Linnæus University
School of Computer Science, Physics and Mathematics

Degree project

Visualization of LNU's Publication Network

Author: Sun Wenyi &Yu Chunmiao
Date: 2011-08-25
Subject: Computer Science
Level: Bachelor Degree
Course code: 2DV00E

Abstract

DiVA, Academic Archive On-line, is a website which can provide the information from the most academic publications of Swedish Universities. The information includes the title, author, publication data, and so on. The aim of this project is to design a tool to visualize the co-authorship publication network and be able to transform the data into a more readable form.

For parsing and visualizing data, the tool adopts the "InforVis Reference" model. Here a set of interaction and visualization technologies are adopted, so that it can create view base on user's query with increasing the usability. In this thesis, we present a "use case" by applying our tool to visualize a part of articles published at Linnaeus University and to illustrate the capability and functions of the tool.

The tool can also provide user a comfortable operating environment with a quick searching speed and high efficiency; it can be used anywhere, if internet is available.

Keyword: Information Visualization, DiVA, co-authorship, Dynamic image, Sweden, CSV file, zoom in, zoom out, spring.

Acknowledgments

Thanks our supervisors Prof. Dr. Andreas Kerren and Ilir Jusufi for supporting us at the University. And Zhou Hong, a software engineer of Jinyuan Company.

Contents

Abbreviations

● **InfoVis:**	Information visualization
● **CSV:**	Comma-separated values
● **DiVA:**	Digitala Vetenskapliga Arkivet
● **URL :**	Uniform Resource Locator
● **PID:**	Publication ID
● **HTML:**	HyperText Markup Language

1. Introduction

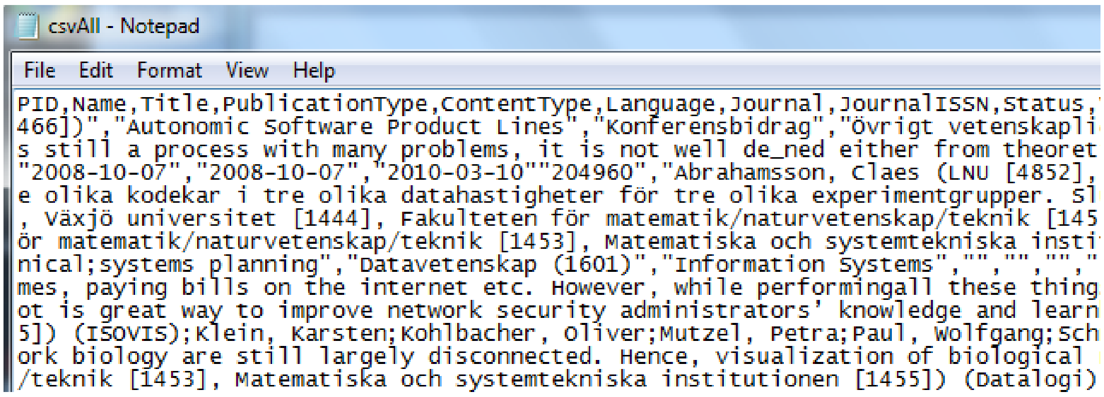
With technological development, the data sets that people get access is becoming larger and larger, so transforming the data into information and make them useful for people is urgent and important. This thesis introduces an example of implementing a visualization approach to help user get useful information from the Academic Achieve-DiVA.

DiVA is an institutional repository for research publications and student thesis written at 28 universities and colleges of higher education (from 2006 till now) in Sweden. It is a powerful publication searching engine that can offer a data file by user's query. In this chapter we will introduce the motivation and objective of this project.

1.1 Motivation

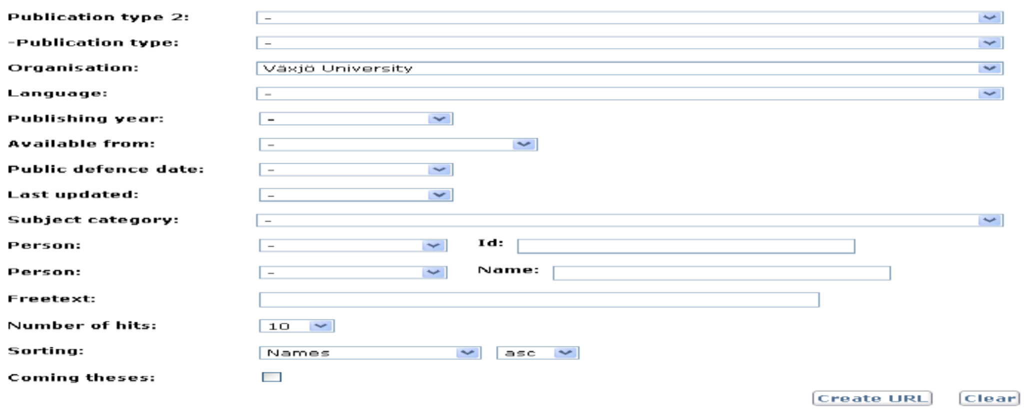
As we mentioned above, DiVA can provide different kinds of files (CSV, RSS, Atom, HTML-div, etc) according to user's query. We select CSV file, because it contains the richest information of publications like title, author, and publication type, etc. But there are two main problems:

- CSV file contains lots of information people do not concern about like publication ID.
- The searching is inconvenient, because the user has to enter information for accurate searching and could not see the comparison between different objects directly.



```
PID,Name,Title,PublicationType,ContentType,Language,Journal,JournalISSN,Status,
466])","Autonomic Software Product Lines","Konferensbidrag","Övrigt vetenskapli
s still a process with many problems, it is not well de_ned either from theoret
"2008-10-07","2008-10-07","2010-03-10""204960","Abrahamsson, Claes (LNU [4852],
e olika kodekar i tre olika datahastigheter för tre olika experimentgrupper. Sl
, Växjö universitet [1444], Fakulteten för matematik/naturvetenskap/teknik [145
ör matematik/naturvetenskap/teknik [1453], Matematiska och systemtekniska insti
nical;systems planning","Datavetenskap (1601)","Information Systems","","","","
mes, paying bills on the internet etc. However, while performingall these thing
ot is great way to improve network security administrators' knowledge and learn
5]) (ISOVIS);Klein, Karsten;Kohlbacher, Oliver;Mutzel, Petra;Paul, Wolfgang;Sch
ork biology are still largely disconnected. Hence, visualization of biological 
/teknik [1453], Matematiska och systemtekniska institutionen [1455]) (Datalogi)
```

Figure 1.1: Part of CVS which offered by Diva



Publication type 2:	-
-Publication type:	-
Organisation:	Växjö University
Language:	-
Publishing year:	-
Available from:	-
Public defence date:	-
Last updated:	-
Subject category:	-
Person:	- Id:
Person:	- Name:
Freetext:	
Number of hits:	10
Sorting:	Names asc
Coming theses:	☐

Figure 1.2: Create link feeds in DiVA [1]

As shown in Figure 1.2, the user has to input some keywords in DiVA's "Create link feeds" and get the download link which can download a CSV file (if CSV is selected in "Format"). Otherwise, Växjö University changed its name into Linnaeus University two years ago, and in DiVA, these two "different" universities exist at the same time, but user can just select one of them each time. The problem is that if an author worked in Växjö University before and is working in Linnaeus University now, and a user wants to search this author's publications, he can just search the publications which are finished in the period of Växjö or the period of Linnaeus separately. The user can not search the publications in both of the two periods at one time. Our project will solve this problem by changing the link which is used to connect to DiVA, see section 5.2.1.

1.2 Goal

Every author may have one or several publications; every publication may have one or several authors; each author may also have co-authors. The goal of this thesis is to implement an interactive widget for visualization of all these relationships in LNU University's publication network. The result is a widget tool which will be opened on web browsers that can automatically connect with DiVA, extract and parse data according users' need, at last visualize an appropriate and understandable form of the whole data set. Then the user can query and search the information which they concern by entering the key words in the "TextArea" of the interface, or directly operate the view by dragging, click and so on. When user click one publication, the publication and its authors will be highlighted, at the same time the information of this publication will be shown in a "TextArea"; when user click one author, his or her publications and co-authors will be highlighted, and the information of this author will be shown in "TextArea", too. And finally, users can have further selections.

1.3 Objective

The solution for LUN's publication network visualization supports a user in doing a high efficiency searching. Its development was driven by the following criteria.

- **Usability**

The program has a good GUI providing functions of searching, querying. Users can accurately find the information they want by directly operating on the visualization view or entering key words and doing selection in control panel, and at last getting results in forms of words and graphs, so that the user can interact with the program easily.

- **Simplicity and efficiency**

The code and operation is not complex, so that it can save time and computational power. The code should be readable for other programmers and users even know nothing about computer technology can use it without much help.

1.4 Used programming environment

In this project we use Java and Processing to implement it.

- Java is a popular Object-oriented programming language, free and open source.
- Processing is based on Java, designed by Caseey Reas and Ben Fry. It is "*an open source programming language and environment for people who want to create*

images, animations, and interactions.”**[2]**

1.5 Structure of the thesis

This thesis is about the visualization of LNU’s public network, it contains three chapters: The theoretical part (chapter 2) introduces the InfoVis Reference method and related work. The second is the practical part. It introduces how we design the whole final plane, what problems we met and how did we solve them, and explains the detail of those functions and codes (chapter 3, 4, 5). The third part (chapter 6) is the conclusion and future work.

2. Theoratical Background

*"Visualization is more than a method of computing! It is a process of transforming information into a visual form enabling the viewer to observe, browse, make sense, and understand the information. It typically employs computers to process the information and computer screens to view it using methods of interactive graphics, imaging, and visual design. It relies on the visual system to perceive and process the information. Visualization is more than pretty pictures. This is not to lessen the importance of visual aesthetics. The latter is quite important in making the user like to look at the information. However, the beauty of an effective visualization is more than skin deep."***[14]**

2.1 InfoVis reference model

The InfoVis reference model is a kind of visualization model. It is a process of transforming information from raw date to view. "*Reference model for visualization. Visualization can be described as the mapping of data to visual form that supports human interaction in a workspace for visual sense making.*" **[4]**

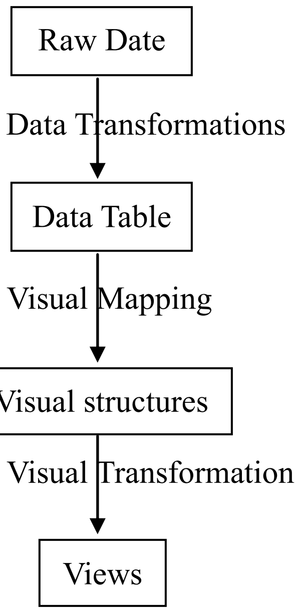


Figure 2.1: InfoVis Reference Model

- **Raw Data**

The Raw data or called specific idiosyncratic formats, collected from the daily life or researches, and not yet organized into a form in which it can be easily used or understood.

- **Data Table**

"*relations (cases by variables) + metadata"***[4]** "*A data table is a visual instrument comprised of labeled columns and rows, used to arrange information contained in a computer's database. It may serve to organize disparate data, as well as to permit data to be easily manipulated and analyzed.*"**[13]**

- **Visual Mapping**

"*A mapping is more effective than another, if it is faster to interpret, can convey more distinctions, or leads to fewer errors.*"**[4]** In this process, the data table will be parsed and useful information will be filtered out and stored.

- **Visual Structures**

"*spatial substrates + visual elements + graphical properties*."**[4]** It is a structure suitable for visual features, to represent data by image to make it vivid and more understandable and impressive.

- **Views**

*"graphical parameters (scaling, zooming, clipping, …)"***[4]**

2.2 Related work

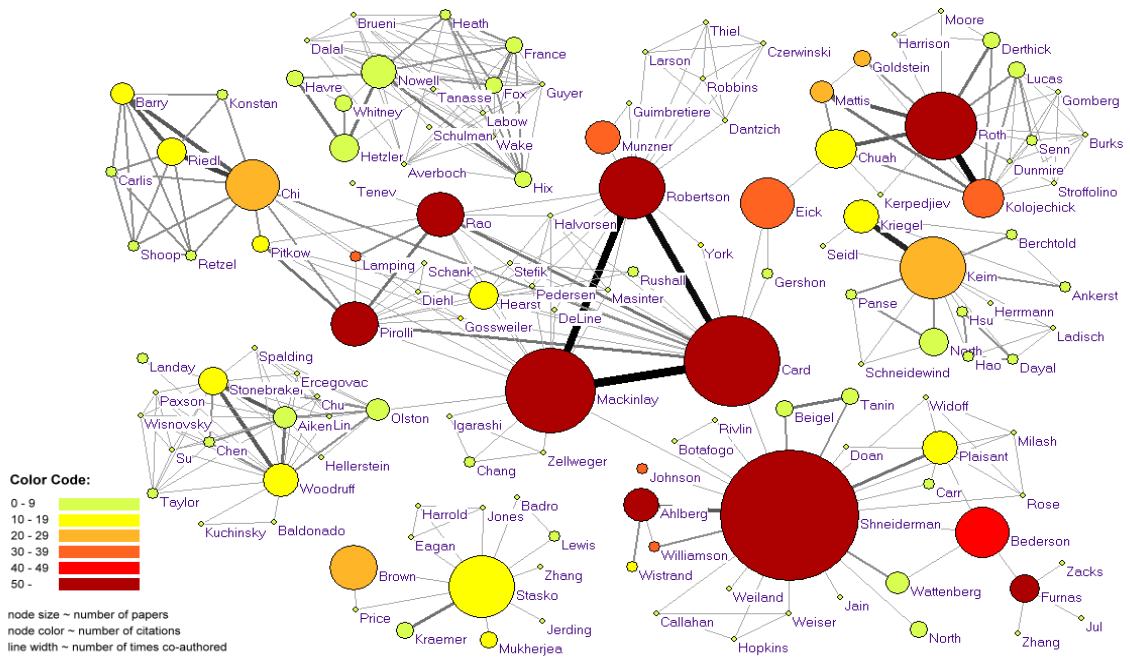


Figure 2.2: Co-author space of highly productive, cited or co-authoring authors. (Taken from **[6]**)

As shown in Figure 2.2, Weimao. Keet al. **[6]** introduce this tool to visualize a co-author network for authors who published no less than 10 papers or got cited no less than 50 times or have no less than 20 occurrences of co-authorship with other authors.

In this view, size of node represents the number of articles, larger the size more articles the author has. The color represents number of citation, warmer the color is, more citations the author has. Lines are being used to connect author with his co-author. And the line's width represents the number of times co-authored.

This view can clearly show the comparison about who has more articles or more citations, and can also show the strength of the relationship between author and co-author. But when we want to absorb this method for our task, there are some problems exist.

- Since the authors scattering on the canvas without discipline, it may take a bit long time to find a certain author.
- Users know line width represent number of co-authors and node size represent number of papers, but the user needs a criterion to tell them how width or how large

the size means to how many.

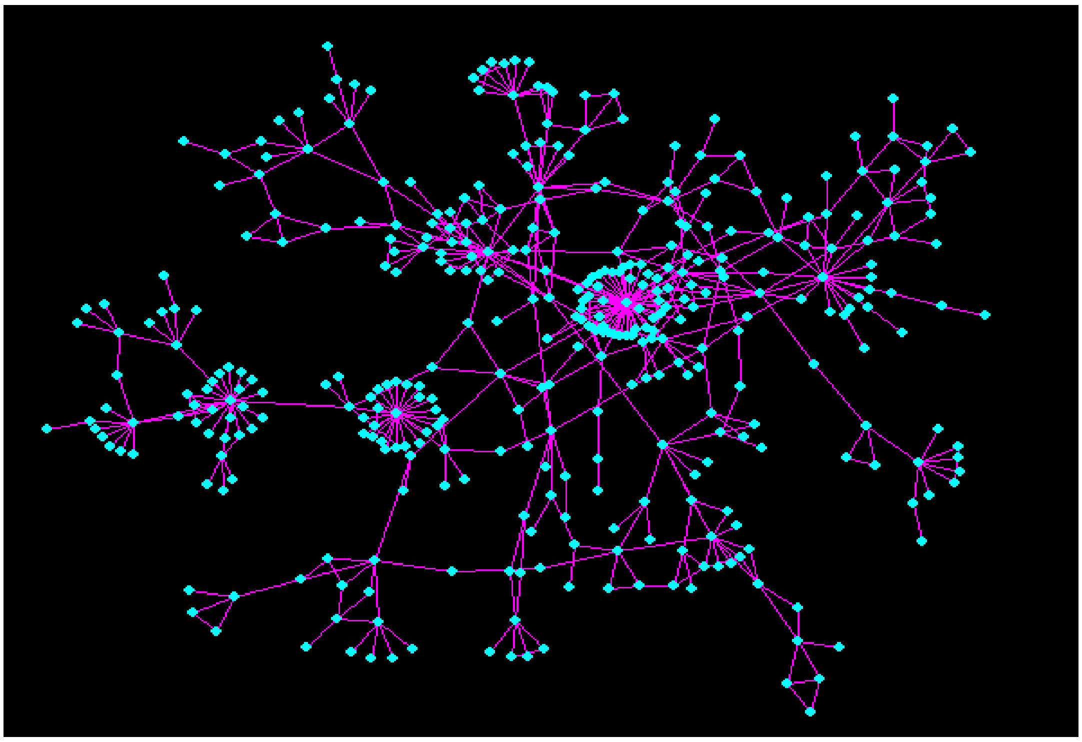


Figure 2.3 A 380-article PFNET of a document co-citation network, containing 523 links (node/link ratio 0.73). The entry threshold is 12 or more citations for each article. (Taken from **[8]**)

As shown in Figure 2.3, Chaomei Chen and Steven Morris compare the visualizations of co-citation networks of scientific publications derived by minimum spanning trees (MSTs) and Pathfinder networks (PFNETS) **[8]**, so they give such an example. It is a 380 node PFNET of author co-citation network, containing 525 links. It used direct co-citation counts as link weights. Nodes represent articles, lines connect to them means they have co-citation relationship. We can see the nodes in the centers of the clusters, means they have lots of co-citations. Meet first citation will lead to meet higher and broader range of network. The relationship between publications shown clear, people can see which one is citied most times and which two publications related to each other, but a low node/link ratio (0.73) means that this PFNET has lots of links, so it will lose the advantage of a Pathfinder network as the PFNET drifts too far from a tree structure.

