



Seeing Further: Extending Visualization as a Basis for Usable Security

Jennifer Rode, Carolina Johansson[†], Paul DiGioia, Roberto Silva Filho,
Kari Nies, David H. Nguyen, Jie Ren, Paul Dourish, and David Redmiles

Institute for Software Research
University of California, Irvine
Irvine, CA

[†]Department of Information Technology
Uppsala University
Uppsala, Sweden

{jen, cjohanss, pdigioia, rsilvafi, kari, dhn, jie, jpd, redmiles}@ics.uci.edu



Outline

- Introduction
- Overview of the Impromptu test bed
- User study design
- User study results
- Design implications
- New Features
- Additional User Study



Introduction

- We see two approaches to usability & security:
 - “strict usability” vs “everyday use”
- The critical concern for usable security is that people be able to make informed decisions about their actions.
- Traditional security is often “automatic” and “transparent”.
- We advocate making security more visible, allowing users to understand the consequences of their actions and empowering them to make “effective” security choices.



Design Approach

In support of “effective security” we are exploring three design principles:

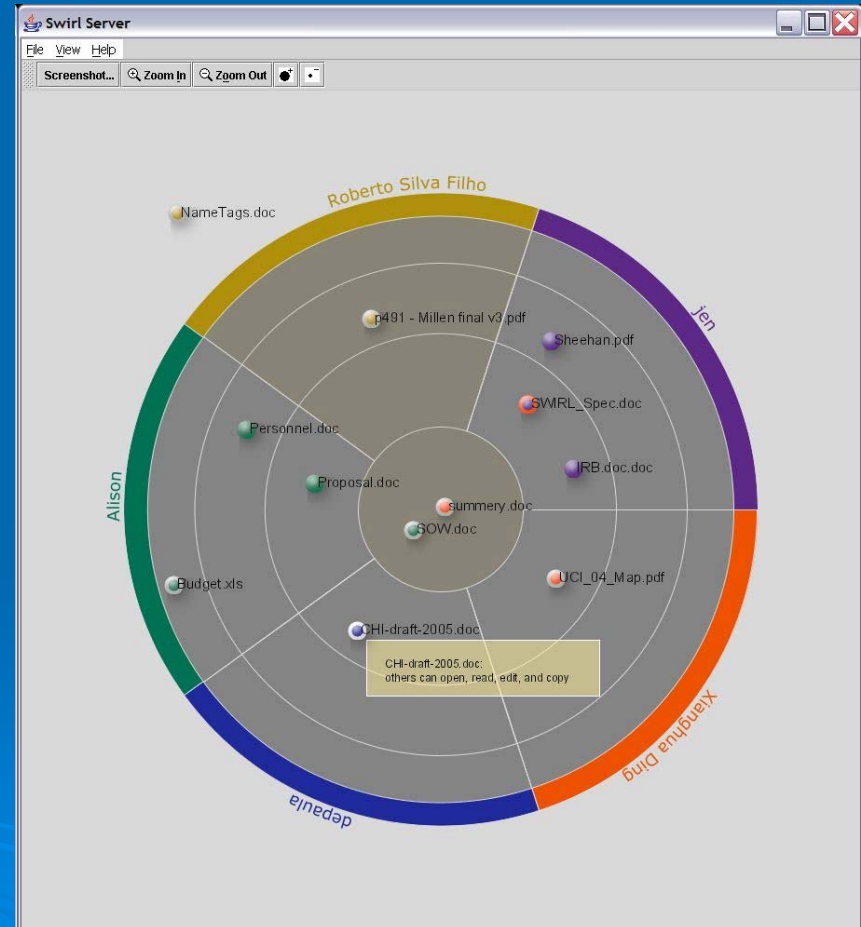
- Dynamic visualization of system activity
- Integration of Configuration and Action
- Event-based architectures



Impromptu Overview

Our testbed is *Impromptu*, an ad-hoc peer-to-peer file sharing application.

- Pie metaphor
- Dots are shared files
- Use of color
- Visualizing user activity
- Sharing levels





User Study

- Wanted to test ‘everyday use’ of our file sharing software
- As this was a prototype we chose to test it in a lab, so we could iterate on the design before investing the effort to make a user installable version



Study Design

- 24 students in 8 small group sessions w/mixture of strong and weak ties.
- In each session 3 participants used Impromptu
- Data:
 - Audio tape of sessions
 - Notetaker- one per users
 - Debrief interview with negative and positive critique of interface



Task Description

- Task: collaborate on a research budget for a grant
 - Create an individual budget & justify expenses
 - Negotiate merging into a group budget
- Budget had a max. Participants received cost estimates.
- Told to imagine it was there one chance to get their advisor to pay for all of the equipment and travel, the everyday financial realities of their research.



