The Web is not a Database

Alberto Mendelzon

University of Toronto

http://www.cs.toronto.edu/~mendel/widm.ps

Joint work with Gus Arocena, Attila Barta, George Mihaila, Tova Milo

Outline

- Introduction
- 1G Queries -- WebSQL
- 2G Queries -- WebOQL
- Applications
- Theory of Web Queries
- The Future

The Web as a database

- Asilomar Report (September '98):
  - "...good news for database systems research: the Web is one huge database"
- What is a database?
  - "A collection of persistent data used by the application systems of some enterprise"
- Implications:
  - Reliability, recovery
  - Concurrency control
  - Common schema
  - Declarative query access
  - ...

- Conclusion: the Web is not a database...
  - But it would be nice if it were
- How is a database used?
  - Application programs are written using a declarative language such as SQL
- Reasons to view the Web this way
  - Too hard to use the "raw" Web by browsing/searching
  - Leverage database techniques and programmer's expertise
  - Fun

WebSQL: A minimalist proposal

- Integrate ***Browsing & Searching***
- Data Model;

  Document (URL, title, type, length, text, modif)

  Anchor (base, label, href)

- Query Language: SQL + regexps
- Semantics:
  - Materialize a fragment of the database
  - Compute the answer on this fragment

Related Work

- Query languages for hypertext

  Beeri and Kornatzky, Minohara and Watanabe

- Queries on structured documents

  Abiteboul et al, Guting et al, Christophides et al

- Web query languages

  Konopnicki and Shmueli, W3QS

  Lakshmanan, Sadri, and Subramanian, WebLog

  Abiteboul and Vianu, theory

- Exploiting link structure

  Spertus, PARASITE

  Kleinberg, hubs and authorities

Search Automation

- Find documents about Toronto that reside in servers in Canada

SELECT d.url,d.title
FROM Document d **SUCH THAT** d **MENTIONS** "Toronto"
WHERE d.url **CONTAINS** ".ca$"

- Find documents about WebSQL that point to U of T

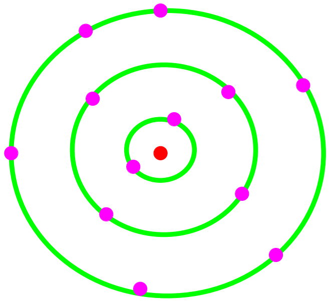
DEFINE INDEX "HotBot";

SELECT d.url
FROM Document d **SUCH THAT** d **MENTIONS** "WebSQL",
Anchor a **SUCH THAT** base = d,
WHERE a.href **CONTAINS** "toronto.edu"
OR a.href **CONTAINS** "utoronto.ca"

Search and Navigation

- Documents about "excursions" near WWW6 home page

SELECT d.url, d.title
FROM Document d
SUCH THAT "www6conf.slac.stanford.edu" **(->| ->-> | ->->->)** d
WHERE d.text **CONTAINS** "excursions"



Path Regular Expressions

- Alphabet (Link types)

#>	interior link: same document
->	local link: same server
=>	global link: different server
=	null path

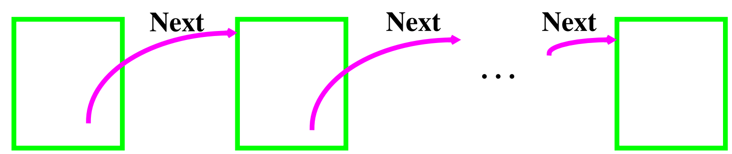
- Regexps Over Link Types

-> | =>	path of length one, either local or global
->*	local path of any length
=>->*	idem, but in other servers
(->|=>)*	the reachable portion of the Web

User-Defined Link Types

DEFINE LINK [next] **AS** label **CONTAINS** “Next”;

SELECT d.url
FROM Document d
SUCH THAT “http://the.starting.document” **[next]*** d,
WHERE d.title **CONTAINS** “Canada”;



WebSQL Semantics

- **Web Calculus**

Set of first-order formulas with equality on:

- Predicates

```
document(url,title,...)
link(from,to,label,...)
path(n1,R,n2)
contains(doc,string)
index(doc,string)
```

Example:

$$\{t \mid \exists u(Document(u,t) \wedge path(\text{www.cs.toronto.edu}, \rightarrow\Rightarrow, u))\}$$

Example:

$$\{x, y \mid index(x, Microsoft), Link(x, y)\}$$

Example:

$$\{x, y \mid Link(x, y)\}$$

Safe Web Calculus Formulas

Document(D,t), Link(D,x), Index(d,String), path(D,R,x), x = C

Conjunctions of safe formulas

Disjunctions of safe formulas with the same free variables

Conjunction of a safe formula with the negation of a safe formula on some of its variables

