Web-Based Information Systems

Prof. dr. Paul De Bra
Eindhoven Univ. of Technology

Topics

• Motivation
• Web Technology
• Design of Web-Based Information Systems
• Automatic Generation of Web-Based Interfaces
Web-Based Information Systems

• The Web brings database information to the world. This offers huge market potential in B2C and efficiency gains in B2B electronic commerce.
• Web-based interfaces for databases need to be designed, because of multimedia objects.
• Heuristics and languages are needed to generate presentations of arbitrary query results.

Applications: electronic commerce

• **product selection**: the range of available products is taken from the production database.
• **pricing information**: presented prices are the same as in the accounting database.
• **on-line ordering**: orders are used to steer production and delivery directly.
• **production and delivery tracking**: customers can follow production and transportation.
• **after-sales support**: documentation (updates) and problem reports are handled through the Web.
Variants of Web database access

• A database can be made accessible to users who do not have a database interface. Server-side scripting is used to send queries to the database and translate answers to HTML.
  – Example: TUE phone book, always up to date
• A Web-site can be populated with data extracted from a database (e.g. at night).
  – Example: UIA phone book, not updated for several years.

Basic Web Standards

• HyperText Transfer Protocol:
  – GET request for information requests;
  – POST request to provide additional information in a request; may have side effects;
  – PUT request to upload information to server.
• HyperText Markup Language
  – Links generate GET requests for other pages;
  – Imagemaps allow the selection of points or regions of images;
  – Forms send attribute/value pairs to the server.
Accessing Databases through www

- The server may forward requests to the database:
  - CGI scripts (Common Gateway Interface)
  - Server-side plug-ins (server dependent)
  - Servlets (Java code executed in the server)
- Browsers can access databases directly through Javascript, VBscript, Java Applets.
- Results are usually converted into HTML to be displayed by the browser.

Architecture using CGI-scripts
Translations in WIS

- Web-server receives forms input, which must be translated into database operations (e.g. into SQL queries or updates).
  - Standards like ODBC and JDBC make it possible to do this in a portable way.
- Database produces results (in a DBMS-dependent format), which must be translated into HTML to send to the browser.
  - When databases can produce results in XML this translation can be done in a portable way.

Architecture with Applets
Presentation of data

• Traditionally tabular data.
• Multimedia data require other presentation:
  – The presentation of an object may require a designed layout for multi-media attributes;
  – Few objects fit on the screen; indirect access may be needed, through links or temporal relationships;
  – Some objects may be too large to fit on the screen; they may need to be split up.

Navigation through data

• Databases consist of objects and relationships.
• Hypermedia applications consist of objects (nodes) and relationships (links).
  – Direct access to objects, as in tables, must be replaced by access through sets of links;
  – Access to different parts of objects must be provided by means of links;
  – Relationships between object types must be translated into link structures.
Design with OOHDM

• Conceptual design
  – Build a model of the application domain using OO modeling principles.

• Navigational design
  – Design how the user can navigate, using indexes and guided tours.

• Abstract interface design
  – Layout through Abstract Data Views (ADV)

• Implementation

Example of conceptual schema
Abstract Interface Design

- An **abstract data view** is a formal, object oriented model of an interface object, showing:
  - the static layout structure, including interface appearance of navigational objects and other interface objects (menu bars, buttons).
  - the static relation to navigation objects.
  - how they behave when reacting to external events; in particular how they trigger navigation. (ADV-charts are a derivative of Statecharts)

Example Configuration Diagram
Example ADV

The Relationship Management Methodology (RMM)

- The name RMM is based on the view that hypermedia is a vehicle for managing relationships between information objects.
- The associated data model is RMDM: Relationship Management Data Model.
- Transformation of data structure into a data and navigation structure.
- RMDM enables to describe information objects and navigation mechanisms in hypermedia applications.
The Relationship Management Data Model (RMDM)

- RMDM’s **domain primitives** model:
  - entity types;
  - attributes;
  - associative relationships.

- **Slices** are groups of attributes, used to split up large groups of diverse attributes into smaller groups of related attributes.
  - Example: a person’s home page can be split up into a main slice, biography slice, publication slice, hobby slice.

