Honors Project CMSC390: The Visual Query Interface; Graphical User Interface Analysis and Redesign

David Cohen

Abstract: Over the last decade, the software market has experienced tremendous growth. As more people are becoming comfortable with and even reliant on using computers in their everyday lives, graphical user interface (GUI) designers are faced with the challenge of providing huge numbers of increasingly complex features bundled in an easy-to-use presentation. In this paper I examine the Visual Query Interface (VQI), a nearly one hundred percent interface driven data mining/modeling application. I construct a study complete with metrics to identify and measure strengths and weaknesses in the GUI from the perspective of usability. I suggest a process through which to improve the software and develop a prototype offering improvements on the interface weaknesses discovered in the study.

1. Introduction

As a product matures, we often learn more about the software and its application than we could have foreseen in our initial planning. In some cases, the internal data structure may have become cumbersome and inefficient, and overhauling it could boost system performance and developer efficiency as well as make the addition of new features easier. Perhaps as the context of the application is better understood, the team realizes that the graphical user interface (GUI) could be redesigned to boost end user efficiency. Or maybe what the software needs to provide an edge over competition is one new amazing feature.

Internal changes, such as streamlining a data structure, are most difficult to justify because, although they may be important from the developer's perspective, there isn't always an obvious and immediate added value from the customer's point of view. More visible modifications, like a redesigned GUI, are easier to defend because a dazzling new interface might be just what the customer wants. More often than not, however, the quickest and easiest way to add value is with the addition of new features.

In our organization we have encountered the problem of architecture degeneration (Lindvall, et al). Through development iterations the same issues continue to arise but are persistently assigned a low priority because of high estimated development costs. Instead, our process has favored quick bug fixes and added functionality – actions that add immediate value to the application; the most "bang for the buck." Over time, the original code base has degenerated (Lindvall, et. al and Van Gurp & Bosch) and the code has become more difficult and more expensive to maintain. It is becoming obvious that these "behind the scenes" problems with the software need to be addressed sooner rather than later. Fortunately with our application, the majority of the issues with our software can be fixed at the same time or with little added cost if the user interface is redesigned.

There's an added problem involved with redesigning a GUI; while internal factors such as data structures can be well designed by a group of experts, GUIs are quite a different matter. To design a good GUI necessitates going to the users themselves and performing some form of usability study. What a developer or even a human-factors expert thinks is a completely natural and intuitive design may, in reality, boggle end users.

In addition, there is a stigma surrounding usability studies; many people feel that only large companies can afford to do studies; that they consume resources, personnel, and take long periods of time to plan and conduct. Furthermore, most studies focus on individual aspects of the GUI, not the application as a whole. In this paper I will concentrate on developing a lightweight model for conducting usability studies for evaluation of GUI-driven applications and apply the technique to our software.

2. Case Study: The Visual Query Interface

The product we are developing is called the Visual Query Interface (VQI). VQI is a data visualization tool that connects to any number of backend databases, combines the information, and presents it on a relational starfield display (Shneiderman, 1999). A screenshot of VQI is displayed in Figure 1.

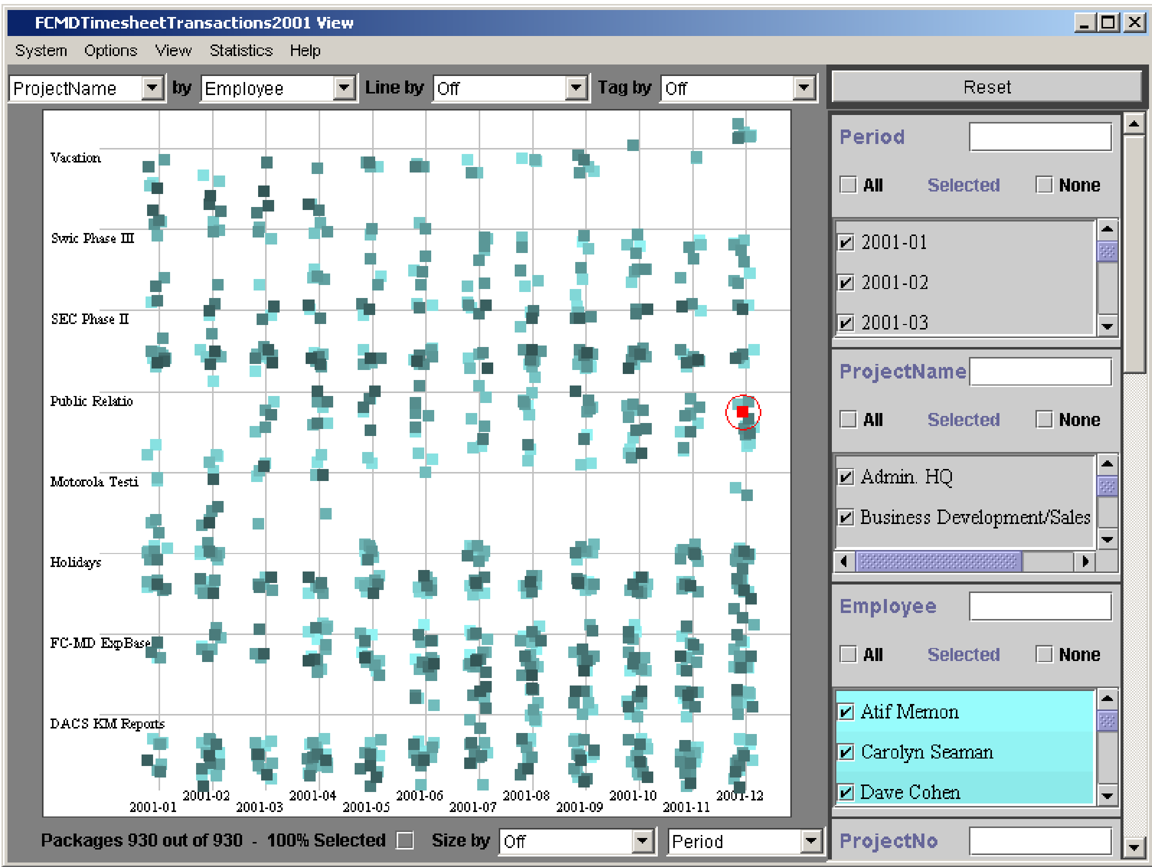


Figure 1: VQI Screenshot

The superset of data is called a package type (in Figure 1, time sheet information for an entire year for every employee in the company), while each collected set of related information is metaphorically referred to as a package (in Figure 1, a single timesheet entry for a single employee). The package type is loaded and sorted based on user-specified values (similar to a dynamic visual SQL query) and packages that meet the specified values are displayed on an X-Y graph. The X and Y values can be chosen from any of the total set of package attributes (in our case: pay period, project name, employee name, hours worked, etc).

The starfield display on the left side of the application is called the "drawing canvas" and all packages meeting display criteria are drawn here. The drawing canvas includes gridlines and grid labels spaced at regular intervals throughout the attribute domain. Around the drawing canvas are options allowing the user to change the X and Y-axes attributes. The user may also associate like values by color and size, connect equal values with a line, and tag packages based on a selected attribute. Users may further select subsets by clicking and dragging the mouse over the set of data they wish to analyze. Packages can be examined in detail by clicking on the desired dot.

Data may be filtered through the use of query devices on the right hand frame of the application. Each package attribute is displayed. In the case of nominal attributes, each value is listed with a checkbox. For continuous attributes such as numbers and dates, attribute values may be selected through the use of a slider bar. Menu-driven functionality allows users to further filter and analyze data.

Despite this heavy reliance on the interface as the primary input device, the VQI is not always simple to use. Early design decisions waste large amounts of display real estate, and make VQI cumbersome to navigate and non-intuitive to operate. From the perspective of the development team, there are four major factors that detract from usability.

