Extending Contextualized Computing in Multiple Institutions Using Threads: Final Project Report

(NSF #CNS-0722177, 10/2007–09/2011)

Prof Elizabeth Sklar
Brooklyn College, The City University of New York
2900 Bedford Avenue, Brooklyn, NY 11210, USA
sklar@sci.brooklyn.cuny.edu
http://bridges.brooklyn.cuny.edu

1 Overview

Our NSF CPATH (CISE Pathways) project was called *EAE: Extending Contextualized Computing in Multiple Institutions Using Threads* or simply *"Threads"*. The primary goal of the project was to adapt Threads-based curriculum in multiple computing departments, following the model developed by the Georgia Institute of Technology (Georgia Tech) and implemented starting in Fall 2007 [1]. The expected projected outcomes were a set of lessons learned and accompanying recommendations, advising tools and evaluation techniques that could be used by other departments interested in adapting Threads-based curriculum. Georgia Tech handled the evaluation component of the project, while four other universities focused on the adaption of Threads in their computing departments:

- Armstrong Atlantic State University (AASU),
- Kennesaw State University (KSU),
- Southern Polytechnic State University (SPSU), and
- Brooklyn College (BC).

The first three, above, as well as Georgia Tech are all part of the University System of Georgia (USG). The last, Brooklyn College, is part of the City University of New York (CUNY).

This final project report summarizes the activities that occurred during the lifetime of the project and focuses on the efforts at Brooklyn College. The Principle Investigator for the Brooklyn College component of the project was Professor Elizabeth Sklar (PI). The Co-PIs were Professors Gerald Weiss and Ira Rudowsky, also of Brooklyn College.

Three types of activities were undertaken in the project at Brooklyn College:

- *Intra-Team Collaboration*
- *Local Threads Adaptation*
- *Local Data Analysis*

Each is described in a separate section of this report.

2 Intra-Team Collaboration

This section highlights Intra-Team Collaborations that took place during the course of our CPATH project. There were meetings in January 2008, July 2009, January 2010, March 2010 and July 2010. Details of these meetings can be found in the project's annual reports for 2008, 2009, and 2010.

This section provides a brief summary of all meetings. The main topics of the meetings included progress adapting threads to each campus, evaluating progress, and project dissemination efforts.

Year 1 (October 2007 — September 2008):
A kick-off meeting was held in January 2008 at Georgia Tech for all project partners. Sklar and Weiss attended this meeting and presented information about the situation at Brooklyn College, our plans for the project and the difficulties that we anticipated for implementing Threads at BC. We projected that our biggest difficulty would be in convincing all faculty members in our department (Computer and Information Science) to support Threads. Any curriculum changes in our department need to be vetted and recommended by the Undergraduate Curriculum Committee (UCC) and then approved by the full faculty, by majority vote. *This prediction was true, and ultimately stood in the way of implementing a full-scale Threads curriculum at BC.*

Year 2 (October 2008 — September 2009):
A second project meeting was held in November 2008, in conjunction with the CPATH PI meeting in Washington. Sklar attended this meeting, at which project team members updated each other on their activities. These primarily consisted of the initial development of Threads proposals on each campus. A third project meeting was held in July 2009 at Georgia Tech. Sklar attended this meeting. The following topics were discussed: project evaluation, planning for future funding, planning a workshop to disseminate project results, and submitting a paper to the SIGCSE conference. *The workshop was held in July 2010. The SIGCSE paper was rejected.*

Year 3 (October 2009 — September 2010):
An informal project meeting was held in January 2010. The PI from the lead institution (Charles Isbell) and BC PI (Sklar) attended the CPATH "Rebooting Computing" meeting. A fourth project meeting was held in March 2010 in Washington at the CPATH PI meeting. A PI and/or Co-PIs attended from each collaborating institution. Sklar represented Brooklyn College. The fifth project meeting was held in July 2010 at Georgia Tech. This meeting was a workshop that included non-project partners as "Listeners". We discussed the experience of trying to adapt Threads on different campuses, and what each group learned from their experiences.

No-Cost Extension Year (October 2010 — September 2011):
There were no project meetings during the no-cost extension year.

In summary, the adaptation of Threads progressed differently to each campus. The annual and final reports of the other project institutions detail their progress. None of the institutions was able to implement a full Threads curriculum, as was done at Georgia Tech. One institution was able to implement a scaled-down version. All of the institutions found that factors such as lack of buy-in from other faculty, lack of support from higher-up administration and lack of resources to teach an expanded curriculum stood in the way of a full-scale implementation of Threads. The next section describes the situation at Brooklyn College.

All the evaluation funding was held by the lead institution (Georgia Tech). Their team focused on evaluating the Threads effort at Georgia Tech and assessing the progress at the collaborating institutions. Their evaluation report was released in Fall 2011.

Dissemination has been a difficult topic to discuss, since the successes were minor compared to what was envisioned when the project began. We all thought that implementing Threads on our respective campuses would be just a matter of working out the logistics, and did not foresee the political and budgetary battles that would need to be fought, many of which were lost or out of the control of the project PIs. The lessons learned were, in part, specific to the peculiarities of particular institutions. Thus it has been difficult to draw general conclusions from the experience and produce reports to disseminate that would be useful to others.

3 Local Threads Adaptation

This section presents the process undertaken at Brooklyn College to introduce Threads.

In the first year, we developed a preliminary proposal for implementing Threads at Brooklyn College and presented this to our Undergraduate Curriculum Committee (UCC) in September 2008. Feedback was received and used as input to revisions developed in the second year of the project. One of the unanticipated difficulties we encountered in developing this proposal was the juxtaposition of this new curriculum proposal with the university's approval of two new degrees within our department which had been proposed several years earlier and had finally made it through the complete approval process (which starts at the departmental level and ends with state approval in Albany). The two new degrees are a BS in Information Systems (IS) and a BS in Multimedia Computing (MC). This made the prospect of Threads more complicated because we need to support three degrees (BS in Computer Science (CS), in addition to the two new ones) while still proposing a coherent Threads curriculum.

In the second year, two revised Threads proposals were presented to the UCC. The third such revision was taken to the full faculty for feedback. Discussion was heated, and an approval vote was not taken. A new department chair and new UCC chairs were elected at the end of the second project year. They did not wish to make major curricular changes too soon after the new revisions that went into effect as of Fall 2008 (including the two new degrees, Information Systems and Multimedia Computing).

Another difficulty is that the CS and IS degrees only have two advanced electives; the MC degree has three. The UCC has been unable to find agreement amongst members to open up the degrees more and drop some of the core requirements, which is necessary to allow more of a Threads curriculum. People argued that "Threads" requires weaving *two* topics together, and that we would need to have at least 4 electives so that students could weave 2 courses from 2 topics. This would greatly increase the number of electives we had to offer, and our department cannot support this expanded teaching.

In the third year, there was limited success in implementing Threads at BC. The central office for the university (CUNY) mandated a new university-wide course numbering scheme, starting in Fall 2010. This required us to align our course numbers based on topics, and we were able to take advantage of the work done on the Threads@BC proposals to organize our courses. We were still unable to pass a Threads-based curriculum, but we were able to develop a steady-state schedule of advanced electives that aligns courses so that students could choose their electives from within a single Thread.

In the no-cost extension year, the UCC finally agreed that we could advise students using the steady-state schedule and suggest that they select their two electives from the same "advisory Thread". At the time of writing, these advisory documents are under preparation and should be voted on and ready to distribute for Fall 2012.

Throughout the process, the following concerns were identified:

- Some faculty complained that the proposed structure was too different from our current degrees. As mentioned above, many faculty were concerned that we had already implemented multiple curricular changes and introduced 2 new degrees since Fall 2006, and they were not anxious to introduce more changes. Most CUNY students do not complete their degrees within 4 years, and so introducing too many changes over a short time period means that we have students in the pipeline each following different sets of degree requirements. This is confusing to both students and faculty advisors.
- Other faculty complain that the proposed structure is too similar to our current degrees, and they would like to see a Threads curriculum that makes sweeping changes to the department's degrees.
- Everyone agreed that the idea of introducing Threads should be "advisory", so students should be able to complete the current majors if they want, without taking any Threads. This means that any definition of Threads has to either be made with the current major as is, or has to incorporate changes to the current major, which have to be approved by the full faculty. This is a significant hurdle.
- There is concern about having software to support advising for students. A student at BC worked with Sklar to

build a prototype “Four-year Planner”. The Georgia Tech team developed software called “Threadspace”, but as of this writing, this application was not adapted to other campuses.