Conjunction of a safe formula on $x_1, x_2, \ldots, x_n$ *with*
$x_i = x_j, x_i \neq x_j, x_i = y$, *contains*($x_i$,*String*), *contains*($x_i$,$x_j$), *Link*($x_i$,$y$), *path*($x_i$,$R$,$y$)

Existential quantification of a safe formula

Theorem: Safe formulas are computable

Example applications

- Indexing an On-line Manual
- Indexing Publication Lists

Index of Online Publications

- Need pairs <URL of .ps, Metadata>

Internet

Alberto Mendelzon and Tova Milo, **Formal Models of the Web,** to appear in *Proc. PODS'97, Tucson, May 1997.*

Gustavo Arocena, Alberto Mendelzon, George Mihaila, **Applications of a Web Query Language,** to appear in *Proc.6th Int'l. WWW Conf., Santa Clara, April 1997.*

Alberto Mendelzon, George Mihaila, Tova Milo, **Querying the World Wide Web,** in *Proc. PDIS'96, Miami, December 1996.*

SELECT a.href, a.label
FROM Anchor a
SUCH THAT base = "http://www.cs.utoronto.ca/~mendel/papers.html"

A (partial) list of publications

- S. Abiteboul, S. Cluet, T. Milo, A Database Interface for Files Update . *Proc. ACM SIGMOD Int. Conf. on Management of [...] 1995* San Jose, May 1995.
- Y. Afek and G. Stupp, Synchronization power depends on th[...] register size. In *Proc. of the 34th Ann. IEEE Symp. on Foundations of Computer Science*, pages 196–205, Novemb[...] 1993.
- Y. Afek and G. Stupp, Delimiting the power of bounded size synchronization objects. In *Proc. of the 13th Ann. ACM Sym[...] Principles of Distributed Computing*, pages 42–51, August 1[...]
- Y. Afek, D. Dauber, and D. Touitou, Wait–free Made Fast. In [...]

DEFINE CONTEXT BEGIN = <LI>, **END** = <LI>;

SELECT e.href, e.context
FROM Anchor e **SUCH THAT**
base = "http://www.math.tau.ac.il/~milo/dept/papers.html"
WHERE e.href **CONTAINS** ".ps"

[Adding Structure to Unstructured Data] (140K)
Peter Buneman, **Susan Davidson**, **Mary Fernandez** and **Dan Suciu**
Technical Report MS-CIS-96-21, CIS Department, University of Pennsylvania.
See here for the abstract.

[A Query Language and Optimization Techniques for Unstructured Data] (144K)
Peter Buneman, **Susan Davidson**, **Gerd Hillebrand** and **Dan Suciu**
Technical Report MS-CIS-96-09, CIS Department, University of Pennsylvania.
An extended abstract of this work appears in *SIGMOD Proceedings*, 1996.
See here for the abstract.

[A Query Language for Multidimensional Arrays: Design, Implementation, and Optimization Techniques] (87K)
Leonid Libkin, **Rona Machlin** and **Limsoon Wong**
SIGMOD Proceedings, 1996.
See here for the abstract.

DEFINE LINK [here] **AS** label **CONTAINS** "here"
SELECT e.url, d.text
FROM Document d **SUCH THAT**
"http://www.cis.upenn.edu/~db/langs/allpapers.html" [here] d,
d [here] e;

Programmatic Interface

```
public static void main(String args[]) {
   String query = "SELECT x.url, x.title, x.length, x.date "+
   " FROM Document x SUCH THAT x MENTIONS\"Java\";";

   try{
      WebSQLServer eng = new WebSQLServer(query, new Mon());
      for (Enumeration e = eng.elements();
e.hasMoreElements(); ) {
         Vector tuple = (Vector) e.nextElement();
         for (int i = 0; i < eng.tupleSize; i++) {
            System.out.print(tuple.elementAt(i));
            System.out.print(" ");
         }
         System.out.println();
      }
   }catch(Exception e){System.out.println("Couldn't create
server.");}
}
```

Query Analysis

- How expensive is a query?
- One measure: number of network accesses
- We want to bound query cost as a function of network parameters:
  - Number of nodes (document) in the network
  - Number of documents per server
  - Number of links per document

Why?

"...I share your concern about what could become a very fragile future if we don't build in mechanisms for constraining the scope of these mobile agents or dealing with the potential resources they might consume. A nightmare situation is that you pose a query and it turns out to be the $64,000 question and you get the bill."

