

Glass Box: Capturing, Archiving, and Retrieving Workstation Activities

Paula Cowley, Jereme Haack, and Rik Littlefield
Battelle – Pacific Northwest Division
902 Battelle Blvd.
Richland, WA 99352
509.375.2282 – Paula.Cowley@pnl.gov
509.375.6350 – Jereme.Haack@pnl.gov
509.375.3927 – Rik.Littlefield@pnl.gov

Ernest Hampson
Battelle – Colonial Place Operations
2111 Wilson Blvd., Ste. 1000
Arlington, VA 22201
571.227.6334
HampsonE@Battelle.org

ABSTRACT

The Glass Box is a computer-based environment that unobtrusively captures workstation activity data from analysts engaged in real intelligence analysis activities, with the aim of supporting research leading to the development of more effective tools for the intelligence community. The Glass Box provides automated data capture, analyst annotations, data review/retrieval functions, and an application programming interface enabling applications to integrate, communicate, retrieve, store, and share Glass Box data. Over 100 gigabytes of data representing eight staff years of analysts working in the Glass Box is regularly distributed to the Glass Box community where it is used for research, testing, and product evaluation. Other community members can also use the Glass Box software to capture data from experiments in their own environments.

Categories and Subject Descriptors

H.3 [Information Systems]: Information Storage and Retrieval

General Terms

Measurement, Design, Experimentation.

Keywords

Instrumentation, integration, capture, archiving, retrieval, workstation activity, activity monitoring, evaluation, API

1. INTRODUCTION

There's a lot happening today on the typical knowledge worker's workstation. A knowledge worker routinely uses email, generates and views documents, prepares spreadsheets and presentations, and uses instant messaging and web browsers. These knowledge workers often have many of these applications

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CARPE'06, October 28, 2006, Santa Barbara, California, USA.
Copyright 2006 ACM 1-59593-498-7/06/0010...\$5.00.

running at the same time and routinely move between them with the click of a mouse. In addition, many use other custom and commercial software applications appropriate to their job requirements. Imagine you needed to record this activity and needed to determine how you would capture and store this kind of activity. To start with, you need to record:

- Keyboard/mouse data
- Window events
- File open and save events
- Copy/paste events
- Web browser events
- Instant Messaging
- Email activity
- Video of what is happening on the screen
- Audio of what the workstation user records using a microphone

If your requirements end here, there are a variety of solutions commercially available. You can procure employee monitoring software that captures much workstation activity including video and audio. What you will find is that these tools capture activity at a relatively low level of detail. They can be readily used to address issues like inappropriate visits to web sites, who sent email to whom, and what transpired during an instant messaging session. These products generally provide mechanisms for retrieving and reviewing captured data. Most record the URL's of web sites visited and the names of documents opened and saved, but not the associated content.

However, in our case, we were required to go beyond this type of monitoring. We needed to capture, archive, and retrieve data to be used to understand what intelligence analysts do, so that existing processes could be modeled and new tools to improve those processes could be developed. Once new applications were implemented and tested, we were to provide mechanisms through which tools developed by separate research projects could communicate and share, and we needed to support the evaluation of these applications and their impact.

This ambitious set of goals was used as the basis for an on-going research effort known as Glass Box. During this four year effort, we have succeeded in meeting the goals previously identified. The rest of this document describes the Glass Box and how it supports the capture, archiving, and retrieval of workstations activities to enable the integration and evaluation of tools intended to improve analysis processes.

2. WHAT IS THE GLASS BOX?

We use Glass Box [2] to capture a lot of data about intellectual activity happening on the workstation by recording tasks assigned, topics being investigated on the web, queries and the sources being examined, search results, documents, what is being discussed using email and instant messaging, what is being worked on as represented by the files accessed and generated, and how applications are being used, both standard knowledge worker applications such as the MS Office Suite and web browsers and specialized applications that have been instrumented to log their activities to the Glass Box. We can actually see the evolution of a person's thoughts and plans when we examine the evolution of documents over time.

