

# Speaker Pages For CoMeT System

Independent Study Report 2930 spring 2013

Name: Yu Xia

Supervisors: Dr. Peter Brusilovsky

Chirayu Wongchokprasitti

## The goal for the independent study

The website is a talk-relative and speaker-relative network. However, it just provided a talk-relative content in the web pages. So the final goal for the study is to add a speaker-relative content to the website and link them together. That is, users can browse the speakers whom are related to the specific talks and view the speakers' detailed profiles on the webpage. There are a lot of works to make the web pages user-friendly. Based on the 'divide and conquer' idea, I divided the big problem into some sub-problem. The list below is the sub-goals:

- 1) Create a webpage to display the speaker list
- 2) Create a webpage for each speaker as the speaker profile
- 3) Design and make the webpages more friendly use
- 4) Complete the speaker data in the database
- 5) (Option) Implement the speaker recommender function  
Or speakers' data visualization

## Introduction to CoMeT

CoMeT is a new website that build at the School of information science by the PAW lab which is leaded by Dr. Peter Brusilovsky. The website is designed for those students who want to attend to a talk, but cannot find one. Also, speakers who give a talk on campus can also go into this system to see their history of the talks. In the website, it will provide a detailed information about where the talk is, what the content is, who is the speaker and the brief information about the talk. Right now, it is added a feature that it can recommend talks to users based on the users' profile. This makes it become a recommender system. Students do not have to go through all the talks and find an interesting one. All they need to do is build their own profile and the RSS feed will give the recommended talks. It is like a social networking for the students in Pittsburgh. Users can subscribe talks and speakers. Also, they can add series and groups. In addition, Users can login the website via different sources, such as LinkedIn, Google+, Facebook etc.

This system benefits the students and the speakers. Firstly, it recommends talks to students that it can give suggestions on talks that are given on campus. In this way, it saves users time since users do not have to go through all the talks. Secondly, speakers do not have to worry about there is no attendance. They don't need to spread sheets. The system provides them a platform that they can post a new talk to the page and every student can see it. In addition, speakers can see their information on the pages. In this way, they can get a landscape about how students like him or her and how popular is his or her talk.

## Technical Detail of the website CoMeT

### *Development Environment and techniques*

- 1) JSP with Struts
- 2) MYSQL
- 3) JQuery
- 4) Ajax

### *Delivery*











<http://halley.exp.sis.pitt.edu/comet/index.do>

## What I am did

What I did is make a page that lists all the speakers that were gave a talk on campus and another page is for display the detailed information of a speaker. There are a lot of work to do in these two page, such as connect to the database, link to different sources of database, information retrieval etc. I will show you the details of what I did.

## Display the speakers with two columns

In the speakers display page, a user will not browse all the information of a speaker. The interface can guide users to the information that can identify a speaker. So a name and an affiliation will almost cover all the situations.

 <p>Name: <a href="#">A. Ercument Cicek</a> Affiliation: Research Assistant, Dept. of Electrical Engineering and Computer Science, Case Western Reserve University</p>	 <p>Name: <a href="#">A. R. Ingraffea</a> Affiliation: School of Civil and Environmental Engineering, Cornell University</p>
 <p>Name: <a href="#">Aabid Shariff</a> Affiliation: Joint Carnegie Mellon University, University of Pittsburgh Ph.D. Program, in Computational Biology</p>	 <p>Name: <a href="#">Aaditya Ramdas</a> Affiliation: Machine Learning Department</p>
 <p>Name: <a href="#">Aapo Kyrola</a> Affiliation: Computer Science Department</p>	 <p>Name: <a href="#">Aapo Kyrölä</a> Affiliation: Computer Science Department</p>
 <p>Name: <a href="#">Aaron B. Phillips</a> Affiliation: Language Technologies Institute</p>	 <p>Name: <a href="#">Aaron Dutton</a> Affiliation: MPIA</p>
 <p>Name: <a href="#">Aaron Henderson</a> Affiliation: Assistant Professor of Studio Arts</p>	 <p>Name: <a href="#">Aaron Katz</a> Affiliation: US Department of Defense, Ames Research Center</p>

However, if the information display in one column, it occupies too many space in that users browse less records in one page, which is less efficiency. So displaying speakers with two columns is a great idea. To achieve this, I used an integer to count the sequence of the record, whether it is an odd or an even. So it can be displayed on the left side or the right side. In addition, the photo of the speaker is based on the URL in the database. If URL does not exist in the database, it will display a default avatar that looks nicer than a unfound picture icon. Another problem is that the photos have different size; I use a function that is developed in Javascript to control the size of photos.

