BRIDGING RELATIONAL TECHNOLOGY AND XML

HiT Software, Inc.

Giovanni Guardalben
VP R&D

gianni@hitsw.com

Evolution of XML-to-RDBMS Integration

- **Main Research Areas:**
  - **Techniques for Storing/Modeling XML in Relational Repositories**
  - **Materializing Relational Data as XML Documents**
  - **Querying XML Views of Relational Data**
  - **Using XML for Data Integration (Mapping Technologies)**
- **HiT Software:**
  - **Allora Framework for XML-to-RDBMS Integration**

Techniques for Storing/Modeling XML in Relational Repositories

- **Problem: use relational storage to save/retrieve/query XML documents.**
- **Generic Technique**
  - **Edge relations**
  - **XML Type/ LOB SQL Type**
  - **The Monet Project**
- **Schema-driven Technique**
  - **Fixed Mapping**
    - **The Hybrid Inlining Algorithm**
    - **The Constraint-preserving Algorithm**
    - **Relational-to-XML Translations**
    - **The Order-preserving Algorithm**
    - **Cost-based Mapping Algorithm**
  - **User- defined Mapping**
    - **The X-Ray Project**
    - **HiT Allora**

Generic Technique

- **Edge Relations**
  - **Store all attributes (elements) in a single table (Edge table)**
  - **Table records the oids of the source and target objects**
  - **Edge Table structure:**
    - **Edge (source, ordinal, name, flag, target)**
    - **Flag is either a data type or inter-object reference**
  - **Attribute approach:**
    - **Group all attributes with the same name into one table (horizontal partitioning)**
  - **Universal table:**
    - **Separate columns for all the attribute (element) names that occur in the XML document**
  - **References**

Daniela Florescu – Donald Kossman

A Performance Evaluation of Alternative Mapping Schemes for Storing XML Data in a Relational Database – *INRIA Rapport de recherche Mai 1999.*

Generic Technique (cont'd)

- **XML Type/ LOB SQL Type**
  - **New XML data type or revisited blob data type**
  - **Supported by most commercial RDBMS XML Extensions**
  - **Limited XML functionality**
  - **Very efficient**
  - **Text Search facilities available**

  - **References**

  See documentation of Oracle, MS SQLServer, IBM DB2 and Sybase ASE.

Generic Technique (cont'd)

- **The Monet XML Project**
  - **Based on the structure of the document at run-time (independent of the DTD)**
  - **An association is either an edge (i.e., parent-child relationship) or an attribute value.**
  - **A path is a sequence of associations.**
  - **Monet uses XML paths to group related associations into the same relation**
  - **Monet enables an object oriented perspective**
  - **Monet combines the elegance of clear semantics with an efficient execution model**

  - **References**

Schmidt A., et al.CWI

Efficient relational storage and retrieval of XML documents. *Proceedings of WebDB, pages 47-52, 2000.*

Schema-driven Technique

- **Fixed Mapping: The Hybrid Inlining Algorithm**
  - **Create a DTD graph (nodes are elements, attributes, operators)**
  - **Treat the | DTD operator as node sequence**
  - **Identify top nodes:**
    - **source nodes**
    - **child nodes of operators * or +**
    - **recursive node with indegree > 1**
  - **Starting from top node T inline all elements and attributes reachable from T unless they are other top nodes**
  - **Attribute names are concatenated using – from the top node name**
  - **Parent_elm and Root_elm can added to improve query efficiency**

  - **References**

Shanmugasundaram, J., et al.

Relational Databases for Querying XML Documents:Limitations and Opportunities. *Proceedings VLDB, Edinburgh, Scotland, 1999.*

Schema-driven Technique (cont'd)

- **Fixed Mapping: The Constraint-preserving Algorithm**
  - **Add semantics constraints**
    - **Domain Constraints -> CHECK (VALUE IN (…,…))**
    - **Cardinality Constraints -> NULL, NOT NULL**
    - **Inclusion Dependencies ->FOREIGN KEY () REFERENCES or CHECK (… IN (SELECT…))**
    - **Singleton Constraint -> UNIQUE**
  - **References**

  Lee.D.,et al.

