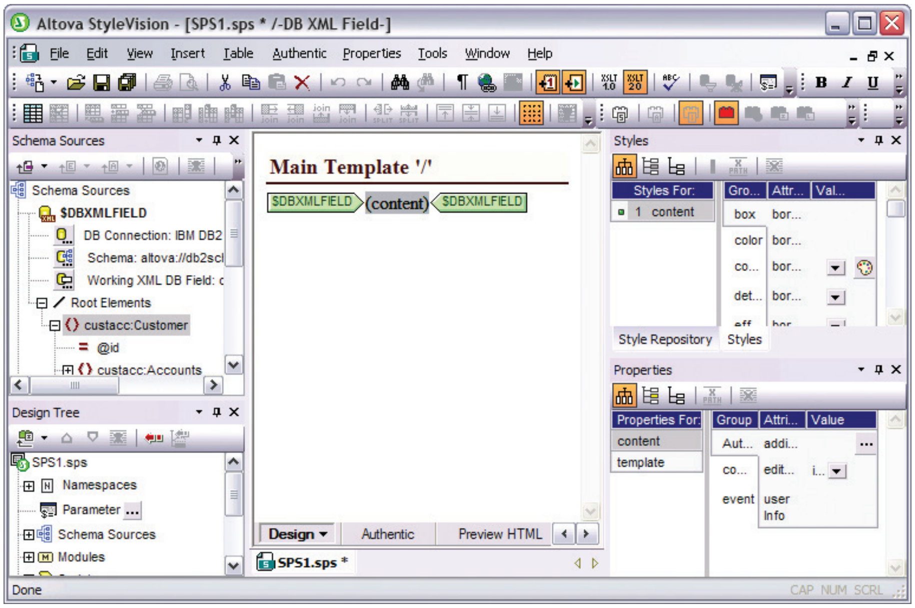
Choose XML Schema

● Schema from Database

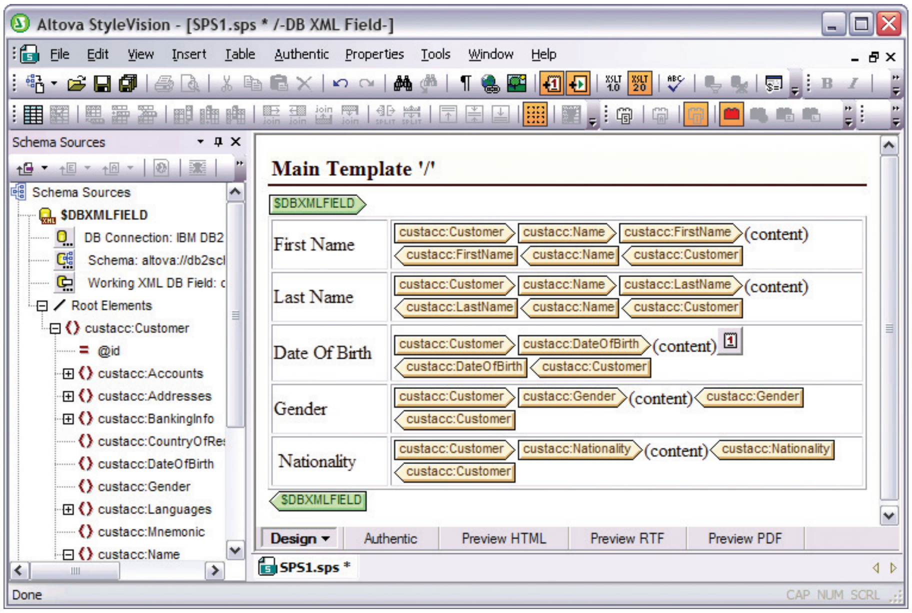
IKOGAN.CUSTACC

○ Schema from File (without DB-Validation)

Once the schema is selected it is displayed in the Schema Sources window:

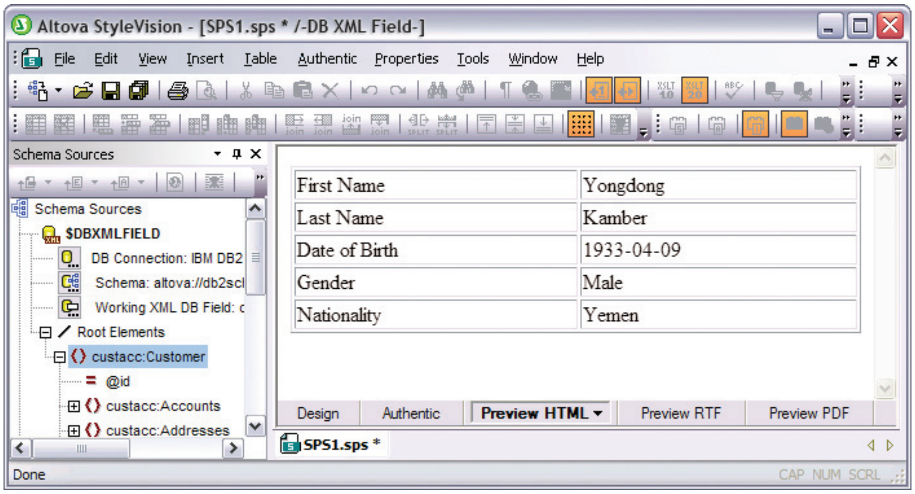


To create a design to render the customer profile in an HTML table, we insert a table into the Main Template, where the content is to appear. In this case, we need five rows and two columns. Then we can manually enter the row names, and drag the content from the XML Schema elements listed in the Schema Sources window to the appropriate cells in the table. The final template is below:



First Name	custacc:Customer custacc:Name custacc:FirstName (content) custacc:FirstName custacc:Name custacc:Customer
Last Name	custacc:Customer custacc:Name custacc:LastName (content) custacc:LastName custacc:Name custacc:Customer
Date Of Birth	custacc:Customer custacc:DateOfBirth (content) 1 custacc:DateOfBirth custacc:Customer
Gender	custacc:Customer custacc:Gender (content) custacc:Gender custacc:Customer
Nationality	custacc:Customer custacc:Nationality (content) custacc:Nationality custacc:Customer

By clicking ‘Preview HTML’, we can view the following output:



First Name	Yongdong
Last Name	Kamber
Date of Birth	1933-04-09
Gender	Male
Nationality	Yemen

StyleVision also includes advanced features for styling our output, including CSS styles, JavaScript functions, templates, etc., but for this example, a simple table is all we require.

5.4 Placing a new order (validation with FIXML)

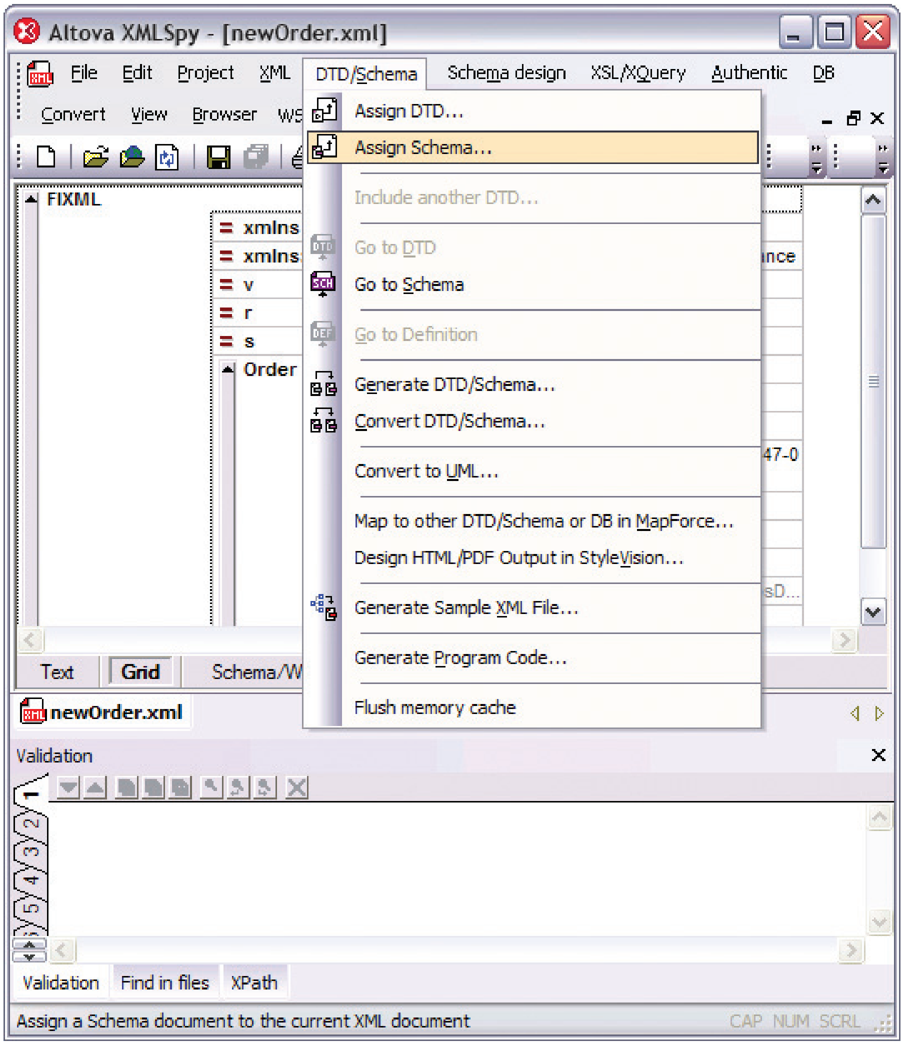
Problem Description:

A request comes in to insert a new order document for a transaction in which a customer sold some shares of the security he/she owned. Validation needs to be performed to ensure that the order document conforms to the FIXML schema.

Solution:

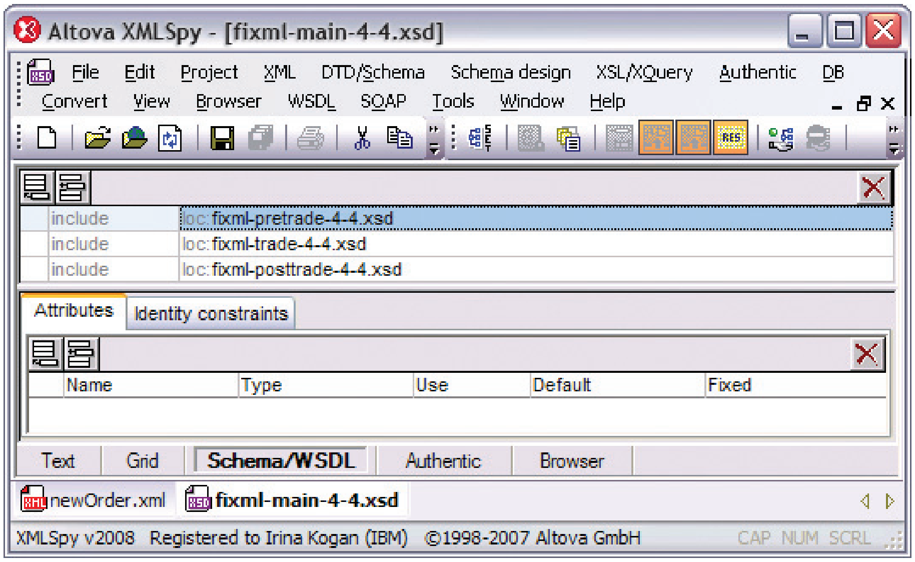
There are two ways to approach this problem. One is to validate the new order document in XMLSpy and then insert it into DB2 9. The other is to insert the document into a DB2 9 table "with validation".

Let's look at the first scenario first. The document is opened in the Grid Perspective in XMLSpy, and a FIXML schema is assigned to it:



Assign a Schema document to the current XML document

The FIXML schema is an industry-standard schema that consists of 41 .xsd files. This is what the primary file looks like when displayed graphically in the Schema/WSDL window:

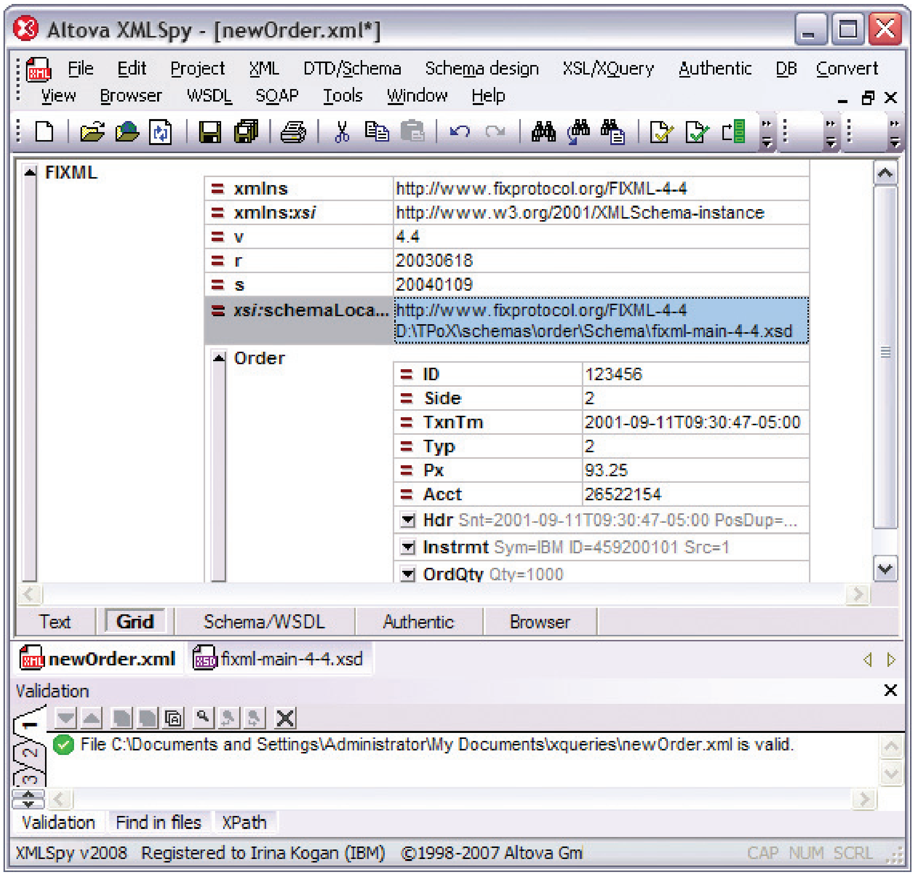

Altova XMLSpy - [fixml-main-4-4.xsd]
File Edit Project XML DTD/Schema Schema design XSL/XQuery Authentic DB
Convert View Browser WSDL SOAP Tools Window Help
Attributes
Identity constraints
Text Grid Schema/WSDL Authentic Browser
newOrder.xml
fixml-main-4-4.xsd
CAP NUM SCRL

include	loc: fixml-pretrade-4-4.xsd
include	loc: fixml-trade-4-4.xsd
include	loc: fixml-posttrade-4-4.xsd

Name	Type	Use	Default	Fixed

XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH

Now that we've assigned the schema, we can validate newOrder.xml by pressing F8, and validation is successful.


Altova XMLSpy - [newOrder.xml*]
File Edit Project XML DTD/Schema Schema design XSL/XQuery Authentic DB Convert
View Browser WSDL SOAP Tools Window Help
Text Grid Schema/WSDL Authentic Browser
newOrder.xml
fixml-main-4-4.xsd
Validation
Validation Find in files XPath
CAP NUM SCRL

FIXML

= xmlns	http://www.fixprotocol.org/FIXML-4-4
= xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
= v	4.4
= r	20030618
= s	20040109
= xsi:schemaLoca...	http://www.fixprotocol.org/FIXML-4-4 D:\TPoX\schemas\order\Schema\fixml-main-4-4.xsd

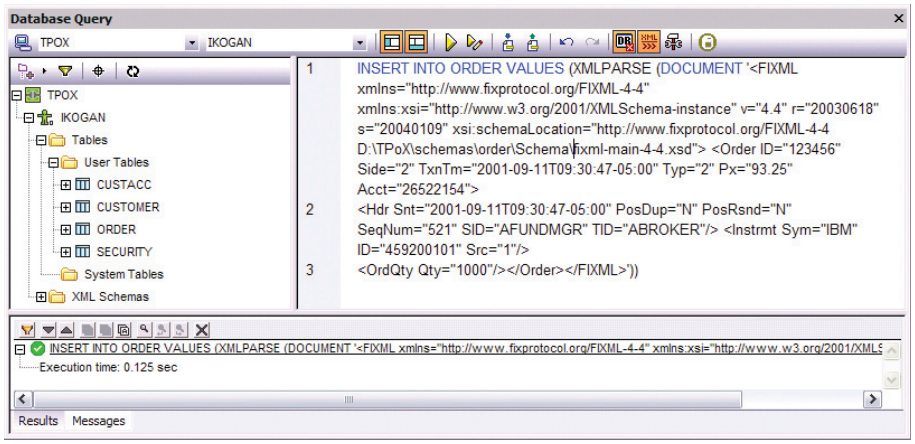
Order

= ID	123456
= Side	2
= TxnTm	2001-09-11T09:30:47-05:00
= Typ	2
= Px	93.25
= Acct	26522154
Hdr Snt=2001-09-11T09:30:47-05:00 PosDup=...	
Instrmt Sym=IBM ID=459200101 Src=1	
OrdQty Qty=1000	

File C:\Documents and Settings\Administrator\My Documents\xqueries\newOrder.xml is valid.

XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova Gm

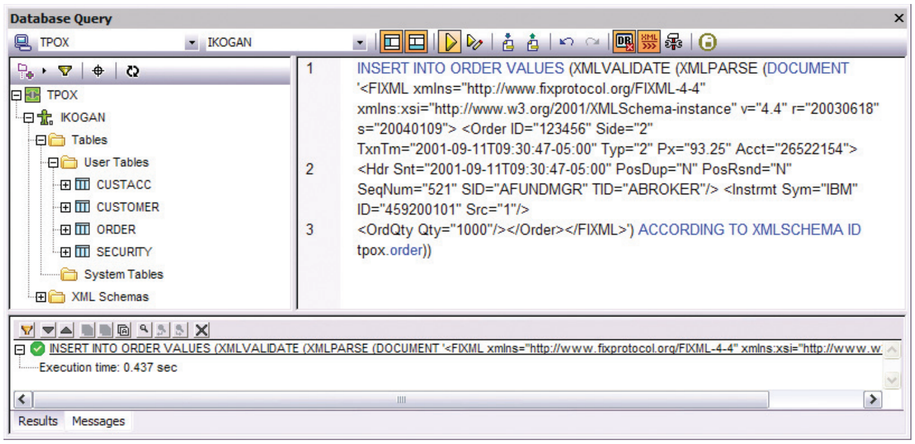
Now the document can be easily inserted into the Order table in DB2 9:



```
INSERT INTO ORDER VALUES (XMLPARSE (DOCUMENT '<FIXML
xmlns="http://www.fixprotocol.org/FIXML-4-4"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" v="4.4" r="20030618"
s="20040109" xsi:schemaLocation="http://www.fixprotocol.org/FIXML-4-4
D:\TPoX\schemas\order\Schema\fixml-main-4-4.xsd"> <Order ID="123456"
Side="2" TxnTm="2001-09-11T09:30:47-05:00" Typ="2" Px="93.25"
Acct="26522154">
<Hdr Snt="2001-09-11T09:30:47-05:00" PosDup="N" PosRsnd="N"
SeqNum="521" SID="AFUNDMGR" TID="ABROKER"/> <Instrmt Sym="IBM"
ID="459200101" Src="1"/>
<OrdQty Qty="1000"/></Order></FIXML>'))
```

```
INSERT INTO ORDER VALUES (XMLPARSE (DOCUMENT '<FIXML xmlns="http://www.fixprotocol.org/FIXML-4-4" xmlns:xsi="http://www.w3.org/2001/XMLS
Execution time: 0.125 sec
```

Alternatively, since the FIXML schema is already registered on DB2 9 (as tpox.order), we can use the XMLSpy Database Query window to perform an “insert with validation” function. The SQL command “XMLVALIDATE … ACCORDING TO XMLSCHEMA ID tpox.order” performs what we need:



```
INSERT INTO ORDER VALUES (XMLVALIDATE (XMLPARSE (DOCUMENT
'<FIXML xmlns="http://www.fixprotocol.org/FIXML-4-4"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" v="4.4" r="20030618"
s="20040109"> <Order ID="123456" Side="2"
TxnTm="2001-09-11T09:30:47-05:00" Typ="2" Px="93.25" Acct="26522154">
<Hdr Snt="2001-09-11T09:30:47-05:00" PosDup="N" PosRsnd="N"
SeqNum="521" SID="AFUNDMGR" TID="ABROKER"/> <Instrmt Sym="IBM"
ID="459200101" Src="1"/>
<OrdQty Qty="1000"/></Order></FIXML>') ACCORDING TO XMLSCHEMA ID
tpox.order))
```

```
INSERT INTO ORDER VALUES (XMLVALIDATE (XMLPARSE (DOCUMENT '<FIXML xmlns="http://www.fixprotocol.org/FIXML-4-4" xmlns:xsi="http://www.w
Execution time: 0.437 sec
```

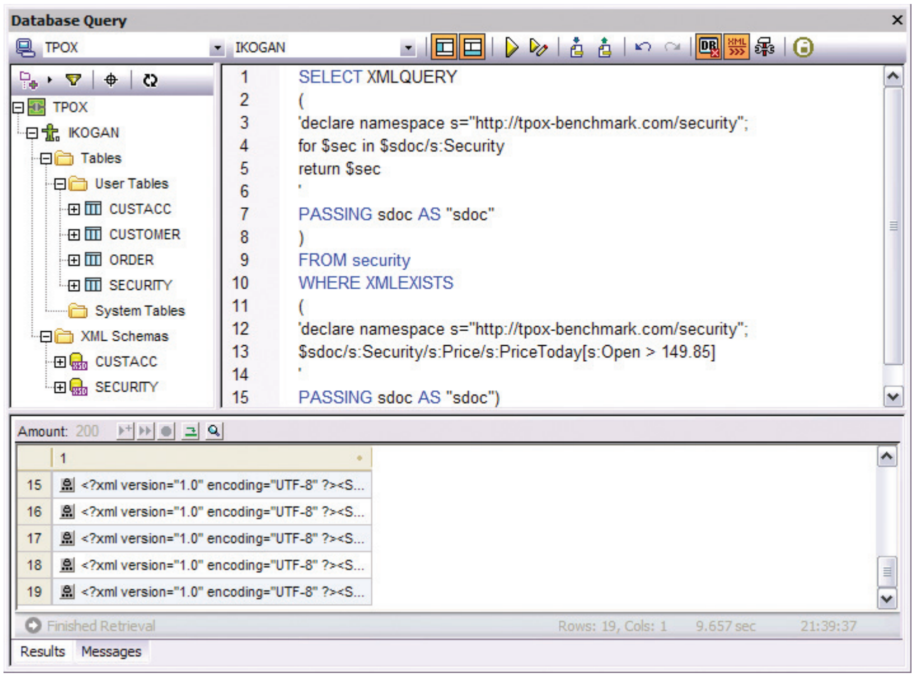
5.5 Producing a list of securities (SQL/XML)