The Glass Box is both software and data. We have intelligence analysts performing real work while the Glass Box captures their activity [5]. This data is routinely distributed to those working to develop new tools for these analysts to use and evaluate. The data can be made available to others funded by the US government. We have over 100 gigabytes of data representing eight staff years of work. Five analysts, two senior and three junior analysts, worked in the Glass Box from one year to 2 ½ years. Another group of analysts spent two weeks each working in the Glass Box to provide another year's worth of data.

We also wanted to record what the analyst wanted to tell us about off-line, non-workstation activity, so we implemented an annotation capability where the analyst could tell the Glass Box about meetings, phone calls, collaborations, documents read, use of hard-copy materials, discoveries, insights, problems encountered, and plans for the day, etc. Figure 1 provides a screen shot of the Glass Box annotation feature. Although we do not expect annotations in the typical production analytic environment, the Glass Box research analysts are expected to provide annotations to better enable the study of the processes being used.

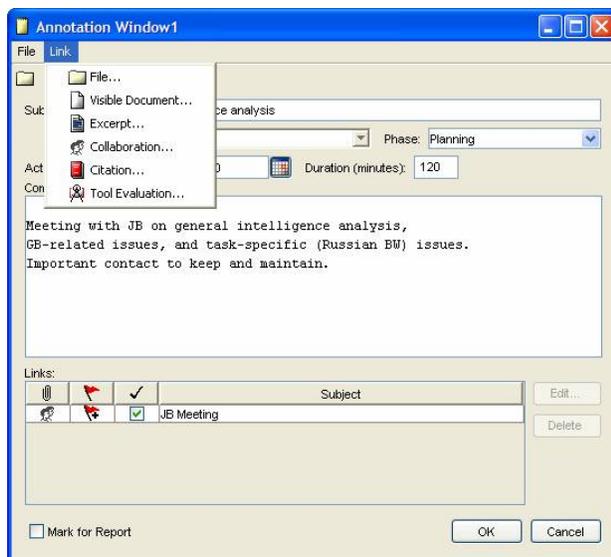


Figure 1. Glass Box Annotation Window

The Glass Box is software that resides on a workstation and quietly captures the parts of the workstation user's process that happen online. The Glass Box serves as the sensory input

mechanism – the eyes, ears, and hands that provide data about user activity. Our challenge was to develop a way to capture the data we needed, organize it in a useful and meaningful fashion, make it available to other applications, and, once captured, enable the data to be used for a variety of purposes, many of which we could not foresee.

3. HOW DOES THE GLASS BOX COMPARE TO OTHER SYSTEMS?

How does the Glass Box differ from other work on capturing, archiving, and retrieving user activity? A few interesting comparisons are provided. The Glass Box is focused on capturing workstation activity while Microsoft's MyLifeBits [3] is focused on personal lifetime storage and stores content and metadata for 25 item types including contacts, documents, email, events, photos, music, and video. MyLifeBits is much more visually oriented rather than the document orientation of the Glass Box. MyLifeBits is research oriented while Glass Box is focused on capturing analytic activity for further study. FolderPredictor [1] captures a user's previous file access behavior for use when the user is trying to locate a file. FolderPredictor only stores a subset of what Glass Box stores (i.e., storing open/save file dialog while not storing the actual file) but is more process focused and organizes and stores that subset to support predicting the file location. VibeLog [6], a windowing system logging tool, is used to research the impact of large monitors and multiple monitors. Like the Glass Box, it logs many kinds of window activity but goes further than the Glass Box in providing a variety of tools to enable users to deal more effectively with greater monitor space. Of all the instrumentation systems we are aware of, the Glass Box captures the widest variety of workstation activity but does not provide aids to the workstation user as other systems do. This aspect is by design, as the Glass Box was specifically charged with collecting data while not changing the user experience in its role as only one component of the larger NIMD¹ research program.