Vinton Cerf

Examples

(Query executed at www.cs.toronto.edu)

SELECT *x*
FROM Document ***x*** **SUCH THAT** "www.cs.toronto.edu" -> *x*

- O(1)

SELECT *x*
FROM Document *x* **SUCH THAT** "www.cs.toronto.edu"(->|=>) *x*

- O(k) = maximum number of links in a document

SELECT *x*
FROM Document *x*
SUCH THAT "www.cs.toronto.edu"->* *x*

- O(1)

SELECT *x*
FROM Document *x*
SUCH THAT "www.cs.toronto.edu"(=>.->*) *x*

- O(ks) = maximum links in a document * maximum documents per server

SELECT *x*
FROM Document *x*
SUCH THAT "www.cs.toronto.edu"(->|=>)* *x*

- O(n) = total number of documents

Cost of Path Expressions

- Expressions with no global links have locality *O(1)*
- Expressions with "starred" global links have locality *O(n)*

- For other expressions, locality is bounded by the minimum of n and

$$m \cdot (l+1) \cdot (k \cdot \min(s, k^l))^m$$

where

m =maximum number of global links on a path

l =maximum number of local links between two global links

s= maximum number of documents per server

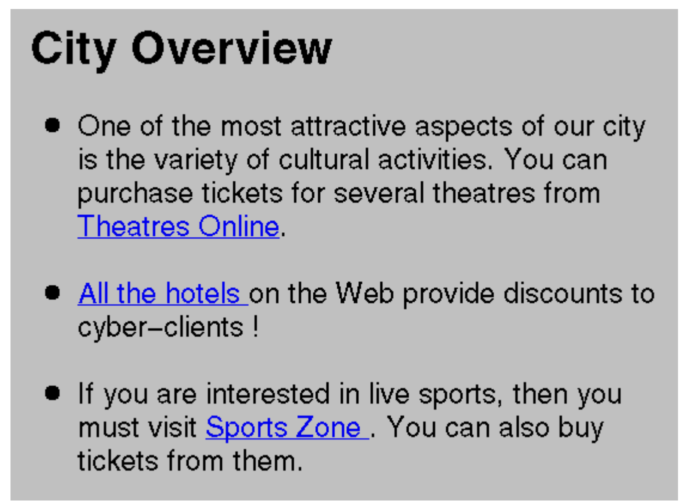
k= maximum number of links per document

WebOQL

- WebSQL: Web as graph of atomic objects
- WebOQL: Web as graph of structured objects
- Query:
  - the Web
  - a single page
  - a set of related pages
- Restructure:
  - HTML to HTML
  - HTML to databases
  - Databases to HTML

Related Work

- Semi-structured Data
  - Buneman et al, UNQL
  - Widom et al, LOREL
- Web site restructuring
  - Atzeni et al, ARANEUS
  - Fernandez et al, STRUQL
- String-to-string mappings
  - Cordy et al, TXL

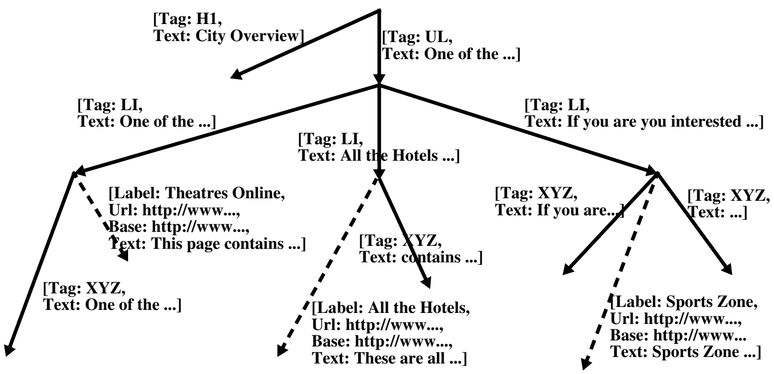


City Overview

- One of the most attractive aspects of our city is the variety of cultural activities. You can purchase tickets for several theatres from Theatres Online.
- All the hotels on the Web provide discounts to cyber–clients !
- If you are interested in live sports, then you must visit Sports Zone . You can also buy tickets from them.