A. A many-windowed interface. Every package type is displayed in a new window outside the original frame. Similarly, menu options create modal dialogs without icons in the system taskbar. Navigating between multiple open windows and dialogs can be difficult and makes finding data cumbersome.
B. Display options are spread around the perimeter of the window. While the original placement of these options was intuitive, the addition of more has caused spaces to become cramped and has negatively affected intuitiveness as a whole.
C. Query devices take up a significant portion of the screen and are sprawled so that only a few options are visible at any one time. This makes selection decisions difficult and slow while at the same time wasting valuable display real estate.
D. Commonly used options are multiple clicks away from the main window, making access to these features more difficult than necessary.

It would be easy just to go ahead and make the changes we feel would remedy these problems, but we want to make sure that the issues we perceive as usability problems are the same ones that the users encounter. All too often developers "improve" their software only to find that the changes they made aren't what the user wanted. This study attempt will bring the developer's and user's perspective together.

2.2 Purpose of the Case Study

The VQI development team agrees that the tool is very useful. In fact, we use it more or less daily. And, while users are consistently impressed during software demonstrations, it is often difficult to get them enthusiastic when they try it themselves. Many of them have expressed the opinion that the interface is intimidating and confusing because they are presented with such a large amount of information. The goal of the case study is to develop and conduct GUI usability studies. Based on these results, I will prototype a new version of the software aimed to reduce the barrier of entry for new users.

3. Study Approach

It is not uncommon for studies to focus mainly on data collection such as number of lines of code or execution times for various modules. When the collection is complete however, it is often unclear how to interpret the data. This paper will use the Goal / Question / Metric model [7] for developing data collection metrics (shown in Figure 2). The GQM model provides a top down approach where a large amount of study-time is spent developing processes that will yield relevant and well-tuned data for the experiment as opposed to mass data collection. This method will be beneficial because the data is easily analyzed and applied toward developing a solution

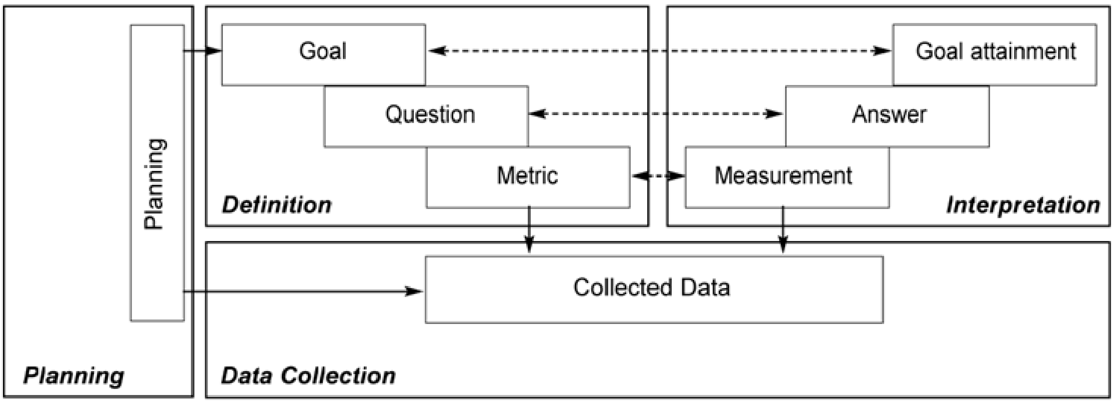


Figure 2: GQM Method from Software Measurement Labs [5]

There are six stages to this project:

A. To establish a reusable Goal / Question / Metric model suitable for evaluating interface-driven applications.
B. Develop a set of hypotheses for improvement of the GUI.
C. Create a set of usability studies and apply the GQM to VQI in order to evaluate the old interface, conduct pilot study, and revise if necessary.
D. Conduct the usability study and validate or adjust hypotheses based on data.
E. Design a prototype and conduct a focus group with a set of developers and study participants. Validate or adjust hypothesis and prototype accordingly.
F. Develop new version of VQI and conduct a second set of usability studies on in order to evaluate and compare it to the old version.

Because of time restrictions, this honors project will complete only the first five stages (A-E) of the outlined project plan.

4. Establishing a GQM

I have attempted to devise a reusable GQM for the evaluation of interface-driven applications that is easily applicable to both VQI and other software applications.

In the GQM, the goal is structured to answer the following questions: what are we measuring, why we're measuring it, what aspect, from who's perspective, and in what context.

> Analyze *the graphical user interface of the application*
>
> for the purpose of *evaluation*
>
> with respect to *usability*
>
> from the point of view of *the end user*
>
> in the context of *typical use of the organization*

Usability is inherently difficult to define – it seems that everyone has his or her own idea of what usability should be (Schneiderman, 1998 et al). For the purpose of this study and in an effort to create a subset of factors that can be tested in a reasonable length of time, I have narrowed the definition of usability to four factors:

A. Intuitiveness
B. Learnability
C. Efficiency
D. Subjective Satisfaction

It is generally agreed that recallability is also an important aspect of a good interface design. Because of the limited timeframe for this study I will be leaving this factor out, but other researchers may find measuring recallability a valuable exercise.

With this definition of usability as a reference point, the set of questions and metrics for the GQM are as follows:

Q1: How intuitive is the interface? (Intuitiveness)
Q2: Is the interface easy to learn? (Learnability)
Q3: How does efficiency increase in relation to experience with the software? (Efficiency)
Q4: What are the user's overall impressions of the software? (Subjective Satisfaction)

The metrics utilize two different types of tasks. Participants will be given *guided tasks* in which they are instructed as to what steps to follow. The metrics for guided tasks will give a measure of intuitiveness and also serve as a quick training exercise to ensure that all participants are familiar with the software. After a number of guided tasks, participants will also be given *unguided tasks* where they are asked to solve a particular problem using any method they prefer. The unguided tasks will give a measure of learnability.

Metrics 1-5 will provide benchmark data by which to compare the current version with the improved version in order to determine whether the new version is improved or not. Metric 6, combined with observational data, provides insight into which features need to be redesigned. The hypothesis will be modified to reflect these results.

M1: Number of unsuccessful attempts to complete a task
    M1.1: Number of unsuccessful attempts to complete a guided task

M1.2: Number of unsuccessful attempts to complete an unguided task
M2: Time in seconds to perform a new task
M2.1: Average time in seconds to perform a guided task
M2.2: Average time in seconds to perform an unguided task
M3: Number of clicks to complete a task.
M3.1: Average number of clicks to complete a guided task
M3.2: Average number of clicks to complete an unguided task
M4: Amount of "wasted time"
M4.1: Percent more time spent on guided tasks than expert user.
M4.2: Percent more time spent on unguided tasks than expert user.
M4.3: Percent more clicks spent on guided tasks than expert user.
M4.4: Percent more clicks spent on unguided tasks than expert user.
M5: Application rating on a scale from 1 to 5; 1 strongly dislike, 2 somewhat dislike, 3 don't care, 4 somewhat like, 5 strongly like.
M5.1: Rating for the software as a whole.
M5.2: Individual feature rating.
M6: Comments and suggestions.

Figure 3 relates the goal, questions, and metrics. It is my hypothesis that a more intuitive interface will allow a new user to successfully complete a task sooner (Q1-M1). An interface that is easier to learn should show an improvement over duration of the study of the new user time as compared to expert time (Q2-M4). An interface that allows a user to work more efficiently will allow users to complete tasks in less time (Q3-M2,M3,M4). And a program that is pleasant to use will, most importantly, keep a user using it and elicit positive feedback (Q4-M5,M6).

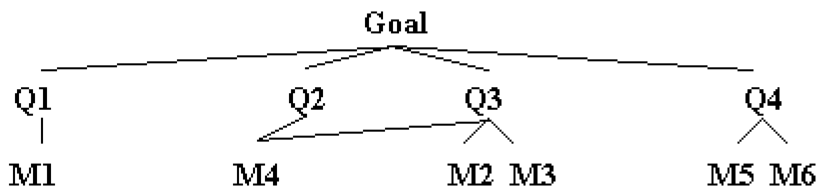


Figure 3: Relation of Goals, Questions, and Metrics

5. Hypothesis

I hypothesize that the following design modifications will improve the usability of the VQI. Table 1 relates these hypotheses to the usability issues outlined in section 2.