Sharing to Accomplish Task

- Asked them to share files to do task, but not required, so they could choose
 - what to share
 - when and under what circumstance
 - level of sharing
- Participants were competing for resources they could create strategies to help maximize the amount of money that would be allocated to them.
- Variety of sharing strategies emerged



Sharing Strategies

- Strategies varied including:
 - free sharing of information from the start (e.g. session 4)
 - hiding personal budget until the last possible minute (e.g. participant A in session 6)
 - sharing despite other's strategies (8b)
 - maliciously editing other budget justifications to help ensure they received more money (7c)
- This meant that privacy in the form of setting access control of one's own files were instrumental to the task.



Findings

- UI and implementation
- Configuration and action
- Dynamic visualization of system activity



Findings: UI & Implementation

- While we had designed a collaboration tool participants viewed Impromptu as a file sharing tool:
 - 9 user complained it didn't update files live
- Suggests that interface succeeded
 - In creating a sense of shared activity
 - That that sharing and interaction was the primary focus— not security



Findings: Configuration & Action

Impromptu allows:

- Context sensitive negotiation of sharing
- Participants to develop explicit strategies of sharing to achieve goals. Recognition of norms relies on configuration being visible to all parties.

Participant 7a: *“Do I have to share?”*

Participant 7c: *“Come on. Put it in the second ring”*

Facilitator: *“Why did you say the second ring?”*

Participant 7c: *“Well, you know. It’s the norm, and you don’t want to share more than necessary, right.”*



Findings: Dynamic Visualization of System Activity

- Gave others a sense of participation:
 - Allowed participant's to know whose files were whose
 - Recognized new files added, changes in permissions, and changes in files
- However, history of interaction provided inadequate as indicated by the rings





Discussion of Study Results

- Integration of configuration and action was successful, as supported by:
 - Subjects ability to master interface and
 - Subjects stated it was easy to use during the debrief, and comments to that effect during tasks
- Concreteness and mutual visibility was successful, as supported by:
 - Emergence of group norms through discussions and uniformity in participant's final permissions
 - Informal conversations about configuration



Design Implications

- 3 findings influence our future work
 1. Understanding of previous activities
 2. Allow participants to assess security risks presented by new users as they join collaboration
 3. Real estate problems
- Remainder of this talk will address 5 features we implemented to address these issues



Design Extensions

- History
 - Rings and ripples
 - History pie
 - Activity wear
- Security risks of unfamiliar users
 - User characterization
 - Media characterization
- Screen Real estate
 - Thin client