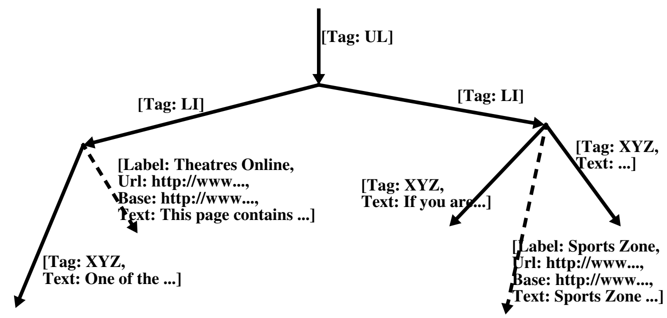
Data Model

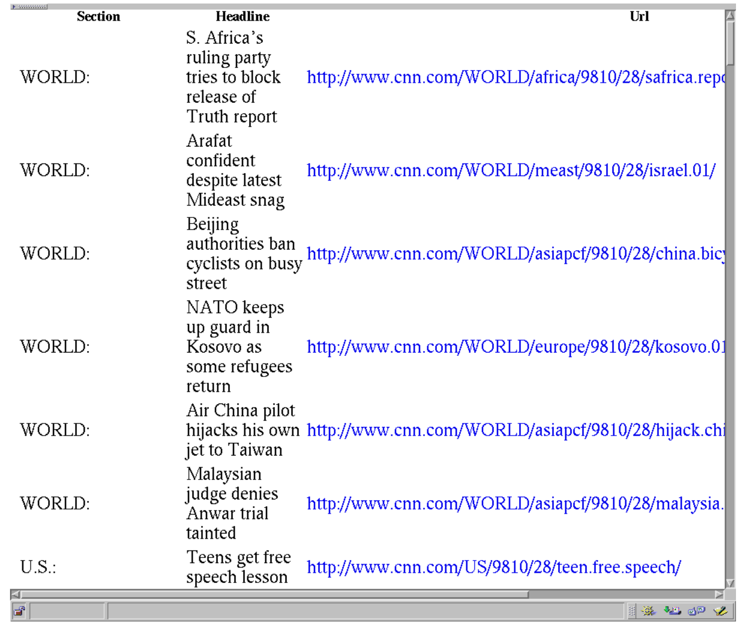
- Records as Labels on Arcs
- Internal and External Arcs



Query: list elements containing "ticket"

```
doc := "http://www.citynet.com/overview.html";
[ Tag "UL" /
        select y
        from y in doc !'
        where y'.text ~ "ticket"
```

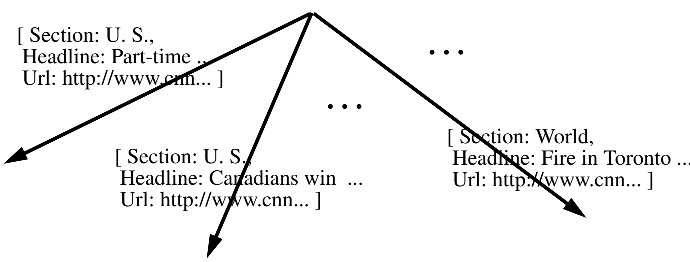




Section	Headline	Url
WORLD:	S. Africa's ruling party tries to block release of Truth report	http://www.cnn.com/WORLD/africa/9810/28/safrica.repo
WORLD:	Arafat confident despite latest Mideast snag	http://www.cnn.com/WORLD/meast/9810/28/israel.01/
WORLD:	Beijing authorities ban cyclists on busy street	http://www.cnn.com/WORLD/asiapcf/9810/28/china.bicy
WORLD:	NATO keeps up guard in Kosovo as some refugees return	http://www.cnn.com/WORLD/europe/9810/28/kosovo.01
WORLD:	Air China pilot hijacks his own jet to Taiwan	http://www.cnn.com/WORLD/asiapcf/9810/28/hijack.chi
WORLD:	Malaysian judge denies Anwar trial tainted	http://www.cnn.com/WORLD/asiapcf/9810/28/malaysia.
U.S.:	Teens get free speech lesson	http://www.cnn.com/US/9810/28/teen.free.speech/

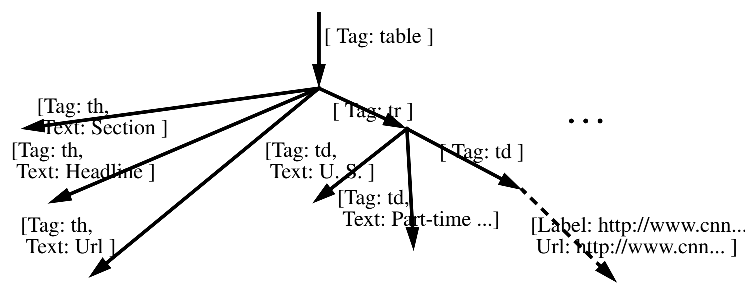
Extracting CNN's Headlines

select [Section:Y.text, Headline:z.text, Url:z'.url]
from X in "http://www.cnn.com" via ^*[text ~ "T O P"],
Y in X!!!' via ^*[tag = "blockquote"],
z in Y!'



