

Visualization of Social Interactions in Facebook

Takashi Okamoto and Kyle Buza

MIT Media Laboratory
Physical Language Workshop
20 Ames Street E15-301
Cambridge, MA 02139
{mud,buza}@media.mit.edu

Abstract

With the popularity of social networks such as Facebook, we have greatly extended our social interactions from the physical into the digital. With social networks we have an increase in interaction opportunities; however, how we experience the interaction is rather bland. Accompanying social interactions with interesting visual feedback is one way to enrich this new space. In our demo, we present E15:FB – a visualization application which shows a graphical representation of social interactions with individual Facebook users. The application also provides alternative methods to navigate Facebook content beyond what is provided through the Facebook website, and creates new opportunities to interact with related content outside of Facebook.

Introduction

Social networks have greatly increased the number of methods we can use to interact with one another. Online interactions have become a commonplace for social interactions, and is an important everyday method for communication which rivals in importance with physical interactions. Our communications have become rich, however the way we experience these interactions is still limited – usually provided through webpages, RSS feeds and email notifications. Creating a graphical representation that visualizes the online interactions can greatly enrich the experience and may even become a useful analysis tool to investigate personal interaction habits and trends (Heer & Boyd 2005).

We have constructed E15:FB, an application visualizing user accessible data and social interactions for Facebook users. E15:FB is a three dimensional environment in which a Facebook user can import and manipulate data provided by Facebook. Users can control the visualization by writing snippets of Python code, evaluated within an embedded Python interpreter. Fig. 1 shows the application window populated with Facebook *friends* and the Python console where users can execute Python code.

Visualization

E15:FB provides the user with visualization flexibility. The visualization is not fixed, instead it is a dynamic system

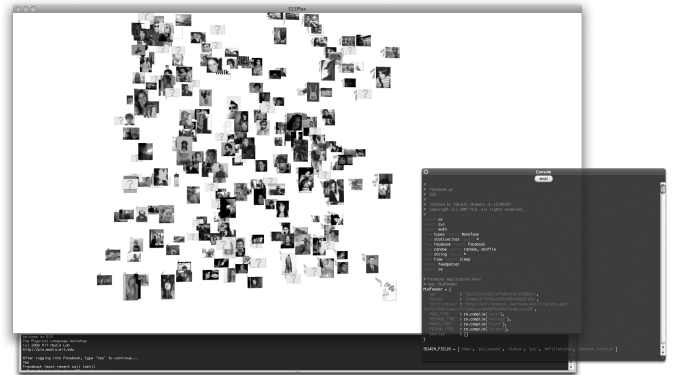


Figure 1: Application window showing the three dimensional environment and Python console

where users dynamically manipulate the data and how it is presented. The system can provide three types of visualization categories:

- 1 *User query-based visualization* Users instruct what data they would like to see and how.
- 2 *Social interaction visualization* The system can run in “watch mode,” where real time events related to the user are animated.
- 3 *Data mashup visualization* Users query additional information from services outside Facebook, to augment the existing information with additional web content such as search results, web pages, etc.

While E15:FB remains flexible enough to support any type of user-specified layout, the default layout we provide has each Facebook user represented as a cube, where the front side is occupied with the profile picture, and the left side displays the name and associated networks. The color of the name represents the cardinality of the set of networks shared with the currently logged in user [Fig. 2]. Videos showing the proceeding scenarios are provided on our server¹.