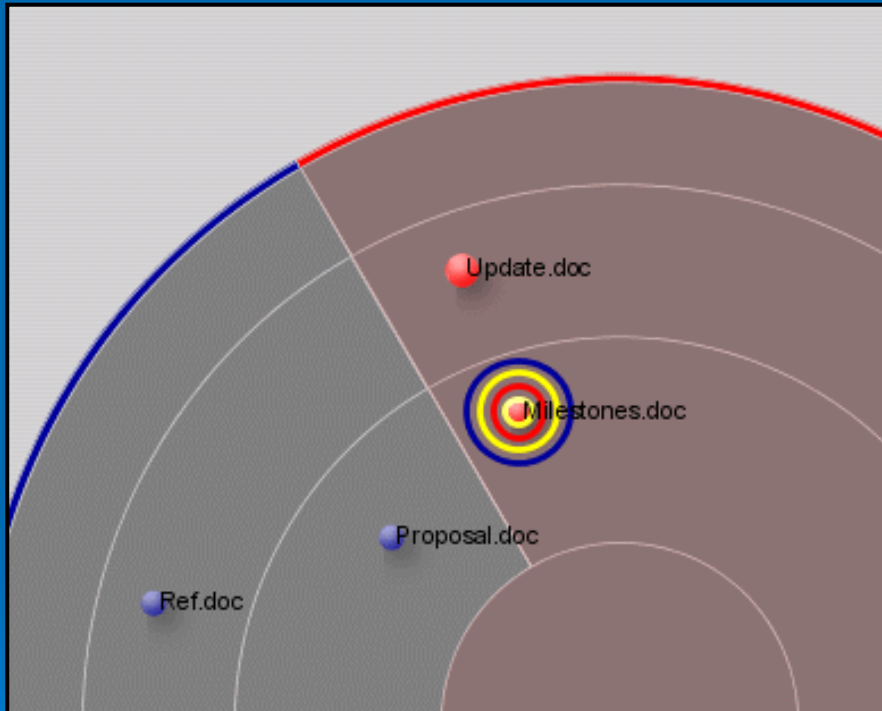


Types of New Visualizations

- History
 - Rings and ripples
 - History pie
 - Activity wear
- Security risks of unfamiliar users
 - User characterization
 - Media characterization
- Screen real estate
 - Thin client



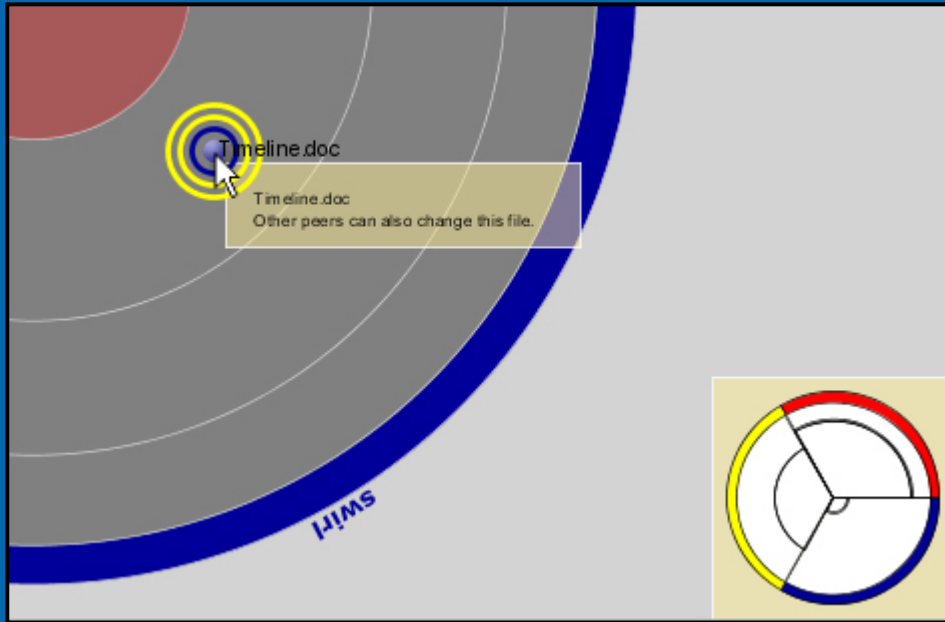
Rings and Ripples



- Problem: Test subjects wanted to see more than simply the most recent action
- Solution: We introduce multiple rings which indicate the 4 most recent activities
 - Rings “ripple outwards,” as ripples in a pond
 - Most recent activity is persistent
 - Older events radiate outwards and eventually disappear



