FLORID: A DOOD-System for Querying the Web*

Rainer Himmeröder Georg Lausen Bertram Ludäscher Christian Schlepphorst

1. Introduction. The limitations of locating and extracting useful information from the Web using current techniques are well-known: By *browsing*, only navigational access is possible, which is quite tedious if information is scattered across different pages. The use of index servers like AltaVista or HotBot for *searching* yields (often overwhelmingly) many results but does not exploit the user's knowledge of the structure of the Web. As a consequence, several Web query languages have been proposed to overcome the shortcomings of simple browsing and searching techniques. Languages supporting deduction and object-orientation seem to be particularly well-suited for querying and reasoning about structure and contents of semistructured Web documents: Deductive rules support a declarative, data-driven programming style for exploring the Web (in particular, navigation along paths of arbitrary length), while object-oriented features allow a flexible modeling of semistructured and partial information.

2. *FLORID* [1] (***F-LO**gic **R**easoning **I**n **D**atabases*) is an implementation of *F-logic* [KLW95], a prominent DOOD-language providing complex objects, uniform handling of data and metadata, rule-defined class hierarchy and signatures, nonmonotonic multiple inheritance, equation of objects by rules, and variables ranging over methods and classes. Moreover, *FLORID* extends F-logic by *path expressions* [FLU94] which turn out to be particularly useful for navigating on semistructured data. All information about a given object is stored in an *object frame* which is identified by the object's unique oid. Several (external) object names may refer to the same oid, thus defining an equivalence relation between objects. Frames consist of *slots* for method applications or class membership enabling schema modifications during the evaluation by deriving new signatures, subclass relationships, and class memberships.

Recently, *FLORID* has been extended to provide a declarative semantics for querying Web documents [HLLS97]. The proposed extension allows extraction and restructuring of data from the Web and a seamless integration with local data. Due to (i) a simple yet powerful predicate for matching regular expressions, and (ii) the (meta) features of F-logic, there is no need for separate languages for wrappers and mediators.

3. The Web Model of *FLORID* distinguishes between the class url of all URLs and webdoc, the class of actual Web documents. For these, the signatures

$$\mathsf{url}[\mathsf{get}\Rightarrow\mathsf{webdoc}] \qquad \text{and} \qquad \mathsf{webdoc}[\mathsf{hrefs}@(label) \Rightarrow\!\!\!\Rightarrow \mathsf{url}]$$

are crucial:

A Web document is accessed and added to the local F-logic database by defining the method get for an instance u of class url in the head of a rule, thereby creating the new oid "u.get". Since exploration of the Web is completely data-driven, a seamless integration with the bottom-up evaluation strategy of *FLORID* is achieved. The fetched Web document u.get becomes an instance of webdoc, and a number of reserved methods like author, modif, and hrefs@($label$) may become defined for it (depending on the result of accessing the document). The multi-valued method hrefs@($label$) holds all outgoing hyperlinks for a given label. After u.get has been created, it becomes an ordinary object in the database and, using all features of F-logic rules including path expressions, the user may define arbitrary new methods for it. To this end, data can be extracted from a Web document using a built-in relation match for matching regular expressions.

*Authors address: Institut für Informatik, Universität Freiburg, Am Flughafen 17, D-79110 Freiburg, Germany; email: {himmeroe,lausen,ludaesch,schlepph}@informatik.uni-freiburg.de

[1] Available from http://www.informatik.uni-freiburg.de/~dbis/florid/

4. Demonstration. The concise, declarative programming style is illustrated by examples like the one sketched in Figure 1, where data from the CIA World Factbook is extracted, queried, and restructured: On the right, some source pages are shown; the corresponding *FLORID* program and some answers are depicted on the left. Other queries are used to illustrate schema browsing and integration of different data sources.