H1. Use of a multi-document interface to hold all open package types and dialogs.
H2. Migration of display options to a side menu that will provide easy access and more intuitive descriptions.
H3. Redesign of query devices to be more intuitive, easier to use, and more compact.
H4. Addition of a menu bar for common operations.

Problem / Goals	H1	H2	H3	H4
A. Multi-windowed interface	**X**			
B. Disperse display options		**X**	**X**	
C. Query device sprawl		**X**	**X**	
D. Hard to access options				**X**

Table 1: Preliminary Goals in relation to Statement of the Problem

6. First Usability Study

The purpose of the first usability study is to characterize the usability of the existing VQI system by to collecting measurements to serve as baselines with which to compare the second version. This study will also generate user-feedback to use as a as a source for improvement.

I have chosen to use our company's 2001 timesheet data for the context of the study to ensure that the context domain is familiar and relatively intuitive for all participants.

The adjusted goal:

> Analyze *the current graphical user interface of the VQI system*
>
> for the purpose of *evaluation and improvement*
>
> with respect to *usability*
>
> from the point of view of *project managers*
>
> in the context of *the Timesheet Application at FC-MD*

Developing experiments with which to examine VQI is an interesting exercise, because the software is unique in the fact that it is almost one hundred percent mouse-driven. At the point of writing, I have been unable to find any published studies similar to or techniques appropriate for this type of application.

6.1 Experimental Design

The user group for testing consists of project managers from a similar domain in the subset of domains targeted by the software. The expert user was chosen as the project manager from the VQI project. He has overseen all the development stages of the software, uses it on a regular basis, and is knowledgeable of all the operations and functionality.

The tasks each user is asked to perform is from within his or her domain in order to minimize the amount of time the user spends thinking about how to solve the problem and maximize the amount of time spent dealing with the application itself. Tasks are standardized over the user group so accuracy of results may later be verified.

To take into account the learning effect as the participant conducts the study, similar tasks will be repeated to explicitly test learning. Increase in efficiency between tasks can be plotted as a learning curve.

To minimize learning between the first and second study, some participants will participate in only the first study, some will participate in only the second, and some will participate in both.

The study design is as follows:

1. Identify the set of GUI features to be tested. Decision is based on usage percentages; due to time constraints, we examine the most commonly used features.
2. Develop a two-part questionnaire to target these features. Because users may have varying levels of familiarity with the software, the first section is guided and will give a good impression of how intuitive these features are. The second section will be unguided, or open-ended questions which will examine the learnability of the software. For examples of guided and unguided tasks, see the questionnaire in 6.2, or Appendix B.
3. Run a pilot study on 3 participants from the user group. Examine difficulties and successes and revise the questionnaire and/or process.
4. Run the study on the expert user to collect base measurements.
5. Run the study on 5 more participants selected from the target user group. The 3 pilot users have not been included in selection.
6. Analyze the study results and construct a mockup of the new and, ideally, improved interface.

Because of time restraints, this paper will conclude with step 6 of the steps outlined above. Were time not an issue, the study would continue:

7. Conduct a focus group on the new interface consisting of VQI project personnel and 3 participants from the study. Informally validate or invalidate the GUI mockup. Make necessary changes and implement the new interface.
8. Using the same questionnaire, run the study on the new interface: First on the expert and then on 5 more participants from the target user group, excluding those who participated in the focus group.
9. Analyze the results and compare the collected metrics to those from the first study to determine success.

6.2 Pilot Study results

I conducted a pilot study on 3 participants from the target user group.

Participants encountered a few minor difficulties:

- All participants had difficulty with the wording of a couple of questions. The questions were rewritten or clarified.
- One participant's transactions, being a part time employee, were all for a small number of hours, which made them hard to see when size by hour was selected.
- One participant was relatively unfamiliar with the interface and initially had difficulty figuring out how access certain features. Once hinted, "there are other features available to examine data" the participant immediately focused on the menu bar and found the features.
- When one participant was asked to select a subset of transactions over 100 hours, there were no transactions left making the next couple questions moot. This question

was changed to reflect a lower range of numbers to ensure that all participants would have relevant data.

Initial feedback from the study was positive, participants thought the questions were “fun” and enjoyed the problem solving aspect of the questions.

One of the things I realized when attempting to collect the metrics for this participant was that it was going to be impossible to time and keep click counts for the tasks. I developed a small frame that loads to the right of the main VQI window that displays the tasks and keeps time and click counts. It then writes the results over the network to my hard drive.

Aside from these small changes;

1. Minor question clarifications
2. Addition of the time and click monitoring frame

The structure and process of the study remained unchanged from the pilot.

6.3 The Questionnaire

The final questionnaire is as follows. Participants were presented data with timesheet information and asked to complete each task. Tasks 1-17 are guided. Tasks 18-25 are unguided. In the event that a user couldn’t complete a task or became overly frustrated, they were gently “coached” in the right direction. Such an approach was necessary because later tasks build off techniques developed in earlier ones.

```
About Yourself
1.     How would you rate yourself as a computer user on
       a scale of 1 to 5 (1 novice, 2 some experience, 3
       average, 4 more experienced than most, 5 expert)?
2.     On the same scale, how would you rate yourself as
       a VQI user?
3.     How many hours a week do you use VQI?

FCMD Timesheets
For the next set of questions you will be using real
data from last year’s timesheets. Each data point is a
transaction, which is one charge to a single project for
a single month.

1.     Color by project name, put employee on the Y axis,
       period on the X, and size by hours.
2.     Color by project name, put employee on the Y axis,
       period on the X, and size by hours.
3.     Show only your transactions using zoom or
       attribute selection
4.     Put hours on Y.
5.     Find by ocular inspection the transaction where
       you spent the most amount of time?  Bring up the
       package viewer. Which project is it and what was
       the cost of the transaction
```

6. Using the summary, summarize by employee. On which project did you spend the most hours over the year? This project will be referred to later as your primary project.
7. Hit reset, and select only the transactions that apply to your primary project by zooming or attribute selection.
8. Put employee on the X axis, hours on Y, and size by cost.
9. Find by right clicking the employee with the costliest charge to the project
10. Select all transactions between 50 and 100 hours for the project using the slidebar.
11. Color and Tag by Employee.
12. Using statistics, who has the most 50-100 hour charges?
13. Turn tags off and hit Reset.
14. Put hours on the Y axis, period on the X, and color by project name.
15. Choose only dots that are related to you.
16. Choose dots related to your primary project and up to 2 other projects you've worked on.
17. Set jitter to 0 and line by project name.
18. Can you see any trends in your work over the year?
19. Hit Reset.
20. Which project was costliest over the year?
21. Which period had the greatest number of charges?\
22. In which period did you work the most total hours?
23. How many hours did you work in October, November, and December combined?
24. How much money did you charge over the year?
25. Who charged the most total hours to your primary project?

<u>About VQI</u>

1. What did you like about the interface?
2. What didn't you like? Did you find anything especially difficult to use?
3. What would you change, remove, or add if you could?
4. How would you rate the interface as a whole, on a scale of 1 to 5 (1 strongly dislike, 2 somewhat dislike, 3 don't care, 4 somewhat like, 5 strongly like)?

Thank you for your participation!

6.4 The Study

The user group chosen for participation in the usability studies consists of project managers who are all computer literate and are familiar with the concept of VQI. They have different levels of familiarity with the software. Some use the software on a regular basis while others haven't used it at all. In order to gauge results based on experience with the software, users are asked to rate themselves in expertise both on computers in general and on VQI. All participated voluntarily.

6.4.1 Data Collected

A complete collection of the raw data collected can be found in Appendix A. A visual bar-chart representation of the collected results can be found in Figure 4.

