INF312 - Advanced Database Systems
Semester Summary, Fall 2002

Contents
A run-through of the lecture themes with focus on the essentials
- Requirements imposed upon DBS technology over time
- Beyond RDBMS’ (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Theme 1
Requirements imposed upon DBS technology over time
- Beyond RDBMS’ (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Comments, questions …
### 3-Step Historical View

**Storage, Retrieval and Exchange Technologies**

- **Step 1:**
  - Own (separate) data management scheme for each application.
  - Not feasible. Also commonalities discovered. Enter DBMS.

- **Step 2:**
  - Lucky strike: Edgar F. Codd introduces and practically establishes the relational DB approach and relational algebra in one go (1970 - 74).
  - Easy, formal and relationally complete.
  - Addresses classical applications with classical requirements.

- **Step 3:**
  - The OO paradigm is introduced (1967) and popularized (1989-90).
  - New applications arise, imposing new requirements. RDBMS’ become insufficient, too restrictive...
  - No longer “data” but “objects” are stored, retrieved and exchanged.

### 3-Step Comparison

**Requirements upon DBS’ of Classical and New Applications**

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Classical</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Structures &amp; Operations</strong></td>
<td>Simple, small &amp; many Generic</td>
<td>Complex, large and few User-defined</td>
</tr>
<tr>
<td><strong>Transactions</strong></td>
<td>Small/simple objects read/modified simply, long, concurrent but not cooperative</td>
<td>Large/complex objects processed complexly, long, concurrent and highly cooperative</td>
</tr>
<tr>
<td><strong>Integrity constraints</strong></td>
<td>DB states consist of small &amp; simple structures, State transitions via txs or generic operations, constraints on DB states</td>
<td>DB states consist of large &amp; complex structures, state transitions also via arbitrary event sequences, arbitrary conditions on state</td>
</tr>
</tbody>
</table>

**Common**

- Persistence management, consistency, ad-hoc queries, …
3-Step Change in Technology
Requirements upon DBS' of the Technical Environment

- Technical environment: Non-stop improvement
  - More power, more intelligence, more mobility, high cooperation etc., encouraging complexity at application, service and base-system levels

- The Internet: A serious challenge and many possibilities
  - Global, very high distribution/heterogeneity and need for integration, availability (7x24), scalability, security, ...
  - With respect to data/information and related operations: More reads than writes, more search-dependent content, ...

- Architecture: Implementing extensibility, scalability etc.
  - From monolithic to component based (CB) architectures
  - CB architecture advantages are obvious, but needs more coordination, management, standardization etc.

INF312 - Advanced Database Systems
Theme 2

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining
Definition: Paradigm, Model

DBS Binoculars to Perceive and Model the World By

- The approach of choice by which the world is modeled is the “paradigm”

- The capabilities as well as the limitations of a DBS is dictated by the paradigm

Database Systems

Alternative DBS Paradigms (Models)

- File-based (files and file-systems)
- Hierarchical
- Networked
- Relational
- Object-Oriented
- Cross-breeds, Extensions and Persistence Services
- Object-Relational (or Extended Relational)
- Real-Time
- Multi-Media
- Document-based

- New/alternative transaction concepts
- Change/version management
- New (data) models, formalisms, languages
- New, better and multi-paradigm DB Management Systems (DBMS) …
Database Systems
Definition, Manipulation and Query Languages

**DEFINITION**
- DDL: Data Definition Language for defining data (schema).
  The CREATE sentence
- ODL: Object Definition Language, the OO counterpart for defining (declaring) objects (classes). ODL is the “schema” language of OO-DBS.

**MANIPULATION**
- DML: Data Manipulation Language for processing/transforming data.
  UPDATE, DELETE etc. sentences, GROUP BY etc. clauses.
  Not computationally complete.
- OML: Object Manipulation Language, the OO counterpart for manipulating objects.
  Programming language binding. Computationally complete.

**QUERY**
- SQL is the name of one specific relational language incorporating data definition, manipulation and querying
- The querying part of SQL is represented by the SELECT sentence
- OQL: Object Query Language, the OO counterpart for querying for objects (collections of them)

INF312 - Advanced Database Systems
Theme 3

- Requirements imposed upon DBS technology over time
- Beyond RDBMS’ (OO-DBS, OR-/ER-DBS, Document DBS)

**Standardization (OO, OMG, ODMG, SQL-99)**
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining
Object Management Group (OMG)
The OO Bases of OO-DBS

- OMG – Existed before ODMG (Object Data Management Group). Standardized the Common Object Request Broker (CORBA) as well as IDL as part of the effort. See: http://www.omg.org/.

