



Προηγμένα Θέματα Βάσεων Δεδομένων

Σύγχρονες Τάσεις και Εφαρμογές

Γιάννης Θεοδωρίδης, Νίκος Πελέκης

InfoLab, Τμήμα Πληροφορικής, Πανεπιστήμιο Πειραιά

<http://infolab.cs.unipi.gr>

version: Jan. 2009

Περιεχόμενα



- Το χθες...
- ... το σήμερα ...
- ... και το αύριο των Βάσεων Δεδομένων
- Αναλυτικότερα θα δούμε μη παραδοσιακούς τρόπους σχεδίασης ΒΔ...
 - πώς υποστηρίζουμε αντικείμενα στις ΒΔ
 - τι είναι μια XML ΒΔ
 - πώς εκτελούμε πολυδιάστατη ανάλυση με στόχο τη λήψη αποφάσεων
- ... και θα συνεχίσουμε με σύγχρονες τάσεις και εφαρμογές ΒΔ όπως:
 - Χρονική, χωρική, πολυμεσική (εικόνες κ.α.) πληροφορία
 - Εν κινήσει και προσωπικές ΒΔ, ρεύματα δεδομένων, ...

Ευχαριστίες: μέρος του υλικού βασίζεται σε διαφάνειες των Τ. Σελλή (ΕΜΠ),
Ι. Ιωαννίδη (ΕΚΠΑ), Π. Βασιλειάδη (Παν/μιο Ιωαννίνων)

Ιστορία των ΒΔ (1)



- Late 60's: **network** (CODASYL) & **hierarchical** (IMS) DBMS.
 - Low-level "record-at-a-time" DML, i.e. physical data structures reflected in DML (no data independence)
- 1970: **Codd's paper -- the relational model**. The most influential paper in DB research (**Turing award**).
 - **Data independence**. Allows for schema and physical storage structures to change under the covers. Truly important theory, led to "paradigm shift" in thinking and in practice.

Ιστορία των ΒΔ (2)



- mid 70's: 2 full-function (sort of) prototypes
 - Ancestors of essentially all today's commercial systems
 - **Ingres**: UCB 1974-77 (Stonebraker & colleagues)
 - Led to Ingres Corp (CA), Sybase, MS SQL Server.
 - **System R**: IBM San Jose (now Almaden)
 - Led to IBM's DB2, Oracle.
- Both were viable starting points, proved practicality of relational approach. Beautiful example of theory → practice !!

Ιστορία των ΒΔ (3)



- 80's
 - commercialization of relational systems
 - 1979: Oracle v.2 is the first commercial RDBMS
 - SQL becomes de-facto standard.
- 90's: the age of maturity
 - network & hierarchical essentially dead (though commonly in use!)
 - relational becomes mainstream
 - improvements in terms of transactional facilities, performance and stability
 - RDBMS's as back-end for internet front-ends
 - Application Servers and middleware
 - object relational (support for ADT's)

Σήμερα: ο όρος-κλειδί είναι ...



- Κλιμάκωση σε τεράστιους όγκους πληροφορίας !
- Παραδείγματα:
 - **EODIS** (*NASA's Earth Observing System Data and Information System*): 1 Tb/day, keep it all for 15 years (they need tertiary storage for that)
 - **WalMart**: 365 node system, 6Tb online, 4billion row table, 200million updates daily, 4000 queries/day, 1500 users/week
 - *"The world produces between 1 and 2 exabytes of unique information per year, which is roughly 250 megabytes for every man, woman, and child on earth"*
 - Source: Lyman & Varian. How much information?
<http://www.sims.berkeley.edu/research/projects/how-much-info-2003/>

Αύριο: ό,τι πουν οι «σοφοί»



- Senior database researchers gather every few years to **assess the state of database research** and to **recommend problems and problem areas that deserve additional focus**.
 - Laguna Beach, Calif. in 1989,
 - Palo Alto, Calif. (Lagunitas) in 1990,
 - Palo Alto, Calif. (Lagunitas II) in 1995,
 - Cambridge, Mass. in 1996,
 - Asilomar, Calif. in 1998,
 - Lowell, Mass. in 2003
 - Claremont, Calif. in 2008

