DATA REPRESENTATIONS AS MATHEMATICAL OBJECTS

Considering Content Compatibility of Relational & XML Data Representations

INTRODUCTION: There is currently some interest in integrated querying of XML represented data and Relational representations of data. Since traditional data access technology relies more on how data is represented then on what relationships are being represented, it is not surprising that current efforts are focused on supporting dual disparate data representations. An alternative approach is to provide a formal operation/operand environment where all data representations belong to the same clan (with set theory being the common formal ancestor) allowing all operations to apply meaningfully to all operands. The theme of this paper is to treat all data representations as mathematical objects instead of as physical structures.

INFORMATION EQUIVALENCE: Data representations that capture identical relationships are called *informationally equivalent*. Operations on informationally equivalent objects must produce informationally equivalent results. This concept is well understood and enforced in RDBMS implementations. SQL-tables with different row-orderings and/or different column-orderings are informationally equivalent when they reflect the same RDM-relation. All operations on informationally equivalent SQL-tables produce informationally equivalent results. XML-documents are somewhat less friendly.

There seems to be no consensus (nor concern) on how to determine if two arbitrary XML-documents are informationally equivalent. Since current query mechanisms of XML-documents are highly dependent on the form of the XML-document and since no path independent mechanism is available for directly accessing the information content of an XML-document, there is no direct means for determining the information equivalence of any two XML-documents or of an XML-document and a RDM-relation (or SQL-table). This places quite a development burden on hybrid XML/RDM system implementors. This burden could be alleviated if the separation of *data content* from *data representation* were formally defined.

RDM-RELATIONS: According to Codd [Co70], the term *relation* is used in its accepted mathematical sense as a set of n-tuples, the term *relationship* refers to an equivalence class of relations, and the term *Table* (an array according to Codd with labeled columns and unlabeled rows) is not an essential part of the relational view, but serves as a conceptually convenient representation of an n-ary relation. For example, the following twelve Tables all represent the same n-ary relation:

$\mathbf{T}_1 =$

A	B	C
a	b	c
x	y	z

$\mathbf{T}_2 =$

A	B	C
x	y	z
a	b	c

$\mathbf{T}_3 =$

A	C	B
a	c	b
x	z	y

$\mathbf{T}_4 =$

A	C	B
x	z	y
a	c	b

$\mathbf{T}_5 =$

B	A	C
b	a	c
y	x	z

$\mathbf{T}_6 =$

B	A	C
y	x	z
b	a	c

$\mathbf{T}_7 =$

B	C	A
b	c	a
y	z	x

$\mathbf{T}_8 =$

B	C	A
y	z	x
b	c	a

$\mathbf{T}_9 =$

C	A	B
c	a	b
z	x	y

$\mathbf{T}_{10} =$

C	A	B
z	x	y
c	a	b

$\mathbf{T}_{11} =$

C	B	A
c	b	a
z	y	x

$\mathbf{T}_{12} =$

C	B	A
z	y	x
c	b	a

All twelve Tables also represent the same unique *Data Content*, expressed set-theoretically by:

$$DC(\mathbf{T}_i) = \Big\{ \{a^{<A>}, b^{<B>}, c^{<C>}\}^{<\emptyset>}, \{x^{<A>}, y^{<B>}, z^{<C>}\}^{<\emptyset>} \Big\}.$$

XML-DOCUMENTS: Unlike structure insensitive RDM-Tables, XML-documents are distinguished by both their data content and their structural representation. Thus, where the twelve Tables above are considered identical because they have the same data content, XML-documents are only considered identical if they have the same data content and the same data structure. When multiple XML-documents contain exactly the same data content there is currently no accepted means (nor apparently any noticeable interest) in determining whether they are informationally equivalent. Following are three very simple different XML-documents, all containing the same data content as that of the twelve RDM-Tables:

XML-DOCUMENT Q1: $< Q1 >$

$$\begin{array}{l} < P > \\ \quad < A > a < /A > \\ \quad < B > b < /B > \\ \quad < C > c < /C > \\ < /P > \\ < P > \\ \quad < A > x < /A > \\ \quad < B > y < /B > \\ \quad < C > z < /C > \\ < /P > \end{array}$$

$< /Q1 >$

Set-theoretic expression for Q1:

$$\mathbf{Q}1 = \Big\{ \{a^{<1,A>}, b^{<2,B>}, c^{<3,C>}\}^{<1,P>}, \{x^{<1,A>}, y^{<2,B>}, z^{<3,C>}\}^{<2,P>} \Big\}.$$

XML-DOCUMENT Q2: $< Q2 >$

$$\begin{array}{l} < P \quad B = \text{“}y\text{”} > \\ \quad < C > z < /C > \\ \quad < A > x < /A > \\ < /P > \\ < P \quad B = \text{“}b\text{”} > \\ \quad < C > c < /C > \\ \quad < A > a < /A > \\ < /P > \end{array}$$

$< /Q2 >$

Set-theoretic expression for Q2:

$$\mathbf{Q}2 = \Big\{ \{y^{<0,B>}, z^{<1,C>}, x^{<2,A>}\}^{<1,P>}, \{b^{<0,B>}, c^{<1,C>}, a^{<2,A>}\}^{<2,P>} \Big\}.$$

XML-DOCUMENT Q3: $< Q3 >$

$$\begin{array}{l} < P \quad A = \text{“}a\text{”} \ B = \text{“}b\text{”} \ C = \text{“}c\text{”} \ / > \\ < P \quad A = \text{“}x\text{”} \ B = \text{“}y\text{”} \ C = \text{“}z\text{”} \ / > \end{array}$$

$< /Q3 >$

Set-theoretic expression for Q3:

$$\mathbf{Q}3 = \Big\{ \{a^{<0,A>}, b^{<0,B>}, c^{<0,C>}\}^{<1,P>}, \{x^{<0,A>}, y^{<0,B>}, z^{<0,C>}\}^{<2,P>} \Big\}.$$

By visual inspection it is obvious that these three XML-documents and the previous twelve RDM-Tables are mutually structurally distinct and collectively informationally equivalent. There are actually 32 distinct XML-documents with the same data content of the twelve RDM-Tables; and as the attributes and number of elements increases, visual validation of equivalent data content becomes impractical.

REAL EXAMPLES: The above examples use dummy variables for clarity of exposition, but more realistic examples can easily be generated. For example let the data content reflect the names(A), ages(B), and email addresses(C) of two people(P) with a=Alan, b=43, c=alan@abc.com and x=Mary, y=37, z=mary@qrs.org, giving for Q1:

$$\begin{array}{l}
< Q1 > \\
\quad < person > \\
\quad\quad < name > Alan < /name > \\
\quad\quad < age > 43 < /age > \\
\quad\quad < email > alan@abc.com < /email > \\
\quad < /person > \\
\quad < person > \\
\quad\quad < name > Mary < /name > \\
\quad\quad < age > 37 < /age > \\
\quad\quad < email > mary@qrs.org < /email > \\
\quad < /person > \\
< /Q1 >
\end{array}$$

Another example could be of invoices(P) reflecting product(A), cost(B), and quantity(C) with a=bicycle, b=99.99, c=3 and with x=yoyo, y=2.99, z=413, giving for Q3:

$$\begin{array}{l}
< Q3 > \\
\quad < invoice \quad product = \text{``}bicycle\text{''} \quad cost = \text{``99.99''} \quad quantity = \text{``3''} \ / > \\
\quad < invoice \quad product = \text{``}yoyo\text{''} \quad cost = \text{``2.99''} \quad quantity = \text{``413''} \ / > \\
< /Q3 >
\end{array}$$

MATHEMATICAL IDENTITY: To mechanically establish data content equivalence between any two data representations requires an operation that can distinguish the *content* from the *structure* of the data representations being compared. This, in turn, requires data representations amenable to algorithmic processing. Having the mathematical identity of any item allows that item to behave as an operand for formal manipulation. The mathematical notation introduced above does in fact capture the mathematical identity of XML-documents. It can be shown that operations do exist that can distill both the pure content and structural skeleton of any XML-document. For such operations to include relational data requires also establishing the mathematical identity for RDM-Tables.