### *Display the speakers by initials of names*

The name initials navigation bar is a common navigation tool to help users find a speaker. The navigation bar displays the speakers separately by initial letter of their names.

Filtered by Name: **A**|[B](#)|[C](#)|[D](#)|[E](#)|[F](#)|[G](#)|[H](#)|[I](#)|[J](#)|[K](#)|[L](#)|[M](#)|[N](#)|[O](#)|[P](#)|[Q](#)|[R](#)|[S](#)|[T](#)|[U](#)|[V](#)|[W](#)|[X](#)|[Y](#)|[Z](#)|[All](#)|

In this way, users can find their interesting speakers more quickly and more accurately. It is also intuitive for users to recognize the current browsing page, when selected a letter, the letter becomes bold, black color and cannot be clicked. This is to imply the users that the letter is being selected and avoid the users clicking it again. In case of some users want to see all the speakers, there is still an option to display all the speakers in one pages. To implement this function, I use a variable to store, which letter is being select and every time a page is opened, the variable will be imbedded into a SQL statement, and display the speakers. Also there is a method to check which letter is selected and changed the letter's font and disable the link of the letter.

### *Implement the paging function*

Even though the initial letter of their names separates speakers, it is still too many speakers for some letters. So I implement a paging function for the web page.

[First Page](#) | [Previous Page](#) |  | [Out Of 17](#) | [Next Page](#) | [Last Page](#)

Users can go to the first page or the last page directly. In another way, they can click the previous page tag or the next page tag to browse the speaker page by page. The core idea of this function is to use the 'LIMIT' command in SQL statement. For each page, it is constricted to display a fixed number of speakers. Then I used a variable to store the page number, every time the 'next page' is clicked, the variable add one. It minuses one in 'previous page'. When clicked 'first page' or 'last page', the variable turn to default value or the maximum value.

In addition, when it is the first page, in terms of simplification, the First Page button would not show up, so is Previous Page Button. It is the same for Next Page button and Last Page button if it hits the last page. The front-end language controls this.

However, it is possible that users want to go to specific page. That is reason why I design an input text. In this way, users can go to the page number that they want to. All of the tools above, navigation bar and paging buttons are to make the page simpler and to let users browse speakers list easily.

### *Display the speakers by different filtering*

Users do not browse speakers with no reason. Users may want to find the speakers by upcoming talks, popular talks, and the users' subscribe, the users' series subscribe or users' groups subscribe. The hardest part is that a user can add several conditions on a display query. To implement this function, I separate the SQL statement into three parts. The first part is from SELECT to JOIN. The second part of the SQL is from JOIN to WHERE. The third part is the GROUP BY and ORDER BY keywords.

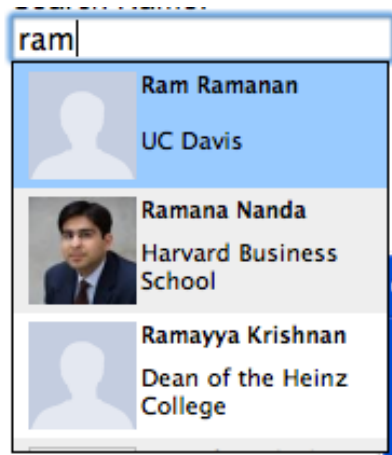
#### **Show Speakers By**

- Upcoming Talks
- Popular Talks
- Your Subscribe
- Your Series Subscribe
- Your Groups Subscribe

To check each of the conditions, I added different conditions into one SQ statement. Then executing the SQL statement, I can get the expected result. However, with multi-constriction, there is probably showing seldom records. So when using this multi-constriction navigation bar, the filtering of initial letter will default set as display all.

### *Implement the search function*

For the purpose of finding a specific person, I add a search text field for users to find the speaker they want.




By using the Ajax, the accomplish function in JQuery. It will display the similar name based on the input field and users can just click it to find the person without typing in the whole name. If a users want to find a person, this is the shortcut.

### Create the speaker profile page

In a speaker profile page, there are the speaker basis information, his talks or upcoming talks and recent activities. In the person information area, there are photo, name, affiliation, Google Scholar link, Microsoft Academic link and speaker's interest. These information are displayed depend on whether it has a record in the database.