- There is a need for advising documents for advisors, faculty and students. As of this writing, the documentation describing “advisory threads” is being prepared and will be inserted into the “Undergraduate Advice Brochure” which gets distributed to students each year. This is anticipated for Fall 2012.

4 Local Data Analysis

This section describes the efforts that were undertaken at Brooklyn College to analyze and visualize enrollment data that is available locally. Gaining a deep understanding of trends in enrollments is helpful in preparing for curriculum change and in evaluating the effects of such changes after they go into place. Development of a robust set of enrollment data analysis and visualization tools could be useful for the entire project team, as well as wider dissemination in the future. Given the myriad data collected by today’s college campuses, the reporting of enrollment data can include not only numbers and demographics of students who major in subjects of interest, but also analysis of relationships between demographics, performance and enrollment trends, examination of groups of courses taken sequentially and groups of courses taken in combination.

In the first year of the project, in anticipation of encountering the difficulties mentioned above (convincing all faculty members to support Threads), we undertook a significant data analysis project examining in detail the enrollment data for students taking computer science courses at Brooklyn College since Fall 2000. The BC administration provided us with this data large data set, and our strategy was to examine historically the enrollments in our courses and determine the interests of students in order to take these into consideration when designing our Threads. This led to the development of a database of enrollment data and the design of a software tool for analyzing and visualizing these data.

In the second year, we prototyped a suite of five analysis and visualization tools, interfaced with the database created in the first year of the project. Several applications were developed as independent tools to view and examine the database in different ways.

In the third year, effort shifted to combining the stand-alone prototypes created in the previous year into a single application and experimenting with innovative ways to visualize the data set.

In the no-cost extension year, the remaining effort was spent refining the prototype and preparing it for dissemination via the project web site (`http://agents.sci.brooklyn.cuny.edu/flora`).

The remainder of this section gives a brief tour of the prototype version of Flora.

4.1 Flora

The Flora visualization comprises two tools for studying cohorts of students:

(a) the Garden tool, and

(b) the Dandelion tool.

The Garden tool presents a zoomable interface for examining a single snapshot in time, across a range of selected dates. This view is intended to allow the user to compare the performance of student groups, filtered by selecting one of several pre-defined sets of courses. The students are displayed in gender and ethnicity subgroups. For example, one could take a snapshot that shows the performance of students who took CS1 in Fall 2003 (Figure 1a). The snapshot highlights the differences between the performance in that semester of different demographic groups. One could then take another snapshot, say of CS1 in Fall 2008, to examine the differences. (Figure 1b). The metaphor for the Garden tool is the idea of flying over farmland where one can look out the window and see fields in which different types of crops are growing. The differences from one field to another are visible, even from the air. We have defined 5 different zoomable view levels. At each level, the cohort of students being studied is divided into 10 groups, based on gender and ethnicity.

The Dandelion tool presents a perspective view over a sequence of semesters, filtered by selecting one of several starting points, where a starting point is defined by a particular course/term combination. This view is intended to allow the user to compare the effect of a particular course (or set of courses) on students' future trajectories. For example, one could look at what students took and how they performed after taking CS1 in Fall 2003 (Figure 2a). One could also look at the same view for students who took CS1 in Fall 2008 (Figure 2b). The metaphor for the Dandelion tool is the idea of flower seeds spreading, blown by the wind from the originating flower, moving on to spawn new flowers. The leftmost column of the Dandelion view shows the originating flower, represented by the term of interest. Subsequent terms are represented in columns moving to the right. The originating flower represents each student belonging to the cohort being examined, i.e., the students who were enrolled in the course and term selected.

Each tool allows selecting of various cohorts of students, such as all students who took a certain course, or students who took particular sections of that course. This is useful for comparing the effects of interventions that were implemented in particular sections. An example is shown in Figures 3 and 4. In both figures, the top view shows the performance of students in all sections of CS1 from Fall 2006 through Spring 2008; the bottom view shows the performance of students in special sections of CS1 in which an intervention was implemented[1]. The first figure (3) illustrates the "summary" view, where the block for each of the 10 demographic groups is color-coded according to the average grade over all the students in that group. The second figure (4) illustrates the "dots" view, where there is a color-coded circle for each student, placed in the appropriate 10th of the field, according to demographic group, and color-coded according to the student's grade.

The next series of figures (5 to 9) examine the Garden tool in-depth.

The most abstract view is the "summary" view (Figure 5). This appears as 10 blocks, each color-coded to the average grade for all students in that block. The grade is computed as the average of all CIS courses taken by students in that group during the time period of interest. This view illustrates the relative grades for students in each gender/ethnicity subgroup.

The next zoom level is the "dots" view. This view is also divided into 10 blocks, the same as above, but each block is comprised of circles of different colors. Each circle represents one student. The color of the circle represents that student's average grade for CIS courses taken during the time period of interest. This view illustrates the relative number of students in each gender/ethnicity subgroup as well as the grade distribution amongst students in the subgroup.

The next zoom level is the "full" view. This view also divides the cohort into 10 subgroups. Instead of circles, each block is comprised of a flower-like shape with petals. The petals represent individual courses taken by each student within the time period of interest, and the color of the petal represents the grade that the student earned for that course. The "full" view is sized proportionately so that all students in the cohort are shown; but this may mean that the individual flowers are quite small and hard to see. More detail is revealed in the two further zoom levels. But even at this zoom level, an abstract impression of the number of students in each subgroup is visible, along with the grade distribution in that subgroup and even the grade distribution for an individual student (i.e., if all the petals on a given flower are the same or different colors).

The next zoom level is the "compare" view. This allows the viewer to see the details of each flower that may be hard to see in the full view, if there are many students to display. All the students will probably not be visible in one screen at this zoom level, but the user can drag the display with the mouse, much like moving around a map in Google maps, in order to look at all the students.

The final zoom level is the "extreme" view (Figure 9). This allows the viewer to see each flower in complete detail. Only about 32 flowers can be seen at once (depending on the size of your browser window). There are two different flower shapes, one for each type of cohort. The "all" cohort flower is like a daisy, with petals arranged around a center. The center of the flower is the average grade over all the petals. The size of the petals varies with the number of courses taken by the student, so one can easily tell how many courses a student took.

[1] The intervention implemented was part of another NSF project, *BPC-DP: Building a Bridge in Brooklyn*, #05-40549.

However, this view does not indicate which courses were taken, only how many. If the “only core courses” filter is selected, then each student is represented by a flower that resembles an abstract tulip, with petals growing up from the bottom (Figure 10). Each of the 10 courses in set of courses required by all CIS majors (i.e., the CIS “core”) has a fixed position in the tulip. The bottom petal represents the introductory programming course. The next two petals above represent advanced programming and data structures courses.

Finally, Figure 11 highlights the options that the user can select with the Garden tool. The user can filter the courses shown by selecting terms within a range and pre-defined sets of CIS courses. The user can also indicate which view should be displayed.

The next series of figures (12 to 15) examine the Dandelion tool in-depth. In all views, the leftmost column shows the cohort selected by the user (i.e., the students who were enrolled in the course and term selected). Each student is represented by an $\times$, color-coded according to the grade they got in the course. Moving to the right shows what courses the cohort took subsequently and how they did, illustrating both the number of students who continued and the spread of different courses they took over time. Each row in the display is labeled with a course-name abbreviation [2].

Figure 12 shows the “random” view, in which the $\times$’s for each student are arranged randomly.

Figure 13 shows the “orbit” view, in which the $\times$’s for each student are arranged in a pattern orbiting in concentric circles around the center. The top view (a) shows the $\times$’s sorted from high grade to low grade; the bottom view (b) shows the opposite: $\times$’s sorted from low grade to high.

Figure 14 shows the “spiral” view, in which the $\times$’s for each student are arranged in a pattern spiraling out from the center. The top view (a) shows the $\times$’s sorted from high grade to low grade; the bottom view (b) shows the opposite: $\times$’s sorted from low grade to high.

