USING ANALYTICS FOR ACTIVITY AWARENESS IN LEARNING SYSTEMS

James M. Laffey, *University of Missouri,* Christopher Amelung, *Washington University*, & Sean Goggins, *University of Missouri*

The purpose of this design case is to tell the story of our work to design context-aware activity notification for learning management systems. The story of the design experience is not simple, in that it includes research and development, conceptual evolution, implementation and extinction, and multiple phases, including inspiration for future products. The main character in our story is the Context-aware Activity Notification System (CANS). CANS was created by researchers and students in a university laboratory and started (as do many designs) while doing something else; became a dissertation; drew FIPSE (Fund for the Improvement of Postsecondary Education) funding; was implemented over a number of years in a localized setting; eventually could no longer be supported; and continues to provide inspiration for visions of new forms of online learning. The design process is described across three phases: 1) initial design as part of a dissertation, 2) extensions to CANS enabled by a grant, and 3) future visions for the impact of CANS.

James M. Laffey is a professor of Learning Technology at the University of Missouri. He has led the research and development of numerous systems to improve social ability in online learning.

Christopher Amelung is the executive director of digital communications at Washington University in St. Louis and original author of CANS. He led efforts to design and develop the CANS implementation.

Sean Goggins joined the University of Missouri's iSchool faculty in 2013. He was previously on the faculty of Drexel's iSchool, from 2009 to 2013, and is leading efforts to conceptualize the use of context-aware activity data in learning analytics.

HISTORY AND PERSPECTIVE

We write this design case to share our experience of designing a system in a university context, which may resonate with many university-based researchers who design pedagogical innovations. The Context-aware Activity Notification System (CANS) was developed over time with the contributions of many colleagues and students. As work at a university, our project began with an integration of academic and practical concerns, was fired along by data and experience which led to dissertation work and eventual external support through a grant. CANS has continued forward as an example and conceptual inspiration for new work on online learning systems. We hope that our design experience can help position other researchers/designers who are on the pathway to envisioning and advancing new ways of enabling online learning. CANS is software that augments a learning management system by identifying social behavior, such as when one student reads the discussion post of another, and provides a representation in the interface to help students be aware of such social activity at times and places where such awareness seems appropriate. Quite frankly, as educators who used CANS in our teaching, we miss the activity awareness and social nature of online teaching enabled by CANS and look for next generation systems that can empower context-aware activity notification in online learning.

As with many stories and designs, CANS began serendipitously. In 1999, we began work on Shadow Networkspace (Laffey & Musser, 2000; Laffey, Musser, Remidez, & Gottdenker, 2003; Laffey & Musser, 2006) that we envisioned as an open source learning management system (LMS) for K-12 schools to match the capabilities that systems, such as WebCT, were beginning to provide to higher education. Along with our goal of developing an open source LMS-type system for K-12, we were developing an interest in social computing (Laffey et al., 2003) and sought ways of making the experience of learning in an LMS more social and engaging. To marry our growing appreciation of the power of social computing with our efforts to develop Shadow, we implemented an Activity Monitor. The Activity Monitor was developed to support social navigation (Laffey & Amelung, 2007; Laffey, Lin, & Lin, 2006; Laffey et al., 2003). Dourish and Bellotti (1992) defined this form of activity awareness as "an
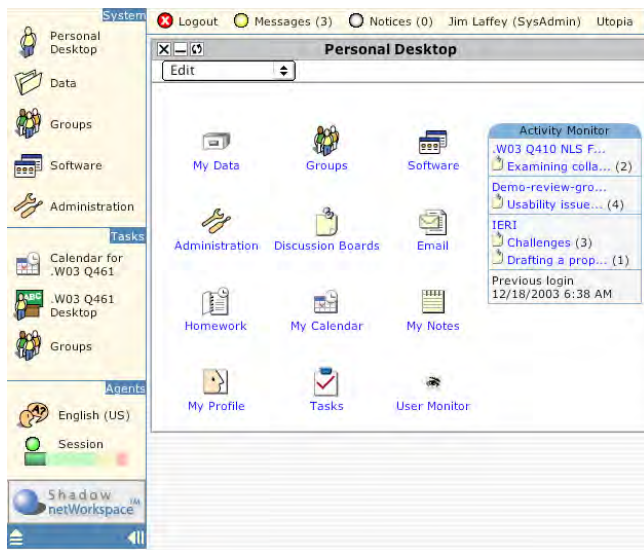
FIGURE 1. A personal desktop in Shadow illustrating the Activity Monitor.

understanding of the activities of others, which provides a context for your own activity" (p. 107). The Activity Monitor appeared on the main interface of Shadow, listing and creating links to the most recent activities in the system, such as posting to the discussion board and uploading a file (Figure 1).

As we developed and tested the system we began using it in our own online teaching. Almost immediately we, and our students, noticed that the information and functionality of the activity monitor was a really interesting part of the learning environment. We nearly always looked to see what had happened and frequently used the links as a way to navigate the course. We recognized that the activity monitor met our expectations for social navigation because we could easily follow the footsteps of others to see what had been produced and to take us to where the action was. In addition, it also created a sense of presence with others in that we could see the social context of our work and workplace by not only seeing the products of others but also by seeing and experiencing the process of others working. We were eager to have more of this! As we began to envision how the activity monitor could be refined and more richly support the social nature of learning in Shadow, we also recognized that the initial implementation would need redesign and reimplementation. Fortunately we had a doctoral student near ready to begin work on his dissertation to take up the challenge of meeting our new vision for activity awareness and social navigation.