Lowell report, 2003



- | | |
|--|----------------------------|
| ▪ Integration of Text, Data, Code, and Streams | ▪ Data Mining |
| ▪ Information Fusion | ▪ Self Adaptation |
| ▪ Sensor Data and Sensor Networks | ▪ Privacy |
| ▪ Multimedia Queries | ▪ Trustworthy Systems |
| ▪ Reasoning about Uncertain Data | ▪ New User Interfaces |
| ▪ Personalization | ▪ One-Hundred-Year Storage |
| | ▪ Query Optimization |

Όχι πια βάσεις «δεδομένων»



it is time to stop grafting new constructs onto the traditional architecture of the past. Instead, we should **rethink basic DBMS architecture** with an eye toward supporting:

- Structured data
- Text, space, time, image, and multimedia data
- Procedural data, that is data types and the methods that encapsulate them
- Triggers
- Data streams and queues

as **co-equal first-class components** within the DBMS architecture - both its **interface** and its **implementation** - rather than as afterthoughts grafted onto a relational core

Claremont report, 2008



- Revisiting database engines
 - broadening range of applicability (streams, read-mostly analytics, ...), improving performance for specific domains, ...
- Declarative programming for emerging platforms
 - data parallelism (e.g. Map-Reduce), enterprise application programming (e.g. Ruby on Rails), etc.
- Interplay of structured and unstructured data
 - e.g. extract structure and meaning from un- and semi-structured data
- Cloud data services
 - manageability /scalability aspects, specialized services, ...
- Mobile applications and virtual worlds
 - heterogeneous data streams management, privacy issues, ...

Βιβλιογραφία



Αξιζεί να διαβάσετε τις 2 πιο πρόσφατες σχετικές αναφορές

- Claremont report (2008)
<http://db.cs.berkeley.edu/claremont/claremontreport08.pdf>
- Lowell report (2003)
<http://research.microsoft.com/~Gray/Lowell/>

Ας ξεκινήσουμε...



- ... με μη παραδοσιακούς τρόπους σχεδίασης ΒΔ...
- ... και με σύγχρονες τάσεις και εφαρμογές ΒΔ...

Αντικείμενα στις ΒΔ - OODBMS



- Κλάσεις (classes), Αντικείμενα (objects), Μέθοδοι (methods), ...

```
class Person : public d_Object {
public:
    d_String  name;
    d_String  address;
};

class Account : public d_Object {
private:
    d_Long    balance;
public:
    d_Long    number;
    d_Set <d_Ref<Customer>> owners;

    int       find_balance();
    int       update_balance(int delta);
};
...
```

Αντικείμενα στις ΒΔ - ORDBMS



- Δομημένοι τύποι (Structured types)

```
create type Publisher as
(name varchar(20), branch varchar(20))
```

```
create type Book as
(title varchar(20), author-array varchar(20) array [10],
pub-date date, publisher Publisher,
keyword-set setof(varchar(20)))
```

- Παράδειγμα ορισμού πίνακα:

```
create table books of Book
```

- Παράδειγμα εισαγωγής εγγραφής:

```
insert into books
values
('Compilers', array['Smith', 'Jones'],
Publisher('McGraw Hill', 'New York'),
set('parsing', 'analysis'))
```

What's XML?



- W3C Standard since 1998
 - Subset of SGML
 - (ISO Standard Generalized Markup Language)
- Data-description markup language
 - HTML text-rendering markup language
- De facto format for data exchange on Internet
 - Electronic commerce
 - Business-to-business (B2B) communication

Core XML Technologies



- XML Validation: Contract for Data Exchange
 - DTD, Relax N/G, XML Schema
- XML API: Programmatic Access to XML
 - DOM, SAX
- Transformation Languages for Data Exchange and Display
 - XSL, XSLT, XPATH, XQuery