Mathematical identities for the twelve RDM-Tables:

$$\begin{aligned}
\mathbf{T}_1 &= \Big\{ \{a^{<1,A>}, b^{<2,B>}, c^{<3,C>}\}^{<1,\emptyset>}, \{x^{<1,A>}, y^{<2,B>}, z^{<3,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_2 &= \Big\{ \{a^{<1,A>}, b^{<2,B>}, c^{<3,C>}\}^{<2,\emptyset>}, \{x^{<1,A>}, y^{<2,B>}, z^{<3,C>}\}^{<1,\emptyset>} \Big\}. \\
\mathbf{T}_3 &= \Big\{ \{a^{<1,A>}, b^{<3,B>}, c^{<2,C>}\}^{<1,\emptyset>}, \{x^{<1,A>}, y^{<3,B>}, z^{<2,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_4 &= \Big\{ \{a^{<1,A>}, b^{<3,B>}, c^{<2,C>}\}^{<2,\emptyset>}, \{x^{<1,A>}, y^{<3,B>}, z^{<2,C>}\}^{<1,\emptyset>} \Big\}. \\
\mathbf{T}_5 &= \Big\{ \{a^{<2,A>}, b^{<1,B>}, c^{<3,C>}\}^{<1,\emptyset>}, \{x^{<2,A>}, y^{<1,B>}, z^{<3,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_6 &= \Big\{ \{a^{<2,A>}, b^{<1,B>}, c^{<3,C>}\}^{<2,\emptyset>}, \{x^{<2,A>}, y^{<1,B>}, z^{<3,C>}\}^{<1,\emptyset>} \Big\}. \\
\mathbf{T}_7 &= \Big\{ \{a^{<3,A>}, b^{<1,B>}, c^{<2,C>}\}^{<1,\emptyset>}, \{x^{<3,A>}, y^{<1,B>}, z^{<2,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_8 &= \Big\{ \{a^{<3,A>}, b^{<1,B>}, c^{<2,C>}\}^{<2,\emptyset>}, \{x^{<3,A>}, y^{<1,B>}, z^{<2,C>}\}^{<1,\emptyset>} \Big\}. \\
\mathbf{T}_9 &= \Big\{ \{a^{<2,A>}, b^{<3,B>}, c^{<1,C>}\}^{<1,\emptyset>}, \{x^{<2,A>}, y^{<3,B>}, z^{<1,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_{10} &= \Big\{ \{a^{<2,A>}, b^{<3,B>}, c^{<1,C>}\}^{<2,\emptyset>}, \{x^{<2,A>}, y^{<3,B>}, z^{<1,C>}\}^{<1,\emptyset>} \Big\}. \\
\mathbf{T}_{11} &= \Big\{ \{a^{<3,A>}, b^{<2,B>}, c^{<1,C>}\}^{<1,\emptyset>}, \{x^{<3,A>}, y^{<2,B>}, z^{<1,C>}\}^{<2,\emptyset>} \Big\}. \\
\mathbf{T}_{12} &= \Big\{ \{a^{<3,A>}, b^{<2,B>}, c^{<1,C>}\}^{<2,\emptyset>}, \{x^{<3,A>}, y^{<2,B>}, z^{<1,C>}\}^{<1,\emptyset>} \Big\}.
\end{aligned}$$

The extracted data content of the three XML-documents is:

$$DC(\mathbf{Q}i) = \Big\{ \; \{a^{<A>}, b^{<B>}, c^{<C>}\}^{<P>}, \{x^{<A>}, y^{<B>}, z^{<C>}\}^{<P>} \; \Big\}.$$