- IDL – Interface Definition Language. Basis for ODMG’s ODL.

- More recently…
  - Standardized UML (Unified Modeling Language),
  - … CWM (Common Warehouse Metamodel),
  - … MOF (Meta Object Facility),
  - … XMI (XML Metadata Interchange),
  - … and initiated the MDA (Model Driven Architecture) effort
  - … and some more (Persistent State Service).

- Has its “own” choice of object model that ODMG builds upon…

The OO Paradigm
Central OO Concepts #1

The OO paradigm offers
- **Classification** – types/classes (user definable, nested); Conceptually with respect to “classical” categorization theory (changing)
- **Encapsulation** – complete, write encapsulation and partial encapsulation
- **Polymorphism** – overloading/overriding, late binding

All objects have
- **Identity** – permanent, immutable and non-reusable identity (OID)
- **State** – i.e., they “remember” through attributes (changing)
- **Behavior** – i.e., they “act” through methods

Objects associate with each other by
- **Exchanging messages** through a link between objects and via interfaces of involved objects
- **Inheritance** – sub-tying/ super-typing, “IS_A”; overriding
- **Aggregation** – composition, containment
The OO Paradigm

Central OO Concepts #2

- **Literal** – Is an object too, but without an OID: A structure for capturing complex values otherwise.

- **Values and Equality** – Same public values (shallow equality), same values regardless (deep equality), same object (equivalence, being “identical”).

- **Collections** – Was already around with Smalltalk (and later C++) before ODMG. There are 5 of them: Set, Bag, List, Array, Dictionary. Used extensively (also) in DBS, especially in managing data-sets (sets of objects).

- **Intension and Extension** – Intension is the definition (class, schema, in a way “code template”) of all possible objects (instances), whereas extension is the collection (set) of actual instances.

Object Data Management Group (ODMG)

Standardizing OO-DBS


- Object Management Architecture (OMA) and Object Data Model (builds upon OMG’s Object Model)
  - Objects with OIDs and literals without, as before
  - An object’s attributes and relationships to other objects are properties that make up the object’s state; Operations are properties as well, and make up the behavior of the object.
  - Objects are instances of types within a super- and sub-type hierarchy; Type of object is known at creation (and does not change); Multiple super-types are allowed, and super-types must be specified explicitly (can not be deduced through signature compatibility).
  - Operations are defined on a single type, are invoked, may have side-effects and are implemented by the methods of the type.
  - NOT INCLUDED: Versions, realization/implementation standardization or specification, distributed systems, transaction mechanisms and other processing aspects, rules etc.

- **Object Specification Languages:**
  - **ODL** (Object Definition Language), based upon OMG’s IDL
  - **OIF** (Object Interchange Format)
  - **OQL** (Object Query Language), based upon SQL (as much as possible)
  - **Language Bindings:** ODL, OML and OQL for C++, Smalltalk and Java
Object-Relational DBS
SQL-99 or SQL-3 (SQL, ISO/IEC 9075-n, 1999)

- ISO/IEC SQL 1999 standards are in many documents, and they cost. Go to http://www.iso.ch/iso/en/ISOOnline.frontpage and search for ‘SQL’ and ‘standards’ to see a list of them (16 documents).
- SQL-99 attempts to address the same requirements that OO-DBS’ have aimed at addressing, but based upon SQL instead (i.e., not from scratch)
- SQL-99 offers:
  - Large objects (BLOBs and CLOBs)
  - Richer types: New basic types, user defined types/ADTs, structured and reference types, distinct types
  - Inheritance, overloading (overriding) of super-type methods
  - Nested types (aggregates)
  - Some amount of encapsulation (inclusion of ADT-methods)
  - Collections and related operations
  - New predicates (SIMILAR, UNIQUE, …)
  - Recursive queries
  - Standardized triggers
  - Improved (and standardized) access control (DCL)

XML and Document DBS
Semi-Structured Databases

- Based upon ISO-standard SGML. To understand the full implication of XML, see (at least): http://www.w3.org/ (and click on XML), http://www.xml.org, http://www.oasis-open.org, http://www.hr-xml.org/channels/home.htm and others...
- Characteristics, advantages and uses
  - With XML, one can define “document types” and schemas
  - One can in principle “structure” data and tell the way it is structured also (metadata), making it ideal for describing and interchanging structured as well as semi-structured data, including objects (where the object’s properties are the structure)
  - Data can be stored as XML documents, DTD and XML Schema provide for schemas, there are a number of query languages and programming interfaces, but...
- Lacks, disadvantages and misuses
  - There is no data integrity, transactions, multi-user access (or access control otherwise), security, indexing, queries across multiple documents
  - XML is hierarchical (back to 60s and the hierarchical DBS)
  - Far too much knowledge – also for constructing, storing and retrieving data – in the application (almost back to square 1 of the DB era)
INF312 - Advanced Database Systems

Theme 4

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)

Active DBS

- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

---

Rules in DBS

Active Database Systems

- What if we wanted to
  - monitor the projects/resources database
  - to check for arrival of unexpected (new) projects,
  - and – if management approval and funding existed,
  - hire in one or more consultants?