Restructuring the Result into HTML

[Tag:"table"/
[Tag:"th", Text:"Section"] + [Tag:"th", Text:"Headline"] + [Tag:"th", Text:"Url"]+
select [Tag:"tr"/ [Tag:"td", Text:Y.text] + [Tag:"td", Text:z.text] +
[Tag:"td"/ [Label:z'.url, Url:z'.url]]
]
from X in "http://www.cnn.com" via ^*[text ~ "T O P"],
Y in X!!!' via ^*[tag = "blockquote"], z in Y!'
]



Easy to do in WebOQL

Extract all headings

Extract all images

Linearize page hierarchy

Flatten hierarchy into table

Censor the Web

Find non-smiling faculty

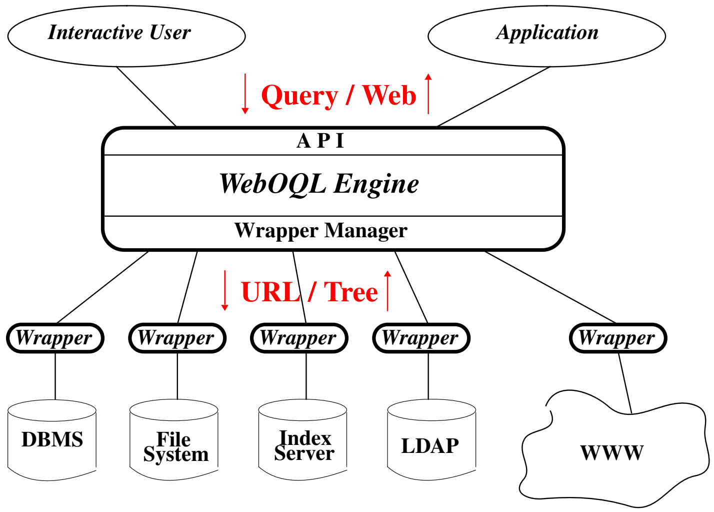
```
SCAN
"http://www.cs.toronto.edu/DCS/People/Faculty/index.html"
USING
    ANY
    <BODY>
       MANY
       <UL>
          {<LI> <A HREF = MemberPage> MemberName </A> </LI>}
       </UL>
    </BODY>
AND
   MemberPage
USING
   ...<IMG SRC = Jpg ".jpg$">
GIVING
  <HTML>
     <TABLE>
        {<TR>
           <TD> text(MemberName) </TD>
           <TD> <IMG SRC = Jpg> </TD>
        </TR>}
     </TABLE>
 </HTML>
```

Generated WebOQL

```
[Tag:"html"/
  [Tag:"table"/
    select [Tag:"tr"/
             [Tag:"td"/[Text:MemberName.text]] +
             [Tag:"td"/[Src:Jpg.src, Tag:"img"]]
           ]
    from V__ is "http://www/DCS/People/Faculty/index.html",
         V_0 in V__!' via [Tag = "ul"] until true,
         V_1 in V_0',
         MemberName is V_1'&,
         MemberPage is MemberName,
         V_2 in browse(MemberPage.url)
             via ^*[Src ~ ".jpg$" and Tag = "img"],
         Jpg is V_2&
    where V__!.Tag = "body" and V_1.Tag = "li" and
MemberName.Tag =
"a"
  ]
];
```

```
<!-- Generated by WebOQL 1.0 -->
<html>
  <table>
    <tr>
      <td>
T.S. Abdelrahman, MSc, PhD
      </td>
      <td>
        <IMG  SRC="http://www.cs.toronto.edu/gifs/Faculty/
tsa.jpg">
      </td>
    </tr>
    <tr>
      <td>
R.M. Baecker, MSc, PhD
      </td>
      <td>
        <IMG  SRC="http://www.cs.toronto.edu/gifs/Faculty/
rmb.jpg">
      </td>
    </tr>
    <tr>
      <td>
A. Bonner, MSc, PhD (Erin) ...
```