Example: E-R Design

![E-R Design Diagram]
RMDM Access Primitives

**E-R Domain Primitives**

- **Entities**
- **Attributes**
  - One-One Associative Relationship
  - One-Many Associative Relationship
- **Slices**

**RMD Domain Primitives**

- **Unidirectional Link**
  - Conditional Index
- **Bidirectional Link**
  - Conditional Guided Tour
  - Conditional Indexed Guided Tour
Example: Slice Design

Example: E-R to RMDM conversion
Example: RMDM Access Constructs

Guided Tour of Associate Professors

(a)

Index of Associate Professors

Index Node

(b)
Example: RMDM Access Constructs

Example: RMDM Scheme
Example: Screen Layout

Applying RMM
Applying RMM

“Real World” Examples

• A simple example with an index structure is the real-estate site of the LMV:
  http://www.lmv.nl/

• An example with indexed guided tours is the real-estate site of the NVM:
  http://www.nvm.nl/
Koopaanbod resultaat

Ontstaan van een goed koopaanbod aan uw specificaties.
KFR voor meer informatie op het object van uw keuze.

Turfveldstraat 2
tuinstad
€790.000 contant koper
nieuw (het)

Reukam 14
huis
€1.250.000 contant koper
nieuw (hdb)

Penningstraat 24
eigenaar, niet afzienbare
€1.399.000 contant koper
nieuw

Rietveld 90
mengingroep, niet afzienbare
€1.119.000 contant koper
nieuw

Laat 36
mengingroep, niet afzienbare
€1.399.000 contant koper
nieuw

Herk de Vrieslaan 64
mengingroep, niet afzienbare
€1.480.000 contant koper
nieuw

Champagnehof 15
mengingroep, niet afzienbare
€1.399.000 contant koper
nieuw

Zandvoort 14
mengingroep, niet afzienbare
€1.442.000 contant koper
nieuw

Bloemhof 11
huis
€1.445.000 contant koper
nieuw

Pagina 2 van 5
Shortcomings of RMM

• For user-defined queries there is **no predefined structure** for which RMM produces a representation.

• An **extension** to RMM is needed to generate hypermedia representations and navigation structures for query results.

• The exact **query specification** needs to be considered as well. It is not feasible to design a presentation for every possible query.
Translation from SQL to RMDM

- Core of the approach is:
  1. Use heuristics to determine a *reasonable* navigation and presentation for query results.
  2. Offer extensions to SQL to allow users to specify alternative navigation and presentation.

- We distinguish three cases:
  - Query result is *single slice*.
  - Query result is *multiple slices* from a *single relation*.
  - Query result is *multiple slices* from *several relations*.

Single slice queries (1)

- Consider relation R with:
  - head slice with attributes ABCD
  - slice Y with EFG, forward link from head slice
  - slice Z with HIJK, forward link from head slice

- **select** A,B,C,D **from** R **where** C=0;

- Heuristics lead to:
  - inter-record navigation as for base relation;
  - intra-record navigation starts at slice “specified” in the select clause;
  - no new volatile slices are created.
Single slice queries (2)

- Extensions to SQL:
  - `select (E,G) asslice X newhead from R;`

- User can connect new slices through links:
  - `select (E,G) aslice X newhead,
    links (X, head(R))
  from R;`
  - `select (H,K) asslice X, (I,J,K) asslice W,
    links (head(R),X), (X,W)
  from R;`

Multiple slice queries

- Consider same R with slices X and Y, with X and Y reachable from head through link.
- `select A,B,I,J,K from R;`

- Heuristics lead to:
  - inter-record navigation same as for base relation;
  - intra-record navigation starts at the head slice;
  - no new volatile slices are created
Multiple relation queries

- Heuristics lead to:
  - inter-record navigation for the resulting set of records is the same as for the first relation in the from clause;
  - intra-record navigation between base records uses indexed guided tour to connect associated base records;
  - intra-record navigation within base record starts at head slice;
  - no new volatile slices are created;

Join structures
Join structures

Example screen shot