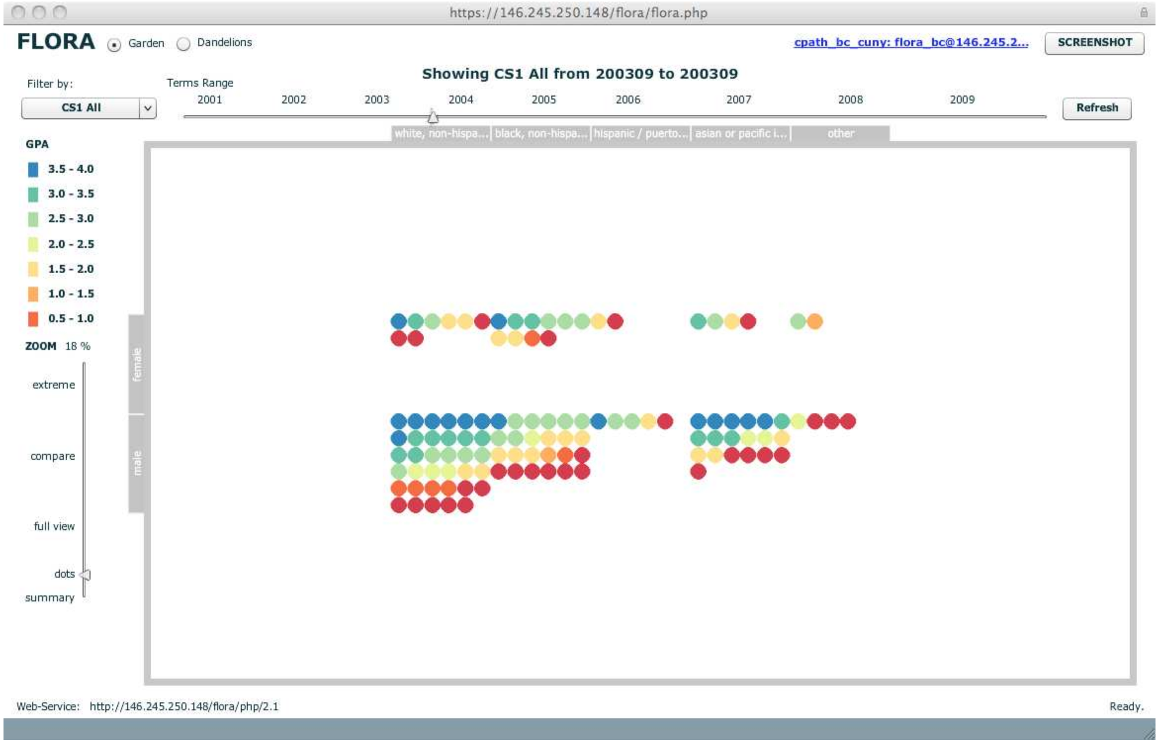
Finally, Figure 15 shows the “bubble” view, in which the students are not shown individually, but are aggregated into a single circle for each flower, color-coded according to the average grade over all the students represented by that circle.

Finally, Figure 16 highlights the options that the user can select with the Dandelion tool. The user can select the starting course (a) and term (b). The user can also indicate which view should be displayed.

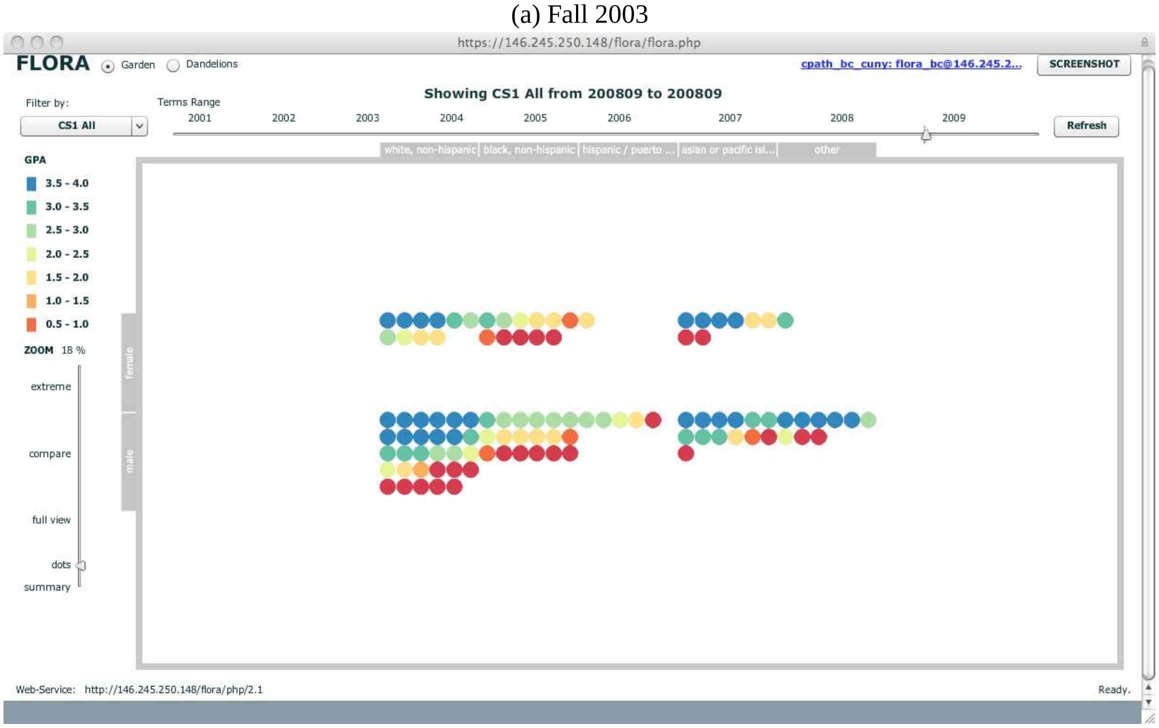
The Flora system was designed to be adaptable to other data sets. A toolkit is available on the project web site, instructing others how to set up a database with their own enrollment data and use the tool to view their data. The system uses a client-multi-server architecture, where the data can reside on a different server from the visualization engine; i.e., the system interfaces a data server, a visualization server and a client. The toolkit also includes a suite of utilities to test the connections between the user’s database and our visualization server.

The system is in an “alpha” development stage, and ready to be tested by others using their own databases. We hope that development will continue, funded through other projects. The documentation will be released as technical reports, and publications are being prepared for submission to conferences and a journal.

[2] os = operating systems, arch = computer architecture, th = theory, dm = discrete math, cs1 = introduction to programming, cs2 = advanced programming, cs3 = data structures, oop = object-oriented programming, sw1 = software applications part 1, sw2 = software applications part 2