Problem Description:

The manager of the online brokerage firm sends a request to the DBA to produce a list of securities that have the current open price higher than or equal to the specified one.

Solution:

The following SQL/XML statement produces the required list of securities. It can be run from the Database Query window in XMLSpy.



```
SELECT XMLQUERY
(
'declare namespace s="http://tpox-benchmark.com/security";
for $sec in $sdoc/s:Security
return $sec
'
PASSING sdoc AS "sdoc"
)
FROM security
WHERE XMLEXISTS
(
'declare namespace s="http://tpox-benchmark.com/security";
$sdoc/s:Security/s:Price/s:PriceToday[s:Open > 149.85]
'
PASSING sdoc AS "sdoc")
```

	1
15	<?xml version="1.0" encoding="UTF-8" ?><S...
16	<?xml version="1.0" encoding="UTF-8" ?><S...
17	<?xml version="1.0" encoding="UTF-8" ?><S...
18	<?xml version="1.0" encoding="UTF-8" ?><S...
19	<?xml version="1.0" encoding="UTF-8" ?><S...

Finished Retrieval Rows: 19, Cols: 1 9.657 sec 21:39:37

The returned XML documents are neatly displayed in a table. In this case, there are nineteen documents in the result set (Rows: 19). By clicking a document, a user can easily view it in XMLSpy.

Note that it would also be easy to retrieve the same set of documents by running XQuery from XMLSpy, similar to how this was done in section 5.3.1.

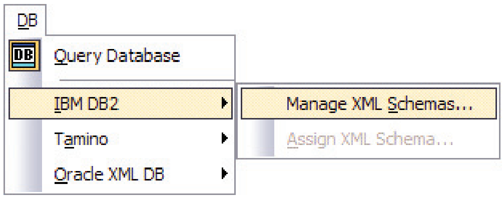
5.6 Issuing a change in the security schema

Problem Description:

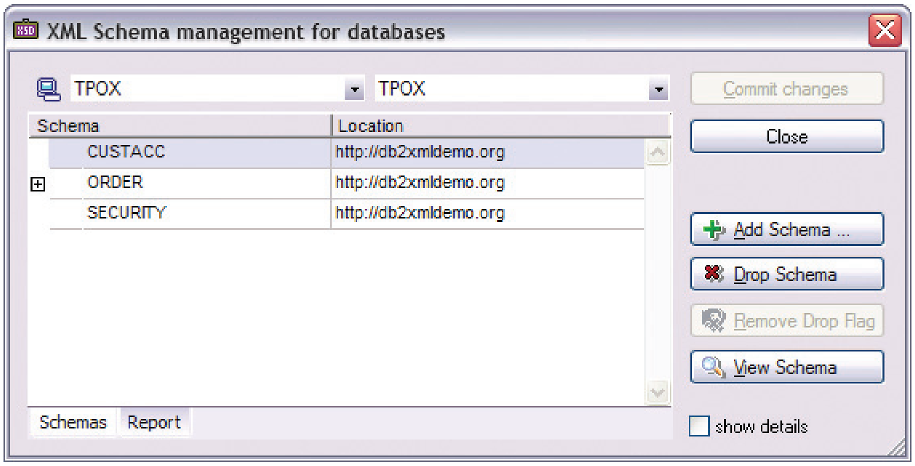
The online brokerage firm requests a change in the security schema. The updated schema needs to be saved back to the database under a different name.