Results after applying the GQM:

```
Expert:
--------------
M2.1 Average guided time:            9.76 seconds/task
M2.2 Average unguided time:          14.66 seconds/task
M3.1 Average guided clicks:          3.39 clicks/task
M3.2 Average unguided clicks:        5.00 clicks/task
                                     -------------------
Totals:                              4.40 minutes total
                                     91 clicks total

Participant 1:
--------------
M1.1 Unsuccessful guided:            0
M1.2 Unsuccessful unguided:          1
M2.1 Average guided time:            18.24 seconds/task
M2.2 Average unguided time:          26.16 seconds/task
M3.1 Average guided clicks:          3.83 clicks/task
M3.2 Average unguided clicks:        5.17 clicks/task
M4.1 Percent more guided time:       46.46%
M4.2 Percent more unguided time:     43.96%
M4.3 Percent more guided clicks:     11.59%
M4.4 Percent more unguided clicks:   3.23%
M5.1 Rating (1-5):                   4
                                     -------------------
Totals:                              8.09 minutes total
                                     100 clicks total

Participant 2:
--------------
M1.1 Unsuccessful guided:            3
M1.2 Unsuccessful unguided:          2
M2.1 Average guided time:            16.31 seconds/task
M2.2 Average unguided time:          37.78 seconds/task
M3.1 Average guided clicks:          3.72 clicks/task
M3.2 Average unguided clicks:        5.50 clicks/task
M4.1 Percent more guided time:       40.15%
M4.2 Percent more unguided time:     61.19%
M4.3 Percent more guided clicks:     8.96%
M4.4 Percent more unguided clicks:   9.09%
M5.1 Rating (1-5):                   3
                                     -------------------
```

```
Totals:                             8.67 minutes total
                                    100 clicks total

Participant 3:
--------------
M1.1 Unsuccessful guided:           6
M1.2 Unsuccessful unguided:         2
M2.1 Average guided time:           60.99 seconds/task
M2.2 Average unguided time:         157.10 seconds/task
M3.1 Average guided clicks:         6.94 clicks/task
M3.2 Average unguided clicks:       18.00 clicks/task
M4.1 Percent more guided time:      83.99%
M4.2 Percent more unguided time:    90.67%
M4.3 Percent more guided clicks:    51.20%
M4.4 Percent more unguided clicks:  72.22%
M5.1 Rating (1-5):                  5
                                    -------------------
Totals:                             34.01 minutes total
                                    233 clicks total

Participant 4:
--------------
M1.1 Unsuccessful guided:           1
M1.2 Unsuccessful unguided:         4
M2.1 Average guided time:           23.98 seconds/task
M2.2 Average unguided time:         88.53 seconds/task
M3.1 Average guided clicks:         3.72 clicks/task
M3.2 Average unguided clicks:       16.00 clicks/task
M4.1 Percent more guided time:      59.28%
M4.2 Percent more unguided time:    83.44%
M4.3 Percent more guided clicks:    8.96%
M4.4 Percent more unguided clicks:  68.75%
M5.1 Rating (1-5):                  3
                                    -------------------
Totals:                             16.05 minutes total
                                    163 clicks total

Participant 5:
--------------
M1.1 Unsuccessful guided:           0
M1.2 Unsuccessful unguided:         1
M2.1 Average guided time:           40.21 seconds/task
M2.2 Average unguided time:         84.52 seconds/task
M3.1 Average guided clicks:         6.17 clicks/task
M3.2 Average unguided clicks:       10.33 clicks/task
M4.1 Percent more guided time:      75.72%
M4.2 Percent more unguided time:    82.65%
M4.3 Percent more guided clicks:    45.05%
M4.4 Percent more unguided clicks:  51.61%
M5.1 Rating (1-5):                  5
                                    -------------------
Totals:                             20.52 minutes total
                                    173 clicks total
```

Study Results

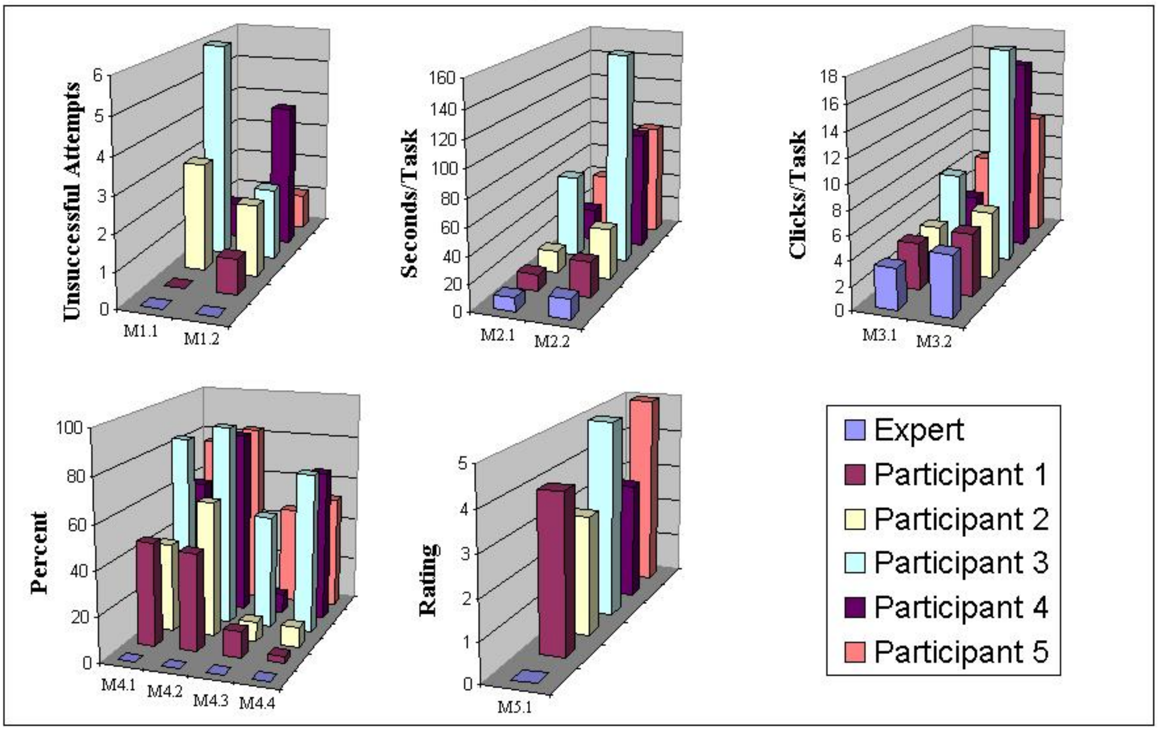
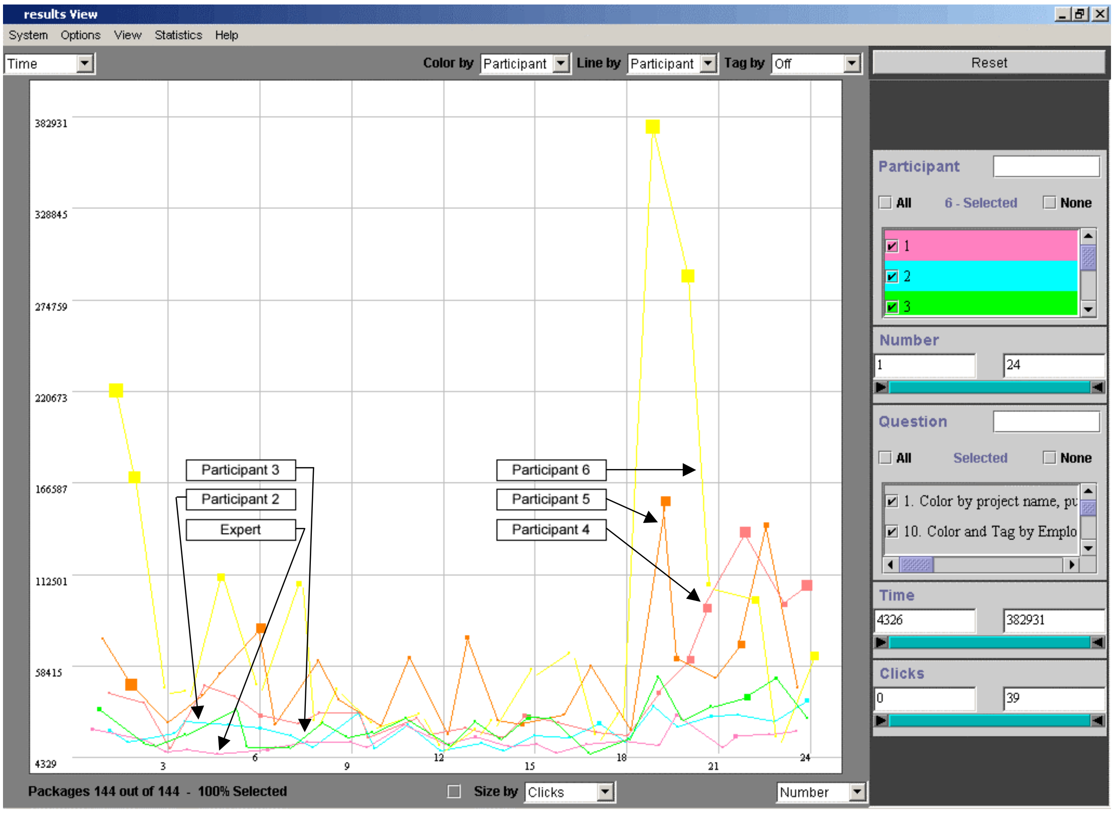


