Querying Bi-level Information

Sudarshan Murthy, David Maier, Lois Delcambre
Department of CSE, OGI School of Science & Engineering at OHSU

http://cse.ogi.edu/sparce
mailto:smurthy@cse.ogi.edu
Superimposed Information*

- People often superimpose new information onto existing information
  - Annotations, summaries, ...
- They use many means
  - Mark up paper
  - Place sticky notes on the paper
- They combine existing information and their interpretations to get “their” view

*Picture courtesy of Prof. James Pankow, Dept. of EBS, OGI
Superimposed information

Item

Group
Some Possible Queries

- Show section headings for Garlic items
- List documents consulted
- Create an HTML table of contents from selections

Bi-level queries operate on superimposed information and base information
Goal

Query superimposed information and heterogeneous base information of varying granularity, with minimal amount of mediated base information
Outline

• Motivation
• Background
  – Superimposed information management, SPARCE
• Bi-level query system
  – An example implementation
• Discussion
• Conclusion
Superimposing Information

- Overlaying new information on top of existing information
  - Add new data
  - Impose new schema or model

- *Mark* is a reference to base element
  - Many implementations, ~ one per base type
  - Addressing scheme depends on base type

Heterogeneous sources: Word, Excel, PDF, HTML,…
Beyond Browsing

• Marks facilitate browsing
• Bi-level querying requires access to base information
  – Context provides access to base information
Excerpts and Contexts

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excerpt</td>
<td>provide applications and users with the benefits of a database with a schema -- similar to … within the Garlic system</td>
</tr>
<tr>
<td>Font name</td>
<td>Times New Roman</td>
</tr>
<tr>
<td>Section Heading</td>
<td>Garlic Overview</td>
</tr>
</tbody>
</table>

- **Excerpt** is the content of a marked region
- **Context element** is one piece of context
- What constitutes a context varies
- A mediator called **context agent** retrieves context of a mark
SPARCE

• The Superimposed Pluggable Architecture for Contexts and Excerpts
  – Middleware for superimposed information management
  – Provides mark and context management services
• Superimposed applications use SPARCE to activate marks and retrieve context
A Naïve Bi-level Query System

Superimposed Info

Mark Info

Transformers

Query Processor

XPath, XSLT, XQuery

Property sets

Context Agents

PDF docs

Word docs

RIDPad Document

XML

XML

XML, HTML, ...

XML, HTML, ...

28-Jun-04

Querying Bi-level Information
<?xml version="1.0" ?>
- <RIDPadDocument name="Data Integration">
  - <Group name="Garlic" index="1" left="2955" top="360">
    + <Item name="Press" index="6" left="3000" top="3990">
      + <Mark ID="AcrobatPDFTextMark20040320105004SURYASMurthy">
      + <Container ID="CClassesCSE606INride-dom-finalpdf">
      + <Application ID="Acrobats5">
      + <Context objecttype="Mark"
        objectid="AcrobatPDFTextMark20040320105004SURYASMurthy">
        <![CDATA[ Provide the benefits of a database with a schema without actually storing data within the Garlic system proper. ]]]>
      </Context>
    </Item>
    + <Item name="Query Optimizer" index="3" left="3000" top="2910">
      + <Item name="Model" index="2" left="2985" top="1785">
      </Item>
    </Group>
  + <Group name="Schematic Heterogeneity" index="7" left="120" top="1320">
    + <Item name="CLIO" index="13" left="120" top="120">
    </Item>
  </_group>
</RIDPadDocument>
Show Section Headings for Garlic Items

//Group[@name='Garlic']/Item/Context/Kind[@name='Containment']/Kind[@name='Section']/Element[@name='Heading']

- <RIDPadDocument name="Data Integration">
  - <Group name="Garlic" index="1" left="3120" top="240">
    + <Item name="Press" index="5" left="3390" top="3915">
      - <Item name="Goal" index="4" left="3270" top="540">
        + <Mark ID="AcrobatPDFTextMark20040528140302TYEEsmurthy">
        + <Container ID="ClassesCSE606INIride-dom-finalpdf">
        + <Application ID="Acrobat5">
    - <Context objecttype="Mark">
        objectid="AcrobatPDFTextMark20040528140302TYEEsmurthy">
        - <Kind id="1" name="Containment">
          - <Kind id="1" name="Section">
            - <Element id="0" name="Heading">
              <![CDATA[ Garlic Overview ]]>
List Documents Selected

```xml
<Paths> {FOR $l IN
    document("src")//Item/Container/Location
    RETURN <Path> {$l/text()} </Path>
} </Paths>
```

- `<RIDPadDocument name="Data Integration">
  - `<Group name="Garlic" index="1" left="3120" top="240">
    + `<Item name="Press" index="5" left="3390" top="3915">
      - `<Item name="Goal" index="4" left="3270" top="540">
        + `<Mark ID="AcrobatPDFTextMark20040528140302TYEEsmurthy">
          - `<Container ID="C:\Classes\CSE606INI\ride-dom-finalpdf">
              `<Agent>AcrobatAgents.PDFAgent</Agent>
              `<Class>PDFDocument</Class>
              `<Location>C:\Classes\CSE606INI\ride-dom-final.pdf</Location>
              `<AppID>Acrobat5</AppID>
          </Container>
        + `<Application ID="Acrobat5">
```
Create HTML Table of Contents
Discussion and Future Work
Query Language and Data Model

- XQuery, etc. may not be appropriate for end users
  - XML may not even be the best data model
Preserve the Layers

```html
<Group name='…'>
<Item name='…'>
<Mark id='…'>
</Mark>
</Item>
</Group>
</Group>
-------------------------------------
<Mark id='…'>…</Mark>
<Mark id='…'>…</Mark>
-------------------------------------
<Context …>…</Context>
<Context …>…</Context>
```

Marks repository
Build dynamically
Why Preserve the Layers

• The information sources are different
  – SI: Superimposed application
  – Marks: SPARCE
  – Contexts: Base applications (via context agents)
• Building a complete hierarchy is unnecessary, and could be inefficient
  – Mark and context information could be replicated
  – Context can be large (broad)
• Joins can provide the same result
  – Views could be defined to “merge” the layers
Add Smarts

Show *section headings* for *Garlic* items

- Minimize amount of information retrieved
  - Only some superimposed information elements and marks might qualify, only some context elements might be needed
  - Push ‘selects’ down
- Pass-through queries
  - Some base applications may have query capability
- Explore parallel and distributed query execution
Exploit Relationships

- Relationships in the superimposed layer may help answer new queries
  - What systems does CLIO use?
  - How is CLIO related to SchemaSQL?

- Issues
  - Some relationships may be multi-way; XML is hierarchical
Conclusion

- Enhancing base information with superimposed information makes possible new queries over base information
- Superimposed information and heterogeneous base information may be queried as one
- We have implemented a naïve bi-level query system, but we have many design factors to consider
Questions?

Contact me for a demo

Visit
http://www.cse.ogi.edu/sparce
Related Work

• Garlic: Carey and others, 1995
• MIX: Baru and others, 1999
• MetaXPath: Dyreson and others, 2001
• CXPath: Camillo and others, 2003
• Dexter: Halasz and Schwartz, 1994
• OLE Compound Documents: Microsoft, 1995
• Multivalent Documents: Phelps and Wilensky, 2000