History Pie



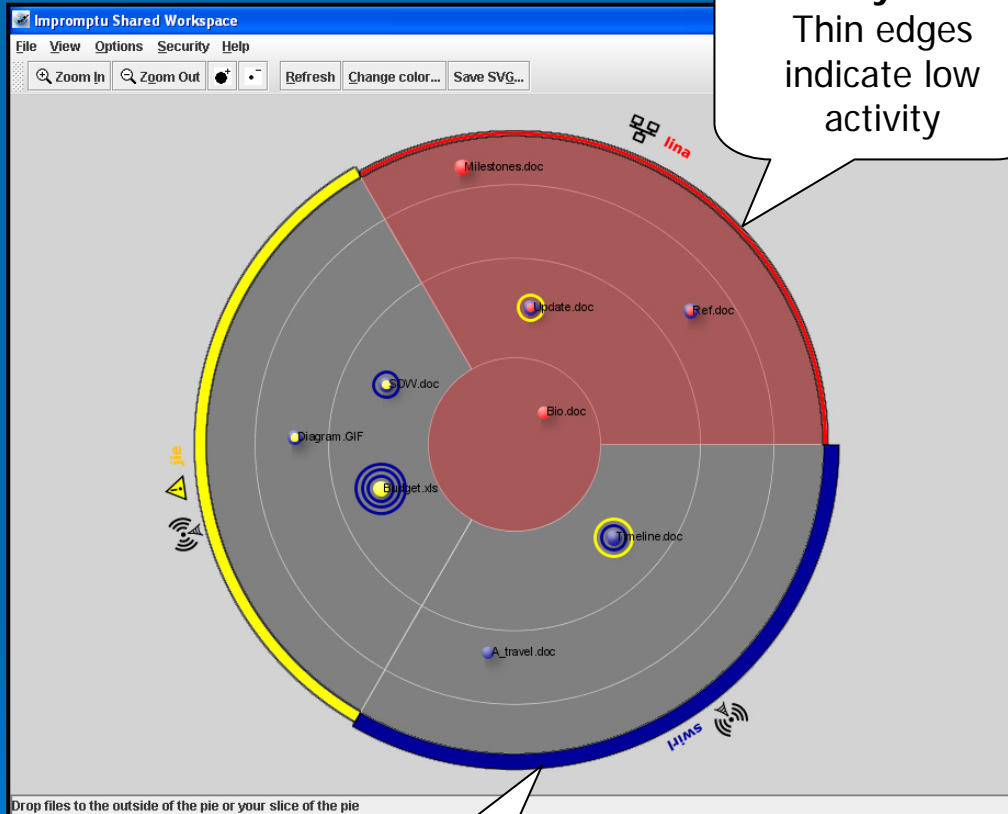
- New Problem: Only the 4 most recent activities are shown
 - Want to show the *entire* history of activity on a file during a user session

- Solution: On a mouseover, provide a complete temporal history for one file

- Layout reflects the spatial arrangement of the “main pie”
- Arcs correspond to a particular user’s activity on that file
- Effect resembles the growth rings on a tree



Activity Wear



Activity wear:
Thin edges
indicate low
activity

Activity wear:
Thick edges
indicate high
activity

- Problem: Need a sense of *user* activity
- Solution: Allow edge thickness to reflect the user's activity level
 - At-a-glance indicator of relative activity
 - We borrow from *Edit Wear* and *Read Wear*, Hill et al.



Types of New Visualizations

- History
 - Rings and ripples
 - History pie
 - Activity wear
- Security risks of unfamiliar users
 - User characterization
 - Media characterization
- Screen real estate
 - Thin client

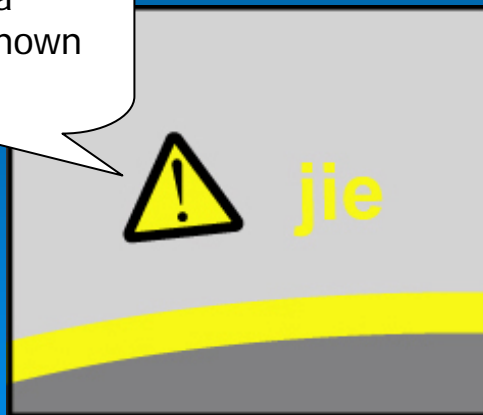


User Characterization

- Problem: Distinguish between familiar and unfamiliar users
 - Convey a sense of prior activity, over multiple sessions

- Solution: Visualize mappings of users to their Ethernet addresses
- Flag unknown or unexpected users with alert icons

Warning symbol
indicating a
previously unknown
user

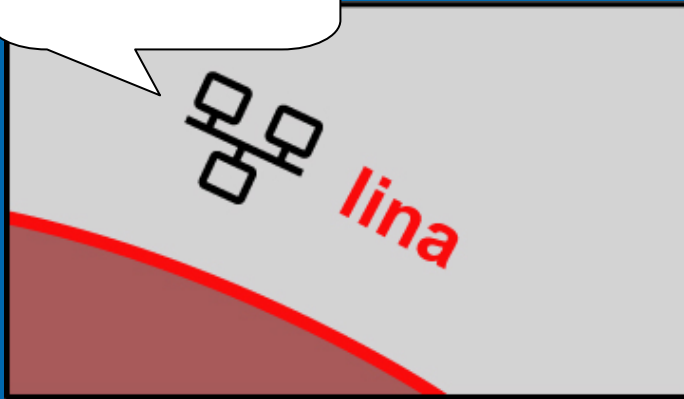


- Unknown user: no established trust
- Familiar username, with a new MAC address: man-in-the-middle attack or masquerading



