Poster B-11

Implementation of a query processing model and view based data integration system

Authors:
Myung-Eun Lim (*Bioinformatics Team, Electronics and Telecommunications Research Institute*)
Myung-Guen Chung (*Bioinformatics Team, Electronics and Telecommunications Research Institute*)
Yong-Ho Lee (*Bioinformatics Team, Electronics and Telecommunications Research Institute*)
Soo-jun Park (*Bioinformatics Team, Electronics and Telecommunications Research Institute*)

Short Abstract: We designed a query processing model to dispersed biological databases and implemented data integration system based on mediator-wrapper concept. To reduce the response time when extracting data from data source, we induced a kind of wrapper cache. By using them, we got a reasonable response time in extracting data.

Long Abstract:
Numerous biological data are being flowed out from laboratories and they are served via web. As each database is built to its own purpose, data have heterogeneity in their formats and expressions. So, there needs to be a method for researcher to collect related information from various data sources without reference to their forms and locations. We designed an integration model based on mediator-wrapper concept and implemented view based integration system to solve out a data integration problem in biological domain.
The system provides functions of defining a view of data, querying to a predefined view, crawling original data, and assembling result data. Before user queries to the system, the target database has to be created. We designed a description language, WDL(Wrapper Description Language), to provide a method to describe the source database as a standard view in the system. Using the expressions of WDL, user can organize data elements and describe nested relations of elements in a database. User can also describe a rule of extracting data from real data source. This rule is applied in wrapper execution time to extract data in a predefined format. After the WDL document is written, wrapper and Meta information is generated. Wrapper is a module that extracts data as is defined in WDL from the real data source. Schema and mapping rule comprise the Meta information. Schema represents the XML Schema of a result XML document extracted by wrapper and is used to check whether generated result is valid or not. Mapping rule is used to separate user query into local wrapper query.
The query language of the system is a kind of simplified XQuery. Original XQuery is powerful in expressing user's requirement of database. But the full specification of XQuery is complicated for biological user, we exclude some features - user defined function, advanced data type, and so on. Still, the basic structure of FLWOR is maintained. When user query is given, the mediator parses it and divides into several wrapper executable queries which are also a kind of simplified XQuery. As we gain data not from the warehouse or any local database but from the wrapping of real-time data, wrapper execution schedule is very important not to spend much time in extracting data. Especially, wrapper should not repeat extraction step of previously extracted data in same querying stage. To overcome the

duplicate extraction using one wrapper, we assigned a wrapper cache to store temporal extracted data. In a certain period, wrapper maintains extracted data and reuse if it is requested. By maintaining the wrapper level cache, we can get better performance in response time. After the extraction stage, the result of the query is made by result assembler. The schema is used to check validity of the result document and assemble results.

In the experiment, we made some scenarios to test the performance of processing complex queries. The complex query, such as 'When unknown amino acid sequence is given by the DNA chip experiment, find related gene and Gene Ontology(GO) information. And then, print out protein information referenced in GO', is tested and the system generate result in a reasonable time.

We designed and implemented query processing model and view based biological data integration system. Still, we need more experiments to test the performance of the system and have to revise a method to reduce the data extraction time from dispersed data source.