- We could check manually at regular intervals, or write a program that polls the DB at regular intervals, or…

- Acquire an Active DB:
  Implies writing in a “rule” that instructs a certain action to be triggered by some event and if certain conditions hold.

- Remember: Event-Condition-Action (ECA) triplets make up the rules.
Requirements imposed upon DBS technology over time
Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
Standardization (OO, OMG, ODMG, SQL-99)
Active DBS

Transaction Management

- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Transaction Management
From Classical to Modern Transactions

- ACID properties of “classical” transaction management:
  - Atomic… Indivisible between states (before/after transaction)
  - Consistent… Produce consistent results or abort
  - Isolated… As if all alone
  - Durable… Result is lasting once transaction is successful

- Classical transactions are typically short

- What happens if we have to deal with long transactions, and have to weaken some or all of the ACID requirements?
Transaction Management
Other (Modern) Transaction Types

- Flat TX with savepoints: Save, not commit. Controlled roll-back to any saved point, but abort returns to starting state.
- Chained TX: Several sub-commits, roll-back/abort to last commit.
- Nested TX: TX within TX, functionally decomposed
  - Closed
  - Open
- Multi-Level TX: TX within TX, several pre-decided levels of abstraction
  Uses compensation (not delete but cancel with a new cancel-TX)
- Distributed TX: Flat TX that runs on a distributed environment
- Long TX:
  - Mini-batch – TX split into shorter TX sequences under program control
  - Sagas – Extended chains, uses compensation
  - Cooperative TX – TXs co-operate to view each others’ partial results.
    Example: Check-in/Check-out
    APOTRAM.

INF312 - Advanced Database Systems
Theme 6

- Requirements imposed upon DBS technology over time
- Beyond RDBMS’ (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
  - Heterogeneous/Federated/Multi-DBS
  - Data Warehouse
  - Change Management
  - XML in Data Management and Data Exchange
  - Multimedia DBS, Digital Libraries and WWW Applications
  - Data Mining
Distributed DBS

Distributing the Processed and the Processing

- Distribution offers enhanced performance, more data volume & extensibility/scalability... Ask yourself:
  - What is it that can be/should be distributed?
  - Data? Processing of data? Both?

- Data distributed:
  - Distinct "rows" of same "table" across the network? Distinct "columns" of same "table"? Same data replicated?
- Processing distributed:
  - Logging/recovery, locking/concurrency control, transaction management, sorting/indexing, access control, cache/buffer management? Application-controlled processing of data? All?

Parts of DB management can and should also be distributed...
Parts may need to stay centralized.

Distributed DBS

Three Client-Server Architectures

- Three C/S architecture alternatives:
  - Object Server, Page Server, File Server

<table>
<thead>
<tr>
<th>Page &amp; File Server</th>
<th>Object Server</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>▶ Simple server design</td>
<td>▶ Complex server design</td>
</tr>
<tr>
<td>▶ Complex client design</td>
<td>▶ Simpler client design</td>
</tr>
<tr>
<td>▶ Fine-grained concurrency control difficult</td>
<td>▶ Fine-grained concurrency control feasible</td>
</tr>
<tr>
<td>▶ Very sensitive to client buffer pool size</td>
<td>▶ Less sensitive to client buffer pool size</td>
</tr>
<tr>
<td>▶ Very sensitive to clustering</td>
<td>▶ Reduces data movement, relatively insensitive to clustering</td>
</tr>
</tbody>
</table>
**Distributed DBS**

**Problem Areas**

- Distributed DB design
- Distributed directory/catalogue management
- Distributed query processing and optimization
- Distributed transaction management
  - Distributed concurrency control
  - Distributed deadlock management
  - Distributed recovery management
- Remember quorums!
  - Coordinator coordinated majority votes.
  - Used in concurrency control, commit/abort, termination and recovery protocols.

---

**INF312 - Advanced Database Systems**

**Theme 7**

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS

**Heterogeneous/Federated/Multi-DBS**

- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining
Heterogeneous-/Federated-/Multi-DBS
The Need and the Solution

What Does (Should) the Integration Layer Provide?