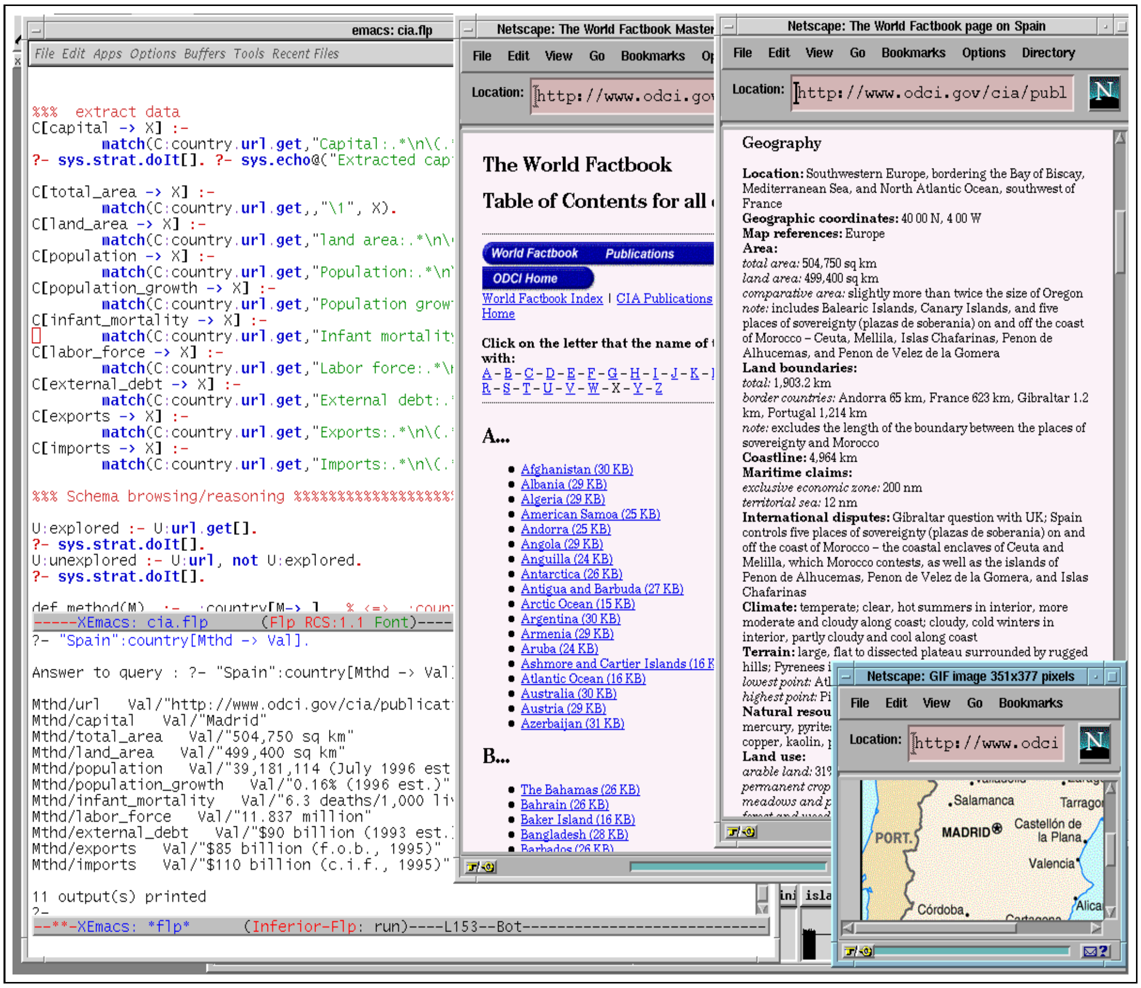

Figure 1: Querying the CIA World Factbook with *FLORID*

References

[FLU94] J. Frohn, G. Lausen, and H. Uphoff. Access to Objects by Path Expressions and Rules. In *Intl. Conference on Very Large Data Bases (VLDB)*, Santiago de Chile, 1994.

[HLLS97] R. Himmeröder, G. Lausen, B. Ludäscher, and C. Schlepphorst. On a Declarative Semantics for Web Queries. In *Intl. Conference on Deductive and Object-Oriented Databases (DOOD)*, LNCS 1341, pp. 386–398, Montreux, Switzerland, 1997. Springer.

[KLW95] M. Kifer, G. Lausen, and J. Wu. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of the ACM*, 42(4):741–843, July 1995.