4. GLASS BOX FUNCTIONS

The Glass Box performs three major functions as shown in Figure 2:

- It supplies the “instrumentation” that captures workstation activity as users work in the Glass Box. Our instrumentation captures data about the context within which the work is taking place and reveals much about the processes being used.
- It serves as an integration platform for research systems so the systems can interact with the Glass Box and with other peer research systems that are handling additional functions. Integration functions and data include a record of current and past activities, logging activities important to the research systems, live event notification of user activity, and enabling communication between collaborating research systems.
- It serves as a tool for evaluating the effectiveness of research products by recording what really happened

¹ NIMD: Novel Intelligence from Massive Data; see acknowledgements.

during their use. Essentially, the Glass Box serves as a surrogate observer, freeing the human observers from having to record detailed workstation activity so they can focus on higher-level cognitive activities.

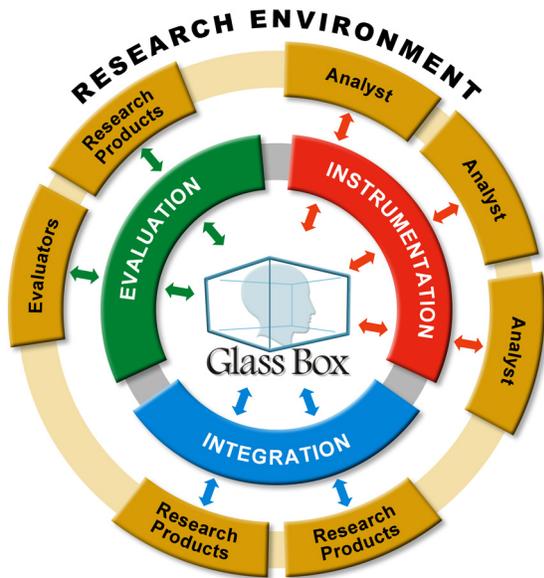


Figure 2. Glass Box Research Environment

By capturing user interactions and system activity, the Glass Box allows the research system to find out what the user is doing in near real time. Then it is up to that research system to determine what that activity really means and respond appropriately. By supporting the system’s ability to retrieve instrumentation data stored from previous activities as needed, the Glass Box also provides access to what the user has done in the past. This allows the system to develop and work from a current model of what the user is trying to do.

5. CAPTURING, ARCHIVING, AND RETRIEVING GLASS BOX DATA

5.1 Passive Instrumentation and Glass Box Review Tools

When running on a user’s workstation, the Glass Box’s instrumentation is passive in that neither the user nor the application has to cooperate with the Glass Box to generate data. The instrumentation captures keyboard/mouse data, window events (active window, active application, location on screen, etc.), file open and save events, copy/paste events, instant messaging, and Web browser activity (URLs, page contents, images, queries, and query results). Except for the responsibility of the user to turn on or turn off Glass Box recording and provide annotations, there is virtually no interference with the user. Most of what this instrumentation captures is obtained by observing the application from the operating system level.

The Glass Box captures the entire multimedia environment being used. We make extensive use of a data repository consisting of an integrated SQL Server database and a hierarchical file store. While most monitoring software

packages only save references to web sites visited and documents viewed and generated, the Glass Box explicitly stores web site and document content. Document content is captured each time the analyst opens a file or does a save. This approach allows us to retrieve what the analyst actually retrieved in spite of the fact that the original content may no longer be available. This is particularly important for web sites that are constantly changing and documents that evolve over time. If a web site is visited many times or a static document viewed more than once, we do not store duplicate files but simply add database records pointing to the existing files.

We also support the capture of screen content and audio recording. By default, we capture screen content at a rate of once per second but other rates can be set as a configuration parameter. Since the associated files are quite large, capturing video and audio is optional. In fact, by volume, video and audio content represents a high percentage of the volume generated from the Glass Box. One significant issue in dealing with screen capture data is that it takes a lot of time for a human to review and process the screen capture data. We provide capabilities to enable this but expect that most people reviewing the data will only go to the screen capture data when they cannot understand what is happening by viewing the other data.