Figure 4: Study Results

6.4.2 Data Analysis

The study data lends itself well to analysis in VQI:



Question number is on the X-axis and time on the Y-axis. Lines are colored by participant and dots are sized by number of clicks. Participant 1 (pink) is the expert control user and has the lowest, flattest line as expected. Users 2 and 3 (aqua and green respectively) have results similar to those of the expert user and rated themselves with a high degree of familiarity with the tool, as the data suggests. Results from participants 4, 5, and 6 (red, orange, and yellow) exhibit more variation and suggests a lower familiarity with the software. In general, response times start out high at the beginning of each section (question #1 starts the guided tasks, question #18 starts the unguided tasks) but drop fairly quickly to a level closer to that of the expert user, suggesting a high degree of learnability.

6.4.3. Observational Data

From observing the participants and from the open-ended questions on the survey, I have compiled a list of common problems and suggestions:

Participants liked:

- The data visualization concept
- Size by, line by, and tag by features
- Summary and statistics concept

Participants disliked or had difficulty with:

- Summary and statistics interface
- Color by gradient (differentiating between shades)
- Default order of attributes in QD (non-intuitive)
- Visibility/Ambiguity of tags for display options (non-intuitive)
- Lots of scrolling for attribute selection and color-by box (not efficient)
- All and None checkboxes (non-intuitive)
- Navigating between the main windows and modal dialogs

In section 8, I will discuss how the proposed design changes affect these issues.

7. Revising the Hypothesis

All of the elements of my hypothesis were supported with the study and the user's dislikes raised a couple new issues I hadn't considered. I have revised my hypothesis to include the following:

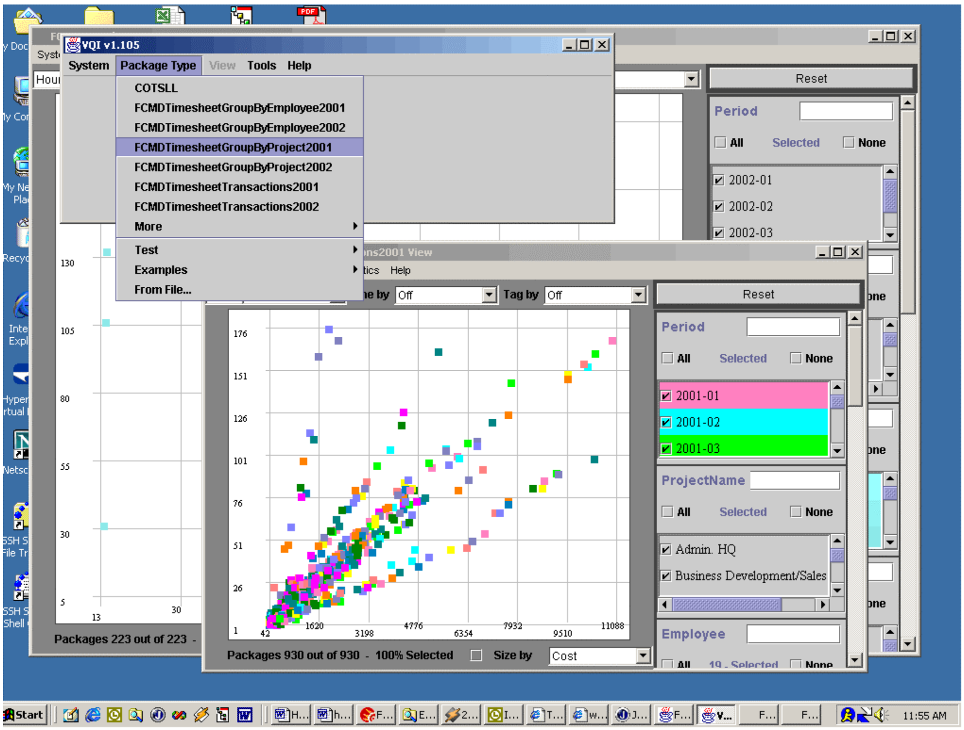
H5: Addition a legend to the graph in order to 1) help clarify color ambiguity and 2) avoid scrolling through the color-by attribute to find color bindings.

H6: Addition "All" and "None" as specific attribute options to increase visibility (as opposed to checkboxes integrated into the query device itself).

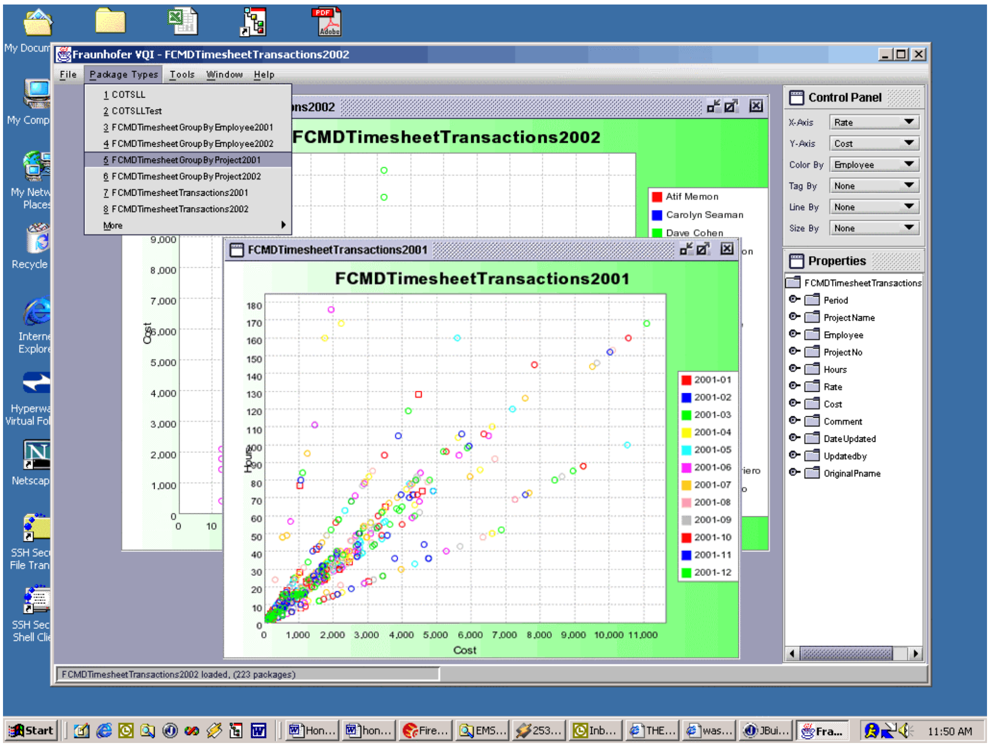
8. Redesigning the Interface

This section discusses the redesigned aspects of the interface and how it compares with the original version of VQI. The screenshots in this section present the comparison between the current version of the software and a semi-functional, redesigned prototype.

