CS348 Project Phase 2

Daniel Burstyn (dmbursty, 20206120)
Yubin Kim (y12kim, 20239608)

November 27, 2009

1 Utility Functions

Instead of working out complex SQL queries, we chose to use very simple queries that we stich together in code. This approach promoted code reuse, and was easier to implement and made the code easier to read. We split up the code and moved a lot of shared functionality into small simple methods in a Utils class.

The Utils class has three distinct components. The first is a simple method to get a Connection object to the db2 database. The other two are collections of methods for retrieving bits of information from the database, and for printing the publication data.

The methods to retrieve data about publication were all non-trivial and used by both bibauthor and bibmaint, so it made sense to factor them out. These methods are:
`determineType(pubid)`: Determines the type (article, journal, book, or proceedings) of a publication by querying each of those tables.
`getTitle(pubid)`: Finds the title of the given publication.
`getYear(pubid, type)`: Determines the publication year of the given publication, and will recurse on appearsin to find the publication year of an article.

For printing, we had a utility function that prints all of the data of a publication according to the specification for bibauthor. The data is passed inside a Publication object so that some analysis can be done in code before printing the data (this was especially useful for sorting.)

2 bibauthor

The bibauthor program somewhat tricky considering the data needed was spread across the entire database schema. In order get the printing right (handling sorting, authors and duplicates) we used a Publication class that stored all the relevant publication data. The code starts by retrieving a list of publications made by the specified author. It then makes a Publication object for each pubid returned, and populates the object through a number of small queries. The authors are sorted by aorder, and then the publications are sorted by first author and year. This is done through comparator classes. Finally, the publications are printed with the duplicates ignored.

3 bibcontent

bibcontent was much simpler than bibauthor. We simply query the article table for all the articles that appear in the given publication. A Publication object is created and populated with the data, and then printed. The SQL `order by` clause was used for ordering, so the code itself ended up being quite short.

4 bibmaint

The output for bibmaint is done through the same method as bibauthor. bibmaint does not do input error checking and will always assume that the input is in the correct format and does not contain inappropriate commands. (For example, inserting a book with aids that do not exist will will cause bibmaint to fail.) In general, bibmaint will check if a given aid or pubid exists, and if it doesn't, inserts a new row, and if it does, it updates the row with the given information. The exception to this rule is authorurl, which can only be used with updates.

5 Work Division

Max Burstyn (dmbursty)

- **Submission**
- bibauthor
- Utility functions and Publication class
- Design document write up

Yubin Kim (y12kim)

- bibcontent
- bibmaint
- Source control and compile/run setup