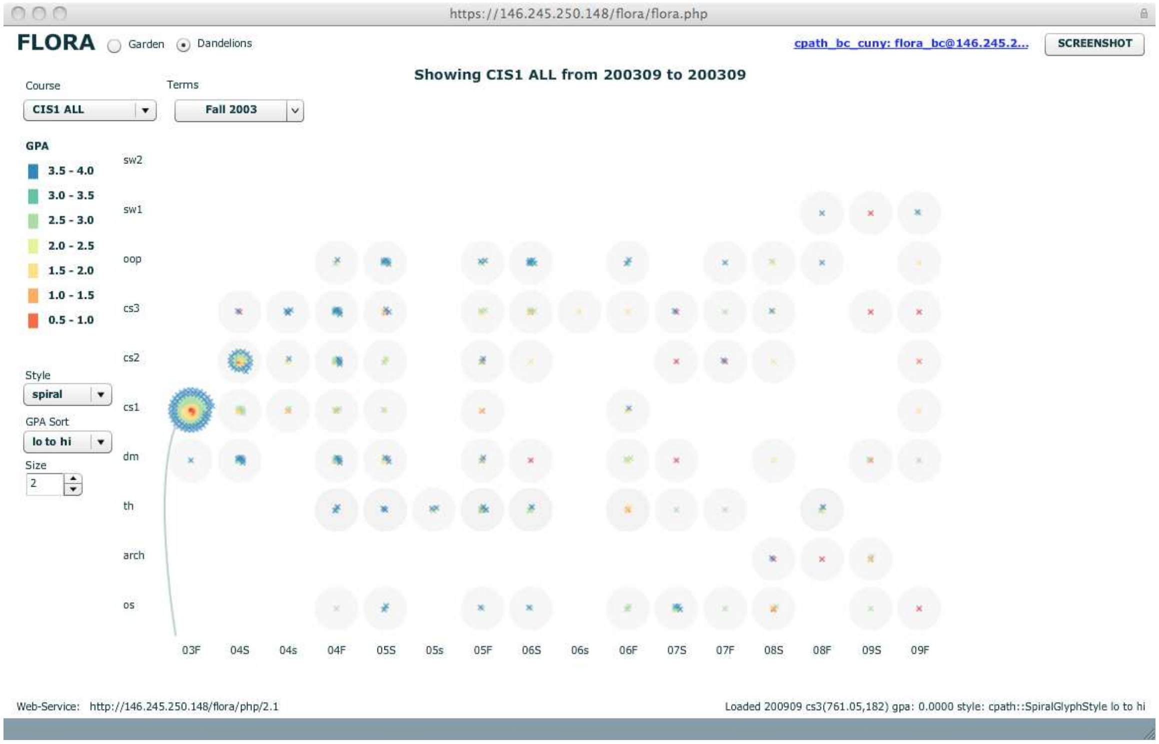


(a) Fall 2003

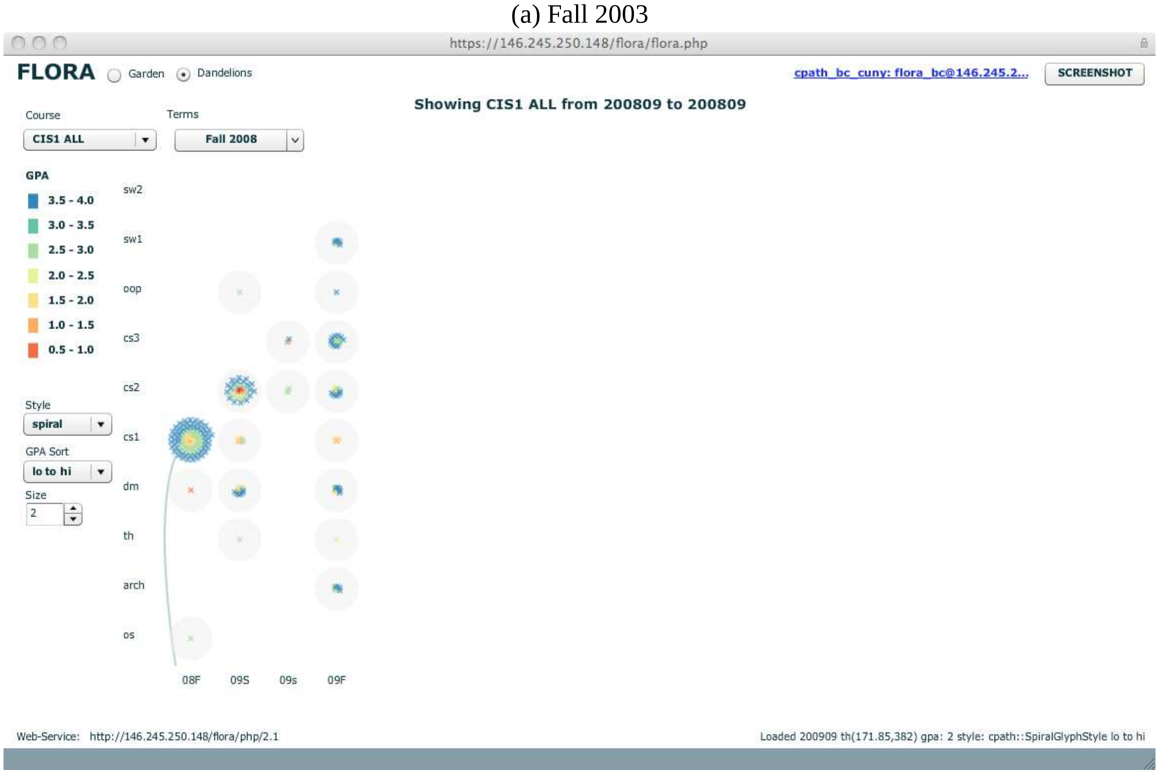


(b) Fall 2008

Figure 1: Garden tool: comparing demographics vs performance in Fall 2003 and Fall 2008 sections of CS1



(a) Fall 2003



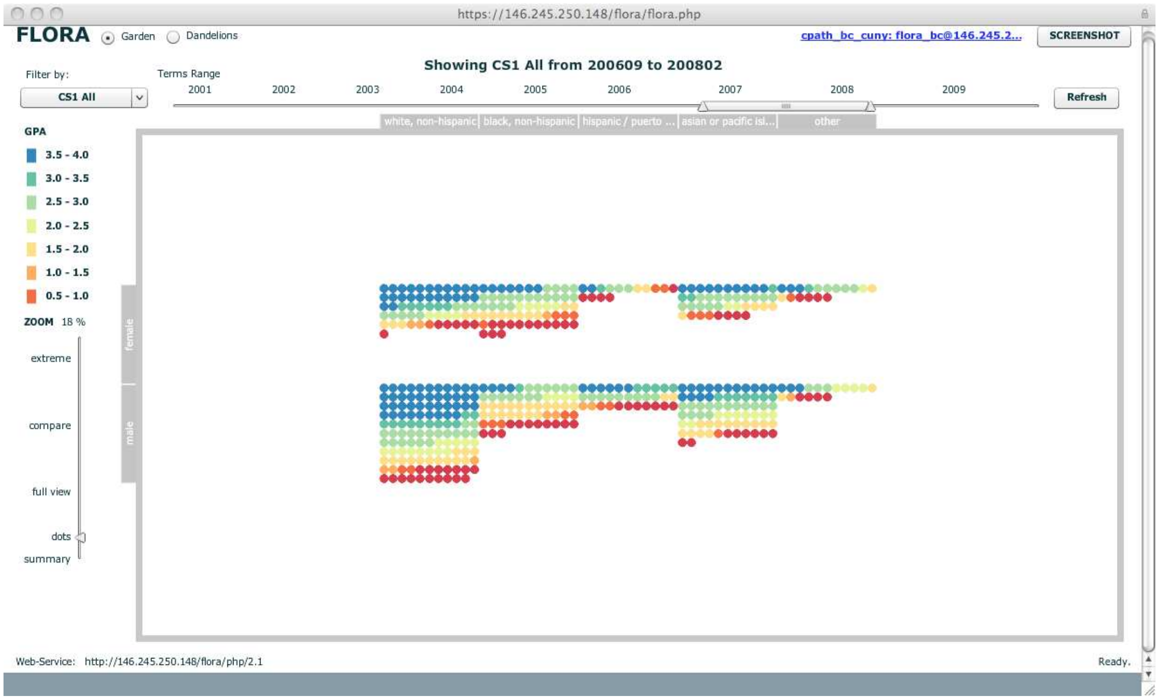
(b) Fall 2008

Figure 2: Dandelion tool: comparing trajectories, starting with Fall 2003 and Fall 2008 sections of CS1