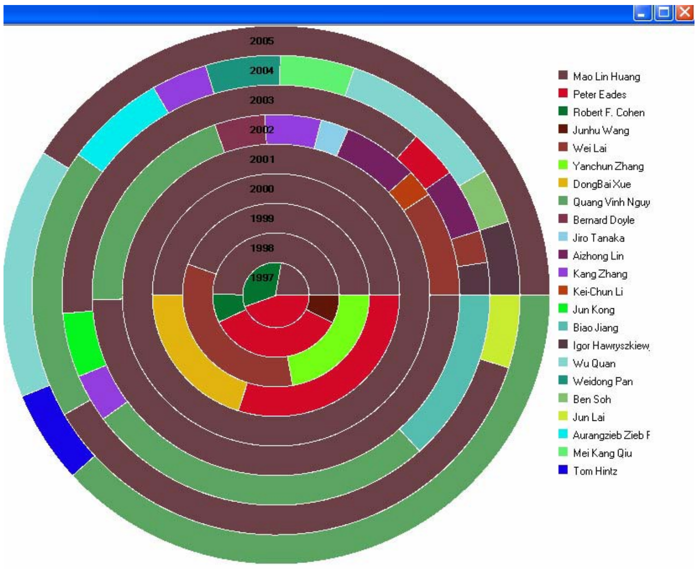
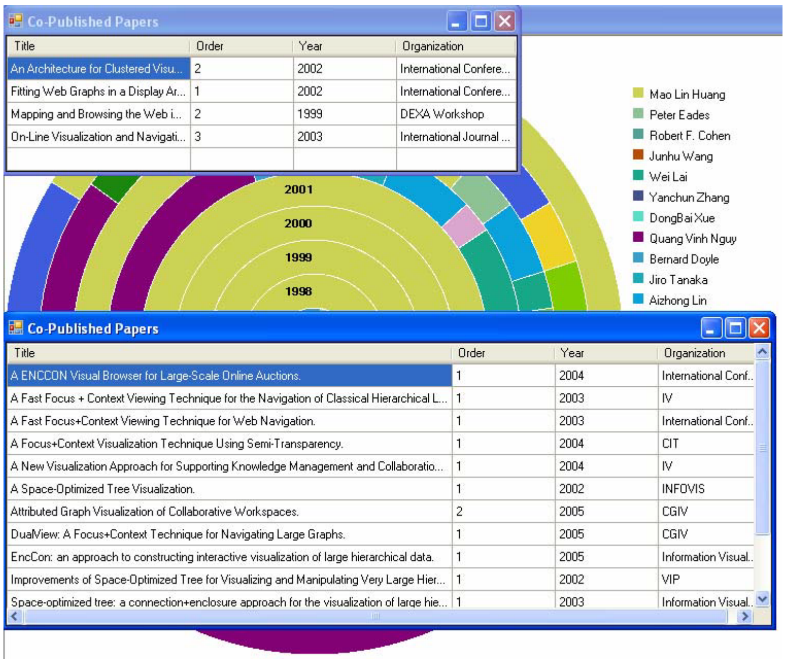


Figure 2.4 Visualization of an individual's co-authorship network using InterRing Visualizer (Taken from **[9]**)



Title	Order	Year	Organization
An Architecture for Clustered Visu...	2	2002	International Confere...
Fitting Web Graphs in a Display Ar...	1	2002	International Confere...
Mapping and Browsing the Web i...	2	1999	DEXA Workshop
On-Line Visualization and Navigati...	3	2003	International Journal ...

Co-Published Papers

Title	Order	Year	Organization
A ENCCON Visual Browser for Large-Scale Online Auctions.	1	2004	International Conf...
A Fast Focus + Context Viewing Technique for the Navigation of Classical Hierarchical L...	1	2003	IV
A Fast Focus+Context Viewing Technique for Web Navigation.	1	2003	International Conf...
A Focus+Context Visualization Technique Using Semi-Transparency.	1	2004	CIT
A New Visualization Approach for Supporting Knowledge Management and Collaboratio...	1	2004	IV
A Space-Optimized Tree Visualization.	1	2002	INFOVIS
Attributed Graph Visualization of Collaborative Workspaces.	2	2005	CGIV
DualView: A Focus+Context Technique for Navigating Large Graphs.	1	2005	CGIV
EncCon: an approach to constructing interactive visualization of large hierarchical data.	1	2005	Information Visual...
Improvements of Space-Optimized Tree for Visualizing and Manipulating Very Large Hier...	1	2002	VIP
Space-optimized tree: a connection+enclosure approach for the visualization of large hie...	1	2003	Information Visual...

Figure 2.5 The textual window (Taken from **[9]**)

Figure 2.4 and 2.5 show the analysis result of a professor for his co-authoring relationship with other 22 researchers during the past 10 years.**[9]** The size of a sector represents the strength of co-author relationship. Different colors are used to represent different co-authors. The open textual windows display the detail of joint publications of two co-authors of an interested sector clicked by the user.

The visualization clearly tells users that this professor has cooperated with what author in which academic year in projects and publications. By analyzing the professor's co-authors' specialties, users can predict the movement of this author's knowledge domain.

But the user can not see how much proportion one relationship occupied in all the co-author relationships. If users want to know who is the most activity co-author, they need to open a large amount of text windows of different co-authors, and then compare them. Another problem is that if an author has too many co-authors, the colors which are used to distinguish them may be not enough. Like Figure 2.6, the user may be confused because some colors are very similar and hard to distinct.

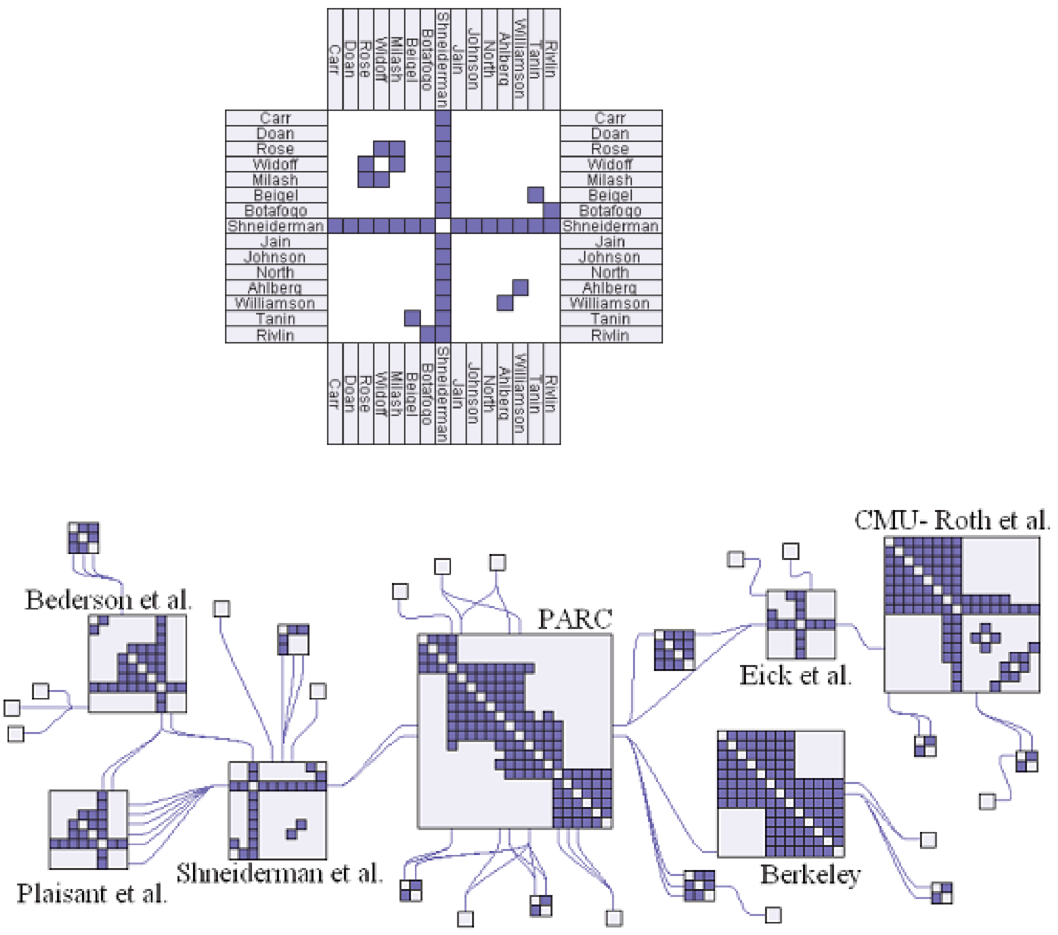


Figure 2.6 NodeTrix Representation of the largest component of the InfoVis Co-authorship Network (Taken from **[10])**

Figure 2.6 shows a visualization called NodeTrix which *integrates the best of the two traditional network representations by using node-link diagrams to visualize the overall structure of the network, within which adjacency matrices show communities.* **[10]**

In the matrix, authors are arranged at four edges, like the upper sub-figure of Figure 2.8. If tow authors have co-author relationship, the node which corresponds to them will be highlight in blue color. Users can split one author or an aggregation of authors out of the main matrix, by dragging it or them to another position then release. Even the two different matrixes can have connections represented by links (lines). In order to show more details to users in the main view, researcher could use color, transparency, shape size, filled area of the shape, border color, width, and labels to represent for example the strength of the co-relationship, domain of the publication, department of authors and so on.

This method is powerful and efficient to display most of the relationships and attributes. But we will still mention some small drawbacks:

- It is hard to show the type of the publications, because the publications between two authors can involve in many domains. If users want to know some details of publications they have to open a text window by some methods (like double clicking nodes), then do further selection. It is a little complex.
- Imagine a large matrix includes 300 authors. User will find sometimes it is easy to get confused to find a node, because it needs to find the corresponding two authors form lots of authors.

3 Visualization Approach

This chapter includes the working flow of the program, the two designs of views and the reason of our selection, then explaining how we arrange every article's position and the solution of decrease overlap. At last, introducing the user interface and its functionality.

3.1 Visualization

As the statement in chapter 1, our objective is to implement a visualization view for LNU's network publications. It connects with DIVA, downloads and transforms the data into readable form. This part indicates the idea of the visualization approach.

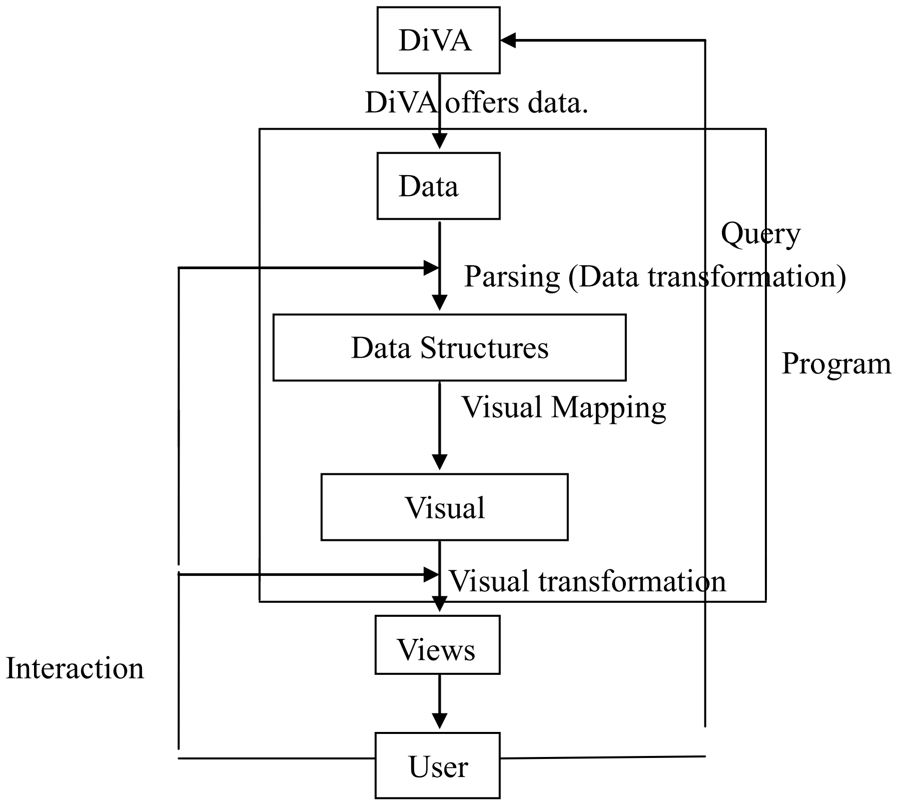


Figure 3.1: The working flow chart of program

1. The program will first connect with DiVA.
2. Program sends the user's "Query" to DiVA.
3. DiVA offers dataset base on the "Query".
4. Program gets the dataset then parses and transforms them into a special "Data Structures". The interaction in this part is brushing
5. Program transforms data structure into visual structure.
6. Program transforms visual structure into view. The interaction in this part is "zoom", "scrolling" and "spring".
7. If user has new query, the program can sent new query to DiVA, and take new dataset for visualizing.

3.1.1 Our two plans

After reference others' solutions and related material, we came up with our two designs. The following content explains both showing advantages and disadvantages.

- **Design 1:**

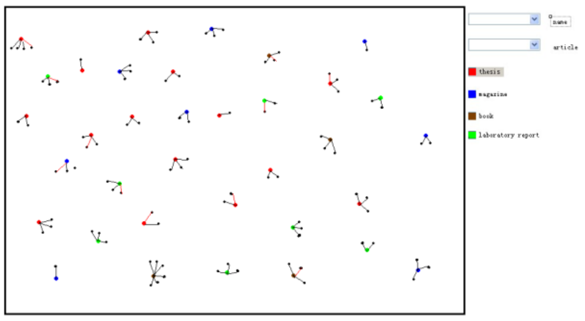


Figure 3.2

In this design, the structural drawing is shown as Figure 3.2. Dots (large one) represent articles, and different colors mean different types of articles. Small black dots represent authors. The lines which connect author with articles represent the relationship between them. When user wants to know the productions of one person, he or she can input the person's name in the message box on the right side, so all the publications of this author will be highlighted. And it is the same when user inputs the article's title. When user clicks the colorful button on the right side, he/she can choose to see specific sorts of articles, for example, if user wants to see thesis, he/she can click red button. All theses and the related people will be highlighted, while other rectangles and dots will disappear.

If a user wants to see the details, he/she clicks the small group in the browser directly, and will see the content of the articles, authors and supervisors of it which will all be shown in another window.

- **Design 2:**

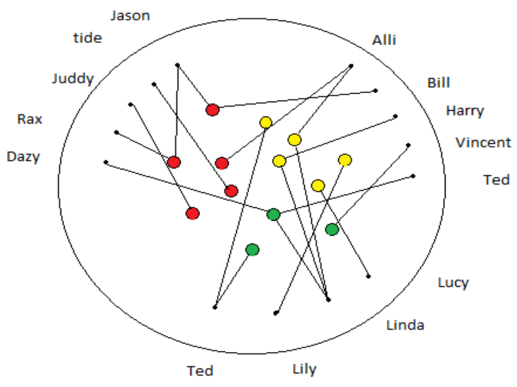


Figure 3.3

The structural drawing is shown in Figure 3.3. Author nodes are placed in circular layout. Authors are represented by small black dots scattered at the outmost layer, and articles are represented by small circle in the center. Red circle represents theses; Pink circle represents books; Blue one represents magazines, and so on. We decide to use different colors to show different types of the publications, because special colors stand out their graphical environment (without any search efforts).

The black lines are used to connect the author and article, and the red lines are used to connect the supervisor and article. "*Connectedness is a powerful grouping principle that is stronger than proximity, color, size, or shape*."**[5]**
Usability: if you click an author's name, the corresponding dots, the relevant article and the line will be highlighted. If you click the article, the authors and the supervisors of this article will all be highlighted. When the mouse pointer moves to the circle, the title of article will be displayed. You can also press and drag it, the line connected to it will go with it, but the dots will not move, so you can see the structure more clearly. When you double click the article, a direct link for the whole article will be offered. Additional, for improving the effect of visualization and dynamic, the lines are retractable just like a spring.

Comparison: The redundancy exists in Design 1, because one author may have several articles, so lots of authors will be repeated several times. But Design 2 does not have the problem of redundancy. So, we decide to implement this one. After improvement of the second plan, we developed a more mature plan, see Figure 3.4.

In the main view, the meaning of the circles and dots are marked at the top left corner. The circles which are surrounded by dots are publications, and dots represent authors. Authors' names are ranged at the outmost layer. Lines are drawn to connect the publications and there authors.

The Control Panel contains three tabbed panes; they are "Query", "Search" and "Result" panels. "Query" is used to query publications or authors according to user's requirement; "Search" is used to search publications or authors in the main view. "Result" is used to show the outcomes in message box, users can also operate in "Result" panel to do further selection.

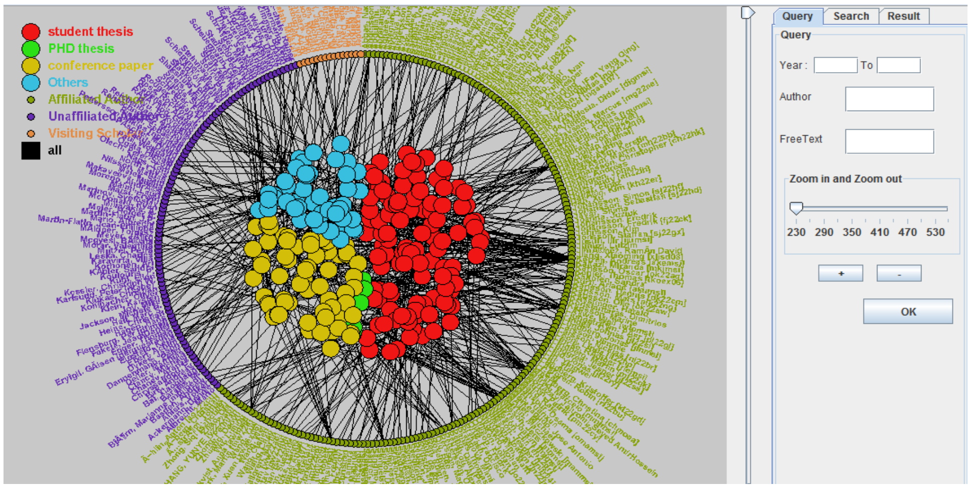


Figure 3.4 Overview

3.1.2 Spatial positioning of articles

As we said in chapter 2, we use different colors to represent different types of publications that scatter in each other's area. The larger the numbers of publications, the larger the area are occupied by the type, see Figure 3.5:

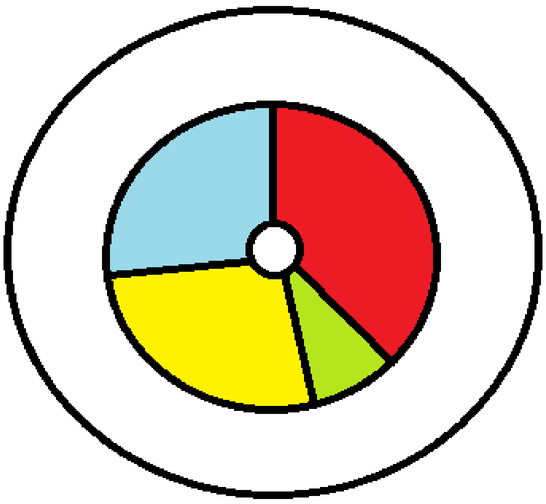


Figure 3.5 Publication distribution

Red region is student thesis; green region is PHD thesis; yellow region is conference paper; blue region is another type includes book, magazine, test report and so on.