8.1. The Multi-Document Interface

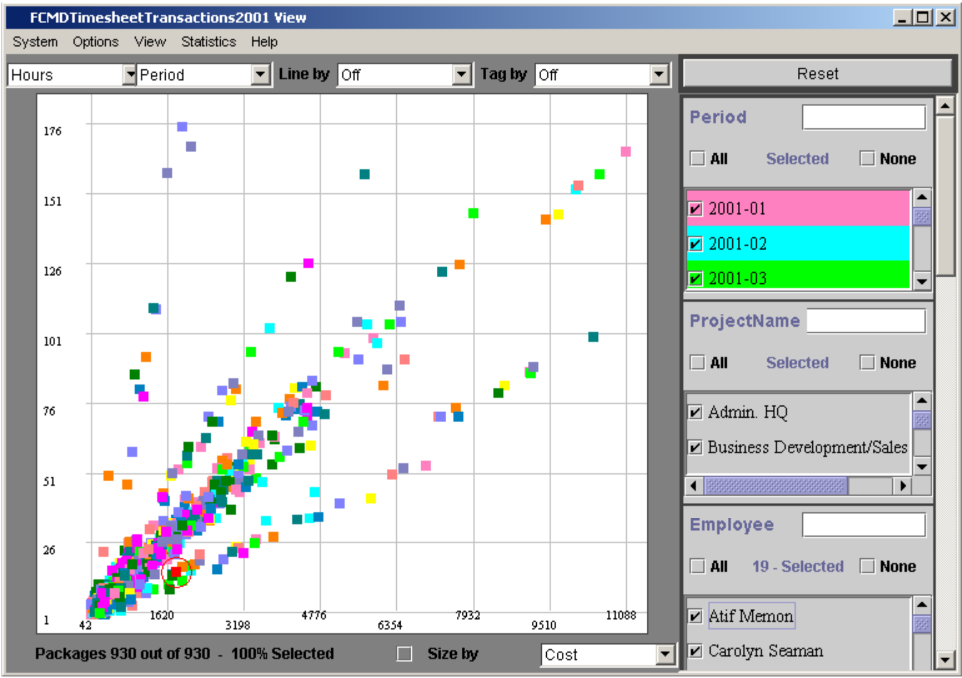


The old interface spawned new windows for each package type and dialog box (such as display options, package type statistics and analysis features, etc). Users complained that they often lost track of which package type they were viewing, and which windows they had open. Windows were easily lost behind other open applications and were mainly accessed through an easily cluttered taskbar. Some modal dialogs didn't produce taskbar buttons and could only be accessed by using the Windows shortcut ALT-TAB.

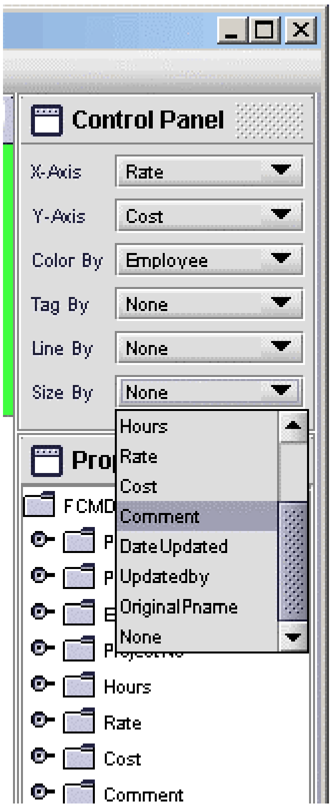


The new interface collects all open windows and dialogs in one common frame. The structure is similar to the one used in many common word processors, and should make navigation more intuitive for the user, a common observed difficulty. Windows can be minimized and maximized within the application frame, and can be intuitively accessed through mouse selection or the Window menu. This design addresses hypothesis H1.

8.2. Display Options and Query Devices

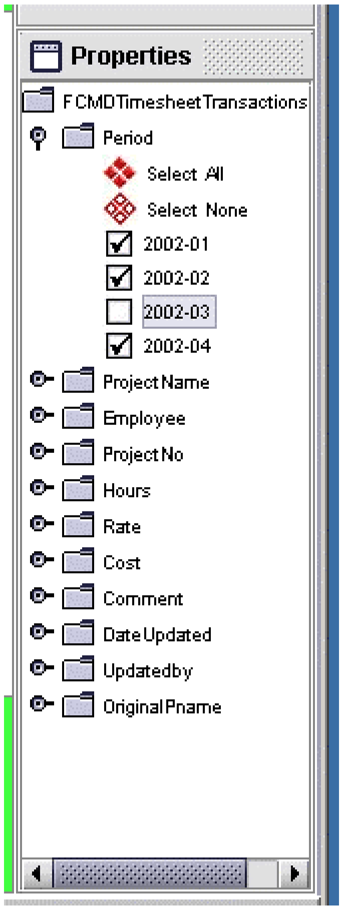


In the previous version, a common source of confusion stemmed from the fact that display options such as X and Y attributes were scattered around the drawing canvas. Resizing the window would hide or obfuscate labels. The query device to the right was bloated and cumbersome to use, only allowing the user to view two to three options in each window. Because of the limited visibility, scrolling in each option box was tricky because it would often scroll too quickly or too slowly making it hard to “browse” the values.



In the new version, the display options are collected into the Control Panel. The Control Panel allows users to view all display options setting at the same time in the same place. Selecting an option creates a drop down list from which the user can select the different values.

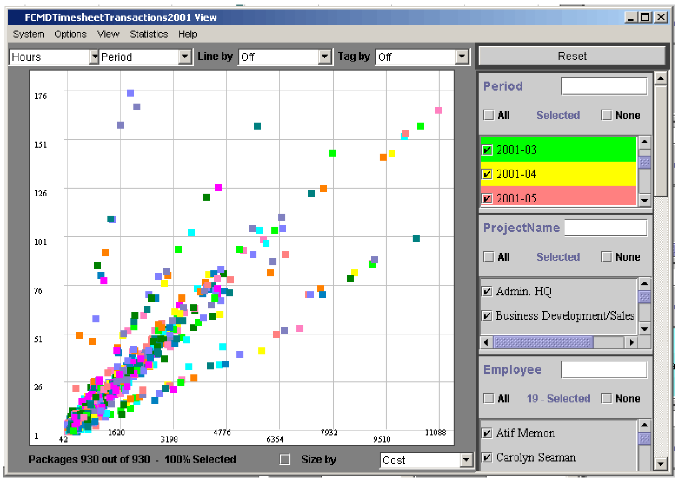
The query device has been modified to utilize a tree structure, keeping the data more compact. Attributes are collected within “folders” giving it a feel similar to that of Windows Explorer. The options “Select All” and “Select None” have been added as explicit entries, rather than peripheral options that were often missed by users in the older version. This addresses hypothesis H2, H3, and H6.