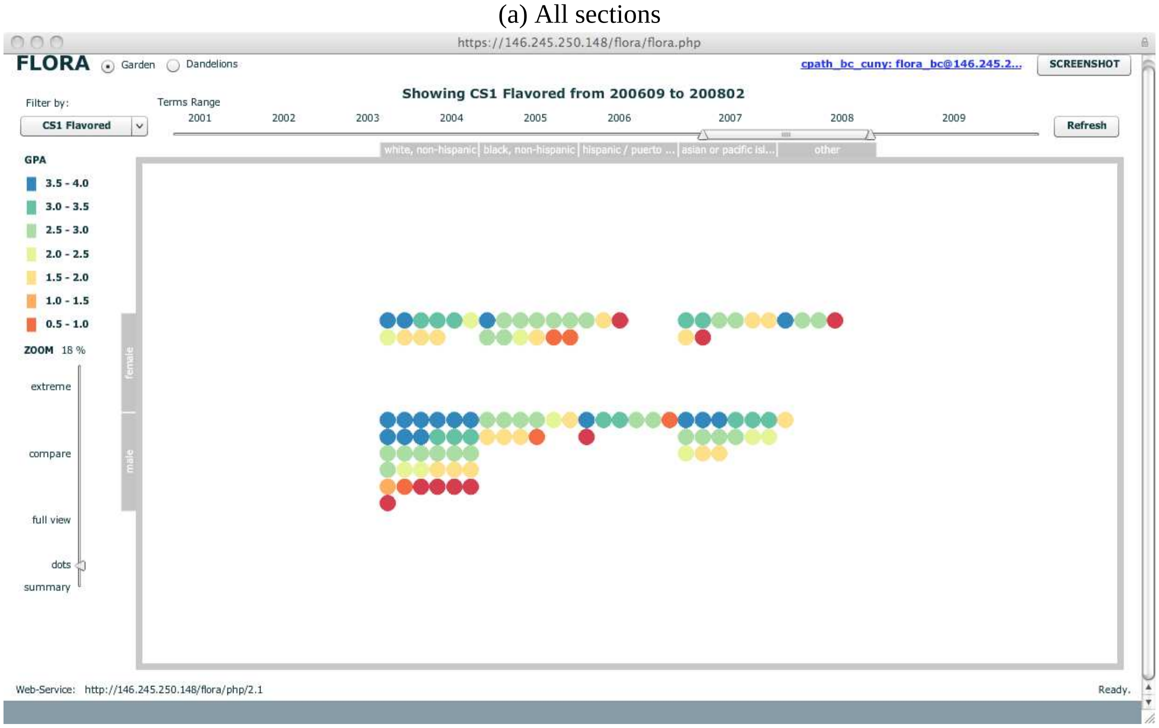
(a) All sections

(b) Intervention sections

Figure 3: Example intervention comparison, shown with Garden “summary” view



(a) All sections



(b) Intervention sections

Figure 4: Example intervention comparison, shown with Garden “dots” view

Figure 5: Garden tool, “Summary” view, all CIS classes (all students, Fall 2000 through Fall 2009)

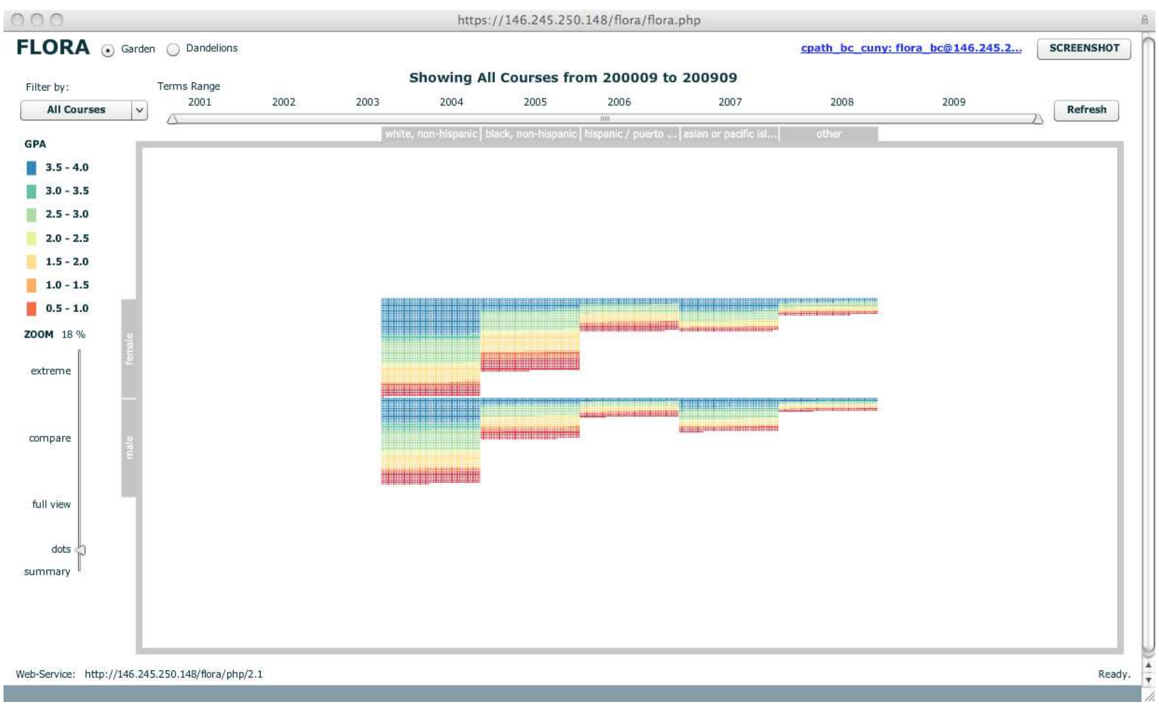


Figure 6: Garden tool, “Dots” view, all CIS classes (all students, Fall 2000 through Fall 2009)

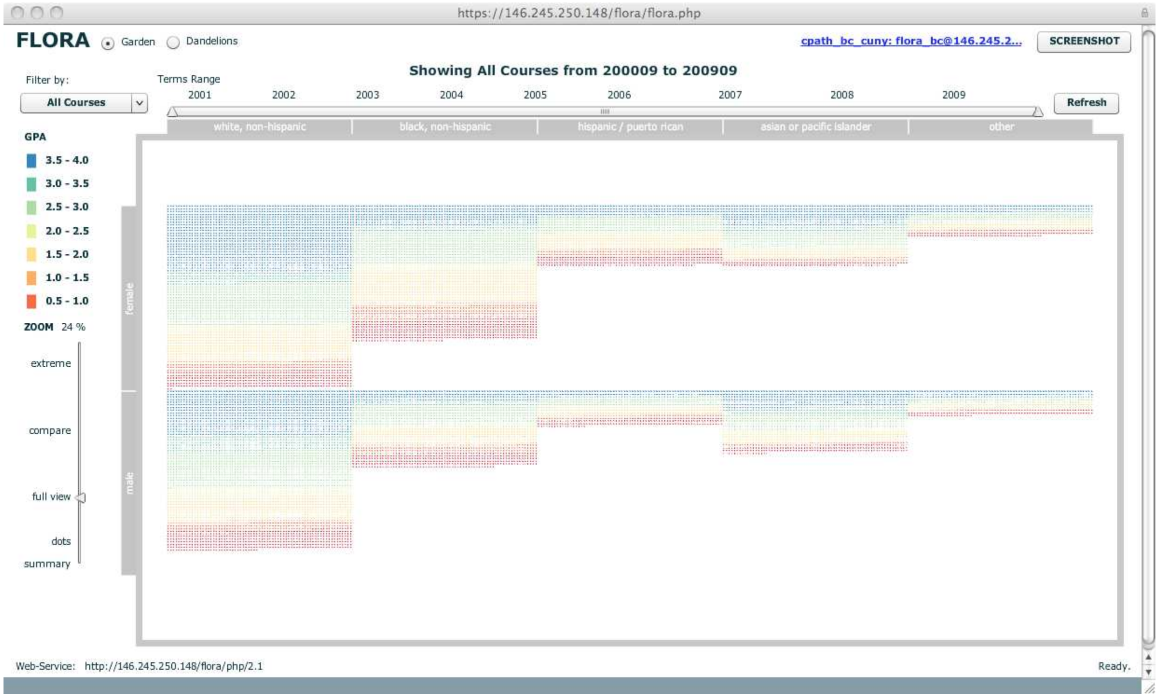


Figure 7: Garden tool, “Full” view, all CIS classes (all students, Fall 2000 through Fall 2009)

Figure 8: Garden tool, “Compare” view, all CIS classes (all students, Fall 2000 through Fall 2009)