We want to create the positions of the publication circles randomly. Because, if we fix the positions, the hole view will be dull.

$$x[i]=x+levellength*\sin(anglesmall)$$

$$y[i]=y-levellength*\cos(anglesmall)$$

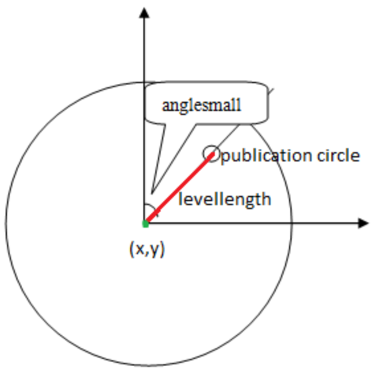


Figure 3.6 publication circle schematic

See the formulas: "x[i]" and "y[i]" are the coordinates of publication circle. In order to create a random position in the big circle we need first create a random angle called "anglesmall" and a random length called "levellength". "x" and "y" are center coordinates of the big circle, as shown in Figure 3.6.

How can we determine the upper limit and lower limit of the angle? We firstly use 2*PI to divide the total number of publications and store the value in "A". Then let "A" multiple the number of publication of each type to know how many angle each type will occupy in a round area. For instance, the lower limit of the red region is 0; the upper limit equals to the number of student thesis divide number of all publications and then multiple 2PI. The lower limit of the green region is the up limit of red region. The upper limit of green one is lower limit plus the angle the green area occupies. "levellength" is also a random integer value, the interval of the value is signed in Figure 3.5.

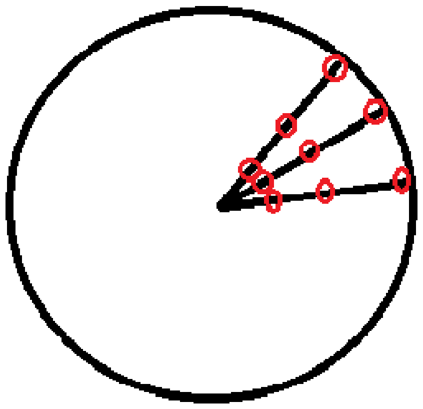


Figure 3.7 Dense of distribution of publications

We can see from Figure 3.7: even though the positions of these points are produced with the same probability, the density of distribution is different. The density of distribution which is closer to the center of the big circle is denser than which is farther. If we split the big circle evenly into 3 parts according to its radius, we can get the area rate is 1:3:5. In order to make the dense equivalent at the three layers, we need to put one ninth of the publications in innermost layer, put three ninths of publications in middle layer, put five ninths of publications in outermost layer, see Figure 3.8.

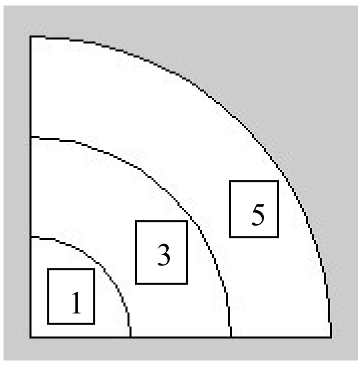


Figure 3.8 Layers

Problem: Here, the one big problem we meet is overlap. The reader of this thesis can imagine that lots of points are produced randomly, so it is inevitable to get some points close to each other. Take these points as centers of circles, will lead some circles to be covered by others.

Solution 1: We can compare the position of current publication circle to the positions of the publication circles that are produced earlier. If current circle overlaps with one of them, drop this position and produce a new one, then compare them again. Repeat the same process over and over until the current circle does not overlap with any other circles, and also repeat this process until producing all coordinates of publication circles.

Advantage: It will totally solve the problem of overlap.

Disadvantage: Consumes a lot of computational power.

Solution 2: Use zoom-in, zoom-out and spring effect to avoid this problem.

Zoom-out makes canvas smaller which means the random number will in a smaller range. It will increase the probability of overlap, but zoom-in will decrease the probability. See Figure 3.9

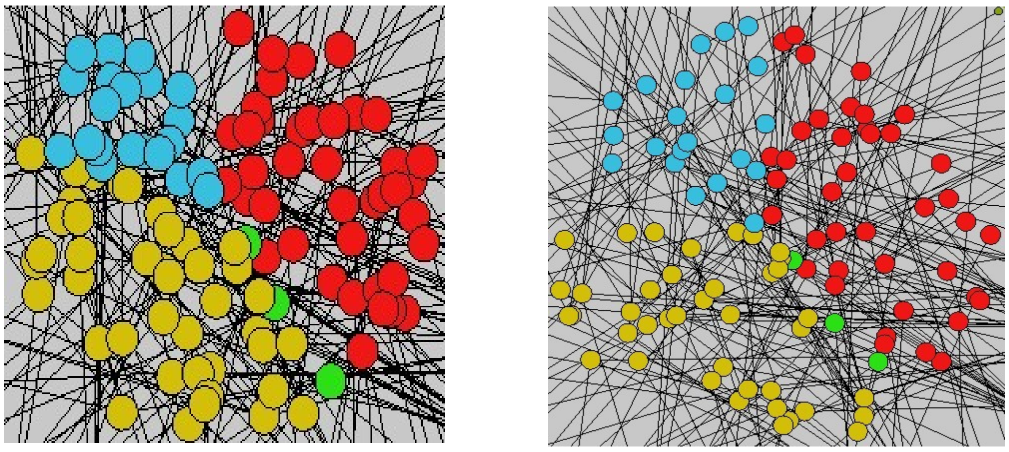


Figure 3.9 zoom-in and zoom-out

Using spring effect can make people drag the circles and avoid overlap. The circles which are covered will stand out. See Figure 3.10

Advantage: It can temperately solve the problem of overlap.

Disadvantage: Because the circle will move like a spring. It will at last stop at the original place and cover other circles. So the problem is not a fundamental solu

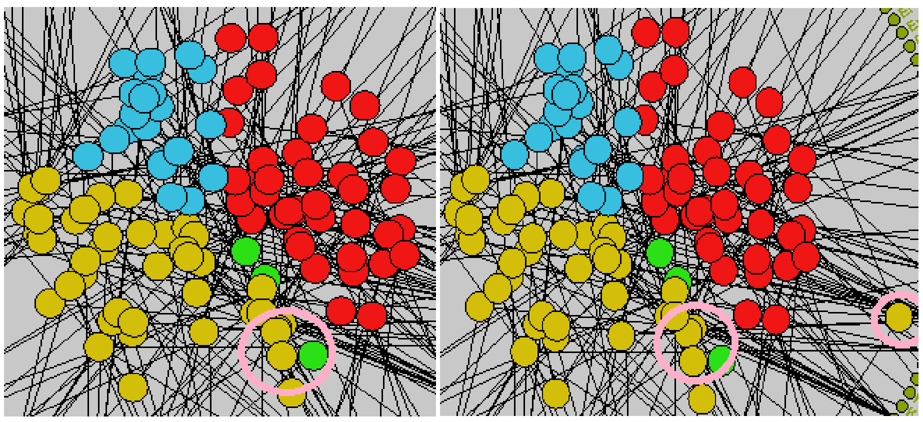


Figure 3.10 Spring effect

After comparing these two solutions, we think that the second on is more suitable for our design.

3.1.3 Determine coordinates of authors

We want to arrange the dots (represent authors) to circle around publications. The easiest way is using "`translate()`" and "`rotate()`" function. "`translate()`" function is used to move the origin to a point we set, so it equals to move canvas. "`rotate()`" function can rotate canvas in any angle you set. See Figure 3.11, by translating the position, we can move the dot to another place of the canvas; by rotating the canvas at PI angle, we make the dots arrange evenly on the half-circle.

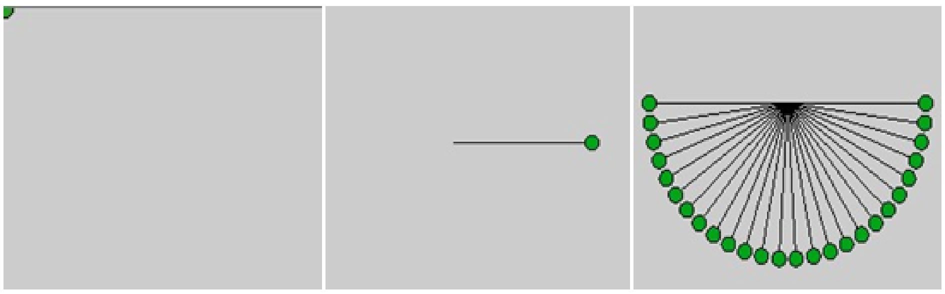
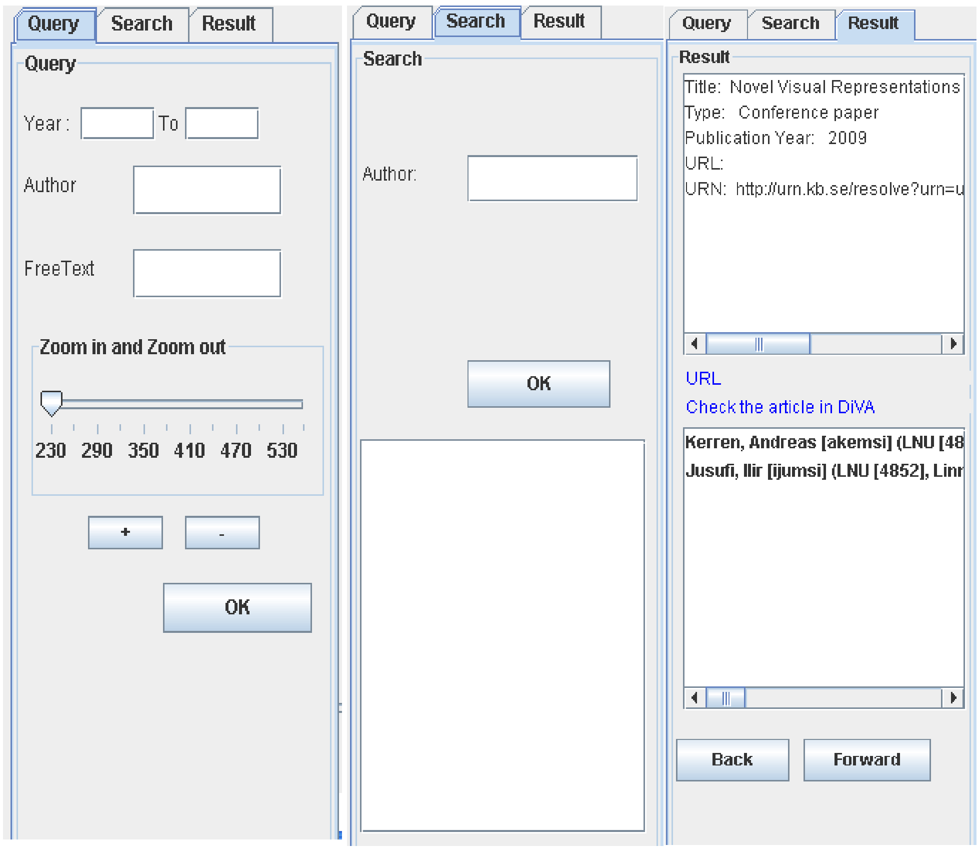


Figure 3.11 Function of "`translate()`" and "`rotate()`"

3.2 User interface

A good user interface is very important in this project, because a good one can improve the program's usability.



Title: Novel Visual Representations
Type: Conference paper
Publication Year: 2009
URL:
URN: http://urn.kb.se/resolve?urn=u

Kerren, Andreas [akemsi] (LNU [48
Jusufi, Ilir [ijumsi] (LNU [4852], Linr

(a) Query panel (b) Search panel (c) Result panel

Figure 3.12 Control panel

- **Select time range**

We provide two text fields for user to input the time. After clicking "OK" button in "Result panel", a dataset which meet the time condition will be downloaded and then be visualized. (Figure 3.12-a)

- **Show article's information**

When user uses mouse to select one article in the views, the article's relevant information will be showed in "TextArea" (Figure 3.12-c).

- **Show author's information**

When user uses mouse to select one author in the views, the author's publication will be showed in "TextArea" (Figure 3.12-c).

- **Hyperlink**

If DiVA offers a URL with one article, and when user selects that article, a hyperlink will be created and user can link to the web page by click that hyperlink (Figure 3.12-c).

- **Search article by author**

User can input the name of author and program can create the views which only have this author's publication and his/her partner. The implementation of this part is seeding a query to DiVA then downloads and visualizes a new dataset.

In this part we do not travel the hashmap and check every article, but change the link and download a new dataset from DiVA (Figure 3.12-a).

- **Search author in one dataset**

User can input the name of author in the "TextField" in the "Search panel", then this author and his publication, co-author will be highlighted (Figure 3.12-b).

- **Scroll bar**

Because when we add the view program in JScrollPane, the scroll bars do not work, so use "Jslider" to instead the "JScrollPane". As shown in Figure 3.13, one is sideway slider, another is vertical slider.

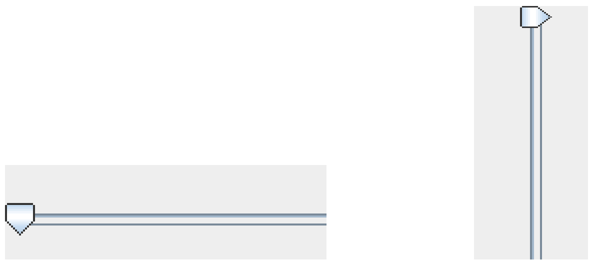


Figure 3.13 Scroll bar

4. Use Case

In this chapter, some examples are used to display how to use this tool and the tool's functionality.

4.1 Initial GUI

When user wants to try to use this widget, firstly need to connect to the server, and it will display such a view in Figure 4.1:

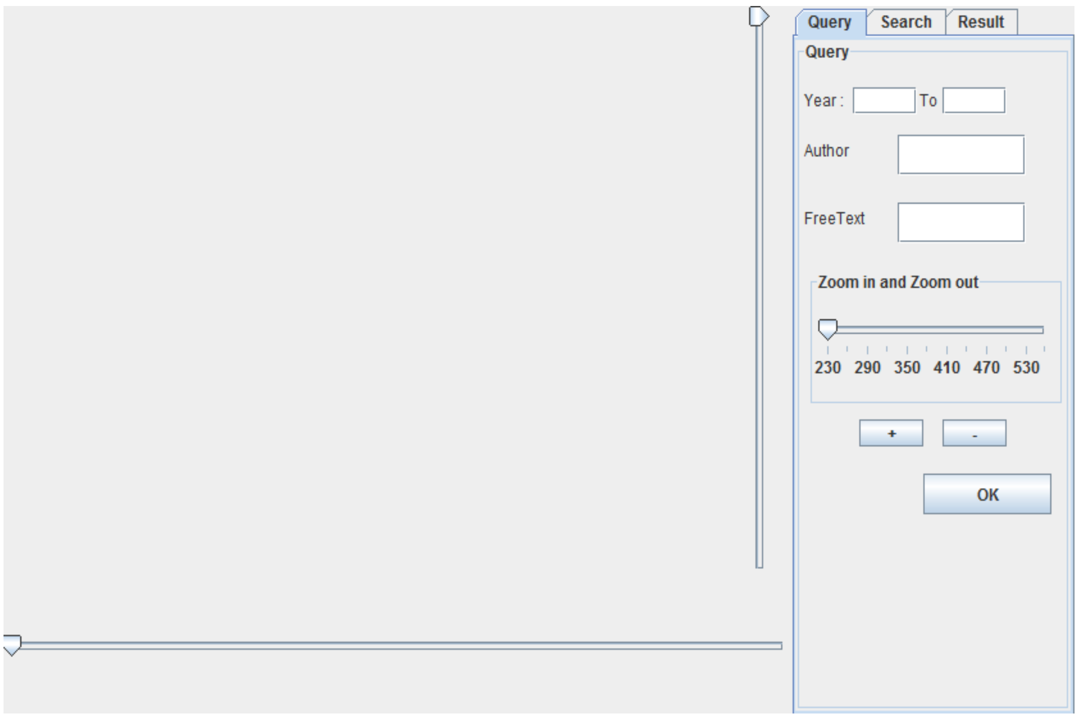


Figure 4.1 Initial GUI View

We can see the view of our widget can be divided into two parts, left part is used to show the Main View, and right part is Control Panel.

4.2 Publication Visualization View

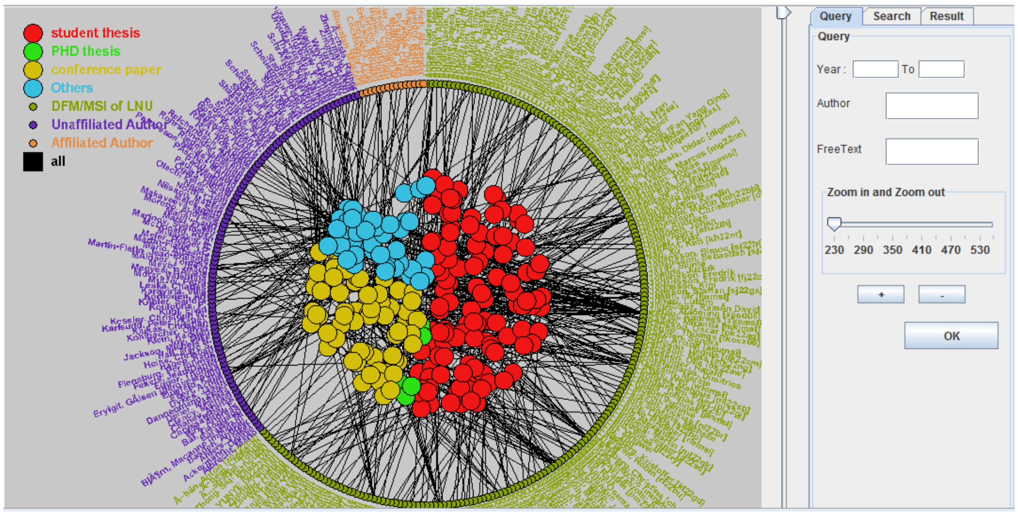


Figure 4.2 The Publication Network

When pressing the "OK" button, we can see in Figure 4.2, the publications are represented by circles in the center part of the circle. Authors are represented by small dots at the circle surrounding the publications. Different colors mean different types. User can check which type each color refers to at the top left corner. Each publication may have several authors, and each author may have several publications. We use lines to connect authors and their publication. Here, there are more than 200 publications and more than 300 authors.

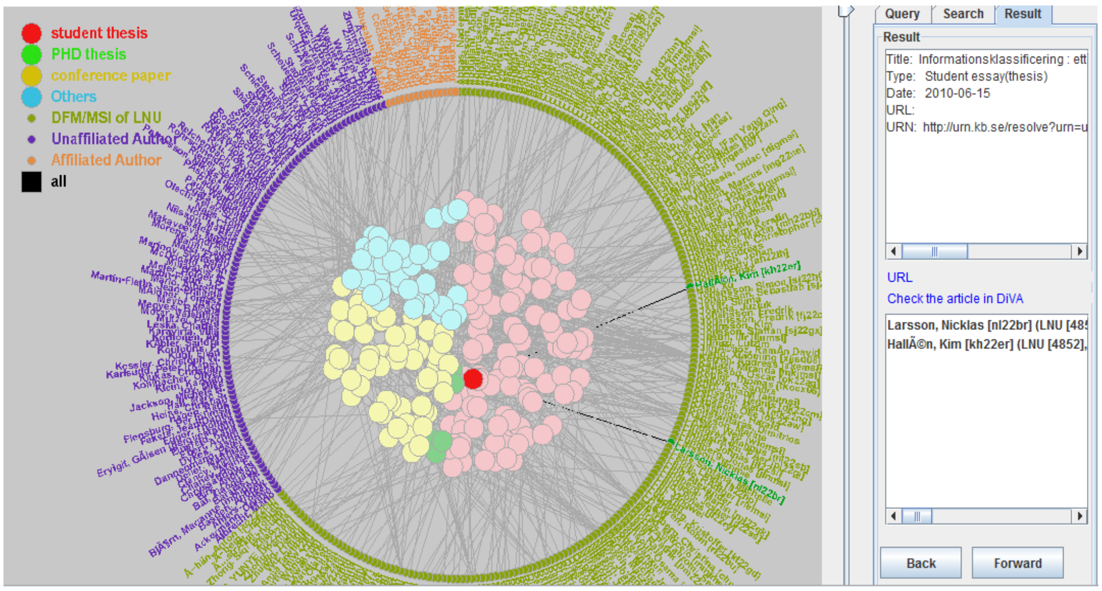
Users might get insight into the detail immediately after loading the data. For instance, if the author who has the most lines connect to, means he or she is the most productive author; if one author has a line connects to a red rounds, means he had gotten a bachelor or master degree here.