¹E15 videos: <http://plw.media.mit.edu/people/mud/E15FB/videos/>

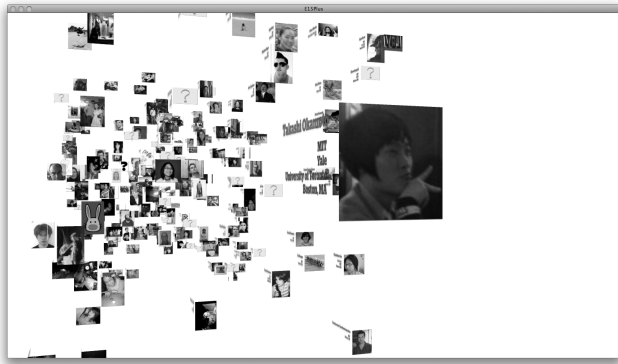


Figure 2: Application window showing current logged in user with friends

User Query

E15:FB begins by authenticating with a Facebook login, then asynchronously populating the three dimensional environment with graphical representations of the user's friends in a random position specified within a cubic region. With this state as the starting point, the user can begin executing commands on the Python console. The user can freely move each friend around in space by issuing a new coordinate, and use two 3D mice to freely fly through the space. Additionally, a command can be executed to animate the camera to put a specific friend into view with proper orientation.

The user can also organize friends spatially. For instance, the user can execute a command that will organize the friends linearly along the z-axis – ordered alphabetically by name. Another method is to cluster friends into different regions by determining how many networks they have in common with the user. This and other arrangements are simple to implement – often in less than five lines of Python code. The user can also display all photographs a specific friend appears in. All photographs can be displayed in the environment at once, which greatly enhances the experience of browsing through photographs compared to viewing them individually through the Facebook site.

Social Interaction

Facebook provides a notification RSS feed for every Facebook user. The notification RSS feed records all events relating to the user, providing a history of transactions within the previous week. The feed is updated as soon as an event occurs, and this provides E15:FB with real time interaction data. By periodically observing the RSS feed, E15:FB can visualize interactions as they occur. This creates a graphical representation of the social interactions that occur within Facebook in real time, and can run on its own without user intervention.

Data Mashup

One of the benefits of using an embedded Python interpreter at the core of E15:FB is the multitude of packages available to extend existing Python functionality. This opens

the door to many new possibilities for exploring new web-centric visualizations that leverage the data obtained from Facebook. For example, we perform proximity-based user name searches to display relevant URLs (and their associated web page) for a given Facebook user. This opens the possibility for forms of *gestural browsing* that do not require keyboard input from the user.

Implementation

E15:FB is a Python script that runs on E15 (Buza *et al.* 2007). E15 is a three-dimensional OpenGL-based web environment, enabling new interactions with web content beyond the traditional browser. E15's main interface is a Python console, and we can use a wide range of Python libraries that are readily available. The communication between Facebook and E15:FB is established using the Facebook REST API (Facebook 2007). The notifications are parsed from the notification RSS feed provided by Facebook for each user. The system currently runs on a 8-core 3.0GHz Mac Pro running Mac OS X 10.5.2, with an ATI X1900 XT graphics card. Users can freely fly through the environment with two 3D mice, where translation and rotation motions are controlled individually by each mouse. Arrow keys are also mapped to move the view from one user to another.

Conclusion

E15:FB attempts to enrich the experience of social interaction methods enabled by popular social networking sites. With data available through web APIs, social networks do not have to be accessed or interacted directly through their websites inside a conventional web browser. Without the need to run inside a web browser, we can use software and hardware facilities available to desktop applications such as hardware accelerated graphics rendering. Currently, we are working to integrate visual effects such as blurring to visually represent intelligent filtering of friends. We are also extending ways to integrate Facebook data inaccessible with the web API, by extracting data available on a web page – such that users can interact with Facebook web pages inside E15 which will directly augment the visual representation in the three dimensional environment. Finally, future development plans include reducing the need to execute Python commands directly by users to minimize stress on novice users and non-programmers.

References

- Buza, K.; Okamoto, T.; Blackaller, L.; and Hollenbach, K. 2007. E15. <http://e15.media.mit.edu>.
- Facebook. 2007. Facebook REST API. <http://wiki.developers.facebook.com/index.php/API>.
- Heer, J., and Boyd, D. 2005. Vizster: Visualizing online social networks. In *Proceedings of IEEE Symposium on Information Visualization (InfoViz05)*.