This next section is written by Chris Amelung and describes the design work that he led as he advanced toward his dissertation and created CANS. This work included moving from our implementation of Shadow to using the Sakai LMS in our own teaching and the migration of CANS from

Shadow to Sakai. Following Chris's implementation and pilot testing we wrote a proposal to the U.S. Department of Education's Fund for the Improvement of Post Secondary Education (FIPSE) and received funding in 2006 to advance CANS. The third section of this article will describe the design work and evolving conceptualization of activity awareness in online learning afforded through the FIPSE funding. In 2013, we shut down CANS because we could not maintain compatibility with new versions of Sakai, and we did not have resources to address bugs in the user-facing CANS tools which were a major outcome of the FIPSE work. The final section of the paper is forward looking in discussing the reimplementation of the CANS design to map to newer forms of LMS and to additional conceptualizations of activity awareness in the context of learning analytics.

DESIGNING CANS

As mentioned in the previous section, with the integration and use of the activity monitor in our LMS, we had experienced a hands-on introduction to the potential of awareness information in online learning. This introduction went beyond reading and writing about social learning theory. We actually experienced the benefits and had data to support future work. We saw that our early work in Shadow and with the Activity Monitor positioned us well to advance how we did online learning and how we could learn more about the social nature of this form of learning. Unfortunately, our homegrown LMS was approaching the end of its life cycle and the architecture of our activity monitor system needed substantial re-implementation. We could not simply build on what we already had. We needed to start from the ground level and build a new framework on which we could base our next incarnation of social computing in online learning.

Activity Monitor Limitations

Before discussing the work that went into CANS, it is worthwhile to briefly explain why the activity monitor needed to be redesigned. The fundamental problem with our activity monitoring and reporting system was that it was too tightly coupled with the LMS. In retrospect, it is easy to see why we ended up this way—we were the authors of our own LMS. We were literally building and evolving Shadow as we were building the activity monitor, so it was natural for us to build the activity monitor within the LMS. The negative impacts of this design process manifested itself in two ways.

1. The life expectancy of the Activity Monitor was wholly dependent on the future of the LMS. The Activity Monitor could not exist without Shadow.
2. The usage of one system inversely affected the performance and effectiveness of the other.

The first point is simple. When designing a new system or feature, such as an Activity Monitor that has the potential for long-term growth and expansion beyond the initial host

environment, one should not design the fate of the monitoring system around the future of the host environment.

On the second point, we found in our design of the Activity Monitor that, as activity within Shadow increased, more and more computing power was required to generate the activity notifications. Because these two systems were reliant on the same processing power, the notification generation process began to negatively affect the use of the LMS. Page load times slowed to a crawl and the system became unusable at a modest level of activity. This was certainly not the experience we were designing for!

Influences on the Design

In 2003, I began work on my doctoral dissertation. The purpose of the study was to advance a theoretical framework for development that could be used by programmers to integrate activity notifications into existing computer supported collaborative environments (CSCE) (Amelung, 2005).

In addition to drawing from our own experiences with the Activity Monitor in Shadow, this research leaned heavily on previous research and existing activity notification design work.

By far, the most influential work guiding the design of this new theoretical Framework for Notification was Geraldine Fitzpatrick's past work on the Locales Framework (Fitzpatrick, 1998). The Locales Framework is based on Strauss' Theory of Action, Vygotsky's Activity Theory, and the experience of Fitzpatrick's team on the development and use of their own activity notification system. Through the five aspects of the Locales Framework, Fitzpatrick championed the importance of realizing how the interaction of users occurs within social worlds and user's actions continuously evolve over time because they are influenced by the actions of others (Fitzpatrick, 1998).

Paul Dourish's concept of "embodied interaction" was another great influencer on the design of the Framework for Notification and consequently, the design of CANS. Dourish (2001) defined embodied interaction as "the creation, manipulation, and sharing of meaning through engaged interaction with artifacts" (p. 126).

In addition to insights provided by Fitzpatrick and Dourish, two existing activity notification systems directly influenced the design of the Framework for Notification and, consequently, the architecture of CANS.

iScent, the InterSubjective Collaborative Event Environment, was built by its authors to be an extremely flexible system through its well designed distributive architecture (Anderson & Bouvin, 2000). While I did not replicate the extent of iScent's extremely distributed nature for CANS, this system did provide the inspiration and leadership required to decouple CANS from the LMS.

Groove, a desktop notification system built by the founder of Lotus Notes, Ray Ozzie, provided the design guidance needed to decouple CANS from the browser and thus gave us the ability to provide activity notifications through any network enabled device.

Principles of the Framework for Notification

The principles of the Framework for Notification are: Social Context, Awareness in Context, Activity Discovery, Trends in Activity, Meaning of Activity, and Notification Customization (Table 1).

The foundational principle behind this Framework is the social context. The social context is "the socially constructed place for user actions and interactions defined by current membership, the collective goals of individuals, recent activity, and the communicative affordances of the technology" (Amelung, 2005, p. 45). The remaining principles of

PRINCIPLE	DESCRIPTION
Social Context	The place where user actions and interactions occur. Social context partially determines the salience of awareness information, the collective goals of individuals, and the recent activity in the context.
Awareness in Context	Deliver notifications to users when the notification is relevant to the user's social context.
Activity Discovery	Allow the discovery of activity outside the user's current context to promote the formation of new social contexts.
Trends in Activity	Maintain activity and notification histories to determine the impact notifications have on user actions and interactions. Trajectory of activity partially determines the salience of awareness information.
Meaning of Activity	Provide mechanisms for users to interpret and construct meaning from the activity occurring in a context.
Notification Customization	Provide notification customization so the user has the final decision on the notifications received.

TABLE 1. The Principles of the Framework for Notification (Amelung, 2005, p. 3).

the Framework articulate how aspects of social action and information about that action moderate this foundational principle. Details about those principles and the Framework can be found in Amelung, (2005, pp. 48-52, and 2007).

