



Departamento de Ciencias de la Computación

UNIVERSIDAD DE CHILE



# XML Databases

## III. Inside databases

B. Piwowski

# Part III: inside XML databases

## ■ Outline

### 1. Introduction

Motivations, influences and problematics

### 2. Storing XML documents

Storing, updating, normalisation and integrity

### 3. Retrieving XML documents

Indexes and evaluation algorithms

## ■ Main sources

☐ R. Bourret web site

☐ XQuery from the experts

# A. Motivations

## Why structured data?

- Data can contain fields not known at design time.
- Data is self-describing.
- The same kind of data may be represented in multiple ways.
- Data may be sparse.
- Natural form for document-centric data

# A. Motivations

Why a database?

- Ease of management
  - Integrity constraints
  - independence from storage
- Enhanced query performance
- Transactional safety (ACID)
- Security
- Managing huge quantities of data
- ...

# A. Motivations

Why an enabled XML DB?

- Enabled XML Database
  - XML is only an “exchange” format
  - Construction of wrappers
  - **Fixed schema**
- XML data-model
  - Storage of XML documents
  - XML-aware full text search
- Structured query languages like Xquery
- Handling large documents

# A. Motivations

Native XML DB (According to the XMLDb mailing list)

- Defines a (logical) model for an XML document (as opposed to the data in that document) and stores and retrieves documents according to that model.
- Has an XML document as its fundamental unit of (logical) storage
- Is not required to have any particular underlying physical storage model.

# A. Motivations

## Why a native XML Database?

- Avoid some bottlenecks
  - A simple XQuery might involve numerous joins
- More flexibility in schema evolution
  - Schemas can be changed
  - Unknown version of a schema
- Links, versioning

# A. Motivations

## XML Applications

- Le Monde (Xyleme Zone Server): >800000 documents, 6 gigabytes
- Flight information (Schiphol Airport in Amsterdam) uses Tamino to integrate data from more than 38 systems in real time.
- Customer profiles
- Sedna: complete web site



# B. Influences on design

## Typical queries

- Queries on text only
  - ☐ Includes keyword, stemming, proximity search
- Queries on text and structure
  - ☐ Content constraints
  - ☐ Structure constraints
- Queries that span structure
  - ☐ Structure might be “superfluous”
  - ☐ User might not know that (or don't want to)

# B. Influences on design

## Two types of documents

- Regular (data-centric)
  - Resembles relational data
  - Regular structure
  - Scalar values
- Mixed (document-centric)
  - Flexible structures
  - Arbitrary depth
  - Sparse data

## B. Influences on design

### Two types of documents (examples)

```
<SalesOrder SONumber="12345">
  <Customer CustNumber="543">
    <CustName>ABC Industries</CustName>
    <Street>123 Main St.</Street>
    <City>Chicago</City>
    <State>IL</State>
    <PostCode>60609</PostCode>
  </Customer>
  <OrderDate>981215</OrderDate>
  ...
</SalesOrder>
```

## B. Influences on design

### Two types of documents (examples)

```
<FlightInfo>
  <Airline>ABC Airways</Airline> provides
<Count>three</Count>
  non-stop flights daily from
<Origin>Dallas</Origin> to
  <Destination>Fort Worth</Destination>.
Departure times are
  <Departure>09:15</Departure>,
<Departure>11:15</Departure>,
  and <Departure>13:15</Departure>. Arrival
times are minutes later.
</FlightInfo>
```

# B. Influences on design

## Two types of documents (examples)

```
<Product>
```

```
  <Intro>
```

```
    The <ProductName>Turkey Wrench</ProductName>  
    from <Developer>Full
```

```
    Fabrication Labs, Inc.</Developer> is
```

```
  <Summary>like a monkey wrench,  
  but not as big.</Summary>
```

```
  </Intro>
```

```
  <Description><Para>The turkey wrench, which  
  comes in <i>both right- and left-  
  handed versions (skyhook optional)</i>, is  
  made of the <b>finest  
  stainless steel</b>. The REDI-grip rubberized  
  handle quickly adapts  
  to your hands, ...
```

# B. Influences on design

## Properties of XML Data and Queries

- Attributes vs. elements
- Heterogeneity: documents can vary in structure even with the same schema
- Identity and structure
- Structure dependence
- Flexibility in mixed content
- Presence of a schema
- Round tripping

# B. Influences on design

## Misc

- Data integration
- XML Data and Web are tightly related
  - Local = efficiency
  - Distributed = up-to-date
- PDOM (Persistent DOM)
  - The DOM tree returned is “live”
  - Similar to one of the roles of object databases
- Content Management System

# C. Problematics

## ■ Representation

- ☐ How to store (speed, size, access)?
- ☐ How to update?

## ■ Schema design (normalisation)

## ■ Transactions: AC – Isolation - D

## ■ Querying

- ☐ How to evaluate e.g. XQuery expressions?
- ☐ What are the appropriate index/representation?



# C. Problematics

## Storing documents

- “Enabled” relational database
  - From XML schemas to relational schemas
  - Wrappers
- “Native” database
  - Text-based
  - Model-based
    - Relational storage: how to capture identity, structure and order?
    - Compressed representation

# C. Problematics

## Normalisation

- Normalisation = do not duplicate the information
- Relational databases
  - Functional dependencies (1-3NF, BCNF)
  - Multivalued dependencies
- XML and normalisation
  - How to extend relational concepts?
  - How to normalise a schema?

# C. Problematics

## Transactions and security

- Relational databases
  - Nothing to do!
- Other native XML databases
  - Durability, Consistence, Atomicity
  - Isolation
- Security
  - How to restrict access?

# C. Problematics

## Query evaluation and indexes

- Evaluating query plans
  - How to predict selectivity of operators?
  - Optimisation
- Indexes
  - Uni-dimensional: structure, content
  - Multidimensional indexing