  Constraint-preserving Transformations from XML Document Type Definition to Relational Schema. I*nt'l Conf. on Conceptual Modeling (ER) Salt Lake City, UT, Oct. 2000.*
- **Fixed Mapping: Relational-to-XML Translations**
  - **Use Inclusion Dependencies from Middleware Catalog Info**
  - **Create an Inclusion Dependency Graph (IND-Graph)**
  - **References**

  Lee.D., et al.

  NeT & CoT : Translating Relational Schemas to XML Schemas using Semantic Constraints. *UCLA CS Technical Report, Feb. 2002.*

Schema-driven Technique (cont'd)

- **Fixed Mapping: The Order-preserving Algorithm**
  - **Support ordered XML data using the unordered relational data model**
  - **Three encoding methods: the Global Order encoding, the Local Order encoding and the Dewey Order encoding.**
  - **Also present algorithms to translate XPath queries into SQL (position) – range predicates in XQuery.**
  - **The order encoding is demonstrated both in a Generic environment (Edge table) and Schema-driven environment (by adding an ordering column to Hybrid Inlining algorithm).**

  - **References**

Tatarinov I., et al.

Storing and querying ordered XML using a relational database system.. *ACM SIGMOD 2002, June 4-6, Madison, Wisconsin, USA.*

Schema-driven Technique (cont'd)

- **Fixed Mapping: Cost-based Mapping Algorithm**
  - **Given an XML Schema generate an initial physical schema**
    - **As expressive as XML Schema**
    - **Contain useful statistics about data to be stored**
    - **There exists a fixed simple mapping from p-schemas to relational tables**
  - **Mapping p-schemas to relations**
    - **One relation per type**
    - **One key to store the node id**
    - **Create all foreign keys as necessary**
  - **Find the most appropriate relational storage based on statistics**

  - **References**

Bohannon P., et al.

From XML Schema to Relations: A Cost-based Approach to XML Storage. *ICDE 2002.*

Schema-driven Technique (cont'd)

- **User- defined Mapping: The X-Ray Project**
  - **Mapping may be defined between XML DTDs and relational schemata preserving their autonomy**
  - **Achieved by storing:**
    - **Meta schema info about the DTD**
    - **The relational schema**
    - **The mapping itself**
  - **Not available a detailed syntax of mapping definitions and notation**

  - **References**

Kappel G., et al.

Towards integrating XML and relational database systems. *Johannes Kepler University Linz Technical Report July 2000.*

Schema-driven Technique (cont'd)

- **User- defined Mapping: HiT Allora**
  - **XML Schema is either DTD or W3C XML-Schema notations**
  - **Relational Schema is based on JDBC/OLEDB middleware**
  - **Mapping definitions is XML-based and loosely based on the XML Schema standard**
  - **Storage technology:**
    - **Sequence of inserts/updates ordered according to referential integrity constraints**
    - **Support auto-increment fields, XML expression, parametrical inserts**
    - **Support n+m or n*m semantics**
    - **Support abstracted, generic, DBMS independent data types**
  - **References**

Guardalben G., et al.

Integrating XML and relational database technologies: a position paper. *HiT Software, www.hitsw.com 2002.*

Materializing Relational Data as XML Documents

- **Problem: publish relational data as XML documents.**

- **XML Default Mapping:**
  - **Default XML View**
  - **IBM Xperanto**
  - **SilkRoute and RXL**

- **User-defined Mapping**
  - **Commercial Products**
    - **IBM DB2 XML Extender**
    - **HiT Allora**

XML Default Mapping

- **Default XML View**
  - **Low-level XML view of the underlying relational database**
  - **Top-level elements correspond to tables with table names as tags**
  - **Row elements are nested under table elements**
  - **Column names appear sub-element tags**
  - **Column values appear as text**
  - **Commercial products implementing the default view:**
    - **ADO (till 2.7)**
    - **Sybase XML Extensions**
    - **Oracle XSU**
    - **IBM XML Extender**
    - **HiT Allora**

