Chapter 15

by
David G. Messerschmitt

Databases

• Treat data as a separate asset
  – May be shared by multiple applications
• Provide protection and integrity features appropriate to mission-critical data
  – Access control
  – Integrity constraints
  – Persistence
  – etc.

Relational table

<table>
<thead>
<tr>
<th>Table</th>
<th>Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Address</td>
</tr>
<tr>
<td>Dept</td>
<td></td>
</tr>
<tr>
<td>Record</td>
<td></td>
</tr>
</tbody>
</table>

Database operations

Each operation results in a new table, so they can be concatenated
**SQL interface**

- Presents single abstract interface to the application logic
- Standardized, not vendor specific
  - “Stored procedures” and “extensions” violate this
- Encapsulates various internal details
  - Data partitioning and replication
  - Host mapping
  - File representation
  - etc.

**Object/table correspondence**

- **Class implementation**
  - **Record-at-a-time program**
  - **Object instance data**

**Object attributes**

- **Rows can be considered object instances with the same attributes**
- **Restriction to simple data types**
- **No encapsulated state**
- **No methods**
Object-relational database

- A column can store object instances of a given class rather than data of a given simple or compound data type.
- Because of the table structure, SQL can be extended to this case.
- Standard SQL queries can be extended to methods returning simple data types.
- Many other good ideas.

Benefits of ORDBMS

- Extension: manage arbitrarily complex data types
- Migration: preserve and extend existing databases
- Preserve SQL interface
  - OR extensions in latest standard
- All the benefits/experience of earlier databases
  - Access control, data integrity, persistence, etc.
- Killer app: Behind Web/CGI
  - Images, video, audio, animation, applets, etc.

Definition

- A markup language describes the structure of a document
  - Based on tags
  - Tags denote structural elements like sections, subsections, figures, etc.
- Internationally standardized, so application independent

Example: HTML

```html
<HEAD>
  <META NAME="GENERATOR" CONTENT="Adobe PageMill 2.0 Win">
  <TITLE>David G. Messerschmitt Homepage</TITLE>
</HEAD>
<BODY ALINK="#c4124e" BACKGROUND="Images/background.jpg">
  <H3>&nbsp;</H3>
  <BLOCKQUOTE>
    <H2><TABLE BORDER="5" CELLSPACING="2" CELLPADDING="0">
      <TR>
        <TH>  
          <P ALIGN=LEFT><IMG SRC="Images/Messer.gif" WIDTH="112" HEIGHT="168" ALIGN="BOTTOM" NATURALSIZEFLAG="3"></TH>
      </TR>
    </TABLE>
    <H2><BR>
    David G Messerschmitt</H2>
    <H3>Roger A. Strauch Professor of <A HREF="http://www.eecs.berkeley.edu/">Electrical Engineering and Computer Sciences</A></H3>
  </BLOCKQUOTE>
</BODY>
</HTML>
```

Family lineage

- SGML
  - Emphasizes formatting and presentation of documents
- HTML
  - Emphasizes structure of documents
- XML
  - Purpose- and industry-specific extensions
Data sharing among applications

• Options include:
  – Messages with defined formats
  – Documents (e.g. XML)
  – Shared databases
  – Remote method invocation middleware

• The first two are the most practical for inter-enterprise applications