System Architecture



Application: WebCat

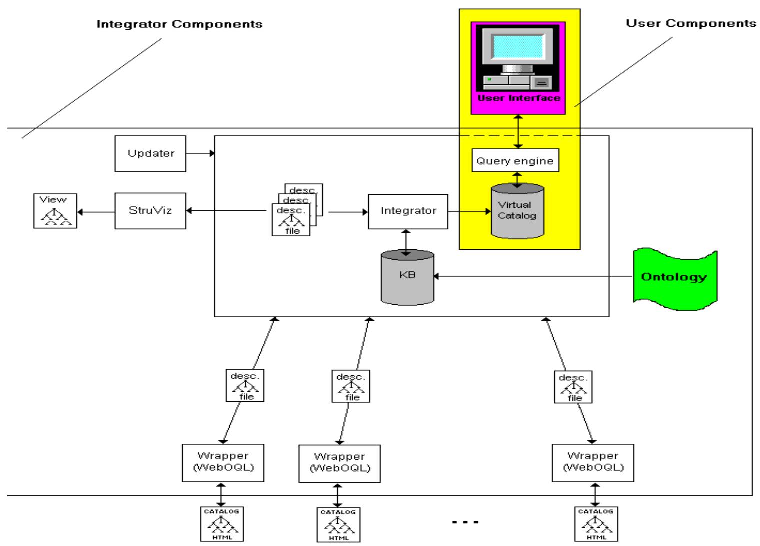
- Web Catalog: a collection of interrelated HTML pages connected by hyperlinks according to a logical hierarchy.
- navigational
- no query capability
- uncooperative
- Examples: on-line product catalogs, library catalog, data sheets,...

WebCat Methodology:

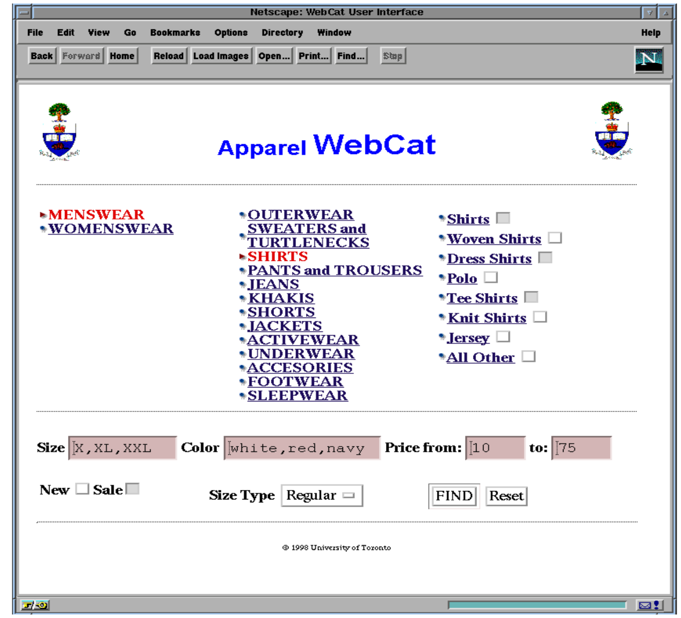
- Catalogs modeled as AST's
- AST's extracted with WebOQL queries
- Integrated using domain ontologies

Implementation:

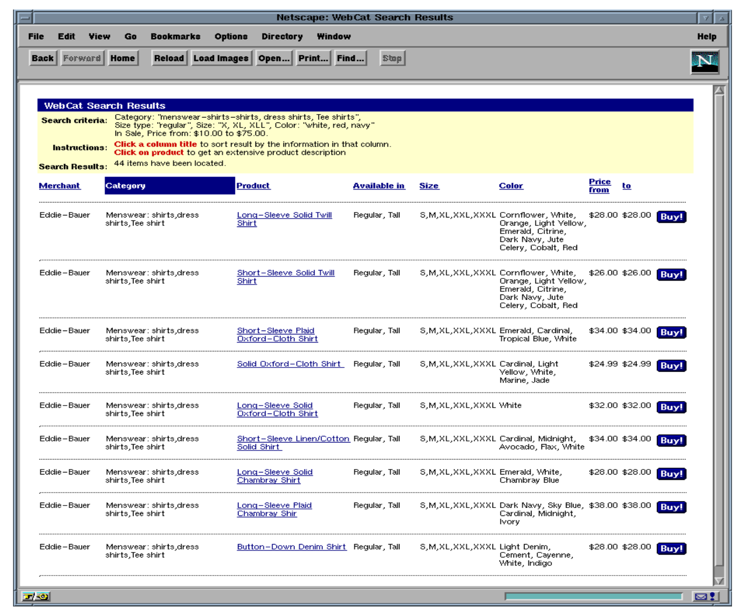
- *Apparel WebCat* integrates J.C. Penney, Eddie Bauer, The Gap