An XML Document



```
<? xml version=" 1.0"?>
<! DOCTYPE sigmodRecord SYSTEM "sigmodRecord. dtd">
<sigmodRecord>
  <issue>
    <volume> 1</ volume>
    <number> 1</ number>
    <articles>
      <article>
        <title> XML Research Issues</ title>
        <initPage> 1</ initPage>
        <endPage> 5</ endPage>
        <authors>
          <author AuthorPosition=" 00"> Tom Hanks</ author>
        </ authors>
      </ article>
    </ articles>
  </ issue>
```

XML and Databases



- Data stored in SQL databases need to be published in XML for data exchange
 - Specification schemes for publishing needed
 - Efficient publishing algorithms needed
- Storage and retrieval of XML documents
 - Need to support mapping schemes
 - Need to support data manipulation XML API-s

Storing XML

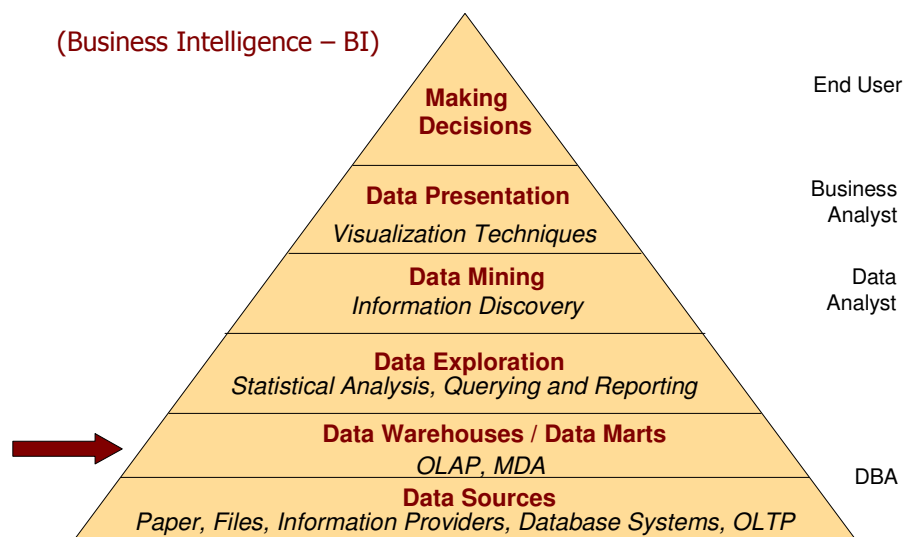


- Storage foundation of efficient XML processing
- XML demands own storage techniques
 - Characteristics of XML data:
Optional elements & values, repetition, choice, inherent order, large text fragments, mixed content
 - Characteristics of XML queries:
Document order & structure, full-text search, transformation
- Goals of tutorial
 - Existing storage features for XML
 - New storage features for XML

Αποθήκες Δεδομένων



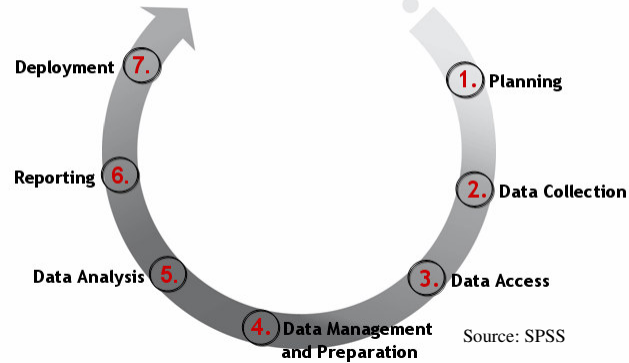
(Business Intelligence – BI)