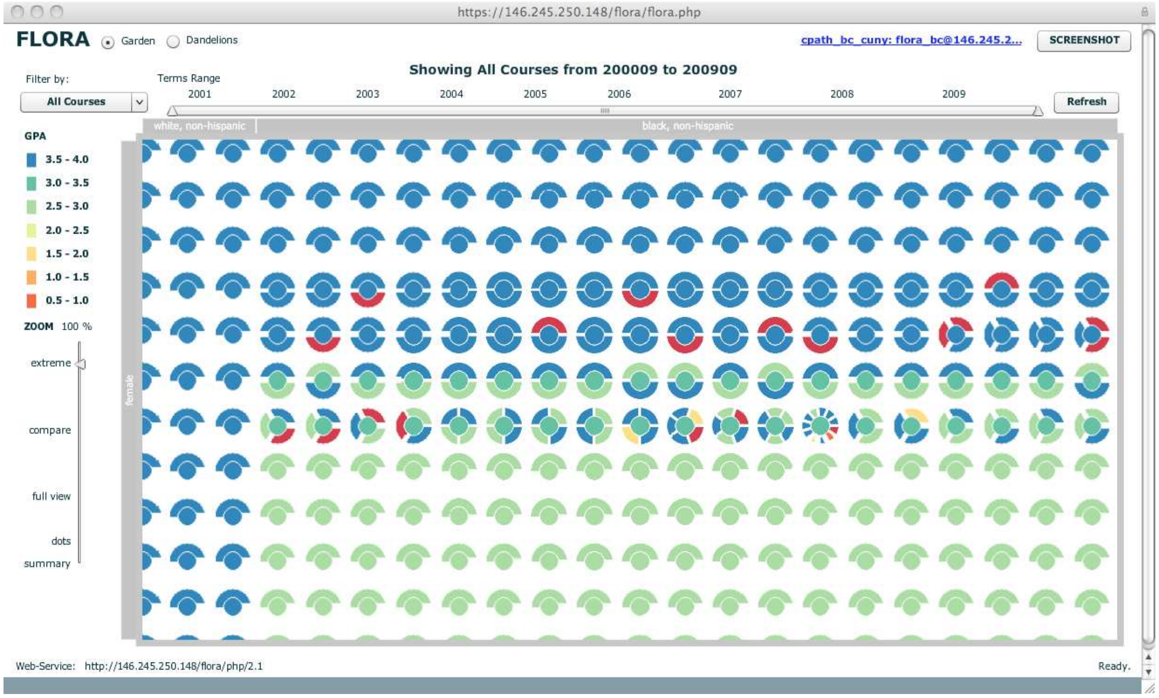


Figure 9: Garden tool, “Extreme” view, all CIS classes (all students, Fall 2000 through Fall 2009)

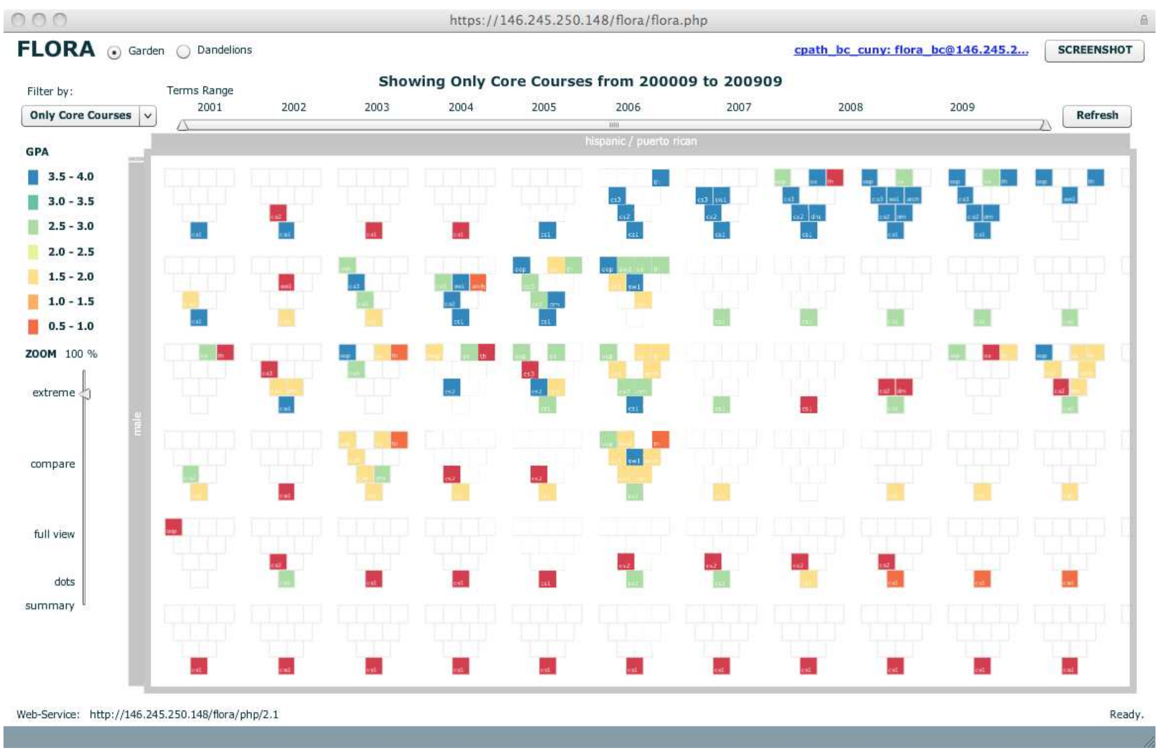


Figure 10: Garden tool, “Extreme” view showing core CIS classes (all students, Fall 2000 through Fall 2009)