The Framework's Influence on CANS

During the dissertation work, the Framework was tested and evolved through eleven iterations of CANS. The final design of CANS, and how it relates to the Framework for Notification, is presented in Figure 2.
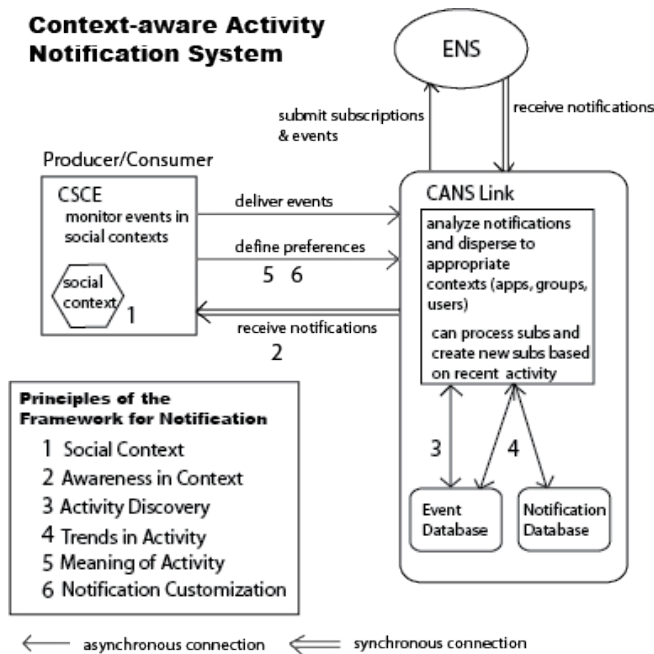

FIGURE 2. Framework for Notification and CANS (Amelung, 2005, p. 101).

After reflection on the types of changes and trajectory of development through the eleven iterations of the design, we have identified some key influences the Framework had on CANS. First, the decision for when a notification should be shared with the user must be based on the user's current social context. This principle helps the designer of future systems keep the focus on the user's current point of view and needs. Second, the Activity Discovery principle guides the developer to realize that even though the current social context has priority for the user, there are activities and conditions that warrant interruption. These notifications should not by default be obtrusive, but they should be presented in a more peripheral way so the user has the information needed to adjust their current activity and goals, if they so choose.

Third, the Framework influenced CANS by "identifying how activity is part of a process and how the knowledge of activity through notifications impacts the goals and outcomes of

those processes" (Amelung, 2005, p. 104). Defined through the Trends in Activity principle, this concept created perhaps the most long-term value for CANS. Because of this guidance, CANS was created with the ability to record the activity history and the notification history of an LMS. CANS became not only a social computing aid to online learning, but more importantly it became a research tool. As will be explained in the section on CANS and Learning Analytics, this design gave researchers the ability to not only influence users' actions and interactions, but to record, measure and analyze those interactions. And, finally, the Meaning in Activity and Notification Customization principles illustrated the need to allow users the ability to configure their own notification preferences because, through that configuration, users are able to develop a better sense of meaning for the notifications when activity occurs.

CANS in Action

All of this work and range of influences led to the CANS and Sakai implementation diagrammed in Figure 3.
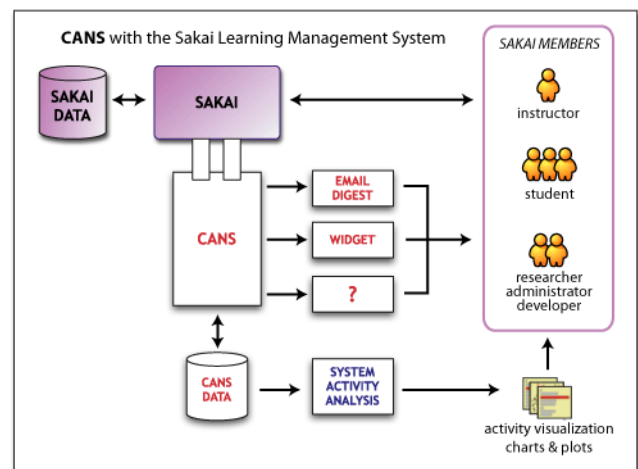

FIGURE 3. CANS and Sakai.

This implementation with Sakai allowed us to monitor user activity and record a history of the activity and associated notifications. This functionality provided a basis for envisioning ways to notify users about activity which could impact the sense of presence and co-presence, as well as provide cues for social navigation with potential to improve teaching and learning. Our first efforts to provide activity notification included email digests and desktop widgets which provided information outside of the LMS and internal to Sakai widgets such as the social comparison widget which would appear on the students LMS homepage.

Email Notification

Figure 4 is an example of a standard CANS email notification that an instructor or student would receive once a day. It is a

CANS notification digest for Thursday, December 7, 2006

History of the Internet
https://sakai.university.edu/portal/site/history_of_internet

Discussion
John Smith posted "When was the Internet invented?" 18:18:03
Robert Jones posted "What is Internet 2.0?" 15:50:10
Robert Jones posted "Re: When was the Internet invented?" 15:47:52
Susie Que posted "Re: When was the Internet invented?" 15:42:35
Susie Que posted "Re: What is Internet 2.0?" 15:37:03
Robert Jones posted "Re: When was the Internet invented?" 15:36:22

Resource
Robert Jones viewed "Orientation Activity" 18:13:05
John Smith created "Orientation Activity" 17:02:45
Robert Jones viewed "Welcome to the Class" 10:01:35
Susie Que viewed "Wecome to the Class" 09:35:22
John Smith created "Welcome to the Class" 09:34:29

Chat
Susie Que created 16:16:14
Robert Jones created 16:15:21

This automatic notification message was sent to users of the CANS Email Digest
To unsubscribe, send an email to request-cans@university.edu

FIGURE 4. CANS Email Digest.

report of the activity that occurred during the previous day. In this example, John Smith is the instructor of a fictional "History of the Internet" course. Robert Jones and Susie Que are two students in that course. From the instructor's perspective, this email notification is a tool for tracking student's activity in the class. The instructor, without ever logging into the actual LMS, can determine who has posted discussion and chat messages or viewed course resources. A student who receives this notification digest can quickly scan the email for activity they overlooked and also evaluate their level of activity compared to the other students. Tools like this email digest were very helpful in having students attend to the ongoing progress taking place in their course as well as provide models for how other students were taking on the learning tasks of posting, replying, etc. This form is quite effective in small courses or when not much activity is taking place but quickly becomes overwhelming when lots of student activity is undertaken and reported.