User can also do lots of operation on the view directly to help us get more information he or her concerns.

4.2.1 Selecting Publication

When user clicks one publication, other publications will fade out, only this publication, its authors and the lines between them will be highlighted. The highlighted authors will be shown in green color.

Click "Result" panel. It will show the authors of this publication in the lower message box, and the information of this publication in the upper "TestArea". The message in the lower "List" can be pressed to select co-authors, see Figure 4.3.

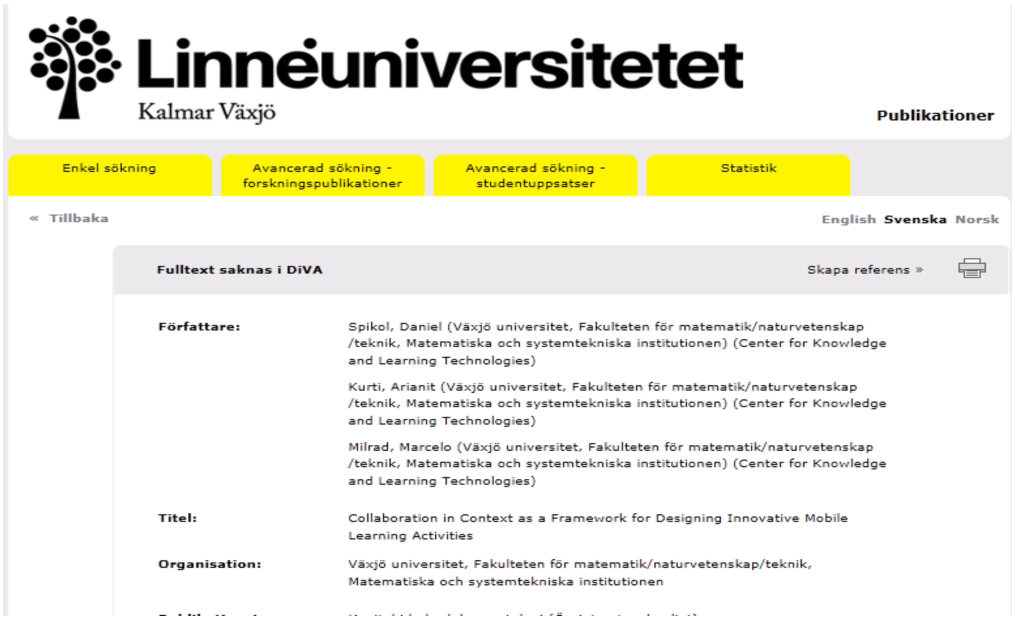


Title: Informationsklassificering : ett
Type: Student essay(thesis)
Date: 2010-06-15
URL:
URN: http://urn.kb.se/resolve?urn=u

Larsson, Nicklas [nl22br] (LNU [485
HallÃ©n, Kim [kh22er] (LNU [4852],

Figure 4.3 View showing one selected publication

Between the two containers, we can see two links. One is connecting to some professional publication website, another one is connecting to publications' database in Linneaus University. All thesis publications can be found on DiVA portal to download. For other publications, users can check only details about that specific publication. Here we select a publication has these both links. When clicking them, the user can get into the website and see such interfaces, see Figure 4.4 and 4.5.


Linnéuniversitetet
Kalmar Växjö

Publikationer

Enkel sökning | Avancerad sökning - forskningspublikationer | Avancerad sökning - studentuppsatser | Statistik

« Tillbaka

English **Svenska** Norsk

Fulltext saknas i DiVA

Skapa referens »

Författare: Spikol, Daniel (Växjö universitet, Fakulteten för matematik/naturvetenskap /teknik, Matematiska och systemtekniska institutionen) (Center for Knowledge and Learning Technologies)

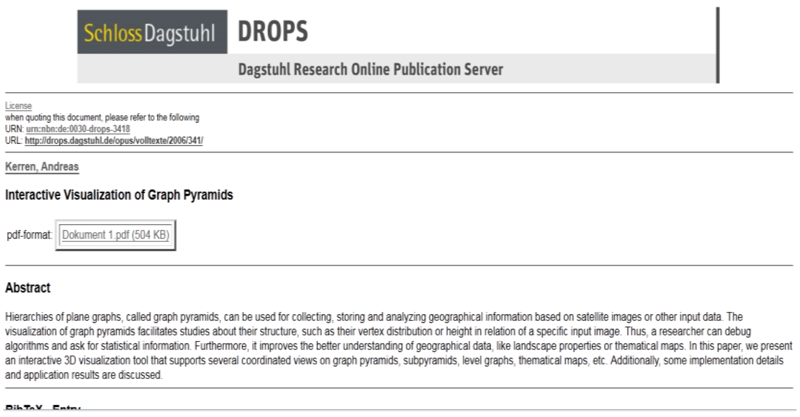
Kurti, Arianit (Växjö universitet, Fakulteten för matematik/naturvetenskap /teknik, Matematiska och systemtekniska institutionen) (Center for Knowledge and Learning Technologies)

Milrad, Marcelo (Växjö universitet, Fakulteten för matematik/naturvetenskap /teknik, Matematiska och systemtekniska institutionen) (Center for Knowledge and Learning Technologies)

Titel: Collaboration in Context as a Framework for Designing Innovative Mobile Learning Activities

Organisation: Växjö universitet, Fakulteten för matematik/naturvetenskap/teknik, Matematiska och systemtekniska institutionen

Figure 4.4 Linneaus University publication databases **[11]**


SchlossDagstuhl DROPS
Dagstuhl Research Online Publication Server

License
when quoting this document, please refer to the following
URN: urn:nbn:de:0030-drops-3418
URL: http://drops.dagstuhl.de/opus/volltexte/2006/341/

Kerren, Andreas

Interactive Visualization of Graph Pyramids

pdf-format: Dokument 1.pdf (504 KB)

Abstract

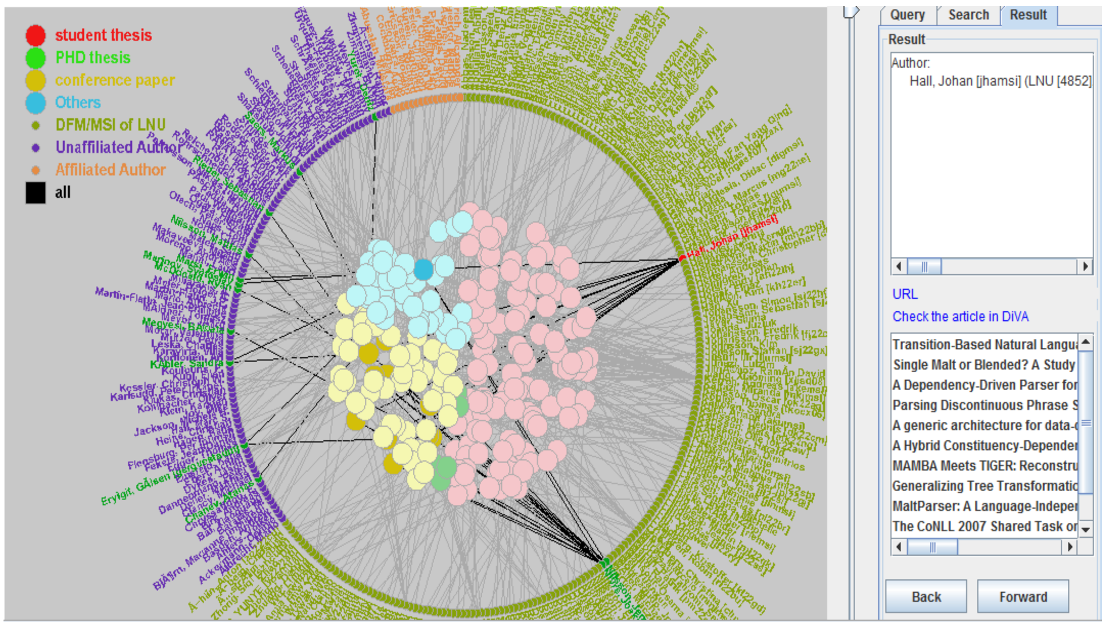
Hierarchies of plane graphs, called graph pyramids, can be used for collecting, storing and analyzing geographical information based on satellite images or other input data. The visualization of graph pyramids facilitates studies about their structure, such as their vertex distribution or height in relation of a specific input image. Thus, a researcher can debug algorithms and ask for statistical information. Furthermore, it improves the better understanding of geographical data, like landscape properties or thematical maps. In this paper, we present an interactive 3D visualization tool that supports several coordinated views on graph pyramids, subpyramids, level graphs, thematical maps, etc. Additionally, some implementation details and application results are discussed.

Figure 4.5 A professional publication website **[3]**

4.2.2 Selecting author

When the user clicks one author, this author will be highlighted in red color. His or her publications will be highlighted, but all other publications will fade out. All of his or her co-authors will be highlighted in green.

This author's publications' titles will be listed in the lower "List" of the Result panel. Users can click these titles to do further selection. The personal information of this author will be shown in the "TextArea", see Figure 4.6.



Result

Author:
Hall, Johan [jhamsi] (LNU [4852]

URL
Check the article in DiVA

Transition-Based Natural Langu
Single Malt or Blended? A Study
A Dependency-Driven Parser for
Parsing Discontinuous Phrase S
A generic architecture for data-
A Hybrid Constituency-Depender
MAMBA Meets TIGER: Reconstru
Generalizing Tree Transformatic
MaltParser: A Language-Indeper
The CoNLL 2007 Shared Task or

Figure 4.6 View showing one selected author.

4.2.3 Select publications according to their type.

Publications can be divided into 4 types. They are student thesis, PHD thesis, conference paper, and others. See the top left corner of the view in Figure 4.7. If user clicks one of the icons of the four types, the publications which belong to this type will be selected. For example, I clicked the yellow one which represents conference paper.

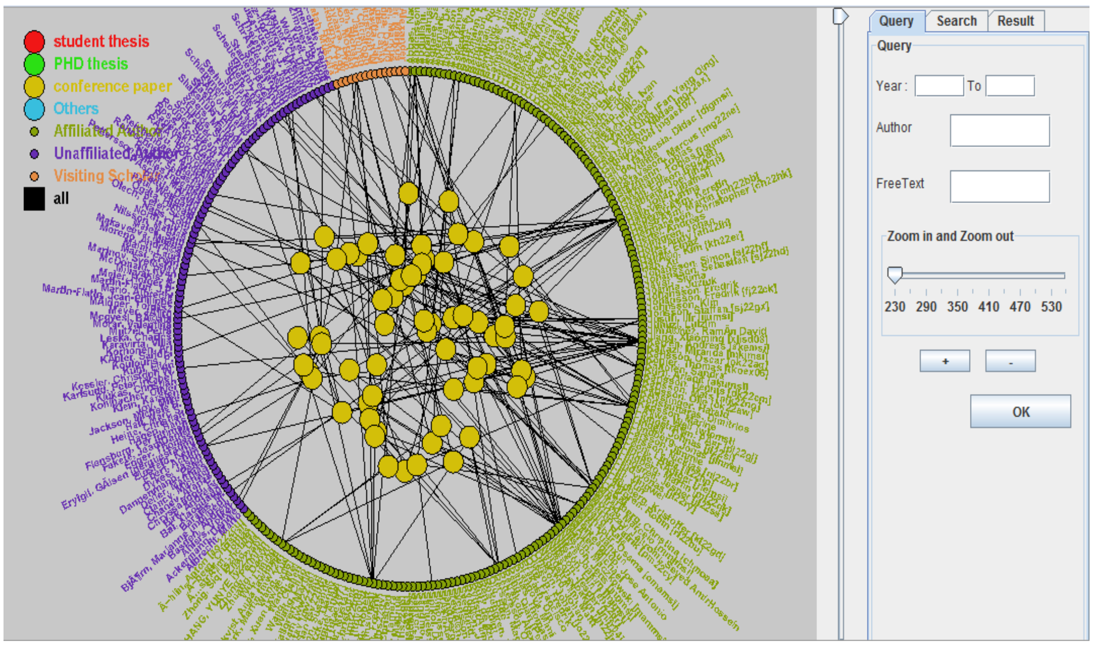


Figure 4.7 Only conference papers are selected

Authors can be divided into 3 types. They are DFM/MSI of LNU, Affiliated Author and Unaffiliated Author. See the top left corner of the view in Figure 4.8. If user clicks one of the icons of the three types, the authors who belong to this type will be selected. The publications which do not belong to these authors will be faded out. For example, we pressed the purple dot which represents unaffiliated author.

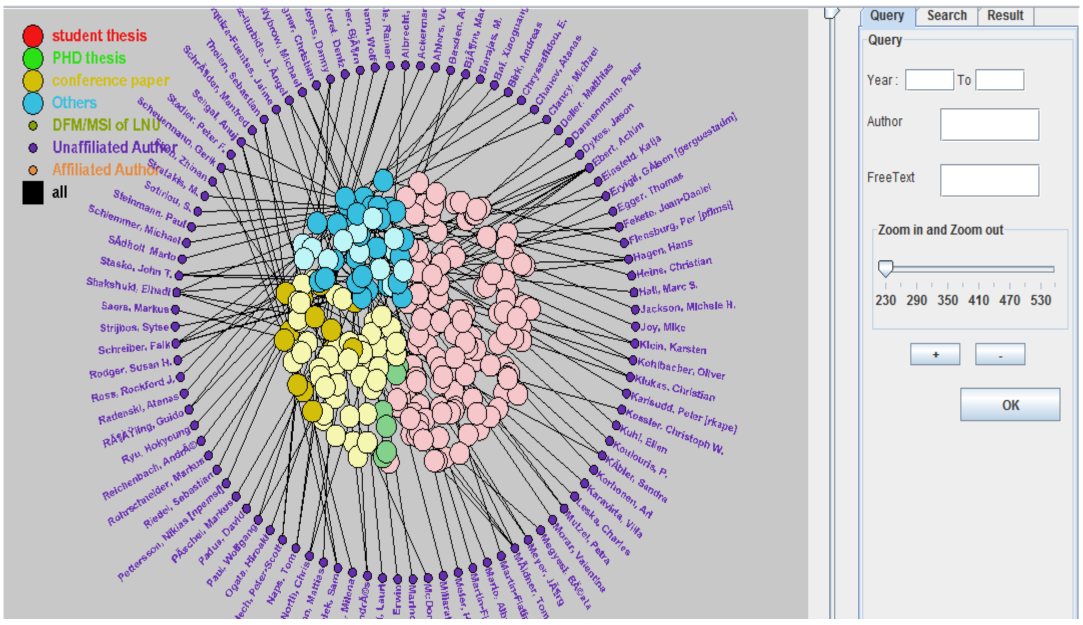


Figure 4.8 Only Unaffiliated Authors are selected.

4.2.4 Show all publications and authors.

Look at the top left corner of the view in Figure 4.8. “All” is used to show all publications and authors. For example, after getting the view of Figure 4.6, when a user presses “all”, the view will go back to Figure 4.2.

4.3 Control Panel

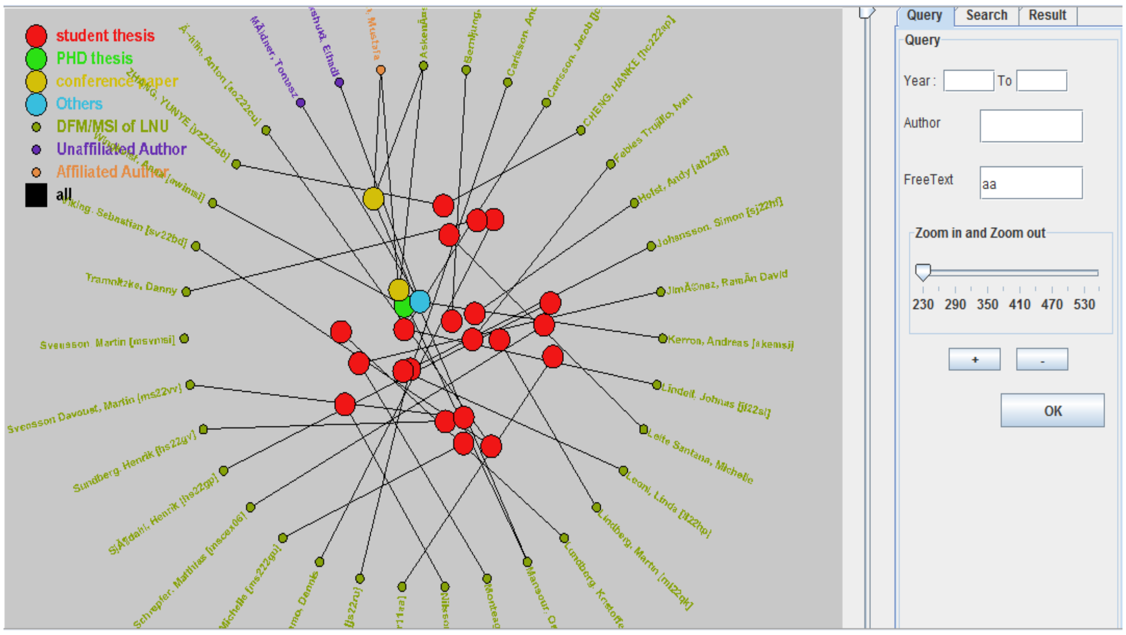
Control panel is a place to realize the functions of querying, searching and showing results in words or links. In this section we will introduce several examples for each function separately to give users detailed knowledge about how to use this control panel.

4.3.1 Query

If a user wants to search the publications from 2007 to 2010, just input “2007” in the left box of “Year” and 2010 in the right box of “Year”. If one just enters the year in left box that means selecting publications at and after this year; if one just enters the year in right box that means selecting publications at and before this year. If the user does not enter any date, it will default to select form all date.

It is easy to see that “Author” box is used to search according authors’ names. For example, if users want to search “Kerren, Andreas”, they enter “Kerren” or “Andreas” or “Kerren, Andreas”, each of the words can search him out, but the first two methods will search out all the people whose names contain the word, no matter whether it is family name or first name. So, if you know the entire name, please input the entire name.