One of our goals is to provide consumers of Glass Box data with information that is meaningful to them. To us, that meant that a person could work with the data of interest at the appropriate level of granularity. One researcher might want to understand the process an analyst was using. Another might want to investigate patterns over time in queries submitted to a search engine. Another might want to investigate keystroke patterns. We recognized that only providing access to one data type at a time would be too limiting for most people so we created a set of review tools that provides different multimedia visualizations of the data and uses a complex set of queries, filters, and heuristics to determine what data will be presented. While our reports are based on SQL queries that run against our database, we use a variety of views and filters to select and perhaps transform the data in some way. In some cases, the views and filters are used together to present a complex view of data from many tables. Figure 3 presents the Glass Box Tabular Review tool that shows how the integrated web, data, file, and keyboard event filter depicts complex activity. We use heuristics (common-sense rules drawn from experience) to infer some relationships that our instrumentation cannot directly observe. These filters can also be accessed and used through the Glass Box Application Programming Interface (API), which is discussed in the section on active instrumentation. This makes data available for use by other software.

Each significant event type as shown in Table 1 has a corresponding filter which allows it to be displayed in the Tabular Review. Web events and Window events also have filters that let the researcher see only the most important events. Two keyboard filters are available: one showing what the user actually typed with all the backspaces and retyping and the other showing only the final string.

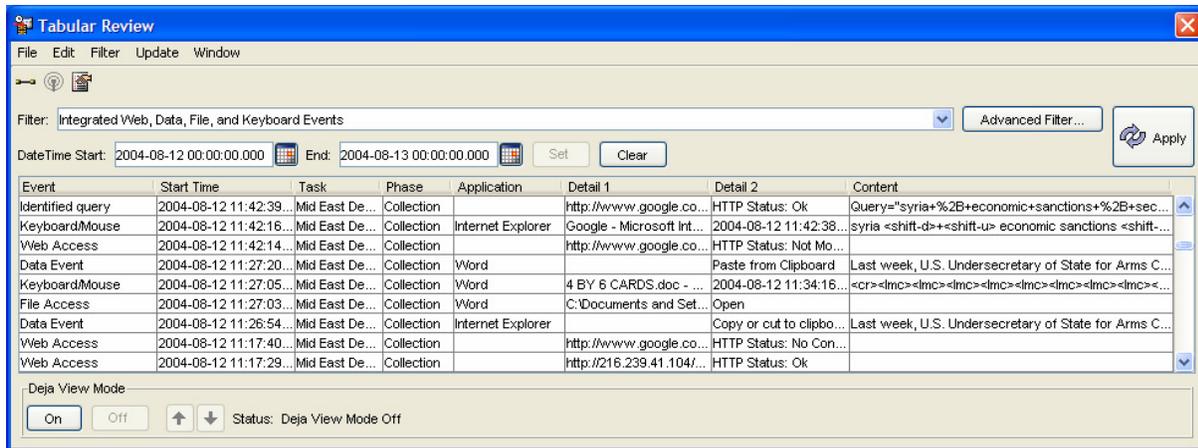


Figure 3. Tabular Review Showing Integrated Web, Data, File, and Keyboard Events

Table 1. Glass Box Event Types

Keyboard/mouse	Browser Bookmarks
Window	Instant messaging
File actions (e.g., open, save)	Mail
Copy/paste	Analyst sessions
Web browser	Annotations
Web queries	Logging

Along with our single event filters, there are a number of filters which combine several event types to give a more holistic view of user activity. The “Integrated Web, Data, File, and Keyboard Events” filter shown in Figure 3 combines filtered web, copy/paste, file, and keyboard events.

Filtered Comprehensive adds session, window, and annotation events. Finally, the all-encompassing “Comprehensive plus Keyboard” filter adds keyboard events for a complete view of user activity.

The Glass Box Review Tool provides analysts and researchers with a quick review and analysis capability, including the capability to observe:

- Progression of activity over time
- Repeated patterns over time
- Information clusters (i.e., areas where analysts repeatedly spend time)
- Breadth of scanning and narrowing of focus.