#### A. Ercument Cicek's Info Subscribe

	Name:	A. Ercument Cicek
	Affiliation:	Research Assistant, Dept. of Electrical Engineering and Computer Science, Case Western Reserve University
	Talks:	1

---

#### Schedule

**Monday, Mar 18**

**23** views [ADEMA: An Algorithm to Determine Expected Metabolite Level Alterations Using Mutual Information](#) 

By: [A. Ercument Cicek](#) [Subscribe](#), Research Assistant, Dept. of Electrical Engineering and Computer Science, Case Western Reserve University at: **11:00 AM - 12:00 PM**

Location: **Reddy Conference Room 4405 Gates and Hillman Centers**

Posted to groups: [Computational/Systems Biology](#), [Bioinformatics](#)

---

#### Recent Activities

[comet.paws](#) posted [ADEMA: An Algorithm to Determine Expected Metabolite Level Alterations Using Mutual Information](#) [Bookmark](#)  
March 12 at 11:01 PM

In addition, if the database has records for speaker's papers, it will shows the top three cited papers in this page. Users can view the speaker's Google Scholar homepage, Microsoft Academic Search homepage, and the other authors in the same area.

The schedule function, recent activities and feed function are quoted from other pages. For schedule function, it shows the talks that belong to this speaker. In the window, users can view the details of the talks, bookmark the talk, subscribe the speaker or view the groups' information. For recent activities, it shows the bookmark behaviors and subscribe behaviors that belongs to the talks and the speaker.

If a user is interested in a speaker, they can use RSS Feed to follow the activities of a speaker.



### *Implement the speaker subscribe function*

Besides the RSS feed with the speaker, users can also subscribe interesting speakers. This function will check a value in database. So once a user subscribes a speaker, the state will be changed on other pages. In case those users may confuse that if they are subscribe a speaker. Once the button is clicked. The value will be changed to 'unsubscribe'. Also the color and the font of the button will be changed.

Name: A. Ercument Cicek [Subscribe](#)

Name: A. Ercument Cicek [Subscribed](#)

[Unsubscribe](#)



### *Correct the data in database*

In the database, there are some flaws with the speakers' names and affiliations. There are duplications in the names, which will confuse users. I delete the same speakers and put the talks into one speaker. In addition, it exists some affiliations with useless prefixes, such as PhD, MD etc. All of those things needed to be deleted.

What's more, there are some complicated SQL statements to achieve a function. However, these SQLs may be too complicated to run it in a short time. So I need to optimize the database, to create the temporary tables to divide the problems.

### *Revise the crawling code for CoMeT*

There are crawlers to retrieve from Google Scholar and Microsoft Academic Search. The crawled data are stored in the scholar database. However, I revise it to store data in the colloquia database which I can do the mapping between the speaker profile id and the speaker id in the system. In this way, I can use the Google and Microsoft data.

In addition, the original code do not crawl the data such as citation number, the trend of paper citations for years, the ranking of the speakers in their own areas. With those data, I can show them in the speaker profile page. Users can have a more general aspect about the speaker. On another hand, those things are the criteria to evaluate speakers. In this way, Users are leaded to the speakers that they are interested in also an authority in their areas. In a way, speakers may also emphasis on those things that can reflect their capability.

However, the speed of crawling data is in a low speed because Google and Microsoft are checking whether it is a bot crawler. So, for now, in the page, I need to check if it has been crawled by the program. If it is, it will combine the information together in one page.

### *Implement the co-author function*

Right now, I added a co-author function into the speaker profile. The data is using the Scopus dataset. However, since the structure of the database is complicated. The query takes a long time to finish the process. So I divided the process into sub-process. Some of the matching work is done advanced. A more accurate mapping between speaker in colloquia and scholar database is still needed. I will improve the function in the future.

---

#### **Co-Authors**

Gary Marchionini  
Ahmed Abbasi  
Eugene Agichtein  
Khurshid Ahmad  
Azzah Al-Maskari  
Gianni Amati  
Sihem Amer Yahia  
Shlomo Argamon  
Daniel Ashbrook  
Paolo Atzeni

Also, there is no homepage for authors in the Scopus database. I think it'd better complete the dataset so that users can view co-authors profile via this speaker's page.

## Conclusion and Future Work

In conclusion, the speaker functions are now working well. The webpages are friendly-design and useful for students. Users now can view more information about a speaker than before. Also, they can focus on a few speakers who are related to a few talks. I learnt a lot from the process of problem shooting. I get the idea that how to build a website and how to make the UI more friendly, as well as the experience and techniques that implement those webpages and design.

According to the webpages that are already built, we can expand functions horizontally. In the speaker profile page, we are trying to link the page with Google Scholar, Microsoft Academic, Mendeley since there is a small dataset for speakers' information in the database. In the future, we need to expand our speakers' information (actually we are doing it right now), which is to display more information on each speaker's profile page and give users a more clearly image on specific speakers. In addition, to link the page with different kinds of academic website and to get more data from those website will make our pages more effective and efficient.

Last but not the least, we can do some profound things if we have enough data, such as speaker recommender function, speakers' data visualization.