“Free Text” box can search the content which contain the character that be inputted to it. For instance, users search “aa” in “Free Text”, no matter it is authors, supervisors or even the titles of publications or departments of authors, if it contains character “aa”, it will be selected out, and be transformed into a view, see Figure 4.9.



Query

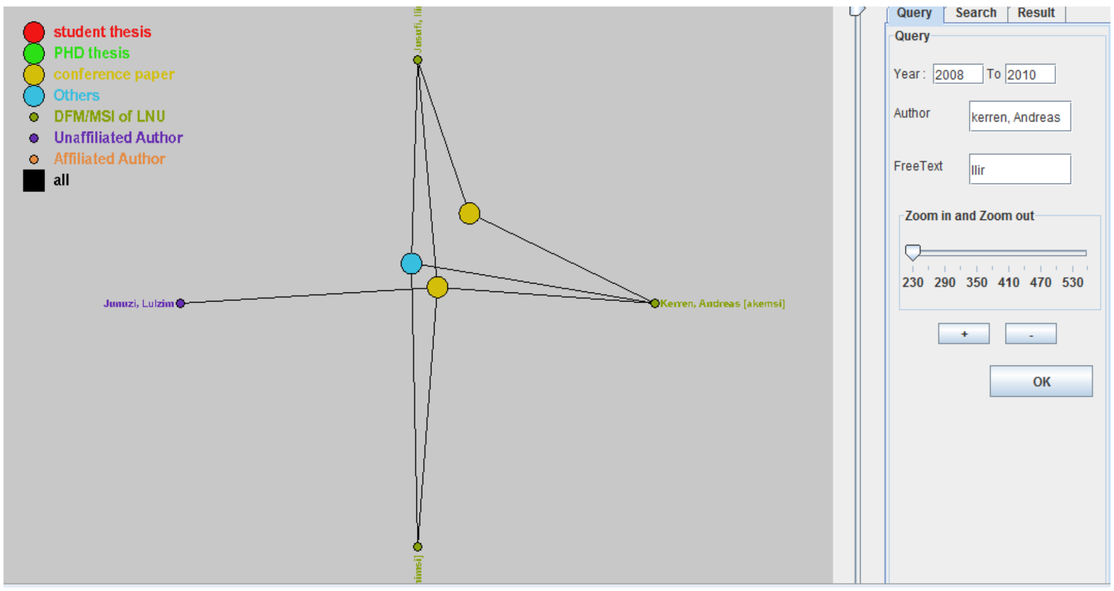
Year : To

Author

FreeText aa

Figure 4.9 Search “aa” in Free Text

When the user clicks “OK” button, only the content which satisfied all the three requirements can be selected. If the box is empty, it will be default to be no limitation. For example, we want to search the publications which published between 2008 and 2010, and include “Kerren, Andreas” as the author, and be contributed by “Ilir”, see Figure 4.10.



Query

Year : 2008 To 2010

Author kerren, Andreas

FreeText Ilir

Figure 4.10 Publications published between 2008 and 2010, authors include “Kerren,Andreas” and “Ilir” also contributed to it.

“Zoom in and Zoom out” slider and two buttons which have “+” and “-” on it separately is used to magnify or minify the view. They have the same function, if users press the “+” button or “-”, the slider will move with it, see Figure 4.11.

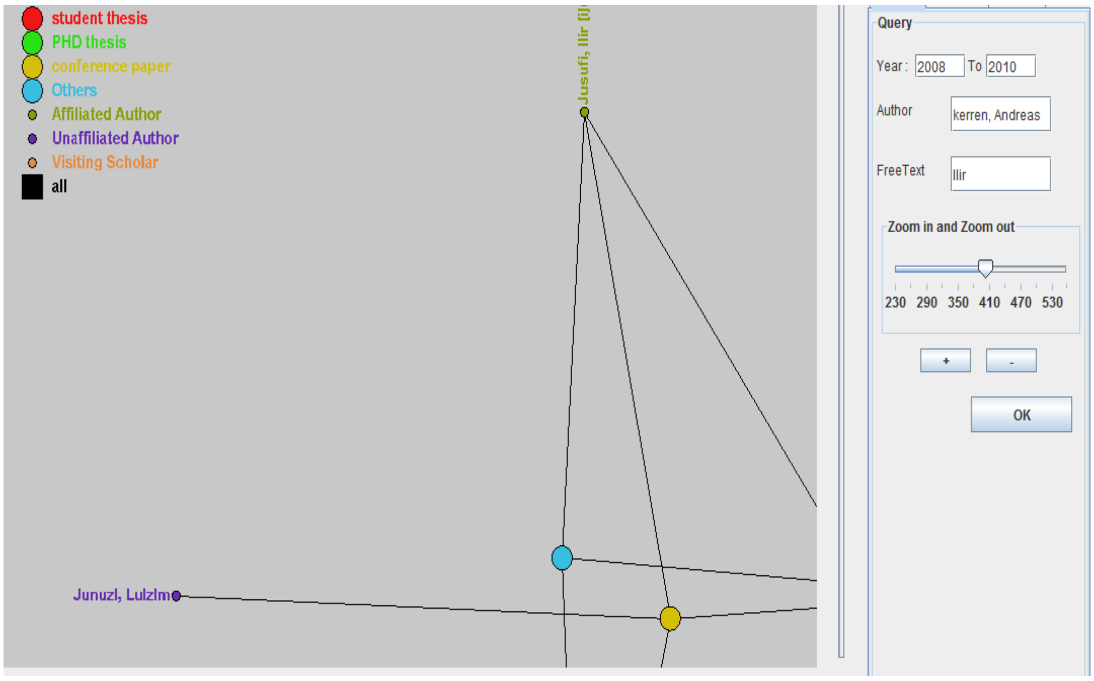


Figure 4.11 Zooming in

To Consider that zooming in will lead to part of the view beyond the screen, we can use vertical scroll bar and horizontal scroll bar to move the canvas, see Figure 4.11.

4.3.2 Search

Search panel is much simpler than query panel, see Figure 4.12.

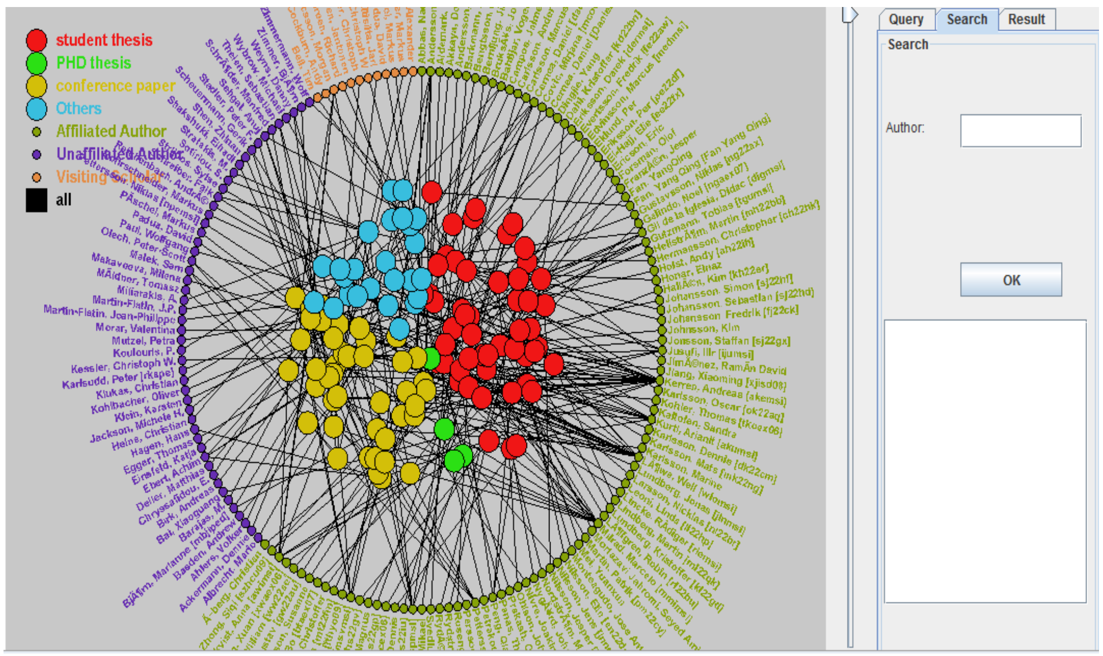
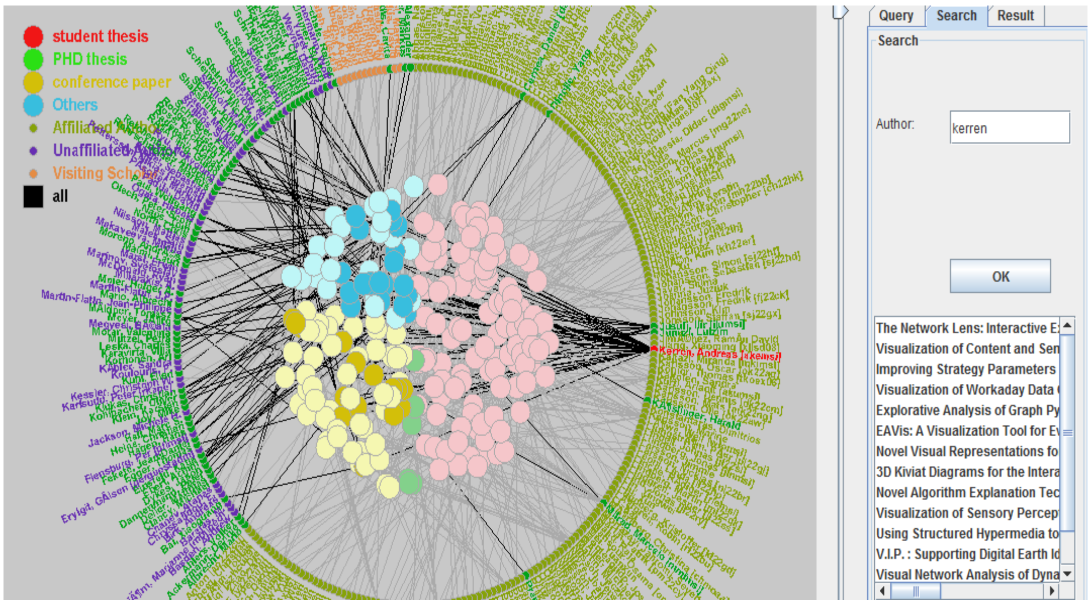


Figure 4.12 Search panel

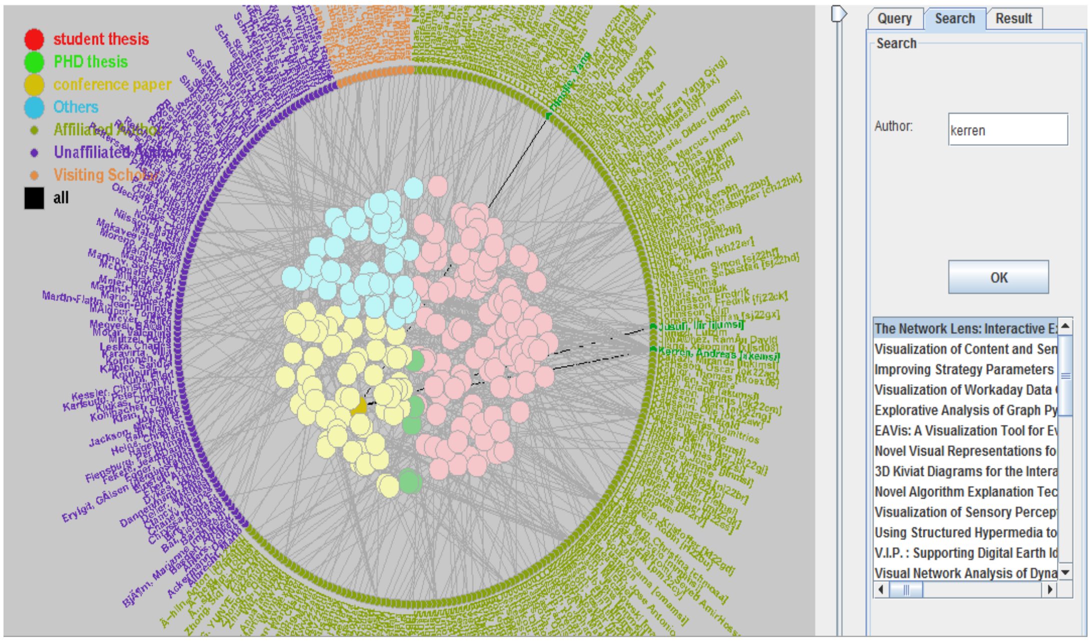
If users want to search one author’s publications, just enter the author’s name and press “ok”. Here I take “kerren” as an example, see Figure 4.13.



The Network Lens: Interactive E
Visualization of Content and Sen
Improving Strategy Parameters
Visualization of Workaday Data
Explorative Analysis of Graph Py
EAVis: A Visualization Tool for E
Novel Visual Representations fo
3D Kiviat Diagrams for the Intera
Novel Algorithm Explanation Tec
Visualization of Sensory Percep
Using Structured Hypermedia to
V.I.P.: Supporting Digital Earth Id
Visual Network Analysis of Dyna

Figure 4.13 Search Kerren's publications.

As shown, the author who is searched is in red color, and all his publications, partners and lines between them are highlighted. His partners are highlighted in green color. In the JList, Kerren's publications are listed. If users click one publication in the JList, for example the first one, this publication will be highlighted and also its authors and lines between them see Figure 4.14.

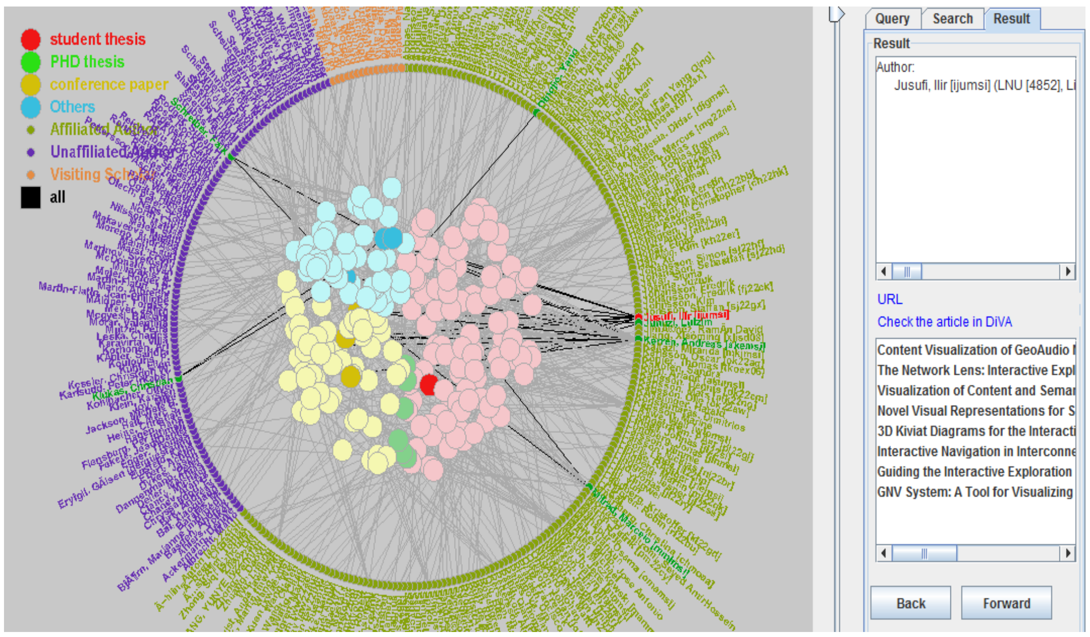


The Network Lens: Interactive E
Visualization of Content and Sen
Improving Strategy Parameters
Visualization of Workaday Data
Explorative Analysis of Graph Py
EAVis: A Visualization Tool for E
Novel Visual Representations fo
3D Kiviat Diagrams for the Intera
Novel Algorithm Explanation Tec
Visualization of Sensory Percep
Using Structured Hypermedia to
V.I.P.: Supporting Digital Earth Id
Visual Network Analysis of Dyna

Figure 4.14 highlight the first publication and its authors.

4.3.3 Result

In result panel, there are "TextArea", "List" and two links, see Figure 4.15.



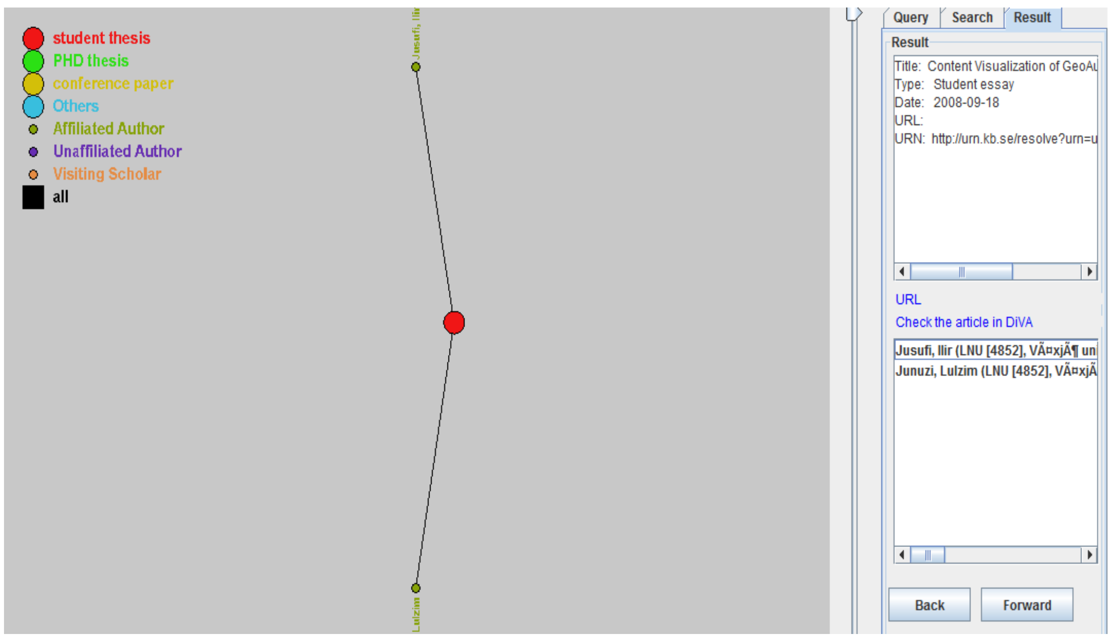
Result
Author:
Jusufi, Ilir [ijumsi] (LNU [4852], Li

URL
Check the article in DiVA

Content Visualization of GeoAudio I
The Network Lens: Interactive Expl
Visualization of Content and Semar
Novel Visual Representations for S
3D Kiviat Diagrams for the Interacti
Interactive Navigation in Interconne
Guiding the Interactive Exploration
GNV System: A Tool for Visualizing

Figure 4.15 Result Panel

If user clicks one author in the Main View, his or her personal information like name, personal ID, university, department and so on will be displayed in the "TextArea", and the "List" below will display the authors' publications' titles. Notice that the messages in the "List" can be pressed. Here we pressed the first publication, and you will find the information about this publication is displayed in the "TextArea", but the author of this publication is displayed in the lower box, see Figure 4.16.