Solution:

First, the original security schema needs to be retrieved from DB2 9. This can now be easily done using XMLSpy:



Once we choose “Manage XML Schemas…”, we see all the XML Schemas available in the database we are working with. Note that the SQL schema name under which the schema is registered can be chosen from a scroll-down menu. In the next snapshot, ‘TPoX’ is chosen:

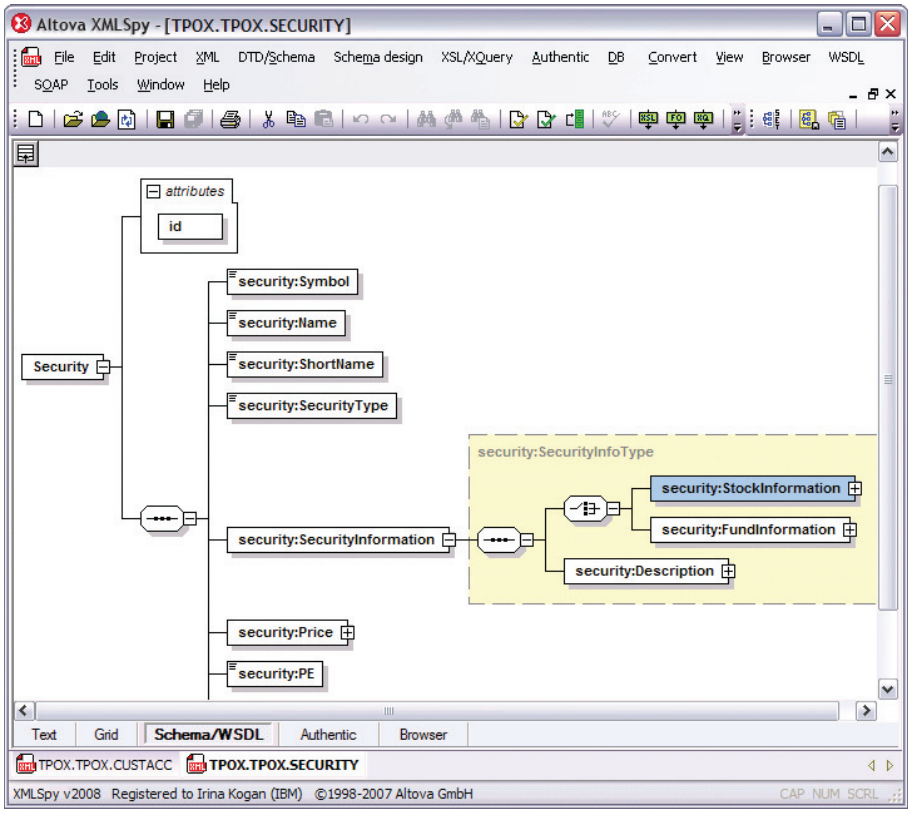


TPOX | TPOX

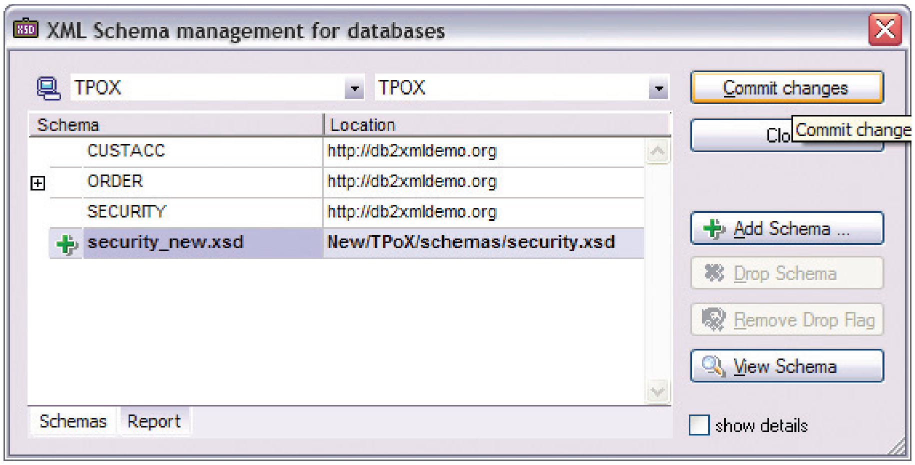
Schema	Location
CUSTACC	http://db2xmldemo.org
ORDER	http://db2xmldemo.org
SECURITY	http://db2xmldemo.org

□ show details

When we select the security schema and press the ‘View Schema’ button, the file is opened in XMLSpy.