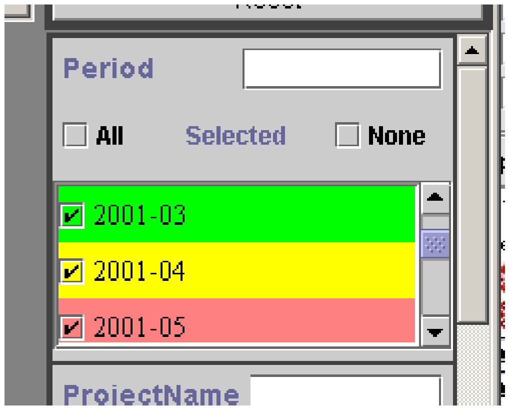


8.3 Menu Bar

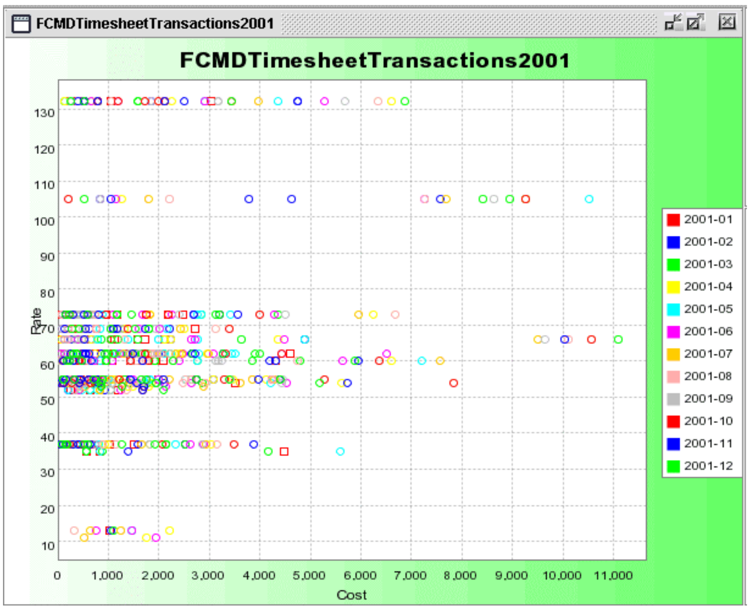
A menu bar was added to provide shortcuts to some of the most common menu-accessible features, such as display options, summary, and statistics. This doesn’t address a specific difficulty encountered by participants, but should lend to a more standardized interface look-and-feel as well as increase user efficiency for commonly performed operations. This addition addresses hypothesis H4.

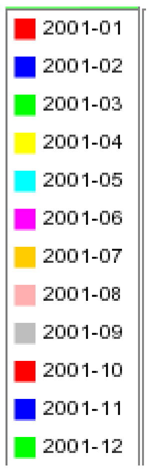
8.4 The Legend





In the old version, the color legend was a common source of complaints. It was often difficult and time consuming to figure out which value corresponded with which color because in order to do so the user had to scroll through the attribute's query device window.





In the new version, a legend has been placed directly on the graph so color associations may be made very quickly. This aspect of the redesign was not in my original hypothesis but was added based on user feedback during the study. It addresses the revised hypothesis H5.

10. Measurement of Success

For the purpose of the honors project, the success of this study is drawn from a number of factors:

1. The development and real-world application of a GUI-centric GQM
2. The successful collection of meaningful metrics
3. The application of collected results to generate an improved GUI prototype.

If I had more time to conduct the study and complete the final version, or for practitioners interested in applying these techniques in the field, success (and meaningful contributions to software engineering literature) can be drawn from any of the following results:

1. An improved interface which generates statistically significant metric improvements

   or

2. Improved subjective user satisfaction, even without significantly improved metrics

   or

3. Lessons learned about specific users groups and/or GUI features (i.e., users from group X found feature Y significantly more difficult to use than feature Z).

11. Works Cited

[1] Lindvall, M., Tesoriero, R., and Costa, P. "Avoiding Architectural Degeneration: An Evaluation Process for Software Architecture", Los Alamitos, CA, IEEE, *In Proceedings of* International Symposium on Software Metrics., pp. 77-86,2002. http://www.cebase.org/0x81020c4a_0x0007a405

[2] Mayhew, D. 1999. *The Usability Engineering Lifecycle*. San Francisco: Morgan Kaufmann.

[3] Shneiderman, B. 1998. *Designing the User Interface*. Reading: Addison-Wesley.

[4] Shneiderman, B. 1999. Dynamic queries, starfield displays, and the path to Spotfire. http://www.cs.umd.edu/hcil/spotfire/ .

[5] GQM Method Application. http://ivs.cs.uni-magdeburg.de/sw-eng/us/index.shtml. Software Application Labs. May 16 2002.

[6] Van Gurp, Jilles, Bosch, Jan. Design erosion: problems and causes, Journal of Systems and Software, Volume 61, Issue 2, 15 March 2002, Pages 105-119. http://www.sciencedirect.com/science/article/B6V0N-44X09DV-1/1/7abede745f1dc5c7ba971124e0a1ad8d

[7] Van Solingen, R., Berghout, E. 1999. *The Goal/Question/Metric Method*. Cambridge: McGraw-Hill.

[8] Zhang, Z., Basili V., and Shneiderman, B. 1999. Perspective-based usability inspection: An empirical validation of efficacy. *Empirical Software Engineering*, 4. 43-69.

Appendix A: Raw Data Collected from First Usability Studies

Participant	Number	Question	Time (ms)	Clicks
1	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	16133	6
1	2	2. Show only your transactions.	13189	5
1	3	3. Put hours on Y.	4326	2
1	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	6890	2
1	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	10716	5
1	6	6. Hit reset, and select only the transactions that apply to your primary project.	12217	6
1	7	7. Put hours on the Y axis, employee on X, and size by hours.	7651	2
1	8	8. Find by right clicking the employee who charged the most hours to the project.	9233	3
1	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	8883	1
1	10	10. Color and Tag by Employee.	6970	4
1	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	22723	4
1	12	12. Turn tags off and hit Reset.	7050	3
1	13	13. Put hours on the Y axis, period on the X, and color by project name.	10275	6
1	14	14. Choose only dots that are related to you.	7150	4
1	15	15. Choose dots related to your primary project and up to 2 other projects youve worked on.	10355	3
1	16	16. Set jitter to min and line by project name.	10765	2
1	17	17. Can you see any trends in your work over the year?	4737	0
1	18	18. Turn lines off, and hit Reset. Feel free to solve the next set of questions in any way youd like.	6480	3
1	19	19. Which project was costliest over the year?	8522	4
1	20	20. Who had the costliest charge to that project?	23253	7
1	21	20. Which period had the greatest number of charges?	9354	4
1	22	21. In which period did you work the most total hours?	19277	8
1	23	22. How many hours did you work in October, November, and December combined?	11717	3
1	24	23. How much money did you charge over the year?	15843	4
2	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	23905	6
2	2	2. Show only your transactions.	14220	5
2	3	3. Put hours on Y.	12067	2
2	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	26498	2
2	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	27310	5
2	6	6. Hit reset, and select only the transactions that apply to your primary project.	24565	6
2	7	7. Put hours on the Y axis, employee on X, and size by hours.	18937	2

2	8	8. Find by right clicking the employee who charged the most hours to the project.	11256	2
2	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	26589	4
2	10	10. Color and Tag by Employee.	13840	3
2	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	21200	3
2	12	12. Turn tags off and hit Reset.	11577	3
2	13	13. Put hours on the Y axis, period on the X, and color by project name.	15612	3
2	14	14. Choose only dots that are related to you.	11186	4
2	15	15. Choose dots related to your primary project and up to 2 other projects youve worked on.	19488	5
2	16	16. Set jitter to min and line by project name.	14101	3
2	17	17. Can you see any trends in your work over the year?	28500	8
2	18	18. Turn lines off, and hit Reset.	7421	3
2	19	19. Which project was costliest over the year?	34360	5
2	20	20. Which period had the greatest number of charges?	15892	2
2	21	21. In which period did you work the most total hours?	26769	7
2	22	22. How many hours did you work in October, November, and December combined?	27920	3
2	23	23. How much money did you charge over the year?	19328	5
2	24	24. Who charged the most total hours to your primary project?	32707	9
3	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	28832	8
3	2	2. Show only your transactions.	12047	4
3	3	3. Put hours on Y.	4997	2
3	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	19418	2
3	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	24846	5
3	6	6. Hit reset, and select only the transactions that apply to your primary project.	14350	5
3	7	7. Put hours on the Y axis, employee on X, and size by hours.	8573	2
3	8	8. Find by right clicking the employee who charged the most hours to the project.	22362	2
3	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	18476	1
3	10	10. Color and Tag by Employee.	13259	4
3	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	23855	4
3	12	12. Turn tags off and hit Reset.	6559	4
3	13	13. Put hours on the Y axis, period on the X, and color by project name.	18657	4
3	14	14. Choose only dots that are related to you.	12208	4
3	15	15. Choose dots related to your primary project and up to 2 other projects youve worked on.	31525	11
3	16	16. Set jitter to min and line by project name.	20369	2
3	17	17. Can you see any trends in your work over the year?	4947	0
3	18	18. Turn lines off, and hit Reset. Feel free to solve the next set of questions in any way youd like.	8342	3