Desktop Awareness Widget

Another form of notification is through a computer's desktop such as the CANS Desktop Awareness Widget shown in Figure 5. A widget is intended to be a small lightweight application that routinely carries out a set of tasks; in this case our widget queries the CANS server for new notification information based on the user's notification preferences. Because the CANS system was designed around the importance of notification customization, the system could be configured, in limited ways, to conform to the user's needs. Users have the option to be notified about new, viewed, edited, and deleted files, discussion and chat messages, assignments, and announcements. The types of notifications are mapped to the types of activity that can take place in the LMS and ideally CANS would be customized to fit with individual course demands and student preferences.

FIGURE 5. CANS Desktop Awareness Widget.

Social Comparison Digest

CANS was designed to be extensible for data processing activity based on context and user preferences to generate activity representations such as the Social Comparison Digest (Figure 6). The social comparison widget is an example of how activity information can be condensed and presented to show how a user's level of activity compares to other students in the class. With a glance a student can see their number of new and viewed resources (files) and discussion messages in a group and compare that amount to the class average and the most active user in the class.
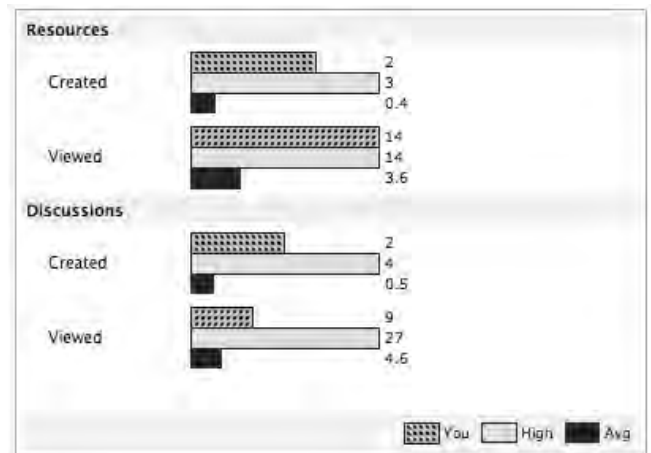

FIGURE 6. CANS Social Comparison Digest.

CANS was designed as an activity monitoring and notification system with a high level of potential for customization and configurability. We saw CANS as a powerful notification system that could impact the social nature of online learning. We also envisioned CANS as a data collection tool for social computing researchers. While discussions of big data and learning analytics were not yet the prevalent conceptualizations that they are today, the potential to use data to predict and shape behavior were beginning to inform our planning.

EXTENDING CANS

In 2006, with partners at the University of Michigan and Virginia Tech, we submitted a proposal to extend CANS by developing: 1) new representations of student activity in online learning which could impact their sense of presence and co-presence as well as support social navigation; 2) activity awareness tools that could assist online instructors; and 3) a notification manager that would allow students and instructors to customize and personalize their awareness representations. Following from the Framework and our experiences using CANS we identified accountability and meaningfulness as central to our continuing design work. We saw that accountability drove action but we also knew that students did not like the feeling of being watched and that it could be fairly easy to thwart an activity monitoring system by simply doing activities mindlessly. Finding ways to make and help students be accountable to their own expectations became a target for our work. Similarly, information is just noise in the learning process unless it means something to the tasks at hand or the social nature of the experience. Making the information meaningful was the premise behind all the trouble we went through to build in the "context-aware" functionality. However, while we could build in functionality for representing courses, tasks and members, how to apply that functionality had to be found in a balance of the course structure and the individual needs and interests of students and instructors.

A key to our new thinking about how to improve accountability and meaningfulness in the basic activity awareness tools developed during the dissertation phase of CANS was a project that I (Laffey) had led in 1992 and 1993 at Apple Computer. The project called LIMB, or Lots of Information Managed Bodaciously, was designed to support the tech support staff who answered customer phone calls needing solutions to technical problems. During that work we had identified "cheat sheets" as a key resource used by all tech support staff. The "sheets" typically were pieces of paper stuffed in a "drawer of knowledge" or tacked to a wall close to their phone and provided tech specs and troubleshooting tips that were created by their tech support peers. One staff member may have created a spec sheet on printers and at some point he had shared a copy of it with a few colleagues. However, subsequently, the original member may have upgraded the spec sheet 4 or 5 times but the colleagues may be still working from the original or other early versions. The documents were extremely valuable because they were created by experts (in a particular domain) to do the type of tasks under the conditions that each tech support person faced. To support the sharing and using of up-to-date information and not make the work of the original expert any more difficult, we conceptualized a subscription service so that once a valuable document had been identified others could subscribe to it and be sure to be kept up to date.

For CANS we reconceptualized the subscription service into a reporter mechanism whereby students and instructors could develop reporters to customize their awareness information. Typically instructors would start by creating some default reporters for students to use and then students could create custom reporters to monitor the activity of team
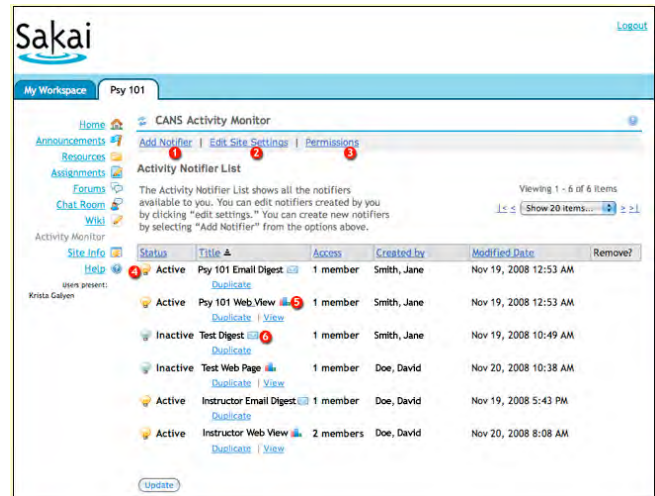

FIGURE 7. CANS Activity Monitor.

members or respected classmates or to be used for one particular assignment that required interdependence. Figure 7 shows the Activity Monitor that enabled the creation (1) and management of reporters/notifiers as an application within Sakai and relevant to a specific course. The application design conformed to the style requirements of Sakai but enabled instructors and students to create and manage reporters. They could set which activities and members to monitor (2) and create permissions for who had access to the reports (3). The list of reporters (4) includes web views (5) as well as digests to be delivered via email (6).