The extracted data content of the RDM-Tables is:

$$DC(\mathbf{T}_i) = \Big\{ \; \{a^{<A>}, b^{<B>}, c^{<C>}\}^{<\emptyset>}, \{x^{<A>}, y^{<B>}, z^{<C>}\}^{<\emptyset>} \; \Big\}.$$

It can now be formally shown that the three XML-documents and the twelve RDM-Tables are content equivalent. In extended set theory, XST, the notation $x \in_y Q$ is read " x is a y-element of Q". The equation

$$(\forall x, i, j)\Big(x \in_{<P>} DC(\mathbf{Q}i) \iff x \in_{<\emptyset>} DC(\mathbf{T}_j)\Big)$$

asserts that for all potential elements 'x', x is a $<P>$-element of all three XML-documents if-and-only-if x is a $<\emptyset>$-element of all twelve RDM-tables.

STRUCTURAL EQUIVALENCE: Though the structural distinctions between RDM-Tables do not impart information about the n-ary relations being represented, specific structure of a RDM-Table representation must be considered when processed by a machine. The organization of data content in a XML-document is integral to its definition and use. In both cases it is desirable to be able to formally separate the *structural-skeleton* of both RDM-Tables and XML-documents. Fortunately, the mathematical identities of RDM-Tables and XML-documents already defined allow a means for extracting the respective structural-skeletons.

Since there are twelve distinct RDM-Tables and three distinct XML-documents, any operation that produces a skeletal-structure must produce fifteen distinct results.

The skeletal-structures of the twelve RDM-Tables are:

$$
\begin{aligned}
SS(\mathbf{T}_1) &= \Big\{ \; \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<1,\emptyset>}, \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_2) &= \Big\{ \; \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<2,\emptyset>}, \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<1,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_3) &= \Big\{ \; \{\emptyset^{<1,A>}, \emptyset^{<3,B>}, \emptyset^{<2,C>}\}^{<1,\emptyset>}, \{\emptyset^{<1,A>}, \emptyset^{<3,B>}, \emptyset^{<2,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_4) &= \Big\{ \; \{\emptyset^{<1,A>}, \emptyset^{<3,B>}, \emptyset^{<2,C>}\}^{<2,\emptyset>}, \{\emptyset^{<1,A>}, \emptyset^{<3,B>}, \emptyset^{<2,C>}\}^{<1,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_5) &= \Big\{ \; \{\emptyset^{<2,A>}, \emptyset^{<1,B>}, \emptyset^{<3,C>}\}^{<1,\emptyset>}, \{\emptyset^{<2,A>}, \emptyset^{<1,B>}, \emptyset^{<3,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_6) &= \Big\{ \; \{\emptyset^{<2,A>}, \emptyset^{<1,B>}, \emptyset^{<3,C>}\}^{<2,\emptyset>}, \{\emptyset^{<2,A>}, \emptyset^{<1,B>}, \emptyset^{<3,C>}\}^{<1,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_7) &= \Big\{ \; \{\emptyset^{<3,A>}, \emptyset^{<1,B>}, \emptyset^{<2,C>}\}^{<1,\emptyset>}, \{\emptyset^{<3,A>}, \emptyset^{<1,B>}, \emptyset^{<2,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_8) &= \Big\{ \; \{\emptyset^{<3,A>}, \emptyset^{<1,B>}, \emptyset^{<2,C>}\}^{<2,\emptyset>}, \{\emptyset^{<3,A>}, \emptyset^{<1,B>}, \emptyset^{<2,C>}\}^{<1,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_9) &= \Big\{ \; \{\emptyset^{<2,A>}, \emptyset^{<3,B>}, \emptyset^{<1,C>}\}^{<1,\emptyset>}, \{\emptyset^{<2,A>}, \emptyset^{<3,B>}, \emptyset^{<1,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_{10}) &= \Big\{ \; \{\emptyset^{<2,A>}, \emptyset^{<3,B>}, \emptyset^{<1,C>}\}^{<2,\emptyset>}, \{\emptyset^{<2,A>}, \emptyset^{<3,B>}, \emptyset^{<1,C>}\}^{<1,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_{11}) &= \Big\{ \; \{\emptyset^{<3,A>}, \emptyset^{<2,B>}, \emptyset^{<1,C>}\}^{<1,\emptyset>}, \{\emptyset^{<3,A>}, \emptyset^{<2,B>}, \emptyset^{<1,C>}\}^{<2,\emptyset>} \; \Big\}.\\
SS(\mathbf{T}_{12}) &= \Big\{ \; \{\emptyset^{<3,A>}, \emptyset^{<2,B>}, \emptyset^{<1,C>}\}^{<2,\emptyset>}, \{\emptyset^{<3,A>}, \emptyset^{<2,B>}, \emptyset^{<1,C>}\}^{<1,\emptyset>} \; \Big\}.
\end{aligned}
$$