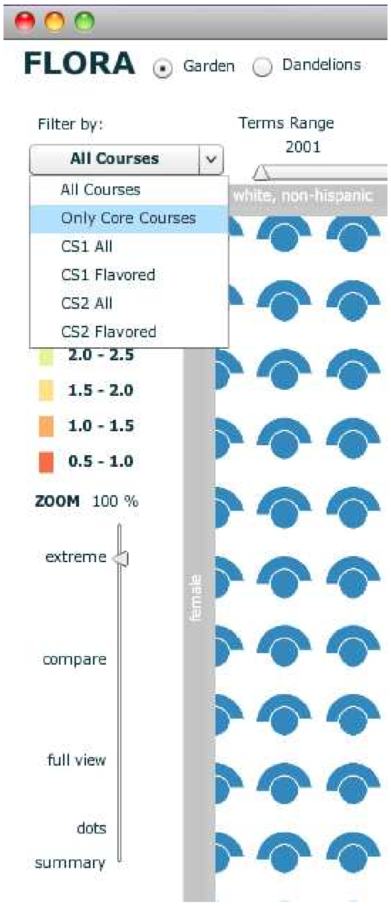


Figure 11: Garden tool, highlighting the user's selection options

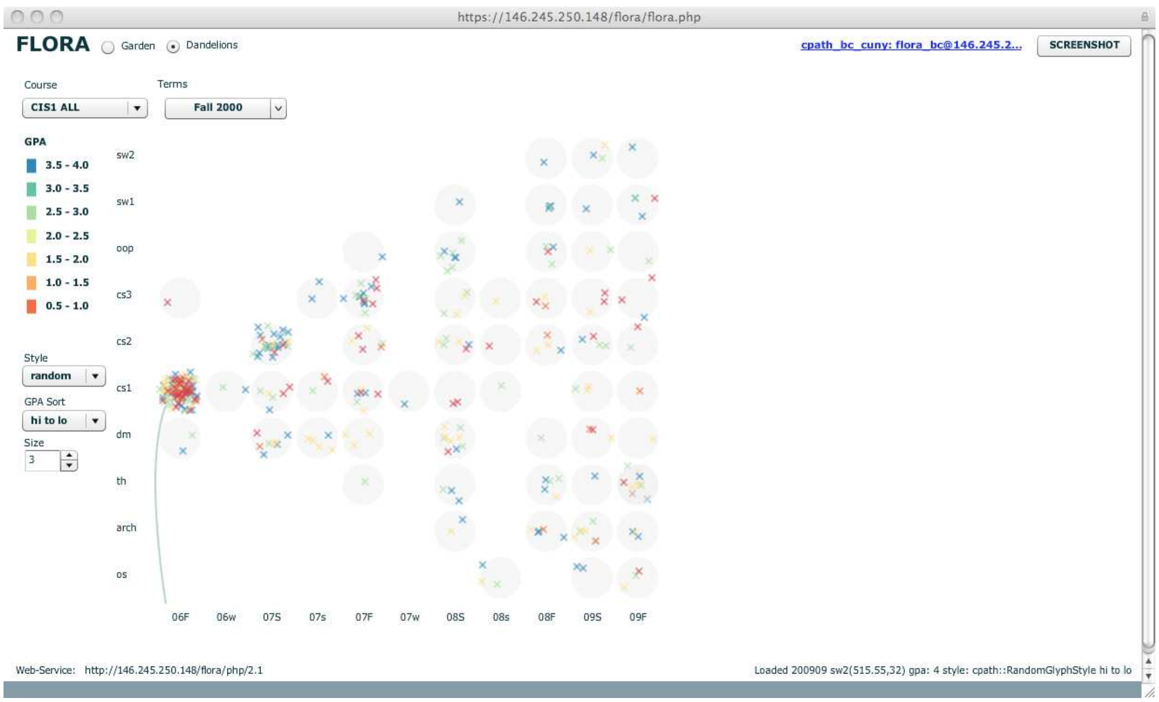
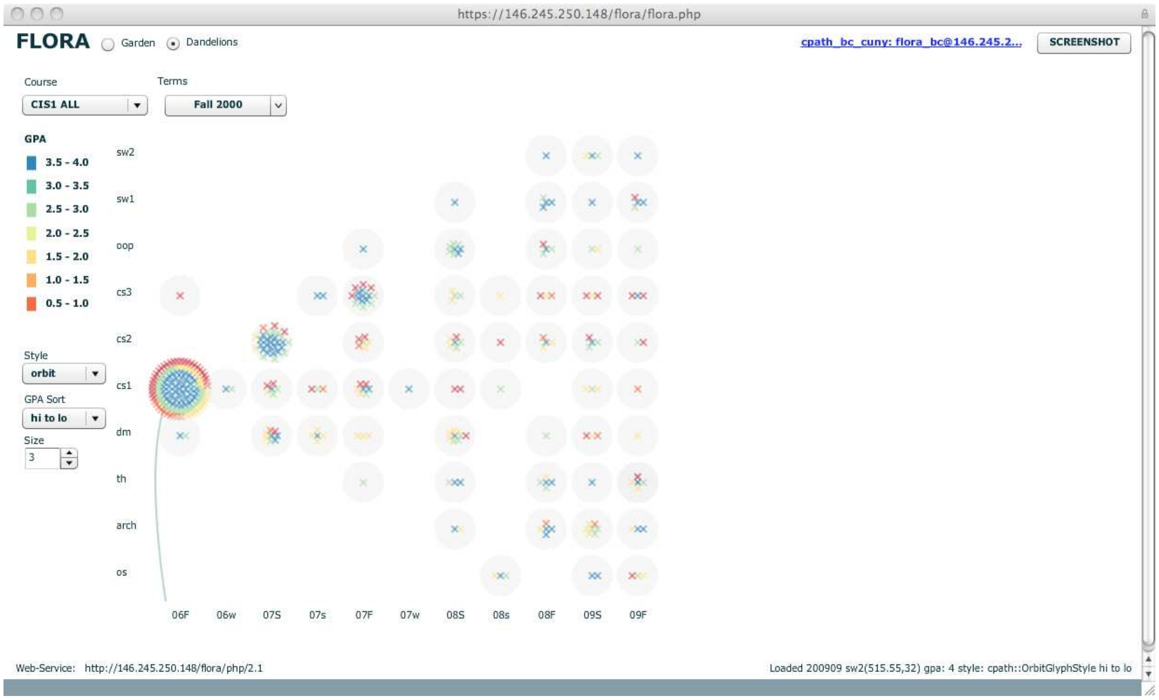
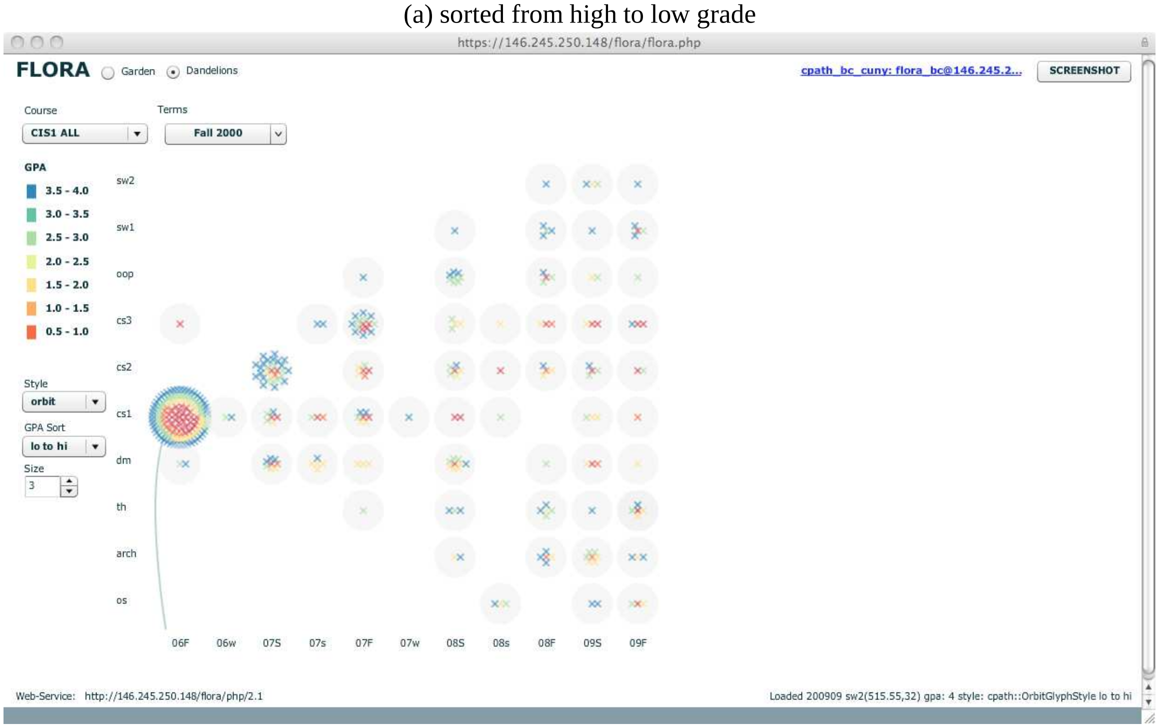


