
BibGallery – A DBLP-based Bibliography Visualization

Group 03 Project Deliverable 1

MINGHAO LU, MINGYU KANG, YICHEN LIANG, YOULYU
ZHANG @ UNIVERSITY OF FLORIDA

Contents

1 Project Overview	2
2 Software Features	2
3 Example Queries	3
4 Potential Challenge	3
5 Division of Labour	4

1 Project Overview

We have reached the consensus the final product of our project will be a single-page application (SPA) that allows the user to search for the bibliographic information on major computer science publications.

All available data are currently parsed and imported from the DBLP Computer Science Bibliography, including all stored articles, journals, conferences, monographs, citations and authors.

The project in general will be divided into two associative sections:

1. Web-based front-end
 1. Simple user interactions with the GUI.
 2. Display data table or diagram in a neatly formatted web page.
2. Back-end
 1. Takes on users' requests from the front-end.
 2. Performs queries and retrieve data from the database.
 3. Generates html segments (to tackle with performance issue in front-end).
 4. Sends the result to the front-end.

2 Software Features

The native users of our application will be able to sign up an account, log in and access to the stored information. For example:

1. Information about a specific author, conference, and paper.

2. Data presenting the relationships between any paper and author, paper and conference, author and conference.
3. Trend of activities of a specific author.
4. Trend of the frequency of a paper cited in the whole database.

The front-end will be implemented by the React framework, in the format of a single-page application that features:

1. A data table or graphic representation of the returned query results (possibly done with Highcharts).
2. Simple and straightforward navigation to page and application features.
3. The ability allowing the user to apply filters to the query results. (This will be eventually achieved by automatically modify generated queries in the back-end.)

The back-end will be implemented in Node.js/Express. When the front-end receives a request from the user, a set of queries will be generated and sent to the database to fetch the stored data. The results will be pipelined from the database back to the front-end afterwards.

The database we use are provided by the course instructor on Oracle CISE.

3 Example Queries

1. Find the author Joe Smith's publications (journals, books, papers) from 2001 to 2007.
2. Find all monographs cited the paper *Water Surface Stimulation*.
3. Find all the coauthor that has collaborated with John Doe.
4. Find the top 10 referred paper, which contains word "cryptography" in their title.
5. Find which conference or journal has the greatest amount of paper about "cryptography" published.
6. Show paper popularity trends, based on the frequency of citation every year.
7. Find the author who published the largest number of publications from 2001 to 2010.

4 Potential Challenge

1. Space limitation: we have ~2.1 GB data in total that needed to be parsed from the DBLP to have a complete database, meaning complete tables that can successfully interact with each other by instructions.

2. Database Schema: a further discussion of the conceptual structure of the database is required. Certain relations can be reduced into simple attributes of an entity. We must thoroughly plan out our design to avoid redundant and unnecessary processes.

5 Division of Labour

There are four tasks, two derived from front-end and two from back-end implementation.

- Front-end tasks:
 1. User login and info storage & user access to data
Assigned member: Yichen
Support member: Youlyu
 2. Display of user request on the single-page web application and filtering
Assigned member: Minghao
Support member: Yichen
- Back-end tasks:
 3. Server DevOps & Database communication
Assigned member: Youlyu
Support member: Mingyu
 4. Potential requirements & API design
Assigned member: Mingyu
Support member: Minghao