The skeletal-structures of the three XML-documents are:

$$
\begin{aligned}
SS(\mathbf{Q}1) &= \Big\{ \; \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<1,P>}, \{\emptyset^{<1,A>}, \emptyset^{<2,B>}, \emptyset^{<3,C>}\}^{<2,P>} \; \Big\}.\\
SS(\mathbf{Q}2) &= \Big\{ \; \{\emptyset^{<0,B>}, \emptyset^{<1,C>}, \emptyset^{<2,A>}\}^{<1,P>}, \{\emptyset^{<0,B>}, \emptyset^{<1,C>}, \emptyset^{<2,A>}\}^{<2,P>} \; \Big\}.\\
SS(\mathbf{Q}3) &= \Big\{ \; \{\emptyset^{<0,A>}, \emptyset^{<0,B>}, \emptyset^{<0,C>}\}^{<1,P>}, \{\emptyset^{<0,A>}, \emptyset^{<0,B>}, \emptyset^{<0,C>}\}^{<2,P>} \; \Big\}.
\end{aligned}
$$

Above are 15 of the 44 possible structures that can be used to capture 2 elements with 3 attributes. All 12 of the possible RDM-Tables are represented, but only 3 of the possible 32 XML-documents are represented.

STRUCTURE PROLIFERATION: The proliferation of possible data modeling representations for a single real-world relationship grows astronomically very quickly as the number of elements and their attributes increase. The number of distinct RDM-Tables representing the exact same data content is expressed in terms of n elements and k attributes by: $T(n,k) \; = \; n! \times k!$.

The number of distinct XML-documents representing the exact same data content grows more abruptly and is expressed in terms of n elements and k attributes by:

$$X(n,k) \; = \; n! \; \times \; \left(\; k! \; + \; \sum_{i=1}^{k-1} \left(\binom{k}{i} \times (k-i)! \right) + \; 1 \; \right) \; and \; \; k \geq 2.$$

For any application involving data with n elements each with k attributes, the total number of traditional data modeling considerations available is $T(n,k) + X(n,k)$. Thus a committee decision or standardized canonical representation is required for optimum operation and data sharing. Since the mathematical identity of any relationship is unique, and since there exists at least one set-theoretic formulation that captures any and all relationships possible to be captured by RDM or XML representations, only one XST representation is required for any specific (n,k)-relationship. The set-theoretic data modeling alternative only requires recognition of the mathematical identity of any given data representation, thus the number of set-processing modeling choices for any given (n,k)-relationship reduces to exactly one.