Figure 8 shows the result of requesting to add a reporter/notifier and shows the selections available for type of notifier (1), events to be monitored (4), members to be monitored (5), who to notify (6) and status as active or inactive (7). As we advanced through design stages, some of the concerns we encountered included addressing who had authority to enable monitoring and the granting of permissions and how to simplify the creation process. We developed a tiered system of authority so that university administrators could make decisions about what was possible across all courses and then within that range of possibilities, course instructors could make decisions about what was enabled for their courses and students. This approach allowed us to address a variety of policies and FERPA concerns across universities as well as allow instructors to be comfortable with the activity awareness practices and customs of their courses. One of the key approaches to simplifying the creation process was to allow members to duplicate and modify reporter/notifiers

FIGURE 8. Form for setting up an activity notifier.

FIGURE 9. An email digest which can be sent to a student each day for each class.

instead of starting over with each new idea for activity awareness.

Figure 9 shows a sample of a new model of a daily email digest. The sample shows the daily activity of the course through a summarization of the key activities and then lists the resources created and viewed by other students. This view of a digest is based on decisions that the teacher and student have made about what is available and presumed useful.

Another view of the same data can be provided through an interactive web page as is shown in Figure 10. This page represents a one-day view, the last three days view, or the prior week's view, and is typically of more interest to the instructor than the students. It allows the instructor to see a visualization of each member's levels of participation in the left column and allows for sorting as well as a mouse over for more detail. The top right display currently showing Resource Views can be switched between events such as discussion posts and views so the instructor can see the actual items the students are posting or accessing as well as being able to click through to the item. These items can also be sorted by title, members, or dates. The bottom right view provides a numeric summary for each member across each event type. Instructors reported this interactive and summary form of data to be of value for identifying students who may be falling behind or for monitoring how students follow instructions in a lesson.

Figure 11 shows a view of data developed for a widget on the home page of the course site in Sakai. The purpose of developing a widget and placing it on the course home page is to make the social information available to the students while they are working on course activities and to put the information in "plain view." The top pane of the widget shows social comparison data from the last 7 days of activity and allows the student to see how their level of activity

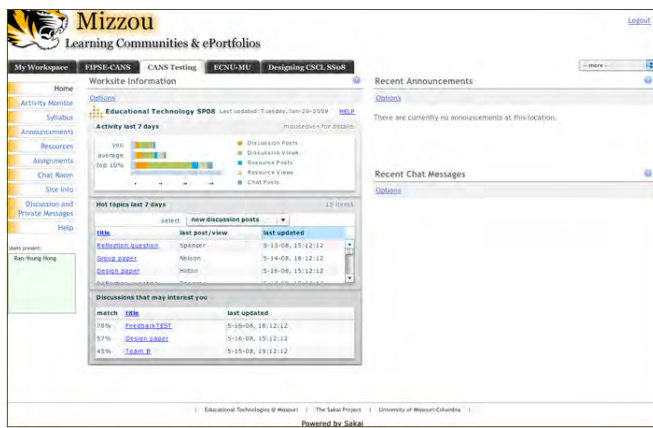**FIGURE 10.** Interactive web page for examining course activity.

FIGURE 11. CANS Widget data presentation in Sakai.

compares with the course average or with the top 10% of active students. This is a form of support for accountability in that it allows the student to compare themselves with others in the class and determine if the comparison data indicate a need for changes in activity level on their part. The middle pane allows selection across event types such as discussion and resource posts and views and shows the most recent and popular events for each category. If a student sees an object, such as a discussion board post, that is of interest to them they can click the name and go directly to the discussion. The bottom pane provides recommendations based on data from the student and peers to suggest potentially relevant and interesting activity.

CANS AND LEARNING ANALYTICS

In the time since the development of CANS a new community of scholars focused on learning analytics in large-scale course management systems and other learning environments has emerged. Sean Goggins, who was a graduate student in our program following Amelung, experienced CANS in his courses and subsequently used CANS data to formulate an approach to understanding online learning in groups. Goggins is now leading several research projects that take insights from the CANS research as a basis for developing new formulations for learning analytics (Paredes & Chung, 2012; Reynolds & Goggins, 2013; Xing & Goggins, in press; Xing, Wadholm, & Goggins, 2014). We believe that the early work on CANS and activity awareness in context, when coupled with new approaches and thinking about learning analytics, serves as a prime example for the kinds of tools that will help instructors manage courses and students to manage their learning.

We have published a number of articles in the learning analytics and related literature, with Goggins' leadership especially focusing on the use of CANS data to understand small group behavior in completely online learning environments (Goggins, Galyen, & Laffey, 2010; Goggins, Laffey, & Galyen, 2009; Goggins, Laffey, Amelung, & Gallagher, 2010;

Goggins, Laffey, & Gallagher, 2011; Goggins, Laffey, & Tsai, 2007; Goggins, Mascaro, & Valetto, 2013; Goggins, Valetto, Mascaro, & Blincoe, 2013).

The future of tools to support design through learning will include a) continued advances in awareness for both teachers and students; b) the design of tools that better support online small group work; and c) the incorporation of learning indicators (analytics) in those tools. Finally, the sustainability of innovative learning designs will benefit from researchers and institutions of learning embracing of open source LMS's such as Instructure Canvas (http://canvas.instructure.com).