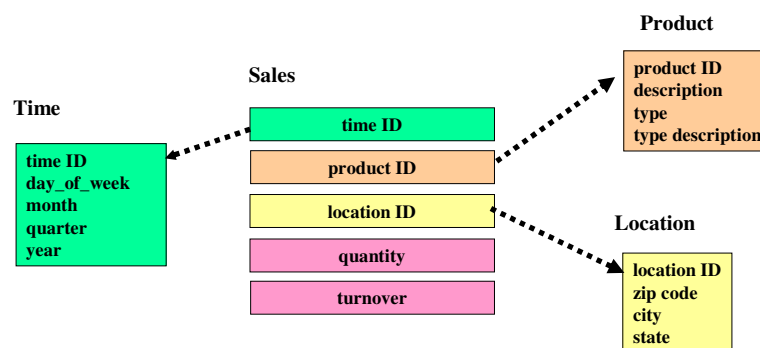
Μεθοδολογία OLAP



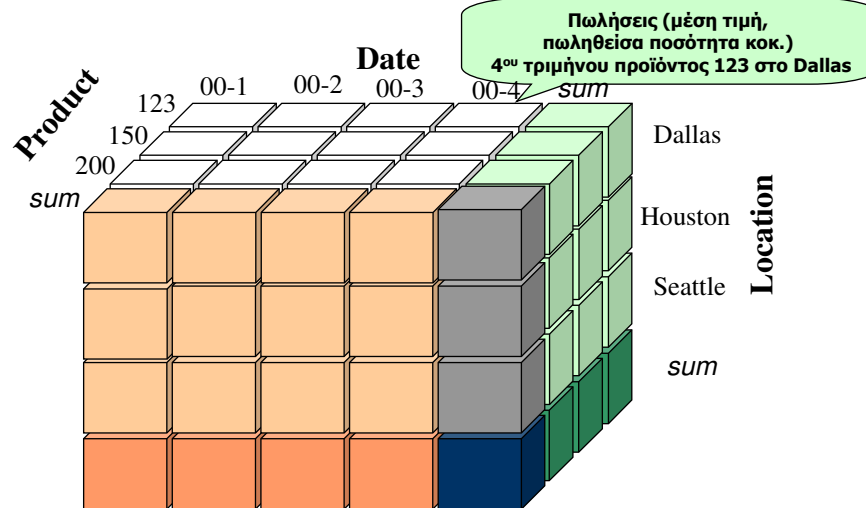
- βήματα αναλυτικής διαδικασίας:
 - από το σχεδιασμό της ανάλυσης,
 - ... στη συγκέντρωση και οργάνωση των απαραίτητων δεδομένων
 - ... την ανάλυσή τους και παρουσίαση των αποτελεσμάτων και τελικά ...
 - την εκμετάλλευση των όποιων συμπερασμάτων



Αποθήκες Δεδομένων – ένα παράδειγμα



Όψη κύβου δεδομένων



Χρονικές βάσεις δεδομένων



Ποια η ανάγκη για χρονικές βάσεις δεδομένων;

- εξαγωγή χρονικών συμπερασμάτων σε βάσεις γνώσεων
- εξαγωγή συμπερασμάτων σχετικά με την ιστορία γεγονότων στο παρελθόν,
- σχεδιασμός μελλοντικών πράξεων και πρόβλεψη συνεπειών,
- λογισμός πάνω στην περίοδο εγκυρότητας της τιμής των δεδομένων.

ΦΥΣΗ ΤΟΥ ΧΡΟΝΟΥ



Τα μοντέλα χρόνου ταξινομούνται ανάλογα με:

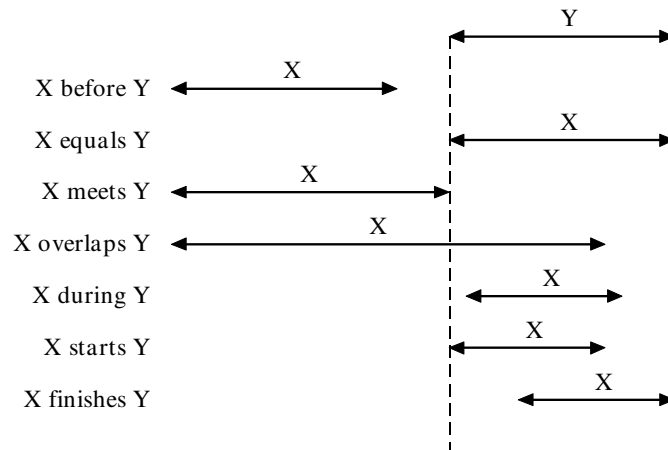
- την πυκνότητα του χρόνου,
- το είδος και το μέγεθος των στοιχειωδών χρονικών μονάδων,
- το μοντέλο εξέλιξης του χρόνου,
- τις διαστάσεις του χρόνου,
- τις πράξεις στο χρόνο.