Figure 12: Dandelion tool, “random” view



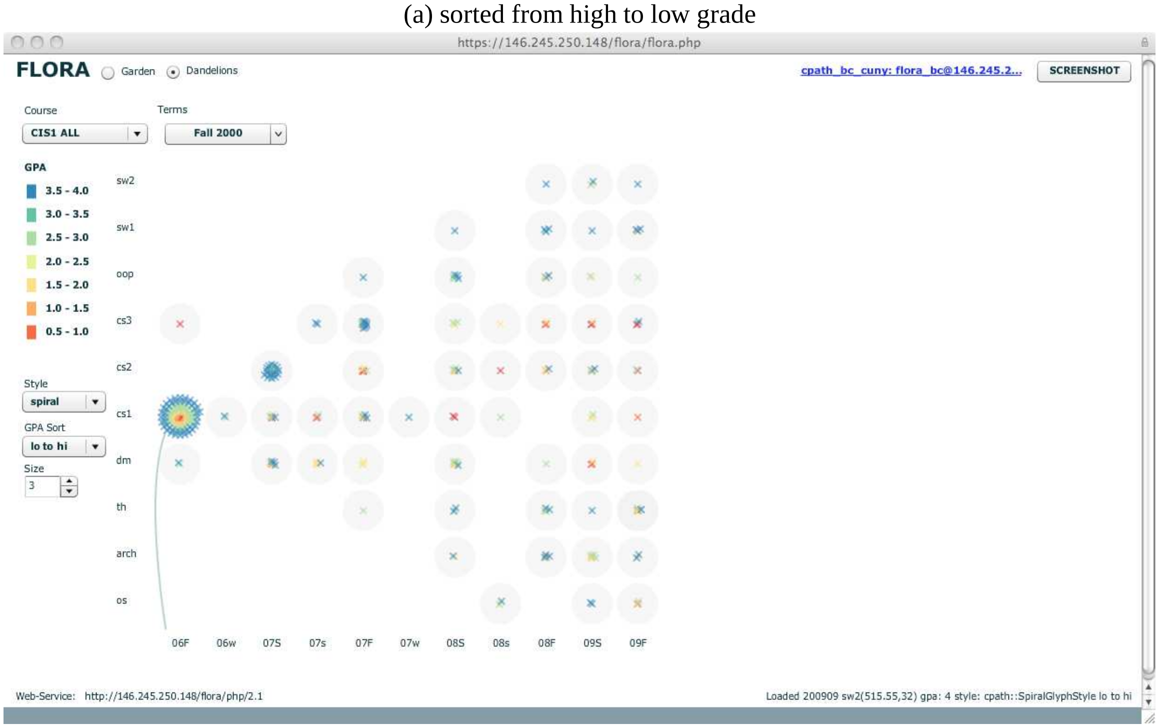
(a) sorted from high to low grade



(b) sorted from low to high grade

Figure 13: Dandelion tool, “orbit” view

(a) sorted from high to low grade



(b) sorted from low to high grade

Figure 14: Dandelion tool, “spiral” view

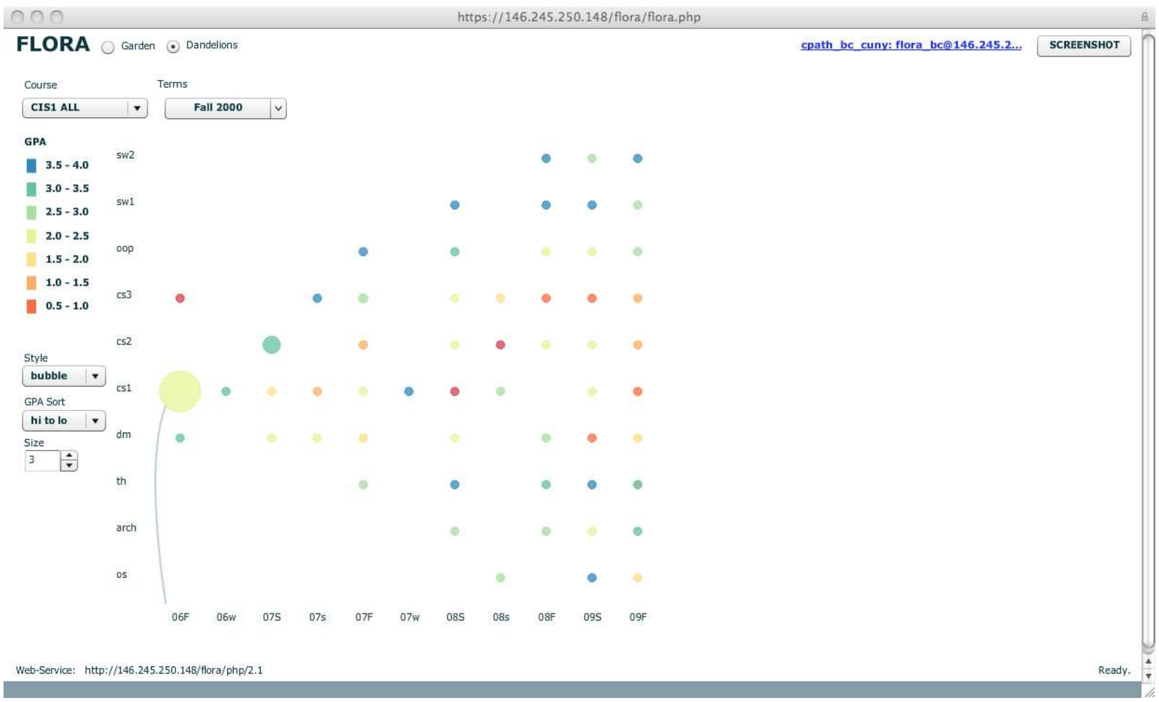
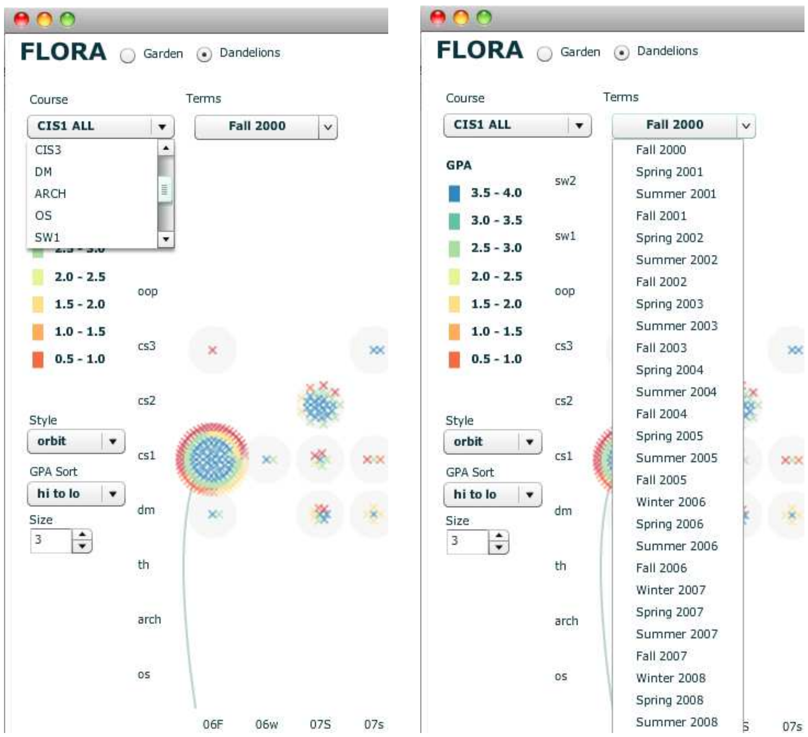


Figure 15: Dandelion tool, “bubble” view



(a) Course selection options (b) Term selection options

Figure 16: Dandelion tool, highlighting the user's selection options

5 Summary

The major findings from this project, at Brooklyn College, were as follows:

1. The overall failure to adapt the full Georgia Tech Threads curriculum at Brooklyn College is finally due to the following combination of factors: (a) the resistance of the CIS department to lessen the number core computing courses (allowing for at least 4 electives so that students could choose from 2 Threads); (b) the resistance of the CIS department to make another curriculum change so close on the heels of changes that went into effect in Fall 2008 (changes that were underway before the arrival of the CPATH grant); (c) the changeover in departmental leadership in Summer 2009, which resulted in a different set of priorities (other than curricular change) taking precedence in the department; and (d) the budgetary cutbacks that occurred in 2011, which makes it impossible to consider curriculum revisions that would expand the teaching requirements of our department.
   However the implementation of a Threads-based steady-state schedule and Advisory Threads for advanced electives for CS majors can be seen as a small victory.

2. Our local data analysis showed the basic trends in enrollments over a 10-year period. We found that there is a significant set of data that can be analyzed to gain better understanding of the reasoning behind these trends. As such, we focused our efforts on development of data analysis and visualization software that would support examination of multiple factors, to look for correlations and causes of trends. The factors include demographic information (gender, ethnicity, transfer status) as well as academic information (year in program, course pool). Our prototype software tool is called *Flora* and can be viewed on our project web site:
   `http://agents.sci.brooklyn.cuny.edu/flora`.

3. Three workshop papers and one technical report were produced using the support provided by this project [5, 3, 4, 2].

References

[1] Merrick Furst, Charles Isbell, and Mark Guzdial. Threads: How to Restructure a Computer Science Curriculum for a Flat World. In *Proceedings of the Thirty-Eighth Technical Symposium on Computer Science Education (SIGCSE)*, 2007.

[2] Ilknur Icke and Elizabeth Sklar. A visualization tool for student assessments data. In *From Theory to Practice: Design, Vision and Visualization Workshop at Design, Vision and Visualization (Vis 08)*, 2008.

[3] Ilknur Icke and Elizabeth Sklar. Visual Analytics: A Multi-faceted Overview. Technical Report TR-2009005, Dept of Computer Science, The Graduate Center, City University of New York, 2009.

[4] Elizabeth Sklar and Ilknur Icke. Using simulation to evaluate data-driven agents. *Multi-agent Based Simulation IX, Lecture Notes in Artificial Intelligence*, 5269, 2009.

[5] Elizabeth Sklar, Chipp Jansen, Jonathan Chan, and Michael Byrd. Toward a methodology for agent-based data mining and visualization. In *Proceedings of the Seventh International Workshop on Agents and Data Mining Interaction (ADMI)*, Taipei, Taiwan, May 2011.