

Results Visualization in the XBrain XML Interface to a Relational Database

Zhengming Tang^{1,3}, Yana Kadiyska³,
Dan Suci³, PhD, James F. Brinkley^{1,2,3}, MD, PhD

¹Departments of Biological Structure, ²Medical Education and Biomedical Informatics, and ³Computer Science and Engineering, University of Washington, Seattle, WA 98195

The University of Washington's XBrain application is used to dynamically export relational data over the web in XML format, as a prelude to data exchange. We describe additional tools to aid the human user in visualizing the dynamically generated XML results returned by the web application.

XBrain is a JSP web application that dynamically publishes data from a relational database containing language map data acquired during neurosurgery.¹ XBrain takes XQueries as input, hands it to a middleware application called SilkRoute, and returns XML results back to the user. Although XML is good for data exchange between applications, it is often not chosen for visualization of the data because it is not very human readable (sample output in Figure 1). Therefore, we focused on the users (neuroscientists) and developed more intuitive ways to visualize the XML output in XBrain.

From initial interviews, we found that users wanted results in HTML and CSV (comma separated values, for import into Excel). We implemented these formats using XSLT, a well-known language for transforming XML into other formats. However, because the DTDs (Document Type Definition) of the XML results are not known beforehand, the XSLT must be generated dynamically, on a per user query basis. The generated XSLT is then used to transform the XML result into either HTML (Figure 2) or CSV (Figure 3).

The advantage of HTML and CSV is that they are fairly general: given any user query input, one can generate the XSLT transformations to convert the results into nested tables for HTML or a flat table for CSV. However, the same generality also revealed a major disadvantage of these formats: by assuming nothing about the results, the visualizations don't offer any interesting analysis of the data, leaving the analysis completely to the users.

In order to further make XBrain useful to the users, we focused on a common set of queries which finds the anatomical regions in the brain where different language errors occur. Figure 4 displays results from such a query using a dynamically generated image. Users may then filter their query results and redraw the images based on the new constraints.

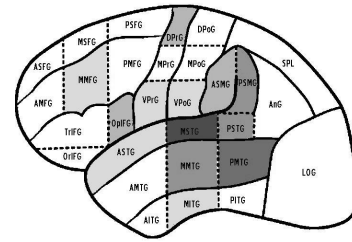


Figure 4. Image Display

We conducted informal user studies on a small number of researchers. All preferred the new visualization formats over XML. When users were asked to compare analyzing results in XBrain against their current manual methods using Excel and Photoshop, all preferred using XBrain. More importantly, users are interested in trends/patterns in the data and in being able to quickly visualize the results and add/update constraints. Thus, they can then explore various permutations and locate new and interesting trends using the new visualizations in XBrain.

Acknowledgements

Supported by Human Brain Project grant RO1 MH/DC02310.

References

- [1] Tang Z, Kadiyska Y, Li H, Suci D, and Brinkley JF. Dynamic XML-Based Exchange of Relational Data: Application to the Human Brain Project. Proceedings AMIA: In Press 2003.

```

<results>
  <patient>
    <last_name>Ciliento</last_name>
    <trial>
      <trial_num>24</trial_num>
      <minim_code>2</minim_code>
      <stmsite>
        <site_label>26</site_label>
        <zone />
        <anatomical_name>middle part of inferior temp</stmsite>
      </trial>
    </patient>
  <patient>
    <last_name>Fulbright</last_name>
    <trial>

```

Figure 1. Original XML Output

last_name	trial	trial_num	minim_code	stmsite	site_label	zone	lobe	anatomical_name
Ciliento	24	2	26	middle part of				
Fulbright	25	2	21	middle part of				
Hollets	111	2	36	posterior part				
Jedley	23	2	34	posterior part				
Steeppig	25	2	31	posterior part				
Steeppig	104	2	26	middle part of				
Lashunda	126	2	30	anterior part of				

Figure 2. HTML

patient last name	trial	trial_num	minim_code	stmsite	site_label	stmsite zone	stmsite lobe	stmsite anatomical_name
1 patient last name	24	2	26	middle part of				
2 Ciliento	25	2	21	middle part of				
3 Fulbright	111	2	36	posterior part				
4 Hollets	23	2	34	posterior part				
5 Jedley	25	2	31	posterior part				
6 Steeppig	104	2	26	middle part of				
7 Steeppig	126	2	30	anterior part of				
8 Lashunda								

Figure 3. CSV