ΕΙΔΟΣ ΤΟΥ ΧΡΟΝΟΥ



- *Γεγονός*: μια απομονωμένη στιγμή στο χρόνο.
- *Διάστημα (interval)*: ο χρόνος μεταξύ δύο γεγονότων.
- *Διάρκεια (duration)*: ένα ποσό χρόνου με σταθερό μήκος, αλλά χωρίς καθορισμένα άκρα.
- *Περιορισμένο χρονικό διάστημα (span)*: μια κατευθυνόμενη διάρκεια χρόνου (θετική ή αρνητική).

ΠΡΑΞΕΙΣ ΣΤΟ ΧΡΟΝΟ - ΣΧΕΣΕΙΣ ΤΟΥ ALLEN



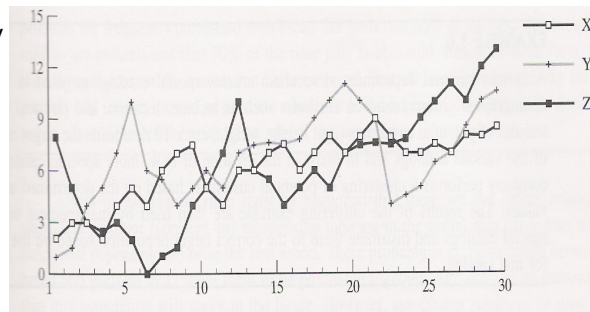
Χρονολογικές σειρές (time series)



- Σύνολο τιμών χαρακτηριστικών στον άξονα του χρόνου
 - $\{ \langle t_1, a_1 \rangle, \langle t_2, a_2 \rangle, \dots, \langle t_n, a_n \rangle \}$

- Παραδείγματα:

- πωλήσεις προϊόντων
- τηλεπικοινωνιακά δεδομένα
- ιατρικά σήματα
- ακολουθίες περιβαλλοντικών μετρήσεων
- ...



Ανάλυση χρονολογικών σειρών



- Ανάλυση χρονολογικών σειρών: η ανακάλυψη προτύπων στις τιμές.
- Είδη προτύπων:
 - Τάση : συστηματική μη επαναλαμβανόμενη αλλαγή στο χρόνο (π.χ. σταθερή αύξηση)
 - Κύκλος : επαναλαμβανόμενη συμπεριφορά
 - Εποχιακό πρότυπο : παρόμοιες τιμές σε συγκεκριμένες περιόδους – μέρα, μήνα, έτος κλπ. - του χρόνου
 - Μη φυσιολογικά δεδομένα (outliers) : στοιχεία που δεν μπορούν να κατηγοριοποιηθούν

Βάσεις Χωρικών Δεδομένων



- Απαιτήση για ειδική διαχείριση των μη-αλφαριθμητικών δεδομένων
 - κατάλληλες γλώσσες επερωτήσεων
 - ειδικές μέθοδοι δεικτοδότησης
 - θέματα βελτιστοποίησης ερωτήσεων
- Υποστήριξη χωρικών τελεστών
 - *"βρες το κοντινότερο νοσοκομείο"*
 - *"βρες όλες τις πόλεις που βρίσκονται σε απόσταση 5Km από ακτές"*

Βάσεις Χωρικών Δεδομένων



Αναλυτικά για ...

- Μοντέλα Δεδομένων
- Γλώσσες Επερωτήσεων
- Μέθοδοι Δεικτοδότησης
- Αρχιτεκτονική Συστημάτων

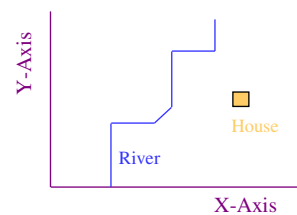
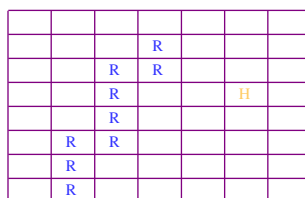
Μοντέλα Δεδομένων