XML Default Mapping (cont'd)

- **IBM Xperanto**
  - **Two phases:**
    - **structuring (i.e., organizing data hierarchically by means of a sequence of SQL queries) Top-level elements correspond to tables with table names as tags**
    - **Tagging (i.e., properly inserting XML tags based on the structured data)**
  - **Early/Late Structuring and Early/Late Tagging**
    - **Late/Late: Path Outer Union Approach eliminates redundancy of complex nested joins**
  - **Conclusions:**
    - **Construct XML inside the relational engine (when feasible)**
    - **Use the Unsorted Outer Join approach (in memory)**
    - **Use the Sorted Outer Join approach (not in memory)**
  - **References**

Shanmugasundaram J., et al.
Efficiently publishing relational data as XML documents. *VLDB Conf. Cairo, Egypt, Sep.2000, pp.65-76.*

XML Default Mapping (cont'd)

- **SilkRoute and RXL**
  - **Maps from the default XML Schema using the RXL (Relational to XML Transformation Language)**
  - **Tried three different SQL generation methods:**
    - **Using the sorted outer join (as in Shanmugasundaram)**
    - **Generate separate SQL queries (and merge them later)**
    - **A combination of the two based on the Execution plan**
  - **Execution Plan Phases:**
    - **View tree**
    - **Partitioned View tree**
    - **Partitioned SQL queries**
    - **Partitioned relations**
    - **Integrated Relations**
    - **XML Document**
  - **References**

Fernandez M. et al.

Efficient evaluation of XML middle-ware queries. *ACM Sigmod 2001, May 21-24, Santa Barbara, California, USA.*

User-defined Mapping

- **IBM DB2 XML Extender DAD**
  - **Textual mapping definitions**
  - **Based on DB2 Stored Procedures**
  - **Works only for single tables (marshal/unmarshal)**
  - **References**

  IBM DB2 XML Extender *http://www-3.ibm.com/software/data/db2/extenders/xmlext/index.html*

- **HiT Allora**
  - **SQL to materialize XML is generated automatically based on the mapping and the relational catalog**
  - **Mapping definitions trigger the usage of referential constraints to define joins/outer joins**
  - **Dynamic SQL creation can use user-defined predicates and parametrical predicates as well as scripts**
  - **Portability across multiple RDBMs**

Querying XML Views of Relational Data

- **Problem: translate XQuery statements into SQL queries and build an XML document based on XQuery templates.**

- **Schema-independent**
  - **Manolescu/Florescu/Kossmann**
  - **The Monet Project**
  - **IBM Xperanto**

- **Schema-based**
  - **CXQuery**
  - **HiT Allora**

Schema-Independent

- **Manolescu/Florescu/Kossmann translation methodology:**
  - **Query normalization: apply equivalent transformations so that translation to SQL is more direct (XQuery constructs are analyzed individually and converted)**
  - **Translate normalized query into a SQL query: using a generic, virtual, relational schema (collection of tables representing a generic XML document)**
  - **Rewrite the SQL query into the equivalent SQL query from the real data source (this is achieved by combining translated queries into a single SQL query).**

  - **References**

Manolescu I. et al.
Answering XML queries over heterogeneous data sources. *VLDB Conf., Roma, Italy 2001.*

Schema-Independent

- **The Monet XML Project :**
  - **Path expressions translate into from clauses where there sets of elements and associations.**
  - **Only elements that belong to paths stored in the database are associated to attribute and element values.**
  - **Predicates are applied either to attribute values or elements, thereby selecting among all returned elements.**
  - **The advantage to this approach is that paths are first-class citizens and do need to be computed by repeated joins.**

  - **References**

Schmidt A., et al.CWI
Efficient relational storage and retrieval of XML documents. *Proceedings of WebDB, pages 47-52, 2000.*