The Review Tool provides three different and complementary ways of visualizing the Glass Box data. The **Tabular Review** window shown in Figure 3 displays a chronological record of recorded events that can be explored in detail. The display can be filtered as described above. From the Tabular Review, you can turn on the **Déjà View** mode to see what the user saw. Events like web pages viewed and documents saved or opened are displayed. The arrow keys allow you to move easily through these events with access to the entire content of the event. However, this does not tell you what portions of the material the user actually saw since the user may have only looked at selected portions. To find this level of detail, you can access the

screen capture data through our review tool called “**Over-the-Shoulder**” that displays captured screens of what the user was seeing on the workstation at the selected point in time. If the mouse trail option is turned on, you will see how the user moved the mouse on the screen. The **Timeline Review** provides a time-based visualization of multiple events in relation to each other. This view shows the flow of information across activities, the specific points in time where the user performed mouse actions and keyboard inputs, and the resulting effects on applications and windows. Our review tools are normally synchronized so the selected view in one tool is automatically reflected in the others. Sometimes researchers turn this off to facilitate their exploration of the data.

Our passive instrumentation has many of the same features found in employee surveillance tools but goes beyond what they provide in several important ways:

- We provide comprehensive views of the data rather than restricting the researcher to viewing one data type at a time.
- Our database is open; you can query into it using SQL to go beyond the review capabilities we provide.
- Our data is focused on understanding the higher cognitive processes being observed.
- We capture web pages and documents as they were at the time they were accessed so we can recreate what was actually retrieved.
- Users are aware that the Glass Box is running and can cooperate with the process. (In fact, our analysts are protected as human subjects during our experiments.)
- Our review tools are intended to be a starting point for examining the data, not the only way the data can be accessed.

We do not try to anticipate all the ways in which the data will be used. We expect the unexpected.

5.2 Active Instrumentation and Integration

Besides the passive instrumentation discussed in the previous section, the Glass Box also provides active instrumentation where applications can interact with the Glass Box through its API. The Glass Box’s active instrumentation supports the ability of applications to use data from the passive instrumentation, log their own events into the Glass Box repository, use the Glass Box as a messaging hub between cooperating applications, and

obtain notification of user and third-party application events through a real-time publish/subscribe mechanism.

This section provides an overview of how the Glass Box uses active instrumentation to support integration and evaluation. Our capabilities for instrumentation, integration, and evaluation are implemented to work together. In fact, our instrumentation and integration capabilities support evaluation and there is no separate evaluation capability. Evaluation is performed using all the features of the Glass Box.

The Glass Box uses a Services Oriented Architecture, in particular, a Web Services design and implementation. This allows instrumented workstations, the data repository, third-party application servers, and review stations to be separated from each other by multiple firewalls and by networks with limited bandwidth and high latency. The entire Glass Box can be installed on a single workstation or components can be distributed across workstations, servers, or networks with one or more workstations writing to a data repository. This flexible architecture has evolved over time, with significant advances and differences from what was originally envisioned as the most likely development path [4]. It allows us to support a “lightweight client” (LWC) that significantly reduces the footprint of Glass Box on instrumented workstations in an environment where the data repository cannot reside on the client machine or perhaps even on the local network.

Glass Box uses Web Service communications internally to implement major data paths between the data repository, the instrumentation, and the review tools. This allows recording and review workstations to be network-remote from the data repository, from each other, and from third-party applications that access the data repository using the API. The Glass Box data repository and publish/subscribe services have been bundled together into what is called an Analytic Event Server

(AES), where events that are deemed important enough to be sent in real time are recorded and made available to queries and subscriptions. Other low level events that are not important enough to be sent to the AES in real time can be recorded locally at the client and loaded to the data repository later if desired. The determination of exactly which events fall in which category is determined by a configuration option. In the most extreme approach, all events down to the level of individual mouse movements can be pushed to the AES in real time for recording and distribution.

The use of web services allows cross-firewall communication. All communication is done through a single port so it is much easier to open a hole in the firewall. We have made sure that the Glass Box always plays the role of server, with all communications initiated by its clients. While this has made some features more difficult to implement (publish/subscribe, for instance), it greatly reduces requirements on the client. In particular, the client does not need to open any ports or change its environment in order to access the Glass Box web services.