Advancing Awareness

Most current LMS systems provide some level of basic awareness of who is posting the most in discussion forums, or downloading which course resources. The instrumentation of contribution provides one, simple view, but leaves out an important dimension of CANS design: analysis of what we call read data. In our work studying awareness in CANS, we found that a critical aspect of the technical implementation was the gathering and presentation of detailed information about who was reading posts made by others. In fact, we found that higher levels of read behavior in online courses is a significant indicator of the level of knowledge construction present in student posts (Goggins, Galyen, & Laffey, 2010). Future awareness systems should leverage these findings and carefully study the social influence of read awareness, as it is long recognized that behaviors change as people gain the ability to compare themselves with others in a social environment (Festinger, 1954).

Tools to Support Group Work

A significant challenge for technology-mediated learning is supporting awareness of both course level and small group level activity without overwhelming students and teachers with information. We showed that learning groups interact with each other to varying degrees, and that these differences are indicators of group cohesion and learning performance (Goggins, Laffey & Gallagher, 2011). Using the read data advantage that CANS offers over existing LMS analytics provides useful group awareness indicators, such as how many students are reading a member's posts or how many students have read the assignment. When read data are presented in the student interface, students see that their work is being used by others. They are not just completing assignments but also contributing to group knowledge. Awareness that others are using your work enhances the social nature of the learning experience, while also emphasizing the need for quality work because there is an audience for the work other than just the instructor. When the instructor sees read data, she has confirmation that her expectations for social learning are being met (or not). Figure 12 illustrates how groups can be viewed as more or less connected. Four of the six groups in this example from an
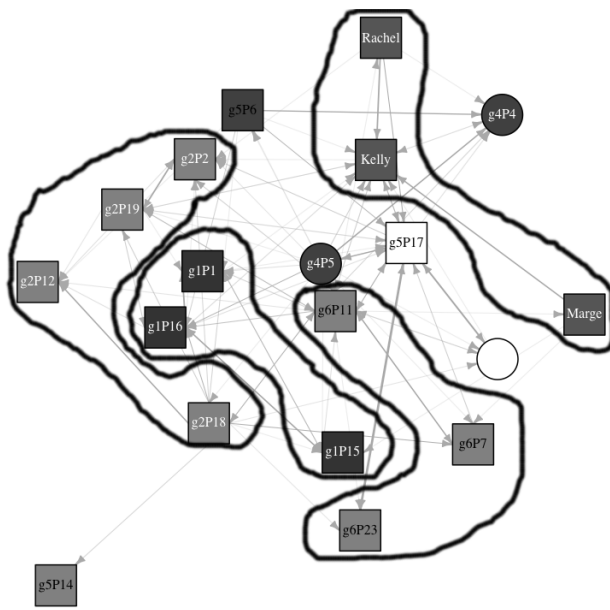
FIGURE 12. Illustration of different levels of group cohesion derived from CANS data.

online course in Sakai are adjacent to each other to varying degrees, and the closeness illustrated in Figure 12 corresponds with interviews with participants, and the different perceptions they had of "group cohesion."

Incorporation of Learning Indicators

Work to date focuses on awareness and discovery of how groups connect with each other in online courses. A next stage in the development of LMS awareness systems will focus to some extent on identifying behavioral indicators of learning. To accomplish this, our work suggests that the content of interaction must be analyzed as well as the CANS record that an interaction occurred.

One such indicator is a shift in the topical focus of discourse. Introne and Goggins (2012) developed a topic modeling algorithm that incorporates network analytic techniques as well as topic models to identify shifts in discourse from one topic to another. This kind of analysis emerges from computational examination of the texts within discussion board messages. Figure 13 is one example of how our topic-modeling algorithm is able to show changes in topic over time in
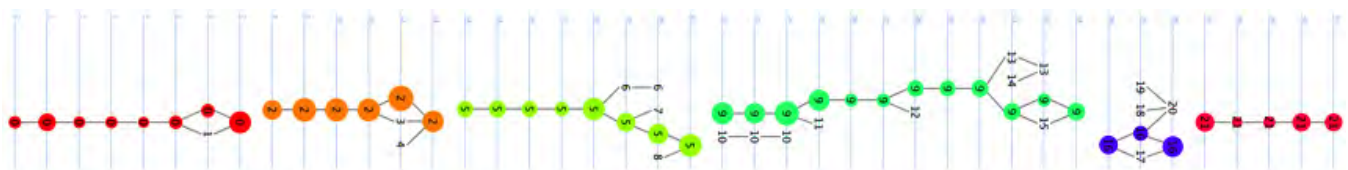
an online news group. The same approach could be applied in a next generation version of CANS.

Importance of Open Source LMS Tools

Sustained innovation and the integration of new analytics features into live course management systems remains a difficult challenge. Like Shadow before it, Sakai is now having a difficult time sustaining itself. Our experience was that Sakai was difficult to deploy and challenging to integrate with, as a complex, Java-based technology. More recently, we have begun to experiment with adding learning analytics to Canvas, an open source LMS. Canvas is based on a popular web framework known as Ruby on Rails, and is hosted on a new type of distributed open source software system called GitHub, which enables anyone to fork, modify, and submit changes to the code (Dabbish, Stuart, Tsay, & Herbsleb, 2012). Creating an easy path for newcomers to contribute code via GitHub give LMS's like Canvas a significant advantage for the introduction of awareness and learning analytics tools like CANS. For these reasons, four major universities recently formed a consortium with Canvas maker Instructure to share in the development of more innovative LMS technology (Instructure, 2014).