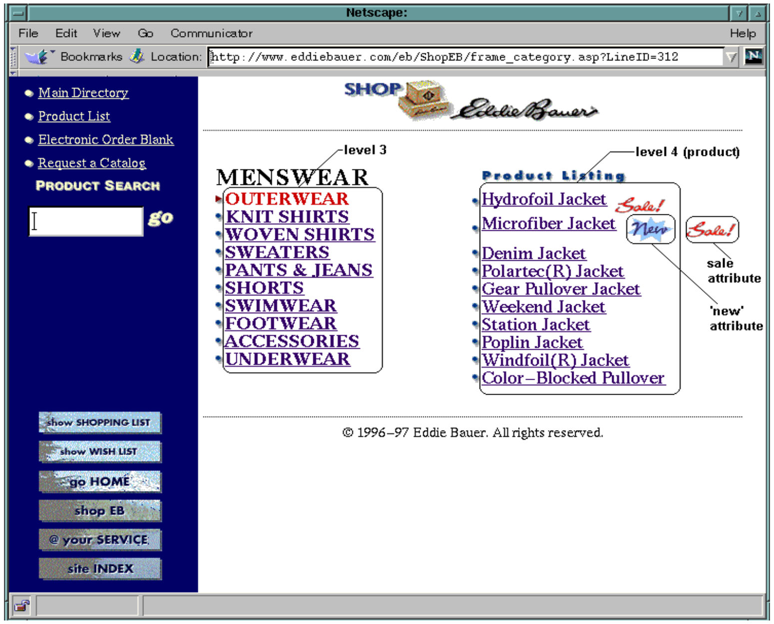
Integrator Components
User Components
User Interface
Query engine
Virtual Catalog
Updater
View
StruViz
desc. file
Integrator
KB
Ontology
Wrapper (WebOQL)
CATALOG HTML
. . .

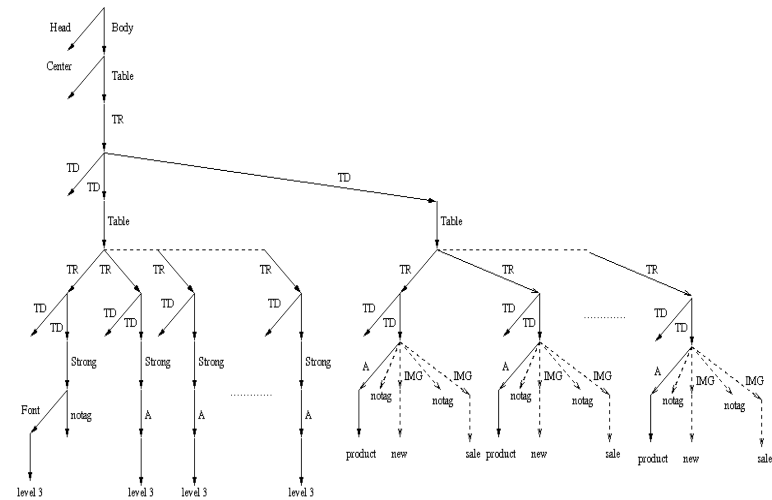


Netscape: WebCat User Interface
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Load Images Open... Print... Find... Stop
Apparel WebCat
MENSWEAR
WOMENSWEAR
OUTERWEAR
SWEATERS and TURTLENECKS
SHIRTS
PANTS and TROUSERS
JEANS
KHAKIS
SHORTS
JACKETS
ACTIVEWEAR
UNDERWEAR
ACCESORIES
FOOTWEAR
SLEEPWEAR
Shirts
Woven Shirts
Dress Shirts
Polo
Tee Shirts
Knit Shirts
Jersey
All Other
Size X,XL,XXL
Color white,red,navy
Price from: 10
to: 75
New
Sale
Size Type Regular
FIND
Reset
© 1998 University of Toronto



Netscape: WebCat Search Results
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Load Images Open... Print... Find... Stop
WebCat Search Results
Search criteria: Category: "menswear-shirts-shirts, dress shirts, Tee shirts", Size type: "regular", Size: "X, XL, XXL", Color: "white, red, navy" In Sale, Price from: $10.00 to $75.00
Instructions: Click a column title to sort result by the information in that column. Click on product to get an extensive product description
Search Results: 44 items have been located.
Merchant
Category
Product
Available in
Size
Color
Price from
to
Eddie-Bauer
Menswear: shirts,dress shirts,Tee shirt
Long-Sleeve Solid Twill Shirt
Short-Sleeve Solid Twill Shirt
Short-Sleeve Plaid Oxford-Cloth Shirt
Solid Oxford-Cloth Shirt
Long-Sleeve Solid Oxford-Cloth Shirt
Short-Sleeve Linen/Cotton Solid Shirt
Long-Sleeve Solid Chambray Shirt
Long-Sleeve Plaid Chambray Shir
Button-Down Denim Shirt
Regular, Tall
S,M,XL,XXL,XXXL
Buy!