Altova XMLSpy - [TPOX.TPOX.SECURITY]
File Edit Project XML DTD/Schema Schema design XSL/XQuery Authentic DB Convert View Browser WSDL SOAP Tools Window Help
attributes
id
Security
security:Symbol
security:Name
security:ShortName
security:SecurityType
security:SecurityInfoType
security:StockInformation
security:FundInformation
security:SecurityInformation
security:Description
security:Price
security:PE
Text Grid Schema/WSDL Authentic Browser
TPOX.TPOX.CUSTACC
TPOX.TPOX.SECURITY
XMLSpy v2008 Registered to Irina Kogan (IBM) ©1998-2007 Altova GmbH
CAP NUM SCRL

Next we can edit and validate the schema. Once the schema is edited, it can be saved to back DB2 9. The 'Add Schema…' button in the window used for managing XML Schemas enables the user to browse for the schema on the file system. Once we've selected the new schema, we simply click 'Commit Changes' to save the changes to DB2 9 (in this case to register a new schema on DB2 9):



Schema	Location
CUSTACC	http://db2xmldemo.org
ORDER	http://db2xmldemo.org
SECURITY	http://db2xmldemo.org
security_new.xsd	New/TPoX/schemas/security.xsd

6 Summary

This paper demonstrates how various online trading brokerage business problems can be easily solved by using DatabaseSpy, XMLSpy, MapForce or StyleVision connecting to the DB2 9 database that contains the XML data and schemas. The solutions are so general that they are clearly applicable to the business problems from any industry that uses XML.

About IBM and Altova

IBM and Altova provide deep integration between the world's first multi-structured data server with the industry's premier XML tools. The combination of Altova tools and DB2 9 pureXML can help different players in the XML community (developers, DBAs, and end users) be instantly productive.

The information contained in this document represents the current view of Altova with respect to the subject matter herein contained as of the date of the publication. Altova makes no commitment to keep the information contained herein up to date and the information contained in this document is subject to change without notice.

As Altova GMBH must respond to the changing market conditions, Altova GMBH cannot guarantee the accuracy of any information presented after the date of publication. The document is presented for informational purposes only.

ALTOVA PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OF IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IBM makes no warranties or representations with respect to the content hereof and specifically disclaims any implied warranties of merchantability or fitness for any particular purpose. IBM assumes no responsibility for any errors that may appear in this document. The information contained in this document is subject to change without any notice. IBM reserves the right to make any such changes without obligation to notify any person of such revision or changes. IBM makes no commitment to keep the information contained herein up to date.

The information in this document concerning non-IBM products was obtained from the supplier(s) of those products. IBM has not tested such products and cannot confirm the accuracy of the performance, compatibility or any other claims related to non-IBM products. Questions about the capabilities of non-IBM products should be addressed to the supplier(s) of those products.

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

References

[1] DB2 pureXML™ Technology, **ibm.com/db2/xml**

[2] DB2 Express-C – free edition of DB2 9 with pureXML,
http://www.ibm.com/software/data/db2/express/

[3] Altova® products and technology, **http://www.altova.com/products.html**

[4] Altova IBM DB2 9 integration,
http://www.altova.com/IBM_DB2_9_pureXML.html

[5] Irina Kogan: *"Using the Altova Tools with IBM DB2 pureXML"*, developerWorks, 2007.
http://www.ibm.com/developerworks/db2/library/long/dm-0712kogan/?S_TACT=105AGX11&S_CMP=FP

[6] M. Nicola, B. van der Linden: *"Native XML Support in DB2 Universal Database"*, VLDB, 2005,
http://www.vldb2005.org/program/paper/thu/p1164-nicola.pdf

[7] Cindy Saracco, *"What's new in DB2 Viper: XML to the Core"*,
http://www.ibm.com/developerworks/db2/library/techarticle/dm-0602saracco/

[8] TPoX XML Database Benchmark **http://tpox.sourceforge.net**, open-source since Jan. 2007.

[9] I. Kogan, M. Nicola, B. Schiefer: *"DB2 9 XML performance characteristics"*, Technical Report, developerWorks, June 2006.
http://www.ibm.com/developerworks/ db2/library/techarticle/dm-0606schiefer/index.html

[10] M. Nicola, I. Kogan, B. Schiefer: *"An XML Transaction Processing Benchmark"*, SIGMOD, 2007.
http://tpox.sourceforge.net/Sigmod2007_TPoX.pdf

[11] The Financial Information eXchange Protocol, FIXML 4.4 Schema Specification 20040109, Revision1 2006-10-06.
http://www.fixprotocol.org/specifications/fix4.4fixml

[12] Toxgene Data Generator. **http://www.cs.toronto.edu/tox/toxgene**

[1] DB2 Viper was a Beta version of DB2 9