The Glass Box architecture divides the software into three components: the Lightweight Client (LWC), the Analytic Event Server (AES), and the API Client, which is used in researcher applications residing on the user workstation or at remote locations. Figure 4 shows how these components can be distributed across firewalls.

The Glass Box allows applications to integrate with each other, not necessarily with the Glass Box itself (i.e., integration does not mean that there will be a menu entry on the Glass Box Control Panel). By using the Glass Box, applications can log their own events into the Glass Box database. Logging focuses on what is happening within an application. Logging can be used for a variety of purposes including evaluation, debugging, understanding how a user is actually using an application, etc.

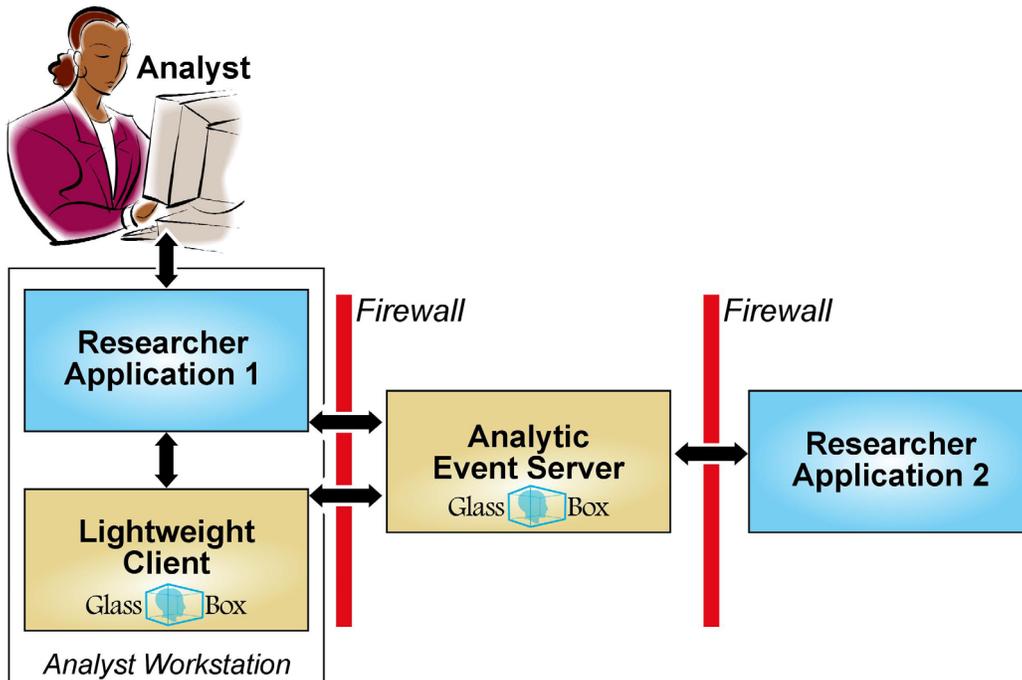


Figure 4. Lightweight Client with Analytic Event Server and Remote Application Behind Firewalls

The Glass Box provides a flexible logging capability to provide each application with the ability to record what its developers and evaluators consider “interesting.” The Glass Box provides the infrastructure to support logging and allows the applications to record data of interest using data formatted with XML.

Applications can subscribe to topics the research projects have agreed upon. When an application publishes an event, subscribers to that type of event are notified. They can then process the data associated with the event, publish a response, and take action as needed. Applications can subscribe to events generated by the analyst through the Glass Box instrumentation, through events logged by applications, or through other events published by the applications.