Schema-Independent

- **IBM Xperanto - the steps to processing an XML Query are:**
  - **XQuery Parser: returns a language neutral intermediate representation of XML queries.**
  - **Query Rewrite: resolves view references, performs view composition**
  - **Computation Pushdown: push all data and memory intensive operations down to the relational engine as SQL**
  - **SQL Translation: translates the intermediate format into SQL**
  - **XML Tagging: this is optimized for efficient in memory processing**

  - **References**

Shanmugasundaram J. et al.
Querying XML views of relational datas. *VLDB Conf., Roma, Italy 2001.*
Carey M., et al.
XPERANTO:Publishing Object-Relational Data as XML.

Schema-based

- **CXQuery:**
  - **Declarative query language based on XML Schema info**
  - **Rule based language (Datalog-style language)**
  - **Ex.: document(campus.xml) //Building(name, dept, spatial)**
  - **Supports updates**
  - **References**

  Chen Y. et al.

  CXQuery: A novel XML query language. *SSGR 2002w, L'Aquila, Italy, Jan 21 2002.*
- **HiT Allora:**
  - **Define a virtual XML Schema-based collection of relation tables (default database)**
  - **Query Normalization as in Manolescu/Florescu/Kossmann**
  - **Translate XQuery constructs into SQL queries based on the default database**
  - **Use the mapping to translate above SQL statements into real data source SQL**

Using XML for Data Integration (Mapping Technologies)

- **Major Issues:**
  - **Schema Management: when mapping heterogeneous data sets mappings are created between their schemas**
  - **Correspondences Management: to integrate data sources correspondences are made. Automating that is called schema matching.**
  - **Mapping Management: to establish a meaning for correspondences, inter-schema constraints are established. Containment constraints are established by mapping.**
- **Projects/Products:**
  - **Clio**
  - **Nimble Integration Suite**
  - **XML Global**
  - **HiT Allora**

Using XML for Data Integration (Mapping Technologies)

- **IBM Clio:**
  - **Support mapping between relational and XML Schemas as well as data translations. For XML, XQuery is supported. For relation, SQL is supported.**
  - **References**

Miller R.J. et al.

The Clio Project: Managing heterogeneity. *SIGMOD Record, 30(1):78-83, March 2001.*

- **Nimble Technology:**
  - **Nimble Integration Suite http://www.nimble.com/**
- **XML Global:**
  - **GoXML Transform 3.0 http://www.xmlglobal.com**
- **HiT Software**
  - **jAllora & winAllora 3.1 http://www.hitsw.com**

HiT Software Allora Framework for XML-to-RDBMS Integration

- **Mapping XML-to-RDBMS (Queries and DBMS Catalogues)**
- **Marshaling & Unmarshaling XML**
- **GUI Mapper**
- **Relational Storage Creation from XML Schema**
- **XML Schema Creation from Relational Catalogs**
- **Data Type Portability Across Heterogeneous RDBMSs**
- **Support for XML & SQL Expressions**
- **Support for Scripting and Parametrical Queries**
- **Support for XML QBE Queries**
- **Future:**
  - **XQuery support based on XML-Schema to RDBMS Mapping**

HiT Software XQuery Support

- **Development Steps:**
  - **a) XQuery parsing: intermediate constructs generation (Manolescu/Xperanto or others ?)**
  - **b) W3C XML Schema translation to virtual relational database**
  - **c) Normalize/Combine XQuery statement for virtual relational database**
  - **d) Choice:**
    1. **Using mapping, materialize virtual database and run SQL-from-XQuery on in-memory relational database, or**
    2. **Using mapping, generate real data source SQL queries and run them**
  - **e) Embed (tag) resulting relational rowset into XQuery XML template.**

HiT Software XQuery Support

- **Publication Opportunites:**
  - **XML Schema to relational schema translation**
  - **XQuery to virtual database translations**
  - **Performance benchmarks on choice d)**
  - **Global project**
- **Any interest in participating ?**