CONCLUSIONS

In reflecting back on our work of designing CANS we think examining two tensions and areas of failure help us make sense of our design process and outcomes. The first tension is contrasting design in the academic world (or at least the way we approach design as university researchers) and design in the commercial world. All software product teams confront the same challenges of user models, systems models, human-computer interfaces, process support, effective code, and testing cycles. In the commercial world these tasks are built around product cycles and must yield software products that optimize all aspects of the product model, meet schedules, and map to profit and loss constraints. Typically the team members are experienced and talented professionals with demonstrated track records for their roles. Failure is not tolerated and leads to negative reviews, possible loss of jobs, and potentially finding a new career better suited to the team members' talents.

FIGURE 13. Topic Model of Online Discourse in Math News Groups. This figure illustrates how a single topical thread in a news group focused on geometry evolves over time. News groups are less structured than course discussion boards, so we can see the red topic evolves until it ends. The second (orange), third (light green), fourth (teal) and fifth (purple) topics each show varying degrees of topical evolution. What begins as one topic evolves into two or more related but connected topics.

In the academic world, the product development process morphs into a research and development agenda. While functional design and working code is needed, the products must optimize the constructs of the research agenda in the sense of making them come to life in innovative ways to impact desired outcomes. Schedules revolve around the academic calendar and are typically seen in years rather than in the weeks and months more common to commercial products. All product development is resource constrained but rather than following market research models, academic development is based on perceived significance of the development to the interests of the researchers and strength of the case researchers can make to funding agencies regarding the intellectual merits and potential social impacts of the work. Commercial developers usually work from a budget, which is set in advance and although budgets are subject to business conditions in a company they usually form a basis for a complete development process. In the academic world, projects may bounce between fitting it in within your other demands to periods with external funding.

In contrast to a team of experienced professionals, academic progress is usually made through student work. Students are often gifted and resourceful but quite frequently it is their first job as part of a development team and the first time they are using knowledge and skills outside of classroom assignments. Student work can lead to many surprises, both to the plus and minus side. An example of the plus side is the computer science student we hired as an undergraduate who then enrolled in our masters program and is now a staff member as a lead programmer who mentors the next generation of undergraduate hourly programmers. On the negative side are the numerous students who we hire with great promise and then cannot be found the next semester or the ones who spend all of their time programming and none of the time needed to pass their other subjects, ending up on probation and unable to be rehired.

Failure is the basis for most progress in design and development. We often talk about "failing fast" in order to get beyond weaker ideas and move to more powerful ones. The initial work referenced in this article of developing ShadowNetworkSpace and the Activity Monitor ended with abandoning those products, but with important insights for new product formulations. CANS iterated through multiple versions until we found the most powerful ways to represent activity in context including the "read" activity which is now a strong basis for new models for analytics.

The early versions of notifications in CANS were weak and mapped to only a few use cases, but led to activity awareness representations that were more robust and meaningful to student and teacher work. In some ways, failure is more tolerable in the academic world than in the commercial world because it is seen as part of the scientific process. Even if the software is abandoned before it is broadly used

or commercialized, the scholarly publications allow for sharing advances in constructs and in the development of new knowledge. Embracing open source LMS systems over proprietary systems creates new possibilities for direct implementation and sustainability of future innovations like CANS into a larger community of developers.

These distinctions between academic and commercial development explain some of the choices we made but they do not obscure the fact that CANS is no longer being used. Failing to attend to key markers of commercial development increases the risk of academic software failure. For example, our failure to account or budget for transitions in technology such as the upgrades to Sakai was a critical error. Similarly, while we did needs assessments to understand how our software functionality could be useful, we did not do market research to determine how best to present and package the functionality for adoption. Finally, as researchers we sought ways to optimize the conceptual reification of our framework for activity awareness, but in doing so may have missed opportunities (as will be discussed in the next section) to meet user needs, which potentially may have led to broader adoption.

The second tension that characterizes the design process for CANS is that social learning, while recognized as desirable, can be hard to achieve in online learning and that efforts to try to support social learning can lead to negative outcomes. Education is a social activity and the ability to engage the social nature of learners is important to teacher-student relationships as well as to support learning through dialogue with others and engaging learners in teamwork that amplifies what can be accomplished in a given time period. The tools for being social in an LMS are pretty limited today and were far worse 15 years ago. So much so that teachers and students often prefer avoiding assignments that call for social interaction—except of the most minimal sort—because coordination, cooperation, and collaboration often lead to frustration and disappointment when they are required parts of online learning assignments. Waiting for others to do their work and not knowing when or if they will do that work so you can continue on with your part of the work is the most common criticism we hear from students in our online courses. Faculty wishing to avoid having to intervene or hearing student complaints often choose to use assignment types which avoid social interdependence.

Our research showed us that the activity awareness afforded by CANS was perceived to be most valuable when more social interdependence was a required part of learning tasks. Thus until the tool sets of LMS include mechanisms to support social learning in ways which avoid the pitfalls of frustration and extra work, instructors will be reluctant to include social learning as key components of online learning. But without assignments that require interdependence instructors and students will not see the benefits of social

information nor receive the benefits of social learning. This chicken and egg problem contributed to the demise of CANS. The operation of CANS required a parallel server configuration be set up to Sakai and this was expensive in hardware and time; so without a compelling demand for these services from faculty, few institutions were willing to implement CANS. Perhaps in the future, with lower costs and more common use of virtual servers, a configuration such as CANS would be more viable.

Earlier we discussed how future versions of learning analytics should focus on indicators of learning behavior, not just indicators of behavior. To the extent that CANS could have filtered through activity to identify the behaviors that were most salient for learning, we would have added more value to instruction and made CANS more valuable to instructors. Hypothetically, learning analytics uses data to identify predictors of learning outcomes but we never achieved sufficiently large numbers of users to meaningfully do predictive analytics. In retrospect we could have approximated some of the benefits of predictive analytics by utilizing instructor expectations for what should happen in their courses as a proxy for the analytics. Mapping behavior against expectations and reporting student performance to the instructor could have been a strong way to make the value and potential of CANS more salient for instructors.

CANS has demonstrated approaches to make online learning more social and we see opportunities for next generation systems that merge activity awareness with learning analytics for more powerful online, context-aware teaching and learning.