5.3 Evaluation

Successfully undergoing a series of evaluations have been important milestones for NIMD projects. An application evaluation is performed using both the instrumentation and the integration features of the Glass Box. The application being evaluated uses Glass Box logging to record what the evaluators and the developers have negotiated to be important. Stability testing is also performed. Once these evaluations are completed satisfactorily, the application is finally ready to be evaluated by intelligence analysts, who work in the Glass Box using the application on assigned analysis tasks. Generally, a week is spent training 2-8 analysts to use the application and practicing on a task designed to exercise that application’s capabilities. A second week is spent working on another task where the analysts are expected to use the application being evaluated. The analysts are also asked to provide feedback on the application. The data is then processed and provided to researchers and evaluators. The evaluators [7] use the Glass Box data, the Glass Box review tools, and their own queries and extraction techniques to perform their evaluation as well as report those results to the application developers and to the NIMD Program. The evaluators consider the Glass Box to be crucial tool for evaluation. The Glass Box quietly collects data and provides integration services so the evaluators can focus on the more subjective portions of the evaluation. Both the evaluators and the developers have access to the Glass Box data so they can see how the application was really used and where the analysts encountered problems.

The Glass Box has participated in two major integration efforts where five and then six research applications including the Glass Box were integrated. These experiments were conducted to demonstrate that applications implemented by diverse research teams could work together through the Glass Box and through a common user interface rather than the analysts being presented with five or six diverse applications. Both integration experiments demonstrated that the Glass Box can successfully be used to support integration between diverse applications and supply the data necessary to evaluate those applications.

6. CONCLUSIONS

The combination of Glass Box’s instrumentation, annotation, and integration capabilities provides a unique environment to support in-depth and continuous archival, retrieval, analysis, and evaluation of workstation activities. From the lowest level keystroke and mouse click to high-level modeling of the analysis

process, Glass Box provides the data that researchers and developers need to understand what the analyst is doing and how the introduction of new tools affects the processes being used. This allows researchers to identify ways in which better tools can be developed so analysts can deal better with information overload and critical deadlines. Our novel combination of workstation instrumentation and integration capabilities has been successfully demonstrated through integration experiments involving multiple, diverse applications. The Glass Box has been crucial in the successful implementation and evaluation of these experiments.

We have provided the Glass Box community with a robust set of data retrieval and display tools that focus on providing meaningful data through our filters and views. However, researchers and developers are not restricted to working with the data through those tools. Our open architecture and documentation, including an extensive data dictionary, enables data access through the user’s tool of choice.

Although specifically designed for the intelligence analyst, we can see using the Glass Box for other tasks: work flow analysis, evaluating other types of software, tracking how a user exercised an application, documenting who was responsible for entering business critical data, and seeing the evolution of thought as documents are generated.

7. ACKNOWLEDGEMENTS

Development of the Glass Box was funded by the U.S. Government’s Disruptive Technology Office as one component of their Novel Intelligence From Massive Data (NIMD) Program.

8. REFERENCES

- [1] Bao, X., Herlocker, J., Dietterich, J. 2006. Fewer Clicks and Less Frustration: Reducing the Cost of Reaching the Right Folder. Proceedings of the 11th international conference on intelligent user interfaces.
- [2] Cowley, P., Nowell, L., Scholtz, J. 2005. Glass Box: An Instrumented Infrastructure for Supporting Human Interaction with Information. Hawaii International Conference on System Sciences, January 3-6, 2005.
- [3] Gemmell, J., Bell, G., Lueder, R. 2006. MyLifeBits: A Personal Database for Everything. Communications of the ACM, January 2006.
- [4] Gorton, I., Haack, J. N. 2004. Architecting in the Face of Uncertainty: An Experience Report, International Conference on Software Engineering.
- [5] Hampson, E., and Cowley, P. 2005. Instrumenting the Intelligence Analysis Process. First International Conference on Intelligence Analysis.
- [6] Robertson, G., Czerwinski, M., Baudisch, P., Meyers, B., Robbins, D., Smith, G., Tan, D. 2006. The Large-Display User Experience. CHI ’06 extended abstracts on Human Factors in Computing Systems.
- [7] Scholtz, J., Morse, E., and Hewett, T. 2004. In Depth Observational Studies of Professional Intelligence Analysts. Human Performance, Situation Awareness and Automation Technology Conference.