3	19	19. Which project was costliest over the year?	48009	4
3	20	20. Who had the costliest charge to that project?	30885	5
3	21	20. Which period had the greatest number of charges?	29793	3
3	22	21. In which period did you work the most total hours?	44524	12
3	23	22. How many hours did you work in October, November, and December combined?	51854	6
3	24	23. How much money did you charge over the year?	21621	3
4	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	225424	31
4	2	2. Show only your transactions.	172057	25
4	3	3. Put hours on Y.	48770	4
4	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	41820	4
4	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	114775	14
4	6	6. Hit reset, and select only the transactions that apply to your primary project.	50403	4
4	7	7. Put hours on the Y axis, employee on X, and size by hours.	103098	11
4	8	8. Find the employee who charged the most hours in a single transaction to the project by right clicking .	21341	4
4	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	39136	1
4	10	10. Color and Tag by Employee.	24315	4
4	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	31916	4
4	12	12. Turn tags off and hit Reset.	10815	2
4	13	13. Put hours on the Y axis, period on the X, and color by project name.	25377	4
4	14	14. Choose only dots that are related to you.	25256	4
4	15	15. Choose dots related to your primary project and up to 2 other projects you have worked on.	54198	4
4	16	16. Set jitter to min and line by project name.	67257	2
4	17	17. Can you see any trends in your work over the year?	17745	0
4	18	18. Turn lines off, and hit Reset. Feel free to solve the next set of questions using any combinations of features.	24115	3
4	19	19. Which project was costliest over the year?	382931	39
4	20	20. Who had the costliest charge to that project?	286712	29
4	21	21. Which period had the greatest number of charges?	104590	8
4	22	22. In which period did you work the most total hours?	95848	14
4	23	23. How many hours did you work in October, November, and December combined?	9494	0
4	24	24. How much money did you charge over the year?	63011	18
5	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	36322	5
5	2	2. Show only your transactions.	33018	4
5	3	3. Put hours on Y.	6820	2
5	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	43552	4
5	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	34870	5

5	6	6. Hit reset, and select only the transactions that apply to your primary project.	27560	9
5	7	7. Put hours on the Y axis, employee on X, and size by hours.	18867	3
5	8	8. Find the employee who charged the most hours in a single transaction to the project by right clicking .	32266	2
5	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	30644	2
5	10	10. Color and Tag by Employee.	14081	4
5	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	24615	3
5	12	12. Turn tags off and hit Reset.	11927	5
5	13	13. Put hours on the Y axis, period on the X, and color by project name.	18106	3
5	14	14. Choose only dots that are related to you.	17395	4
5	15	15. Choose dots related to your primary project and up to 2 other projects you have worked on.	25597	8
5	16	16. Set jitter to min and line by project name.	20490	2
5	17	17. Can you see any trends in your work over the year?	22472	0
5	18	18. Turn lines off, and hit Reset. Feel free to solve the next set of questions using any combinations of features.	13029	2
5	19	19. Which project was costliest over the year?	46006	9
5	20	20. Who had the costliest charge to that project?	59736	15
5	21	21. Which period had the greatest number of charges?	92583	16
5	22	22. In which period did you work the most total hours?	136162	21
5	23	23. How many hours did you work in October, November, and December combined?	89513	11
5	24	24. How much money did you charge over the year?	107194	24
6	1	1. Color by project name, put employee on the Y axis, period on the X, and size by hours.	76520	5
6	2	2. Show only your transactions.	49071	26
6	3	3. Put hours on Y.	17255	2
6	4	4. Find the transaction where you spent the most time and bring up the package viewer. Which project is it and what was the cost?	34039	3
6	5	5. Open summary and summarize by project name. On which project did you spend the most amount of time? This is your "primary project."	51974	5
6	6	6. Hit reset, and select only the transactions that apply to your primary project.	75639	20
6	7	7. Put hours on the Y axis, employee on X, and size by hours.	25267	4
6	8	8. Find the employee who charged the most hours in a single transaction to the project by right clicking .	56601	4
6	9	9. Select all transactions between 50 and 100 hours for the project using the slidebar.	34600	0
6	10	10. Color and Tag by Employee.	16503	4
6	11	11. Open statistics and organize by Employee. Who has the most 50-100 hour charges?	58394	7
6	12	12. Turn tags off and hit Reset.	11156	4
6	13	13. Put hours on the Y axis, period on the X, and color by project name.	70862	9
6	14	14. Choose only dots that are related to you.	22082	3
6	15	15. Choose dots related to your primary project and up to 2 other projects you have worked on.	26949	8
6	16	16. Set jitter to min and line by project name.	23213	2

6	17	17. Can you see any trends in your work over the year?	60387	2
6	18	18. Turn lines off, and hit Reset. Feel free to solve the next set of questions using any combinations of features.	13249	3
6	19	19. Which project was costliest over the year?	155163	19
6	20	20. Who had the costliest charge to that project?	64823	10
6	21	21. Which period had the greatest number of charges?	47038	4
6	22	22. In which period did you work the most total hours?	65965	14
6	23	23. How many hours did you work in October, November, and December combined?	135935	11
6	24	24. How much money did you charge over the year?	38225	4

Appendix B: Task Examples

Participants were presented guided and unguided tasks.

Guided tasks were explicit in explaining how the user was to manipulate the software:

- Color by project name, put employee on the Y axis, period on the X, and size by hours.
- Using statistics, who has the most 50-100 hour charges?

Unguided tasks asked the user to solve problems without offering the method by which to do so:

- Which project was costliest over the year?
- Which period had the greatest number of charges?

Appendix C: Goal Abstraction Sheet

Measurement Goal				
Object	**Purpose**	**Quality focus**	**Viewpoint**	**Environment**
the current graphical user interface of the VQI system	Evaluation	usability	project managers	timesheet application at FC-MD

Quality focus	**Variation factors**
What factors define the quality focus?	*Which factors have an impact on the quality focus?*
Q-QF1 How intuitive is the interface? M-QF1.1 Number of unsuccessful attempts to complete a guided task M-QF1.2 Number of unsuccessful attempts to complete an unguided task Q-QF2 Is the interface easy to learn? M-QF2.1 Percent more time spent on guided tasks than the expert user	Q-VF1 How much computer experience does the user have? M-VF1.1 How would you rate yourself as a computer user on a scale of 1 (novice) to 5 (expert)? Q-VF2 How experienced is the user with VQI? M-VF2.1 How would you rate yourself as a VQI user on a scale of 1 (novice) to 5 (expert)? M-VF2.2 How many hours a week do you use VQI?

M-QF2.2 Percent more time spent on unguided tasks than the expert user M-QF2.3 Percent more clicks spent on guided tasks than the expert user M-QF2.4 Percent more clicks spent on unguided tasks than the expert user Q-QF3 How does efficiency increase in relation to experience with the software? M-QF3.1 Percent more time spent on guided tasks than the expert user M-QF3.2 Percent more time spent on unguided tasks than the expert user M-QF3.3 Percent more clicks spent on guided tasks than the expert user M-QF3.4 Percent more clicks spent on unguided tasks than the expert user M-QF3.5 Average time in seconds to perform a guided task M-QF3.6 Average time in seconds to perform an unguided task M-QF3.7 Average number of clicks to complete a guided task M-QF3.8 Average number of clicks to complete an unguided task Q-QF4 What are the user's overall impressions of the software? M-QF4.1 Rating of the software as a whole on a scale from 1 (strongly dislike) to 5 (strongly like) M-QF4.2 Individual feature rating on a scale from 1 (strongly dislike) to 5 (strongly like) M-QF4.3 Comments and suggestions	
Baseline hypothesis	**Impact on baseline hypothesis**
What is the current expectation wrt. the quality focus?	*How do the variation factors influence the quality focus?*
H-QF1 Modifying the current design of the VQI interface will improve usability.	H-VF1 A more experienced computer user will have an easier time with the application. H-VF2 A more experienced VQI user will perform better than a less experienced user.