- Δύο βασικές προσεγγίσεις

- raster model

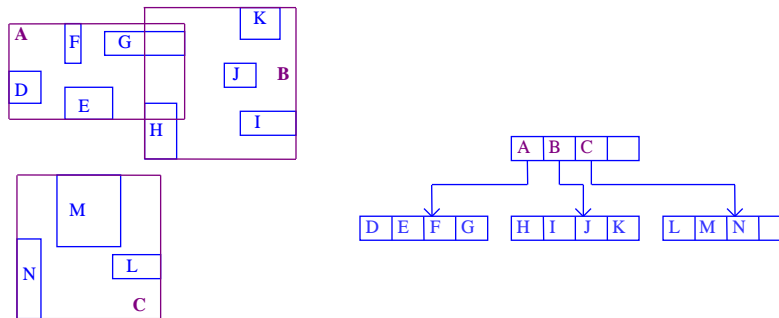
- vector model



Μέθοδοι Δεικτοδότησης



- R-tree



Παράδειγμα



- Consider a spatial dataset with:
 - County boundary (dashed white line)
 - Census block - name, population, boundary (dark line)
 - Water bodies (dark polygons)
 - Satellite Imagery (gray scale pixels)
- Storage in a SDBMS table:

```
create table census_blocks (  
  name string,  
  population number,  
  boundary polygon );
```



Ρεύματα δεδομένων



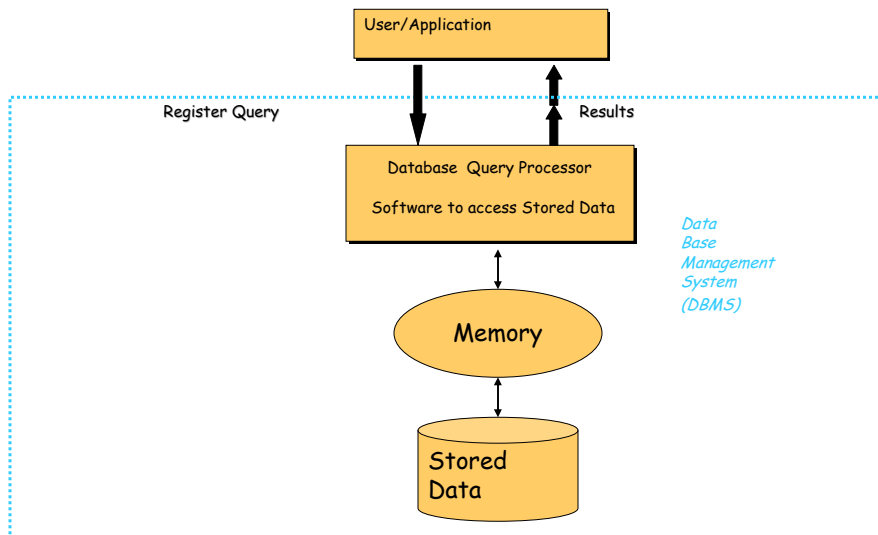
- Data streams differ from conventional stored relation model:
 - Data elements in the stream arrive online
 - System has no control over order in which data elements to be processed
 - Data streams are potentially unbounded in size
 - Once an element from a data stream has been processed, it is discarded or archived. It cannot be retrieved easily unless it is stored in memory, which is small relative to the size of data streams
- Operating in data stream model does not preclude use of data in conventional stored relations.

DBMS versus DSMS



- | | |
|---|---|
| ▪ Persistent relations | ▪ Transient streams |
| ▪ One-time queries | ▪ Continuous queries |
| ▪ Random access | ▪ Sequential access |
| ▪ "Unbounded" disk store | ▪ Bounded main memory |
| ▪ Only current state matters | ▪ History/arrival-order is critical |
| ▪ Passive repository | ▪ Active stores |
| ▪ Relatively low update rate | ▪ Possibly multi-GB arrival rate |
| ▪ No real-time services | ▪ Real-time requirements |
| ▪ Assume precise data | ▪ Data stale/imprecise |
| ▪ Access plan determined by query processor, physical DB design | ▪ Unpredictable/variable data arrival and characteristics |

The Database Model



The Data Stream Model