ALTERNATE XST REPRESENTATIONS: Though the above XST representation of RDM and XML structurings demonstrate the XST ability to capture the respective information content and structural skeleton, the XST representation is not rich enough to adequately support desired interrogation operations on XST representations.

```
<BIB>
   <BOOK  YEAR="y1">
      <TITLE>
         t1
      </TITLE>
      <AUTHOR>
         <LAST>
            l1
         </LAST>
         <FIRST>
            f1
         </FIRST>
      </AUTHOR>
      <EDITOR>
         <LAST>
            l2
         </LAST>
         <FIRST>
            f2
         </FIRST>
         <AFFILIATION>
            a1
         </AFFILIATION>
      </EDITOR>
      <PUBLISHER>
          p1
      </PUBLISHER>
      <PRICE>
          $1
      </PRICE>
   </BOOK>
</BIB>
```

The above XML-schema is intended to be a conceptual representation capturing relationships between properties of books. The book properties being related are: YEAR of book publication, TITLE of book, AUTHOR of book where there may be more than one, EDITOR of book, PUBLISHER of book, and PRICE of book. Some of these properties themselves have properties. LAST for the last name of an author or last name of an editor, FIRST for the first name of an author or first name of an editor, and AFFILIATION for the affiliation of an editor.

Any proper choice for an XST representation of an XML-document must allow set-theoretic operations on that representation to extract the properties of any relationship or sub-relationship. For example, given a relationship representation for BOOK there needs to be defined a corresponding operation that extracts all properties represented as valid for BOOK. Assume that 'PROPS' is just such an operation, then for the representation above the operation on BOOK, provided BOOK is a legitimate relationship, must yeild:

$$\mathbf{PROPS}(BOOK) = \{ YEAR,\ TITLE,\ AUTHOR, EDITOR,\ PUBLISHER,\ PRICE \}.$$

For the operation $\mathbf{PROPS}(BOOK)$ to be of any formal value, it has to be known that *BOOK* is indead a legitimate XST expression for an XML representation of a relationship. Assume that 'RSHIP(x)' has been formally defined to determine that 'x' is a legitimate XST expression for an XML representation of a relationship, then:

$$\mathbf{RSHIP}(BIB) \implies \mathbf{PROPS}(BIB) = \{ BOOK \}$$
$$\mathbf{RSHIP}(BOOK) \implies \mathbf{PROPS}(BOOK) = \{ YEAR,\ TITLE,\ AUTHOR, EDITOR,\ PUBLISHER,\ PRICE \}$$
$$\mathbf{RSHIP}(AUTHOR) \implies \mathbf{PROPS}(AUTHOR) = \{ LAST,\ FIRST \}$$
$$\mathbf{RSHIP}(EDITOR) \implies \mathbf{PROPS}(EDITOR) = \{ LAST,\ FIRST,\ AFFILIATION \}$$

CONCLUSION: This paper is an introduction to an extensive examination of the position that systems designed to efficiently support the integrated querying of XML represented data along with Relational representations of data having a homogeneous formal foundation for the representation and manipulation of data could aid in the design, implementation, and operation of systems for storing and processing both Relational and XML data.

References

[Ch68] Childs, D. L.: *Feasibility of a Set-Theoretic Data Structure: A General Structure Based on a Reconstituted Definition of Relation*, Proc. IFIP Congress 1968

[Ch77] Childs, D. L.: *Extended Set Theory: A General Model for Very Large, Distributed, Backend Information Systems*, Third International Conference On Very Large Databases, Tokyo, Japan, 1977

[Ch86] Childs, D. L.: *A Mathematical Foundation For Systems Development*, NATO ASI Series, Vol F24, Database Machines, Edited by A. K. Sood and A. H. Qureshi, Springer-Verlag, 1986

[Co70] Codd, E. F.: *A Relational Model of Data for Large Shared Data Banks*, CACM 13, No. 6 (June) 1970, [p. 379-380]

[Sk57] Skolem, Thoralf: *Two Remarks on Set Theory,* Mathematica Scandinavica 5 (1957), p.43-46.

[Su60] Suppes, Patrick: *Axiomatic Set Theory*, Van Nostrand, 1960, p.20 & 141.

[W3C] W3C: *XQuery Use Cases*, xxx, 2006, p.20 & 141.