WebOQL query to extract the links to products

```
target := browse("http://www.eddiebauer.com/eb/ShopEB/
       frame_category.asp?LineID=312&LineName=MENSWEAR&CatID=1275");

select [z.base, z.url, z.text]
from x in target via ^[tag = "table"],
y in ((((x'')!!)')!!!)', z is (((y')!)')&;
```

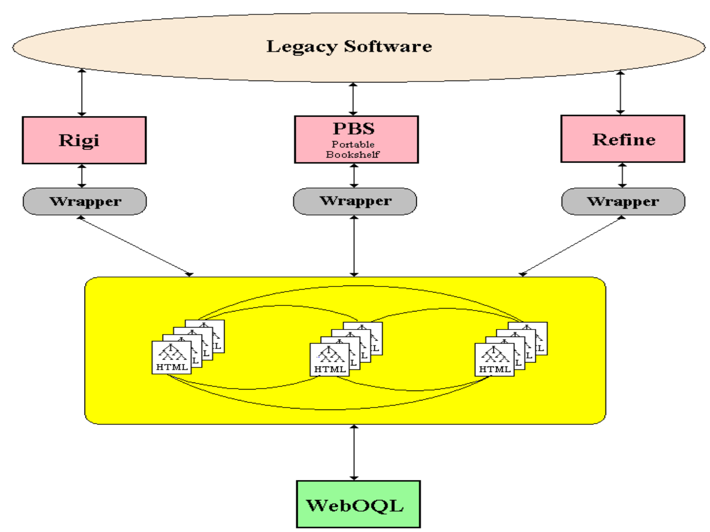
WebOQL query to extract the sale attribute

```
target := browse("http://www.eddiebauer.com/eb/ShopEB/
category.asp?ProdID=&CatID=1309&LineID=312&LineName=MENSWEAR");

select   [z.text,  t.src]
from x in target via ^[tag = "table"],
y in (((x''!!)')!!!)', t is ((((y')!)')!4);
```

WebOQL query to extract the new attribute

```
target := browse("http://www.eddiebauer.com/eb/ShopEB/
category.asp?ProdID=&CatID=1309&LineID=312&LineName=MENSWEAR");

select   [z.text, w.src]
from x in target via ^[tag = "table"],
y in (((x''!!)')!!!)', z is (((y')!)')&, w is ((((y')!)')!!)&;
```

Application: Software Repository



Theory of Web Queries

Which queries are "computable"?

- All documents my home page points to
- All documents reachable from my home page
- All documents such that *at least* 3 other documents point to them
- All documents such that *at most* 3 others point to them

Static Web

A **Web machine** is a Turing machine augmented with a Web oracle.

A **Web query** is a total relational query on the Nodes, Links database schema.

A web query is **computable** if it is computable by a terminating Web machine.

Theorem: *A web query is computable iff there is a finite set of URL's C such that the query only depends on the reachability closure of C.*

Example: find documents reachable from my home page.

Computable Query Languages

The Web calculus can express non-computable queries.

The safe Web calculus cannot.

Theorem: A Web calculus query is computable iff it is expressible in the safe Web calculus.

Static Web (cont.)

A web query is **eventually computable** if a Web machine will eventually produce every tuple in the answer.

Theorem: A web query is eventually computable iff it is monotone with respect to addition of unreachable nodes to the Web.

Example: find documents that at least three others point to.

But: "find documents that at most three others point to" is not eventually computable

Dynamic Web

"Find documents reachable from my home page" is no longer computable

Dynamic Web machine:

- runs on an infinite sequence of (Node,Link) databases
- the oracle can switch from one database to the next at any time

A Web query is now a non-deterministic mapping from sequences of Webs to (possibly infinite) relations

Dynamic Web (Cont.)

Computable: computable by a dynamic Web machine

(nodes at distance 2 from my home page)

Semi-computable: computable by a dynamic Web machine that is fast enough

(nodes reachable from my home page)

Eventually computable: eventually computable by a dynamic Web machine

(nodes containing the string "Java")

Summary

- High level abstraction of the Web
- Combined searching / browsing: *WebSQL*
- Access to internal structure: *WebOQL*
- Theory of Web queries

The Future

- XML, Metadata
- Web/Internet Information Systems
- *Hidden Web*
- Query optimization

Asilomar Report:

> "The majority of human information will be on the Web in ten years. It will be an exabyte spread across the planet in many formats. Absent new tools, finding and understanding answers to our questions will be even harder than it is today."