- Global data-model
- Global schema and meta-data management
- Global, distributed transaction management
- Global, consistent recovery

- Support for global/distributed DDL, DML, …
- … and DQL, of course (distributed/global query processing/optimization)

- Distribution transparency (transparent integration of the DBSs/DBAs)
- Extensibility

- Tools, techniques (always forgotten), for example for (local) schema homogenization, export/integration and global schema construction
INF312 - Advanced Database Systems
Theme 8

- Requirements imposed upon DBS technology over time
- Beyond RDBMS’ (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS

Data Warehouse

- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Data Warehouse
The Value and Whereabouts of Information in Data

- Large DB, storing (tera-bytes of mostly static) data from multiple sources
- For generating information, i.e., for
  - Decision Support,
  - On-Line Analytical Processing,
  - Data Mining etc.

Summary Table
  (data-cube, multi-dim.)

Dimension Table
  (attr. of one dim. of FT)

Fact Table
  (timed for validity)

4-Step Life Cycle

- Global Schema Definition & Design
- Data Extraction & Loading
- Query Processing
- Data Update

Extract & clean data, materialize views and measures, store in DW

Roll-up, Drill-down, Pivot/Rotate,
Slice/Dice with Data-Blade
Sort, Select, Derive (attributes/new queries)

Monitor/track data sources, refresh DW
(creating diffs & deltas)
INF312 - Advanced Database Systems

Theme 9

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse

Change Management

XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Change Management
A World in Parts, Versions and Configurations

- Objects, parts, schema are versioned,
  - either because different configurations are required,
  - or because of collaborative work, where access to same object/part is necessary
  - or because of the need for modifications/evolutions (for example on schema) while ensuring “backwards compatibility”
- Workspaces are (often individual) areas for keeping ‘own’ copies/versions (usually ‘checked-out’ prior to work, and ‘checked-in’ after work)
- A configuration is selection of constituent versioned objects/parts
- Other kinds of versions: Revisions, alternatives, variants, representations (equivalences)
- Conflict resolution, for example on merging different versions of same object (for example due to parallel modifications on the same object)
INF312 - Advanced Database Systems

Theme 10

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management

XML in Data Management and Data Exchange

- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

XML in Data Management and Data Exchange

The Conveyor Belt of Data in the WWW Age

- Allows for interchange and interpretation of structured and semi-structured data
- XMI (XML Metadata Interchange adopted by OMG) is one example
- Note: Remember the concept of a “namespace”

- XML is hierarchical

- See XML in theme 3, “XML and Document DBS”, slide 16
Theme 11

Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange

Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Multimedia DBS (+Digital Libraries and the WWW)
The Art of Exact Copying

- The major issue in multimedia (for example in transmitting MM data) is the issue of copying the source to the destination as truthfully as possible, while maintaining full control of the data so as to be able to manipulate the data in various ways
- MMDBS offers (or should offer) support for:
  - "Almost" real-time storage/retrieval and processing
  - Temporal concepts
  - Representing and processing various data types uniformly
  - Representing and processing large amounts of data uniformly
  - Managing various data storage devices/units, tertiary storage, multi-level storage uniformly
  - Abstract operations on MM data
  - Storage and processing parallelism
  - Distribution/synchronization
Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications

**Data Mining**

Querying for What You Don’t Know is There

- Extraction/discovery of potentially useful (implicit) information form existing data (for example from a Data Warehouse): Knowledge Discovery in Databases (KDD)
- **OLAP**: On-Line Analytical Processing (estimation/planning, discovery of multi-dimensional data relationships)
- Data mining techniques require a good mastery of statistical/analytical techniques (statistical/mathematical modeling and a good deal of AI techniques)
  - Neural Networks, Training & Mining, Genetic Algorithms, Bayesian Statistics, Regression Analysis, Pattern Discovery ...
- DBS support is as for a “Programmable Data Warehouse”
- See also theme 8, Data Warehouse, slide 30
INF312 - Advanced Database Systems
Theme 12

- Requirements imposed upon DBS technology over time
- Beyond RDBMS' (OO-DBS, OR-/ER-DBS, Document DBS)
- Standardization (OO, OMG, ODMG, SQL-99)
- Active DBS
- Transaction Management
- Distributed DBS
- Heterogeneous/Federated/Multi-DBS
- Data Warehouse
- Change Management
- XML in Data Management and Data Exchange
- Multimedia DBS, Digital Libraries and WWW Applications
- Data Mining

Comments, questions...

Comments
On Exam Style

- List up and then explain!
- Stay in dialogue!
- Draw, demonstrate!

- And good luck!

- Questions ???