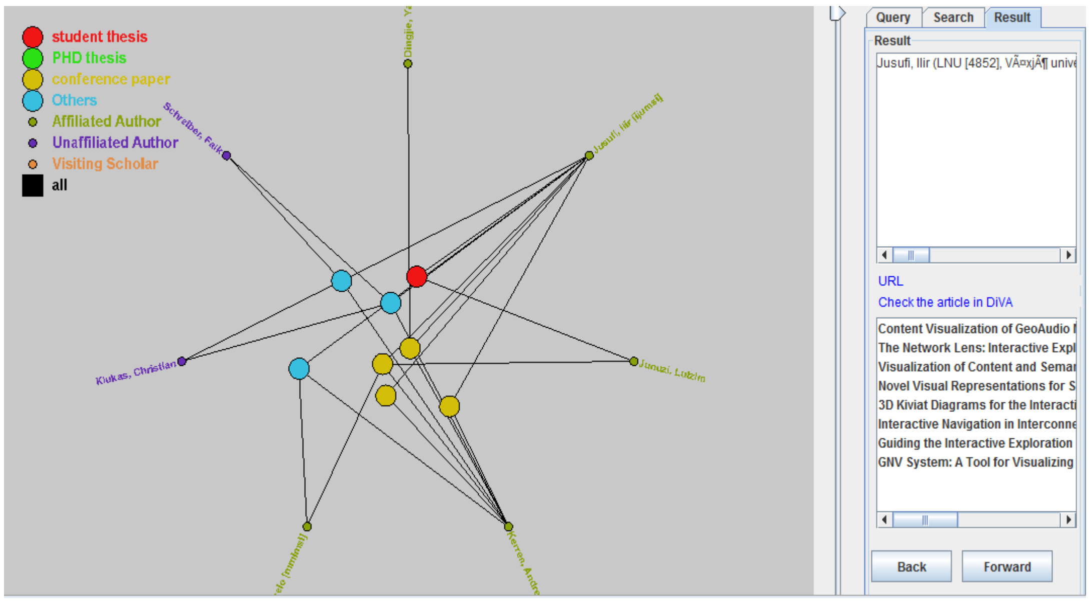
Result
Title: Content Visualization of GeoAu
Type: Student essay
Date: 2008-09-18
URL:
URN: http://urn.kb.se/resolve?urn=u

URL
Check the article in DiVA

Jusufi, Ilir (LNU [4852], VÃ¤xjÃ¶ uni
Junuzi, Lulzim (LNU [4852], VÃ¤xjÃ

Figure 4.16 Press the first publication in the "List".

No matter it is publication title or author name in the "List", it can be selected. If it is author, after pressing it, the author's publications will be displayed in the lower box and the author's information will be displayed in the "TextArea"; if it is publication, after pressing, the names of its authors will be displayed in the "List" and the publication's information will be also shown in the "TextArea". In Figure 4.16, if we press the first author called "Jusufi, Ilir", we will see such a view, see Figure 4.17



Result

Jusufi, Ilir (LNU [4852], VÃ¤xjÃ¶ unive

Content Visualization of GeoAudio I
The Network Lens: Interactive Expl
Visualization of Content and Semai
Novel Visual Representations for S
3D Kiviat Diagrams for the Interacti
Interactive Navigation in Interconne
Guiding the Interactive Exploration
GNV System: A Tool for Visualizing

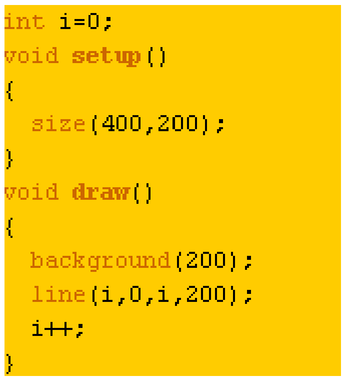
Figure 4.17 Selecting the first author's name called "Jusufi Ilir".

As shown, "TextArea" shows his personal information, and the "List" shows his publications. The visualization view will show him, his publications and his co-authors.

5 Implementation

In this chapter, we will discuss about how to download data from DiVA (in section 5.2), how to parse the data (in section 5.3) and our data structure (section 5.3.5), and how to solve those challenges in visualization (in section 5.4). Before we talk about these, we will briefly introduce how to use the software-Processing we used in our implementation. So, what is Processing? "*Processing is an open source programming language and environment for people who want to create images, animations, and interactions. Initially developed to serve as a software sketchbook and to teach fundamentals of computer programming within a visual context, Processing also has evolved into a tool for generating finished professional work.*"[2]

Processing mainly provides "`setup()`" and "`draw()`" functions to draw picture. "`setup()`" runs one time, so it is usually used to draw static picture like background. While "`draw()`"executes continuously with high frequency, which means we can take advantage of it to give people a illusion of movement by quickly change the coordinates of objects. For example, we will use the follow program to draw a dynamic picture:



```
int i=0;
void setup()
{
  size(400,200);
}
void draw()
{
  background(200);
  line(i,0,i,200);
  i++;
}
```

Figure 5.1 Processing program

Here, we set the size of the canvas to be 400 pixels' width and 200 pixels' height. Then we use "`draw()`"to draw a line, the X-coordinate of which is the variable "i". When "draw" executes over and over, "i" will growth continuously, which leads the line move from the left side of the screen to the right, see Figure 5.2:

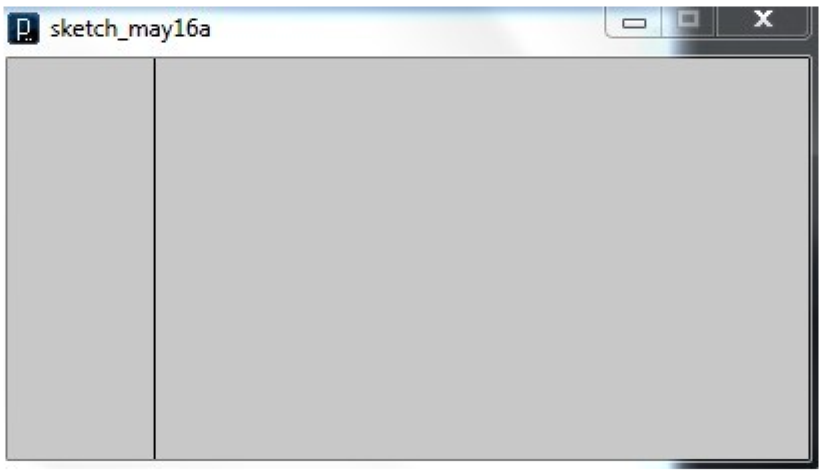


Figure 5.2 moving line

5.1 Analyze CSV file

The CSV is provided by DiVA. It is about article and article's related information, for instance author, publication data, type and so on; the total number is 52. Two different articles are separated by two double quotation marks. Different related information or called attribute are separated by one comma and two quotation marks surround this comma. Different authors are separated by semicolon.

5.2 Connect program to DiVA

Why our programs need to connect with DiVA? Consider DiVA may update its database at any time and our project will be published on line, so that our program should not use the CSV file which has been downloaded on the hard disk, but should connect directly with DiVA.

5.2.1 Create link

On the DiVA's "Create link feeds" **[1]** page, you must input several keys information, for example, the university's name and the major's title. Click "OK" after that. And DiVA will offer a link for downloading. But here we have a problem: Växjö University has change its name into Linnaeus University, but on DiVA, Växjö University and Linnaeus University exist at the same time (it also leads another problem that we will explain in section 5.3.4). And, if users just select one of the universities, the information is incomplete, which means we must select both of them.

At first, we change the URL: "organisationId: 4853", one part of the download link, to "organisationId:(4853OR1444)". We found this link works perfect on the browse, but in Java, nothing had been downloaded. Afterward we discover that the browse will change the URL: "4853OR1444" into "1444%20OR%204853" automatically, but Java can't. So, the effective URL is "organisationId:(1444%20OR%204853)". The final link is:
http://www.diva-portal.org/dice/csvAll?query=+organisationId:(1444%20OR%204853)
%20+subjectId:401&start=0&rows=500&sort=author_sort%20asc

5.2.2 Download CSV

As an Academic Archive On-line system, DiVA provides "Create link feeds" that can offer a link to download the CSV file base on user's query. And, our program uses the URL class and BufferedReader to get the information from DiVA.

Code:

```
/*download data from Diva and store in a array*/
public String Load()
{   String line=null;
    String k="";
    i=0;
    try
    {
        URL url1=new URL(x);
        BufferedReader br=new BufferedReader(new InputStreamReader(url1.openStream()));
        while ((line=br.readLine())!=null)
        {
              /*Add a mark "$" to separate contiguous article*/
              k+=line+"$";
              i++;
        }
        br.close();
        i--;
    }
    catch (MalformedURLException e)
    {
        e.printStackTrace();
    }
    catch (IOException e)
    {
        e.printStackTrace();
    }
    return k;

}
```

Code interpretation:

In CSV file, different article are separated by two double quotations, such as "2010-06-07""206842". "2010-06-07" is one article's "LastUpdated", and "206842" is another article's "PID" (publication ID). And mark "$" is used to separated two contiguous articles so that in String "k" the data is "2010-06-07"$"206842". String x is the link. Integer "i" is a static variable used to count the number of articles.

5.3 Parsing and Data manipulation

After getting the information from DiVA, we must extract information which will be used. In our design, the article's authors, title, publication data, URL, NBN, PID (being used to search article in DiVA) and public type will be used in our program. So our parsing program will split this information out by delimiters, like parentheses, square brackets, double quotation and comma.

5.3.1 Extracting single article

At the beginning, the whole information is stored in the string "L" (contiguous articles are separated by mark "$"). First, we use "`substring()`" function to extract single article, and store them in the array "parse1".

5.3.2 Extracting useful information

We use "`split()`" function to split string, and then store the information what we need in array "parse2".

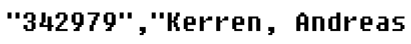

"342979","Kerren, Andreas

Figure 5.3 PID and author

As shown in Figure 5.3, “342979” is a PID, publication ID, and "Kerren, Andreas” is author. Every attribute is separated by a comma and two double quotations surround this comma, and we have to know the location of every attribute that stored in array “k” first. After the counting artificially, we know that author locates at `k[1]`, title is the `k[2]`, publication type is `k[3]`, URL is `k[22]`, NBN is `k[28]` and publication year is the `k[14]`. And `k[0]` is PID, `k1[1]` is the PID which has remove the double quotation.

Code:

```
/*Select the useful information from data */
for(m=0;m<i;m++)
{
    copy=parse1[m];
    String[] k=copy.split("\",\"");
    String[] k1=k[0].split("\"");//Remove the double quotation
    parse2[m][0]=k[1];//Author
    parse2[m][1]=k[2];//Title
    parse2[m][2]=k[3];//PublicationType
    parse2[m][3]=k[22];//Urls
    parse2[m][4]=k[28];//NBN
    parse2[m][5]=k[14];//PublicationYear
    parse2[m][6]=k1[1];//PID Publication ID
    k=null;
}
```

But, because one article may have two or more authors, we have to parse author again and store author in a new two-dimension array “Name”.

Code:

```
/*Parse the author again*/
String Name[][]=new String[i][maxau]; //maxau is the maximum number of authors, one article may have
for(j=0;j<=i-1;j++)
{
    copy=parse2[j][0];
    String[] k=copy.split(";");
    l=k.length;
    for(m=0;m<l;m++)
    {
        Name[j][m]=k[m];
    }
    k=null;
}
```

Code interpretation:

In the CSV file the different authors are separated by semicolon, such as “Mario, Albrecht;Kerren, Andreas”, so that use “`split()`” function to split string by semicolon

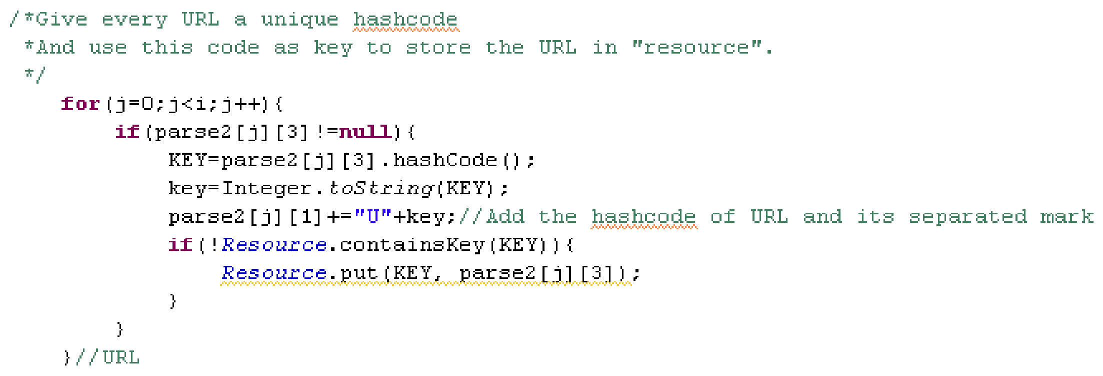
5.3.3 Internal data storage

Now we can get the information by traversing those two arrays. However, there are several shortcomings:

- We have to traverse two arrays at the same time.
- Because one author may have several publications, so the repetitive content exists
- The memory must provide a special and consecutive area for creating array, which will reduce the efficiency of using memory space.

We use a HashMap instead array to solve these problems. Firstly, the program provides a unique hashcode for one information which is in one article, then storing information in the hashmap "resource" by use the hashcode. In our program all the information (title, author, url, publication year and publication type) are stored in "resource". The code of storing title, publication year and publication type are similar.

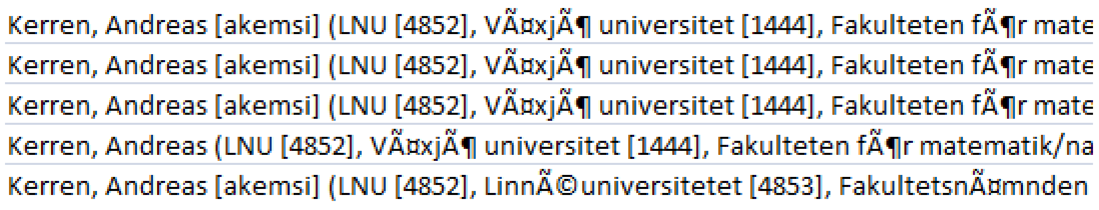
Code:



```
/*Give every URL a unique hashcode
 *And use this code as key to store the URL in "resource".
 */
    for(j=0;j<i;j++){
        if(parse2[j][3]!=null){
            KEY=parse2[j][3].hashCode();
            key=Integer.toString(KEY);
            parse2[j][1]+="U"+key;//Add the hashcode of URL and its separated mark
            if(!Resource.containsKey(KEY)){
                Resource.put(KEY, parse2[j][3]);
            }
        }
    }//URL
```

5.3.4 Combining the repetitive author

Because some human errors, such as author forgot to input their whole information or update the old ones in the DiVA, the data in CSV is not regular. For example there are three professors Kerren (as show in Figure 5.4), but actually they are the same person. Otherwise, the data include the old Växjö University and the new Linnaeus University, so the program will judge one person as two or more people! The only solution is using personal ID to be primary key, but unfortunately that some authors forgot to input their personal ID.



```
Kerren, Andreas [akemsi] (LNU [4852], VÃ¤xjÃ¶ universitet [1444], Fakulteten fÃ¶r mate
Kerren, Andreas [akemsi] (LNU [4852], VÃ¤xjÃ¶ universitet [1444], Fakulteten fÃ¶r mate
Kerren, Andreas [akemsi] (LNU [4852], VÃ¤xjÃ¶ universitet [1444], Fakulteten fÃ¶r mate
Kerren, Andreas (LNU [4852], VÃ¤xjÃ¶ universitet [1444], Fakulteten fÃ¶r matematik/na
Kerren, Andreas [akemsi] (LNU [4852], LinnÃ©universitetet [4853], FakultetsnÃ¤mnden
```

Figure 5.4: Three Kerren, in Växjö university with personal ID, in Linnaeus with ID and in Linnaeus doesn't have ID

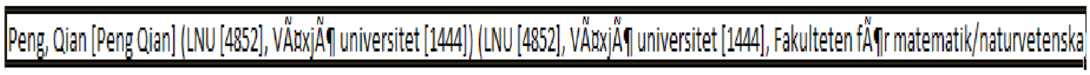

Peng, Qian [Peng Qian] (LNU [4852], VÃ¤xjÃ¶ universitet [1444]) (LNU [4852], VÃ¤xjÃ¶ universitet [1444], Fakulteten fÃ¶r matematik/naturvetenska

Figure 5.5: One person with repeated information

We do not have any perfect solution for this problem. And for decreasing the errors as more as possible, we first check the personal ID. If there is no ID we will check the name. But this method has a hidden trouble: If two people with the same name are working in the same department in the same university and one guy forgets to input his/her personal ID, the program will judge them as one.

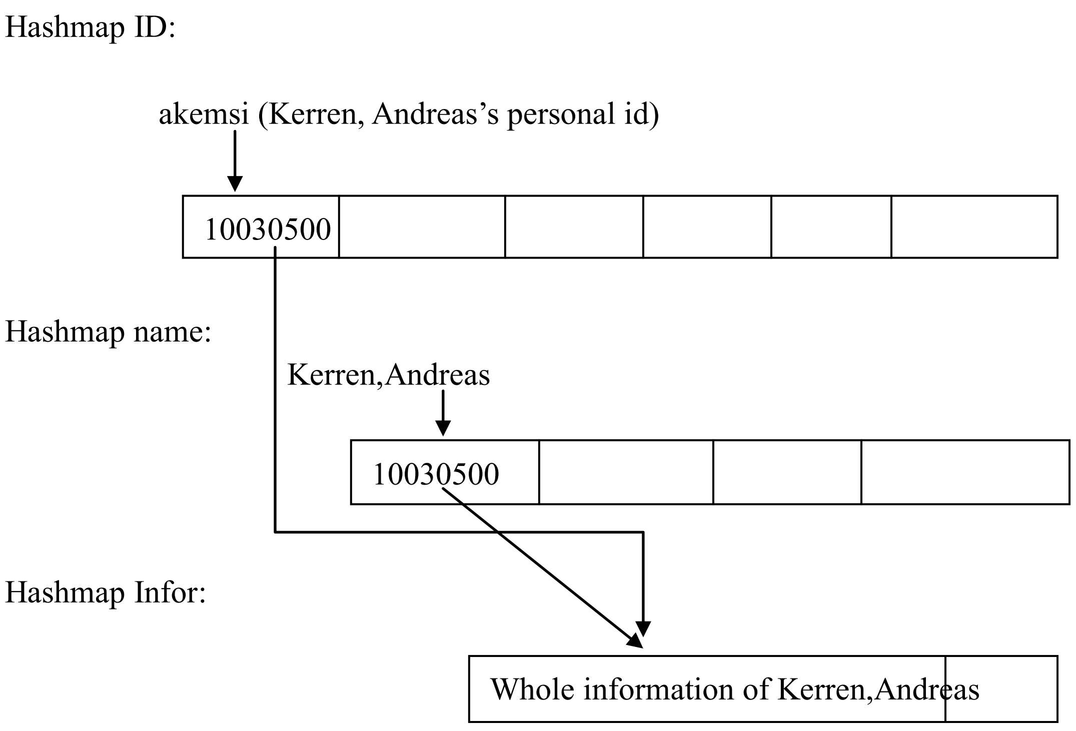
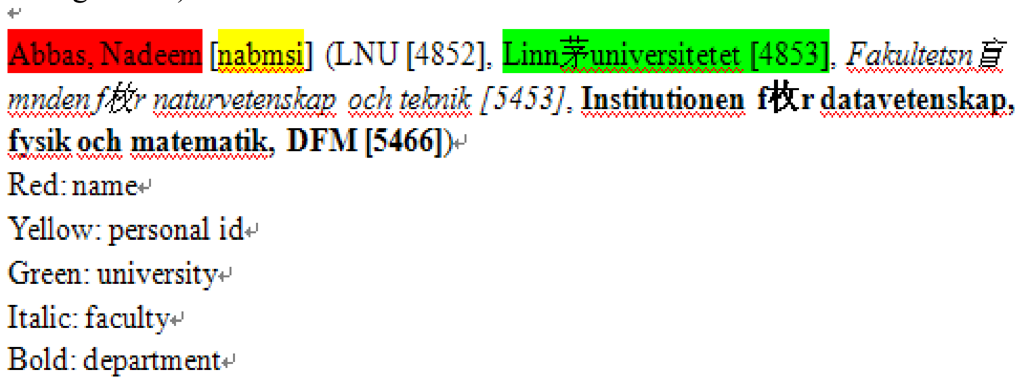

Hashmap ID:
akemsi (Kerren, Andreas's personal id)
10030500
Hashmap name:
Kerren,Andreas
10030500
Hashmap Infor:
Whole information of Kerren,Andreas

Figure 5.6 sketch map

As shown in Figure 5.6, we create three new hashmaps(Infor,ID,name), then use t
he "`hashcode()`" function to give every author (just includes the people who has additional information except name and surname) a unique hashcode(we call them author's hashcode). Afterwards, we use personal ID and author name, this name include both name and surname, as the key to store the author's hashcode respectively in hashmaps "ID" and "name". But, if an author does not have personal ID, we will only store his hashcode in "name".