REFERENCES

Amelung, C. (2005). *A Context-Aware Notification Framework for Developers of Computer Supported Collaborative Environments.* (Unpublished doctoral dissertation). University of Missouri, Columbia, MO.

Amelung, C. (2007). Using social context and e-learner identity as a framework for an e-learning notification system. *International Journal on E-Learning, 6(4), 501-517.*

Anderson, K. M., & Bouvin, N. O. (2000). Supporting project awareness on the WWW with the iScent framework. *ACM SIGGROUP Bulletin*, *21*(3), 16-20. http://dx.doi.org/10.1145/605647.605650

Dabbish, L., Stuart, C., Tsay, J., & Herbsleb, J. (2012). *Social coding in Github: Transparency and collaboration in an open software repository*. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work* (pp. 1277-1286). New York, NY: ACM Press. http://dx.doi.org/10.1145/2145204.2145396

Dourish, P. (2001). *Where the action is: Foundations of embodied interaction*. Cambridge, MA: MIT Press.

Dourish, P., & Bellotti, V. (1992). *Awareness and coordination in shared workspaces*. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work* (pp. 107-114). New York, NY; ACM Press. http://dx.doi.org/10.1145/143457.143468

Festinger, L. (1954). A theory of social comparison processes. *Human Relations*, *7*, 117-140.

Fitzpatrick, G. (1998). *The Locales Framework: Understanding and designing for cooperative work.* (Unpublished doctoral thesis). The University of Queensland, Brisbane St. Lucia.

Goggins, S., Galyen, K., & Laffey, J. (2010). *Network analysis of trace data for the support of group work: Activity patterns in a completely online course*. In *Proceedings of the 16th ACM international conference on Supporting group work* (pp. 107-116). New York, NY: ACM Press. http://dx.doi.org/10.1145/1880071.1880089

Goggins, S. P., Laffey, J., Amelung, C., & Gallagher, M. (2010). *Social intelligence in completely online groups*. In *Proceedings from IEEE International Conference on Social Computing* (pp. 500-507). IEEE. http://dx.doi.org/10.1109/SocialCom.2010.79

Goggins, S. P., Laffey, J., & Gallagher, M. (2011). Completely online group formation and development: Small groups as socio-technical systems. *Information Technology & People*, *24*(2), 104-133. http://dx.doi.org/10.1108/09593841111137322

Goggins, S., Laffey, J., & Galyen, K. (2009). *Social ability in online groups: Representing the quality of interactions in social computing environments*. In *Proceedings from IEEE Conference on Computer Science and Engineering* (pp. 667-674) IEEE. http://dx.doi.org/10.1109/CSE.2009.339

Goggins, S., Laffey, J., & Tsai, I.-C. (2007). *Cooperation and groupness: Community formation in small online collaborative groups*. In *Proceedings of the 2007 international ACM conference on Supporting group work,* (pp. 207-216). New York, NY: ACM Press. http://dx.doi.org/10.1145/1316624.1316654

Goggins, S. P., Mascaro, C., & Valetto, G. (2013). Group informatics: A methodological approach and ontology for sociotechnical group research. *Journal of the American Society for Information Science and Technology*, *64*(3), 516-539. http://dx.doi.org/10.1002/asi.22802

Goggins, S., Valetto, G., Mascaro, C., & Blincoe, K. (2013). Creating a model of the dynamics of socio-technical groups. *User Modeling and User-Adapted Interaction*, *23*(4), 345-379. http://dx.doi.org/10.1007/s11257-012-9122-3

Instructure. (2014). *Unizin, a New Higher Ed Consortium, Chooses Canvas by Instructure as its Foundational Learning Platform*. Retrieved from http://www.instructure.com/press-releases/unizin-chooses-canvas

Introne, J., & Goggins, S. (2012). *Tracing knowledge evolution in online forums*. In *Proceedings from ACM Web Science Conference: Words and Networks Workshop*.

Laffey, J., & Amelung, C. (2007). *Cues and mechanisms for improving the social nature of online learning*. In *Proceedings from World Conference on Educational Multimedia, Hypermedia and Telecommunications* (pp. 1277-1282). Chesapeake, VA: AACE.

Laffey, J., Lin, G. Y., & Lin, Y. (2006). Assessing social ability in online learning environments. *Journal of Interactive Learning Research*, *17*(2), 163-177.

Laffey, J. M., & Musser, D. (2000). *Shadow netWorkspace learning systems project*. In *International Workshop on*

Advanced Learning Technologies (pp. 188-189). IEEE. http://dx.doi.org/10.1109/IWALT.2000.890605

Laffey, J. M., & Musser, D. (2006). Shadow netWorkspace: An open source intranet for learning communities. *Canadian Journal of Learning and Technology/La revue canadienne de l'apprentissage et de la technologie*, *32*(1).

Laffey, J. M., Musser, D., Remidez, H., & Gottdenker, J. (2003). Networked systems for schools that learn. *Communications of the ACM*, *46*(9), 192-200. http://dx.doi.org/10.1145/903893.903937

Paredes, W. C., & Chung, K. S. K. (2012). *Modelling learning & performance: a social networks perspective*. In *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge* (pp. 34-42). New York, NY: ACM Press. http://dx.doi.org/10.1145/2330601.2330617

Reynolds, R., & Goggins, S. P. (2013). *Designing socio-technical systems to support guided "discovery-based" learning in students: The case of the Globaloria game design initiative*. In *Proceedings from International Workshop on Teaching Analytics*, Leuven, Belgium.

Xing, W., & Goggins, S. (in press). A candidate participation based student grade prediction model: Integrating learning analytics, educational data mining and theory through genetic programming. *Computers in Human Behavior*.

Xing, W., Wadholm, B., & Goggins, S. (2014). Learning analytics in CSCL with a focus on assessment: An exploratory study of activity theory-informed cluster analysis. In *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge* (pp. 59-67). New York, NY: ACM Press.