Media Characterization

Wired network connection icon



Wireless network connection icon



- Problem: Connection details usually made transparent in the interface
 - Different media have different security repercussions
- Solution: Allow connection methods to be apparent in the interface
 - Display wireless & wired Ethernet icons adjacent to usernames



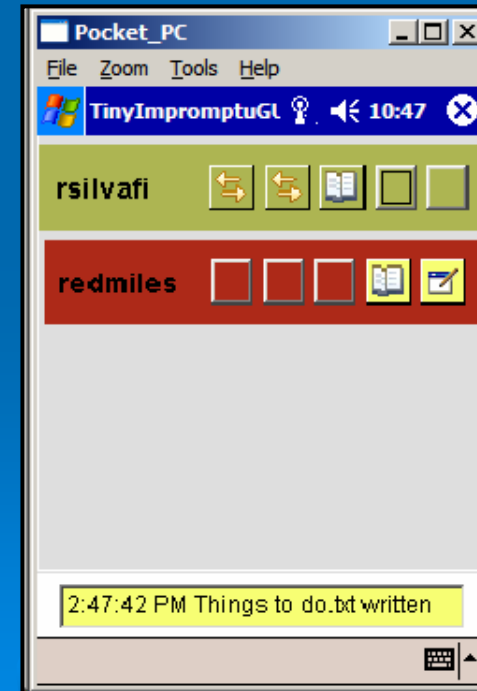
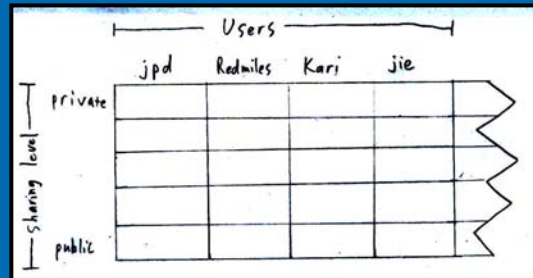
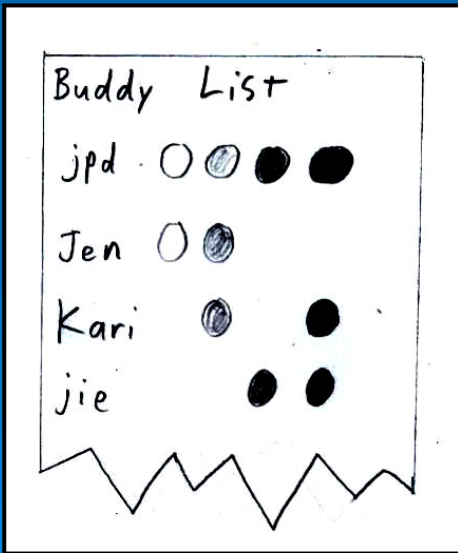
Types of New Visualizations

- History
 - Rings and ripples
 - History pie
 - Activity wear
- Security risks of unfamiliar users
 - User characterization
 - Media characterization
- Screen Real estate
 - Thin client



Thin Client

- Problem: Applications can obscure *Impromptu*
- Iterative design: Performed task analysis, followed by paper mockups
- Solution: PocketPC implementation
 - Peripheral *Impromptu* visualization





Evaluation of History Features*

*Carolina Johansson's Master's Dissertation work

- 6 groups evaluated (total of 12 subjects)
- Results:
 - Ripples and history pie were understood by study participants
 - ripples/rings: from 65% to 88%, $p < 0.05$
 - history pie: from 49% to 82%, $p < 0.05$
 - On a Likert scale, most users agreed/strongly agreed:
 - That they knew when others had interacted with their files
 - That others could see what *they* were doing



Evaluation of History Features*

**Carolina Johansson's Master's Dissertation work*

- However, users wanted more fine-grained activity information
 - Ability to tell exactly where in the file the activity was taking place
- Re-affirms the our success
 - Users were focusing on task instead of security



Conclusions

- Further evaluated our interface from SOUPS '05
- Extended our interface as part of our ongoing iterative design process
- Evaluated our extended visualizations
- Provided evidence for our 'everyday use' approach by establishing the need for:
 - Dynamic visualizations of system activity
 - Combining configuration and action



Acknowledgements:

This work is supported in part by the NSF
and Intel Corporation

Project Website:

<http://www.isr.uci.edu/projects/swirl>