At the same time we use author's hashcode as key to store information of author in hashmap "Infor". We use "`containsKey()`" to check whether the ID or name is exist or not in the hashmap "ID" and "name". If it exists, we do nothing. If not, store the author's information in "Infor" (the key is author's hashcode) and store the author's hashcode in "ID" (the key is personal ID) and "name" (the key is name). So that we can remove the repetitive author, for instance we know that in the CSV there are three professors Kerren (they are same person with different information) and after we store the first Kerren the

other two will use the first Kerren's information.
Otherwise, we use "try, catch" function to judge whether the author is a person who has information that can prove he/she is working or studying in Lnu (This information we call it additional information, it includes the university name, department and faculty; as shown in Figure 5.7) .



Abbas, Nadeem [nabmsi] (LNU [4852], Linn茅universitetet [4853], *Fakultetsn盲mnden f枚r naturvetenskap och teknik [5453]*, **Institutionen f枚r datavetenskap, fysik och matematik, DFM [5466]**)

Red: name

Yellow: personal id

Green: university

Italic: faculty

Bold: department

Figure 5.7: The information of one author

When we use "`split()`" to split author's information by comma, then store the information which has been split in a array. If author has additional information the third element of array should be the university's name, but if author does not have additional information the third element is "null" and a "nullpointexception" will be thrown.

The next step is to combine the repetitive author. At first, we check the personal ID. If this author has personal ID, the program will use this personal ID to get the author's hashcode and then get the author's information. Afterward use the author's hashcode as a key to store the author's information in hashmap "resource". If the author doesn't have an ID, we have to compare the name and combine them.

Then we check whether the author have been store in "name" or not. If he/she has, we compare the author's whole name and combine (the process is the same as deal with author who has ID). If not, we will store this author in the hashmap "Resource" directly.

5.3.5 Date Structure

Program will provide a unique hashcode to every title, author, URL, publication type and publication date. Afterward we create a string by use those hashcode, and use several special marks to separate different attributes.
"A": mean the hashcode is authors.
"U": URL's hashcode.
"P": publication type's hashcode.
"D": publication year's hashcode.
"T": Title's hashcode.

The following is one element in `parse2[j][1]`.

-770653127T-388656740U0D-1611050409A-511509699A323460424A1550574020

The letter is the separate mark; numbers are the hashcode that represent different information.

And then give a unique hashcode to every element in `parse2[j][1]`, these hashcode are not provided by `hashCode()` function but manually, the hashcode is from 1 to i (the number of articles).

But, how to get the information from the HashMap? Use the spilt function to cut the string. For example, we want to know the title of this article "-770653127T-388656740U0D-1611050409".

```
k = "-770653127T-388656740U0D-1611050409";
 String[] c = k.spilt("T");
```

The `c[0]` is "-770653127", now we have get the hashcode for this article's title. The next step is use the `get()` function to take information from HashMap.

```
String title = resource.get(Integer.parseInt(c[0])
 //resource is the HashMap
```

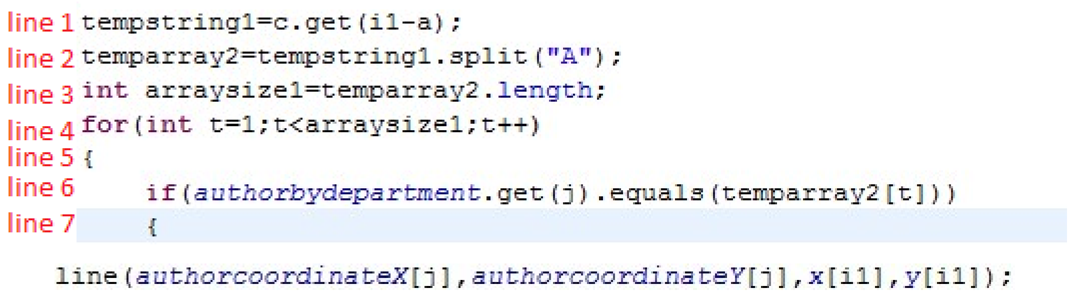
And now the string "title" stores the title of this article.

5.4 Implementing Visualization and GUI components

5.4.1 Connect authors to their publications.

We want to connect authors with their publications by line. First we take out an author to judge whether he or she writes one publication. If he did, we draw a line to connect them, if not, do nothing, then we compare it with the next publication till go over all of the publications, and do the same to all authors. So we finish the connection from all the authors to all of their publications.

Code to judge whether the author have written the publication:



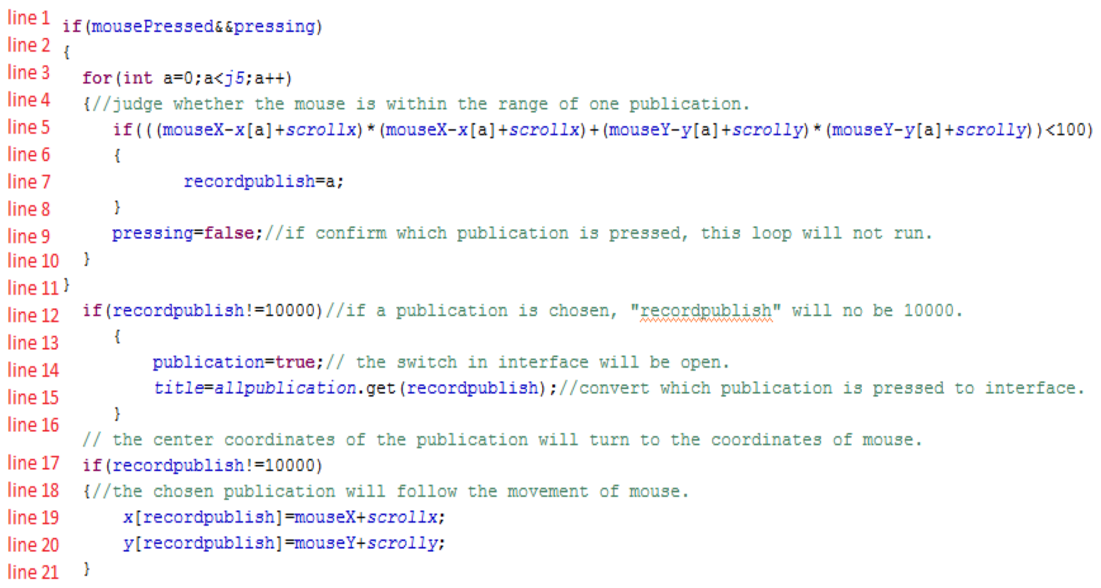
```
tempstring1=c.get(i1-a);
temparray2=tempstring1.split("A");
int arraysize1=temparray2.length;
for(int t=1;t<arraysize1;t++)
{
    if(authorbydepartment.get(j).equals(temparray2[t]))
    {
line(authorcoordinateX[j],authorcoordinateY[j],x[i1],y[i1]);
```

Code interpretation:

Here, we firstly parse the information in hashmap, to select all the authors of one publication and store them in an arraylist called "`temparray2`". In the "for" loop, all authors' hashcodes of one publication will be traversal, for judging whether this author is involved in it. If he is, it means the author has written the publication, so we will connect the author with this publication by line.

5.4.2 Spring effect

Code of spring effect:

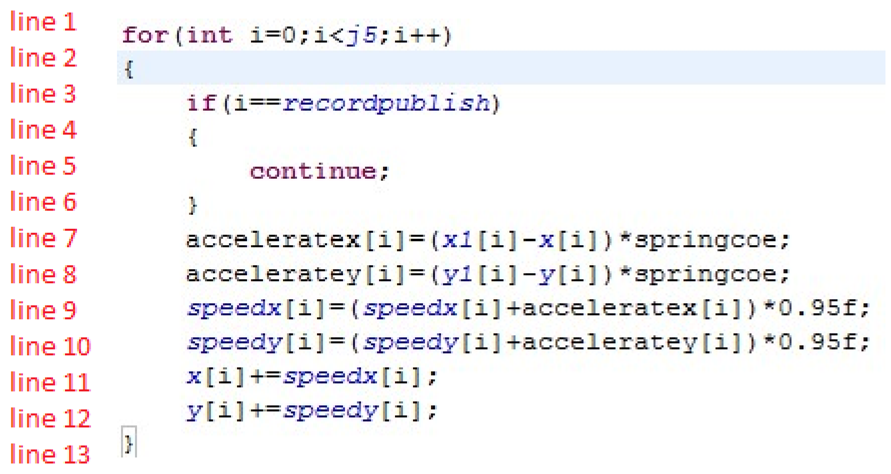


```
if(mousePressed&&pressing)
{
  for(int a=0;a<j5;a++)
  {//judge whether the mouse is within the range of one publication.
     if(((mouseX-x[a]+scrollx)*(mouseX-x[a]+scrollx)+(mouseY-y[a]+scrolly)*(mouseY-y[a]+scrolly))<100)
     {
            recordpublish=a;
     }
     pressing=false;//if confirm which publication is pressed, this loop will not run.
  }
}
  if(recordpublish!=10000)//if a publication is chosen, "recordpublish" will no be 10000.
     {
         publication=true;// the switch in interface will be open.
         title=allpublication.get(recordpublish);//convert which publication is pressed to interface.
     }
  // the center coordinates of the publication will turn to the coordinates of mouse.
  if(recordpublish!=10000)
  {//the chosen publication will follow the movement of mouse.
      x[recordpublish]=mouseX+scrollx;
      y[recordpublish]=mouseY+scrolly;
  }
```

Code interpretation:

If mouse pressed one publication, the number of this publication will be recorded in "`recordpublish`". This publication will move with the movement of mouse, because the center's coordinates of this publication will be set as the coordinates of mouse, in other words users can drag the publication.

Code of spring effect:



```
for(int i=0;i<j5;i++)
{
    if(i==recordpublish)
    {
        continue;
    }
    acceleratex[i]=(x1[i]-x[i])*springcoe;
    acceleratey[i]=(y1[i]-y[i])*springcoe;
    speedx[i]=(speedx[i]+acceleratex[i])*0.95f;
    speedy[i]=(speedy[i]+acceleratey[i])*0.95f;
    x[i]+=speedx[i];
    y[i]+=speedy[i];
}
```

Code interpretation:

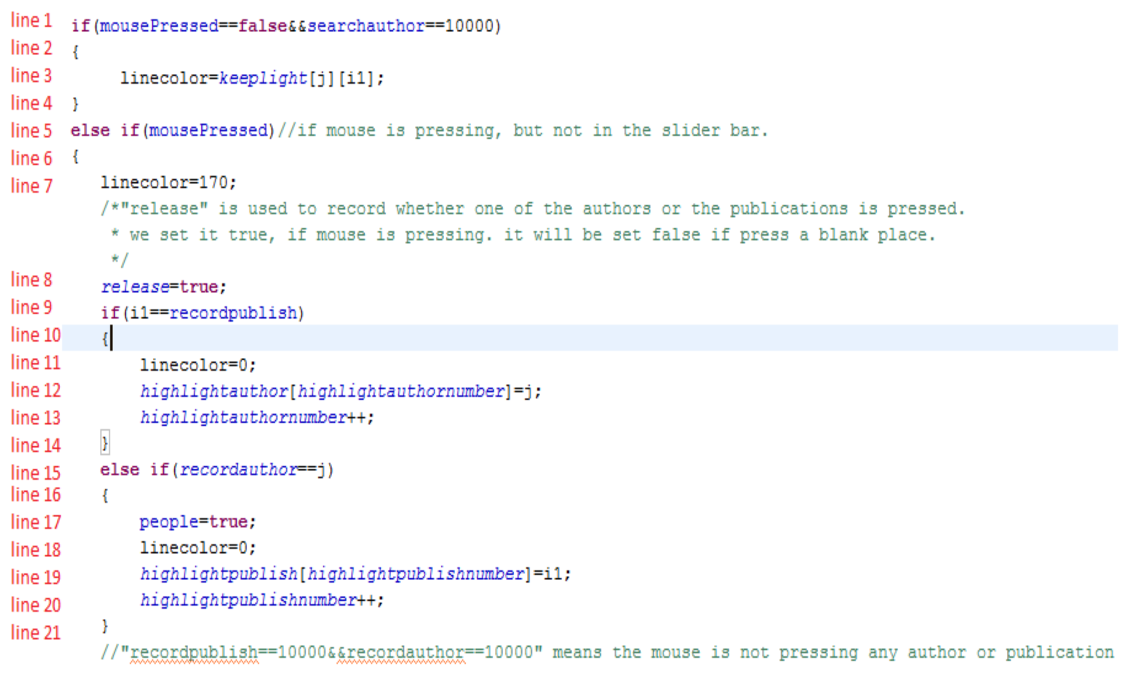
This code is used to make publications move like springs. "`springcoe`" is elasticity. "`x1[i]`" and "`y1[i]`" are the coordinates of original center; "`x[i]`" and "`y[i]`" are the present coordinate of center. So we can see that, farther the round to the original place, greater the acceleration speed is. The speed of the movement equals to previous speed pluses accelerate speed then multiples 0.95. That is because we suppose that there are 5 percents resistance needed to be subtracted. "`x[i]`" and "`y[i]`" change their value with speed in x-axis direction and y-axis direction, and will at last return to the original value and rounds will stop at the original place.

5.4.3 Highlight

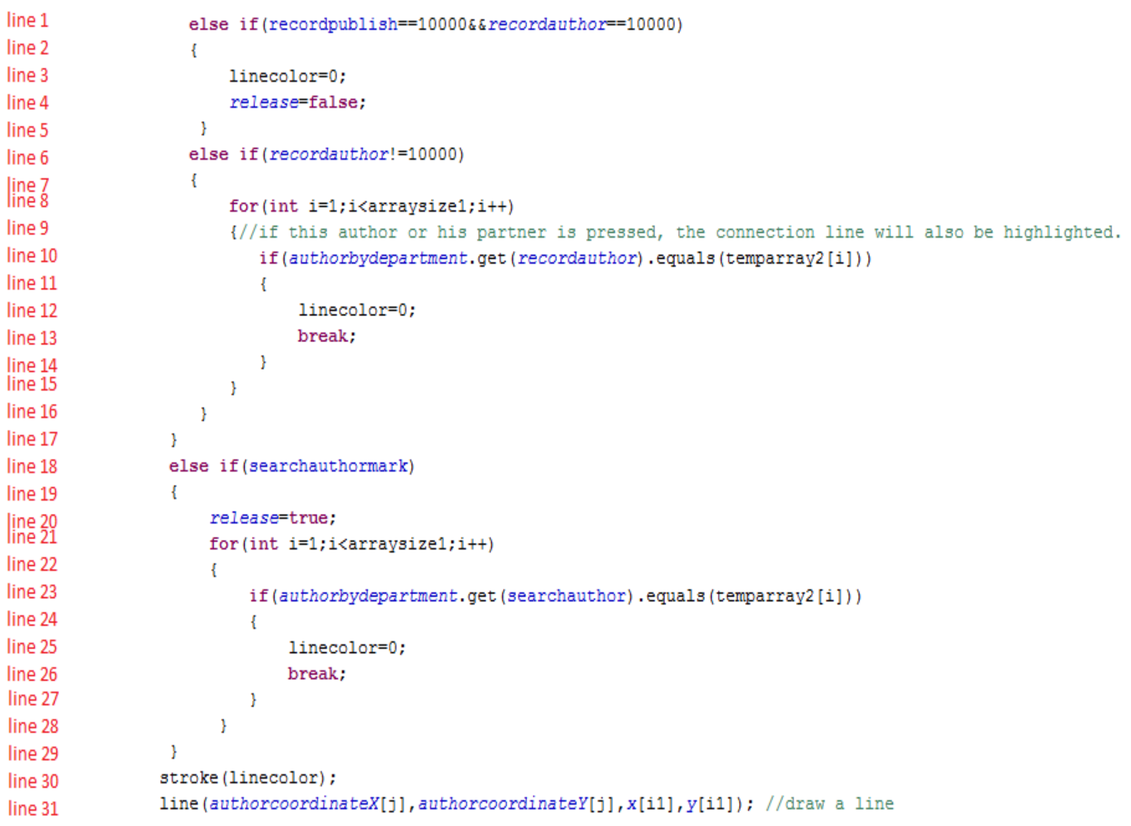
How to highlight the dots, rounds and line is the hardest and most complex part in the visualization. We will talk about each necessary function and how did we realize it. Let's illustrate the program part by part.

- **Highlighting the lines between publications and authors**

Code:



```
if(mousePressed==false&&searchauthor==10000)
{
     linecolor=keeplight[j][i1];
}
else if(mousePressed)//if mouse is pressing, but not in the slider bar.
{
   linecolor=170;
   /*"release" is used to record whether one of the authors or the publications is pressed.
    * we set it true, if mouse is pressing. it will be set false if press a blank place.
    */
   release=true;
   if(i1==recordpublish)
   {
       linecolor=0;
       highlightauthor[highlightauthornumber]=j;
       highlightauthornumber++;
   }
   else if(recordauthor==j)
   {
       people=true;
       linecolor=0;
       highlightpublish[highlightpublishnumber]=i1;
       highlightpublishnumber++;
   }
   //"recordpublish==10000&&recordauthor==10000" means the mouse is not pressing any author or publication
```



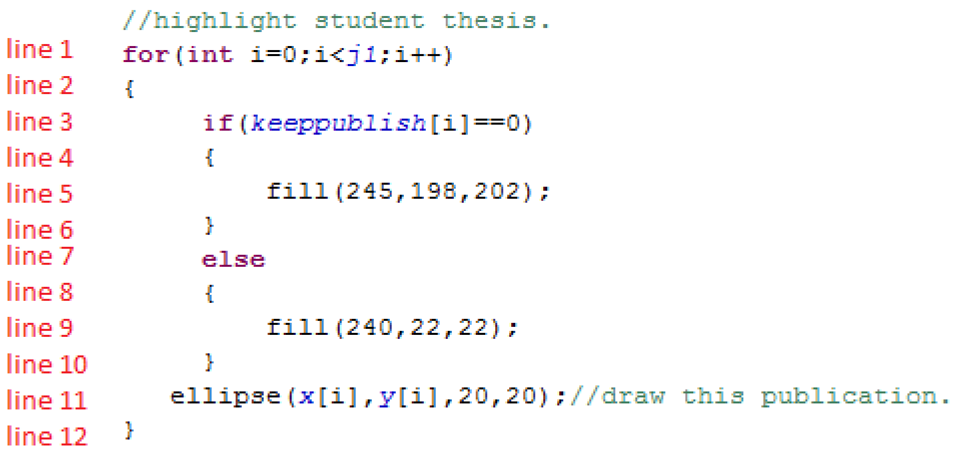
```
            else if(recordpublish==10000&&recordauthor==10000)
            {
                linecolor=0;
                release=false;
             }
            else if(recordauthor!=10000)
            {
                for(int i=1;i<arraysize1;i++)
                {//if this author or his partner is pressed, the connection line will also be highlighted.
                   if(authorbydepartment.get(recordauthor).equals(temparray2[i]))
                   {
                       linecolor=0;
                       break;
                   }
                }
             }
         }
         else if(searchauthormark)
         {
             release=true;
             for(int i=1;i<arraysize1;i++)
             {
                 if(authorbydepartment.get(searchauthor).equals(temparray2[i]))
                 {
                     linecolor=0;
                     break;
                 }
              }
         }
         stroke(linecolor);
         line(authorcoordinateX[j],authorcoordinateY[j],x[i1],y[i1]); //draw a line
```

Code interpretation: First we should know in which situation the line should be highlighted:

- At the beginning, all lines are highlighted.
- A publication is pressed, the lines between this publication and authors will be highlighted.
- When an author is pressed, the lines between this author, his co-authors and this author's publications will be highlighted.
- When user is searching on author, the author and his publications will be highlighted.
- When mouse is not pressing any place, all the lines will keep the previous color.

- **Highlight publications**

Code of highlighting publications (take student thesis as an example):



```
//highlight student thesis.
for(int i=0;i<j1;i++)
{
     if(keeppublish[i]==0)
     {
         fill(245,198,202);
     }
     else
     {
         fill(240,22,22);
     }
   ellipse(x[i],y[i],20,20);//draw this publication.
}
```

Code interpretation:
If a line is highlighted, the publication which is connected to it will also be highlighted. So just record the corresponding publications in array "`keeppublish`".

- **Highlight authors**

Code of highlighting authors (take authors who belong to DFM Department as an example):

```
for(int t=0;t<authornumber;t++)// traversal Math and Computer Science department
{
    if(t<Mainumber)
    {
        if(keepauthor[t]==0)
        {
            fill(134,162,8);// fill grass green color
        }
        else if(keepauthor[t]==250)
        {
            fill(250,10,10);
        }
        else
        {
            fill(0,164,23);// highlight (green)
        }
    }
    else if(t>(Mainumber-1)&&t<MaiOth)
    {
```

- **Highlighting authors' names**

Code:

```
if(t<halfcircle)
{
     ellipse(bigradius,0,8,8);
     text(namearray[t],bigradius+6,4);
}
else if(t==halfcircle)
{
     rotate(PI);
     ellipse(-bigradius,0,8,8);
     textAlign(RIGHT);
     text(namearray[t],-bigradius-6,4);
}
else if(t>halfcircle)
{
     ellipse(-bigradius,0,8,8);
     text(namearray[t],-bigradius-6,4);
}
rotate(angleauthor);
```

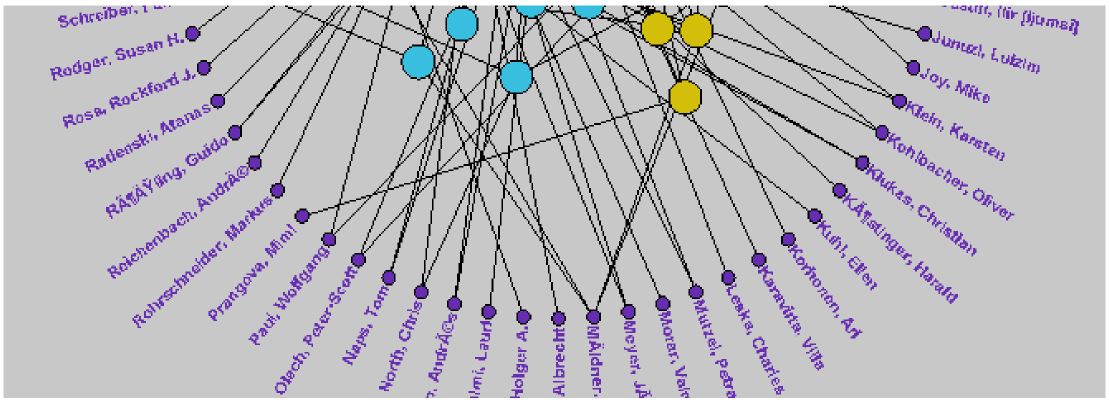


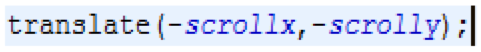
Figure 5.8 Overturn of authors' name

Code interpretation:

We can see Figure 5.8. When name is at the right half circle, the name has the same color of the dot. When name is at the left half circle, the name will be overturned to make it more readable.

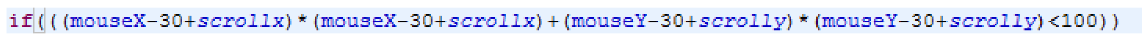
5.4.4 Scroll bar

When the horizontal scroll bar is dragged, the volume of movement will be converted to “`scrollx`” and the volume of movement of vertical scroll bar will be converted to “`scrolly`”. By using the function:

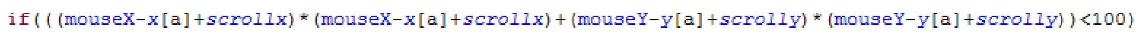


```
translate(-scrollx,-scrolly);
```

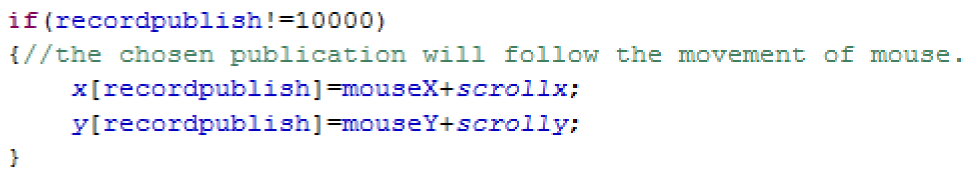
The whole canvas will move with the scroll bar.
But we should notice that, when the canvas moves, the coordinates will also change, so some functions and judgments requirement should be adjusted. For example:



```
if(((mouseX-30+scrollx)*(mouseX-30+scrollx)+(mouseY-30+scrolly)*(mouseY-30+scrolly)<100))
```



```
if(((mouseX-x[a]+scrollx)*(mouseX-x[a]+scrollx)+(mouseY-y[a]+scrolly)*(mouseY-y[a]+scrolly))<100)
```



```
if(recordpublish!=10000)
{//the chosen publication will follow the movement of mouse.
    x[recordpublish]=mouseX+scrollx;
    y[recordpublish]=mouseY+scrolly;
}
```

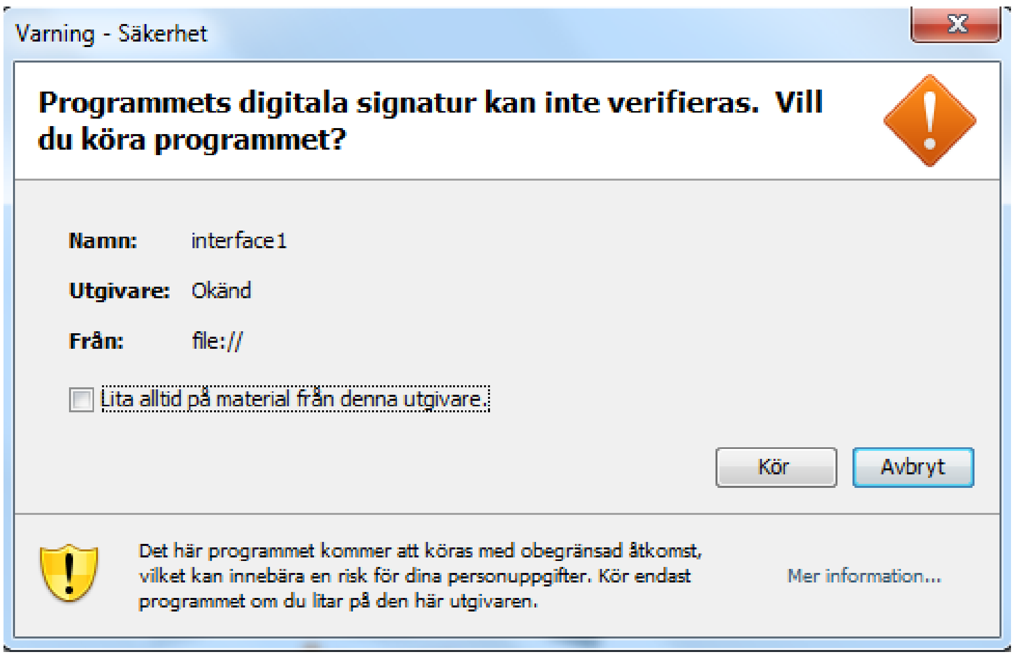
Here we just give some examples, there are not all the changes, but we can see that a lot of calculations add or subtract “`scrollx`” or “`scrolly`”.

5.5 Publishing on line

As we said in the beginning, we will publish our program to the internet. The tool should run in web browsers. So we should make our program to be applet, because applet can run in web browsers using the JVM (Java Virtual Machine). But if we add the applet in the HTML directly, a “java.security.AccessControlException” will be thrown. The reason of this exception is because “*Applets are started automatically by the browser after it downloads and displays a page.*” **[12]** And for the security reason, JVM on the browsers will not allow the applet to access the local resource or the internet resource. There are three solutions to solve this problem.

- Change the JVM’s security policy to give the applet all permission, but this solution has a problem, because we have to change the security policy in every client machine.

- Sign the applet; create a digital certificate, it includes the digital signature and where this applet came from, to sign this applet. And when user want to run the applet, the browsers will pop-up windows and ask user to trust this program or not.



Varning - Säkerhet

Programmets digitala signatur kan inte verifieras. Vill du köra programmet?

Namn: interface1

Utgivare: Okänd

Från: file://

☐ Lita alltid på material från denna utgivare.

Det här programmet kommer att köras med obegränsad åtkomst, vilket kan innebära en risk för dina personuppgifter. Kör endast programmet om du litar på den här utgivaren.

Figure 5.9 Warning windows (This picture is in Swedish, “Kör” means run, “Avbryt” means cancel)

As show in Figure 5.9, warning windows will be shown before running the applet. And if the user clicks “run”, the applet will run.

- The third solution is to use both change the security policy and sign applet, but as we discuss on the above, we cannot change the security policy in every client so that we cannot use this solution either.

At last, we decided to adopt the second solution.

6 Conclusion and Future work

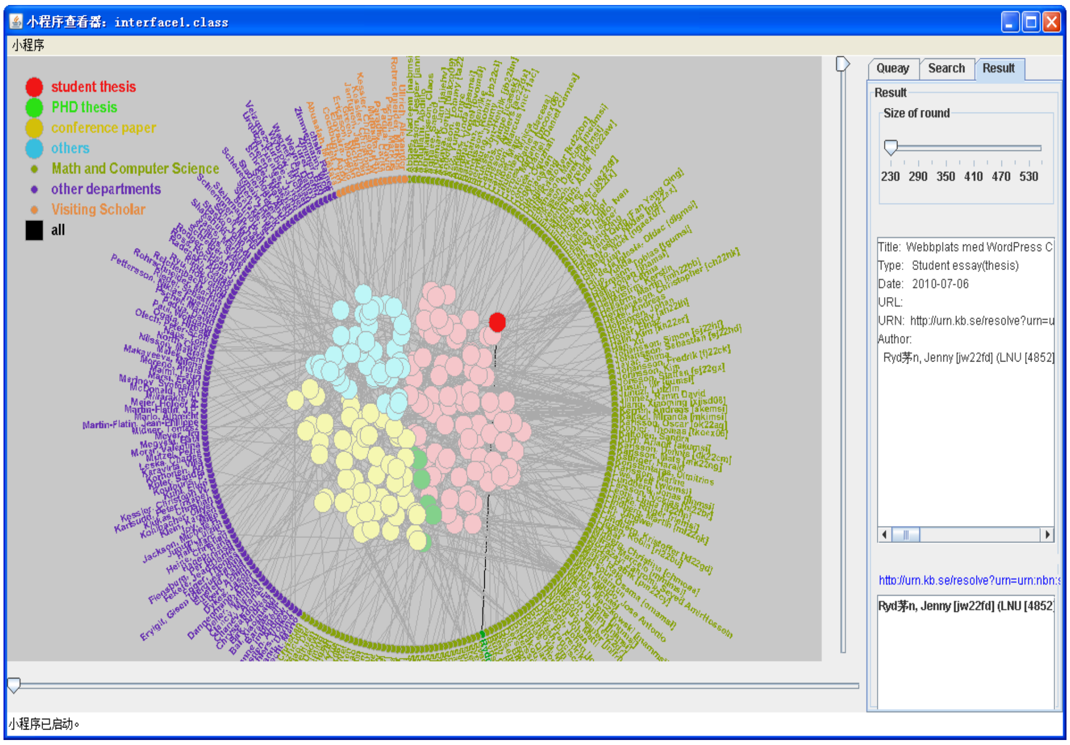
This chapter is to conclude the whole thesis and also to discuss the work which should be done in the future.

6.1 Conclusion

Chapter 1 introduces the motivation, goal and objective of this project. Chapter 2 and 3 belong to theory part, we introduced the InfoVis Reference Model, related material we referred and the visualization method we used in our design. We also trace the process of making plan of this project. We had made two initial plans, by analyzing and comparing their advantages and disadvantages, and select the most suitable one.

In chapter 4 and 5 belong to practical part, it includes Use Case which is used to tell user how to operate this widget and Implementation which introduces how to realize all these functions in code step by step. We introduce the challenge and problem we had met and introduce the process of solving them is aim to help reader absorb experience.

Until the last step of our program, we have realized the searching and querying functions, and users can get accurate information they want both in forms of words and views. The program has been noted at each line of code. And compare with searching article in DiVA directly, our tool is friendly and can show more information about the author, for example from the views (see Figure 6.1) user can know who has written the most articles. In a nutshell, the program satisfies the goal. While there are still some bug exist, and we may use more efficient method to replace the old one. Since the bugs existing in DiVA's, the "repeated author" cannot be solved by a perfect solution for the time being.



Title: Webbplats med WordPress C
Type: Student essay(thesis)
Date: 2010-07-06
URL:
URN: http://urn.kb.se/resolve?urn=u
Author:
Ryd茅n, Jenny [jw22fd] (LNU [4852]

http://urn.kb.se/resolve?urn=urn:nbn:s

Ryd茅n, Jenny [jw22fd] (LNU [4852]

Figure 6.1 Whole view

6.2 Future work

- There are some bugs needed to be corrected. For instance, the DiVA can not distinguish publications which have colon in the titles, that lead to users couldn’t search publications with a colon in it. Some authors don’t have or forget to put the personal ID in DiVA database, so that sometimes the software will take different authors who have same names as one person.
- The data structure is complex, if we want to get the useful information, we must cut the strings. It will spend lot of time, and slow the speed of running. So, we need to improve the data structure.
- In order to improve the operability, we will build a friendlier and beautiful user’s interface after receiving the advices of user and supervisors.
- We will rich the functions to make it more powerful at querying and searching. For example we want the software tells user how many partners the author which is clicked by user has without counting one by one by the users selves.

Reference:

[1] DiVA Create link feeds. Last access: June 20 2011, from < http://www.diva-portal.org/smash/builder.jsf?type=createLink>

[2] Processing. Last access: May 20 2011, from < http://processing.org/>.

[3] Drops. Last access: June 25 2011, from < http://drops.dagstuhl.de/opus/volltexte/2006/341/> .

[4] R. Spence. *Information Visualization – Design for Interaction*.3nd Ed. Prentice-Hall, ISBN 978-0132065504, 2007. page 17

[5] C. Ware. *Information Visualization: Perception for Design. 2nd Edition*, Morgan Kaufman, San Francisco, 2004, pages 149,192

[6] Weimao. Ke, Katy. Borner, Lalitha. *Viswanath. Major Information Visualization Authors, Papers and Topics in the ACM Library*. IEEE Symposium on Information Visualization, 2004.

[7] Chen. YongYue, Zhang. HuiPing, Xia. HuoSong. *Knowledge Acquisition and Knowledge Innovation Based on Visualization Technology in Digital Library*. 2008 International Symposium on Knowledge Acquisition and Modeling. KAM '08, page(s) 74-78.

[8] Chaomei. Chen, Steven. Morris. *Visualizing Evolving Networks: Minimum Spanning Trees versus Pathfinder Networks.* IEEE Symposium on Information Visualization, 2003. INFOVIS 2003. page(s) 67-74.

[9] Tze-Haw. Huang, Mao. LinHuang. *Analysis and Visualization of Co-authorship Networks for Understanding Academic Collaboration and Knowledge Domain of Individual Researchers*. International Conference on Computer Graphics, Imaging and Visualization, 2006. page(s) 18.

[10] Nathalie, Henry. Jean-Daniel, Fekete. Michael J. McGuffin. *NodeTrix: A Hybrid Visualization of Social Networks.* Transactions on Visualization and Computer Graphics, IEEE. page(s) 1302-1309.

[11] Linnaeus University Publications. Last access: July 1 2011, from<http://urn.kb.se/resolve?urn=urn:nbn:se:vxu:diva-3700>

[12] Larry. Siden. *Signed Applet Tutorial*. Last accessed : 24 June, 2011, from <http://www-personal.umich.edu/~lsiden/tutorials/signed-applet/signed-applet.html>

[13] wiseGEEK. What is a Data Table? Last access: 8 July, 2011, from <http://www.wisegeek.com/what-is-a-data-table.htm>

[14] Information Visualization Resources. Last access: August 15 2011, From < http://www.infovis.org/>