CAR-TR-826 May 1996
CS-TR-3637

Life Cycle of User Interface Techniques: The DJJ Information System Design Process

Anne Rose, Jason Ellis, Catherine Plaisant, Stephan Greene

Human-Computer Interaction Laboratory
Center for Automation Research
University of Maryland, College Park, MD 20742-3255
http://www/cs.umd.edu/projects/hcil/
rose@cs.umd.edu

Abstract

To take advantage of today's technology, many organizations are migrating from their legacy systems. With help from the Human-Computer Interaction Laboratory (HCIL) and Cognetics Corporation, the Maryland Department of Juvenile Justice (DJJ) is currently undergoing an effort to redesign their information system to take advantage of graphical user interfaces. As a research lab, HCIL identifies interesting research problems and then prototypes solutions. As a project matures, the exploratory prototypes are adapted to suit the end product requirements. This case study describes the life cycle of three DJJ prototypes: (1) LifeLines, which uses time lines to display an overview of a youth in one screen, (2) the DJJ Navigator, which helps manage individual workloads by displaying different user views, and (3) the ProgramFinder, a tool for selecting the best program for a youth.

INTRODUCTION

Social service agencies are often plagued with legacy software systems that are difficult to use and provide inadequate functionality. One cause of these problems is poorly designed user interfaces. A good user interface can increase performance, reduce errors, and enhance user satisfaction. Advances in user interface technology coupled with the affordability of personal computers are motivating many organizations to redesign their outdated systems (Sawyer, 1995).

In 1994, the Human-Computer Interaction Laboratory (HCIL) began working with the Maryland Department of Juvenile Justice (DJJ) in an effort to redesign the user interface of their current information system, ISYS (Information System for Youth Services). ISYS is a terminal based system used to handle 50,000 juvenile complaints per year (Figure 1). It is used by about 600 DJJ employees in various offices and facilities across the state.

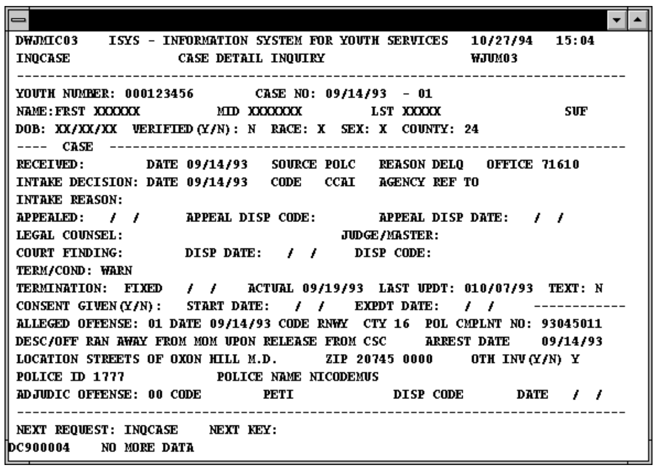

Figure 1. Sample ISYS screen

NISYS, the next generation ISYS, will run on PCs in a Windows™ environment. Visualization techniques are used to display youth record overviews, and worker-specific screens facilitate task scheduling and document management.

The Redesign Effort

To learn and assess ISYS, we conducted 22 field visits, attended training sessions, and administered the Questionnaire for User Interaction Satisfaction (QUIS) (Rose, Shneiderman & Plaisant, 1995; Slaughter, Norman & Shneiderman, 1995; Vanniamparampil, Shneiderman, Plaisant & Rose, 1995). Based on our findings, we proposed 28 short term recommendations to improve ISYS while NISYS was being developed. Our recommendations focused on making system access easier, improving data accuracy, making information retrieval easier, increasing the usefulness of the system, and enhancing user satisfaction. DJJ's initial response was to take action on all 28 recommendations. However, because of the effort dedicated to designing NISYS only a subset of the recommendations have been implemented to date.

Next we developed three NISYS prototypes:

- LifeLines, which uses time lines to display a youth's history with DJJ in one screen (Plaisant, Milash, Rose & Widoff, 1995; Milash, Plaisant & Rose, 1995),
- the DJJ Navigator, which helps manage individual workloads by displaying different user views, and
- the ProgramFinder, which can be used to find the best program for a youth.

Early prototypes have been shown to approximately 60 DJJ personnel, ranging from case workers to supervisors. Overall their reactions have been very positive. We are refining the prototypes and integrating them into NISYS based on their feedback.

In 1995, Cognetics Corporation joined our team. HCIL and Cognetics are designing NISYS jointly using Cognetics' Design Methodology (CDM) for quality usability engineering (Kreitzberg, 1995). Cognetics is responsible for preparing the request for proposal (RFP) for NISYS and for developing a testable prototype. The RFP will contain the user interface specifications in addition to the functional specifications.

In addition to redesigning the user interface, it has become evident that many of DJJ's processes need to be re-engineered. Specifically, the role of documents in process re-engineering has emerged as a major focus of this project (Greene, 1996). We compiled a comprehensive inventory of DJJ documents, analyzed their reporting needs, and are currently helping them redesign their documents. Many of the new documents will be incorporated into NISYS.

LIFE CYCLE OF USER INTERFACE TECHNIQUES

In this paper, we describe the life cycle of three NISYS prototypes from problem to product (Figure 2). The cycle begins with a problem that needs to be solved. Exploratory prototypes, ranging from paper mockups to Delphi applications, are then developed to illustrate possible solutions and communicate ideas among the designers, users, and implementors. For researchers, the next stage typically involves generalization or conceptualization of the techniques used. This is when most of the research papers are written. The other path is more commercial. It leads to the refinement of the prototypes and integration into a final product. HCIL's primary role in the NISYS project has been to design exploratory prototypes for key functional areas while Cognetics is responsible for integrating these ideas into the NISYS design.

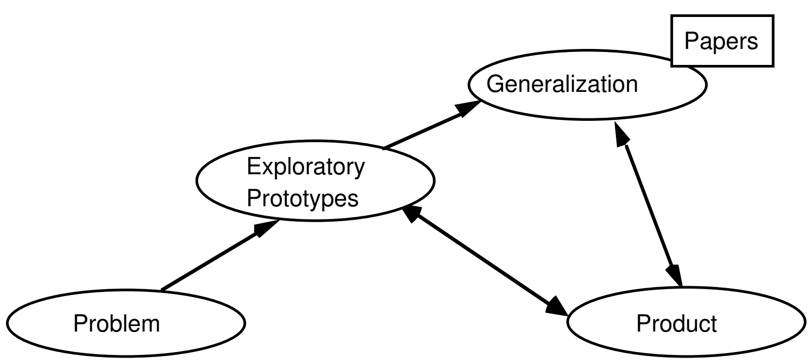

Figure 2. Life cycle of user interface techniques

The exact sequence of these stages varies from project to project (Figures 3, 6, and 11). The white ovals represent phases accomplished during the NISYS project, the gray ones were by done previously, and the hashed ones remain to be done. The LifeLines and Navigator designs started at the problem stage while the ProgramFinder is an application of a general visualization technique that had been developed in a previous project. The NISYS project is ongoing so the product designs are not final.

LifeLines

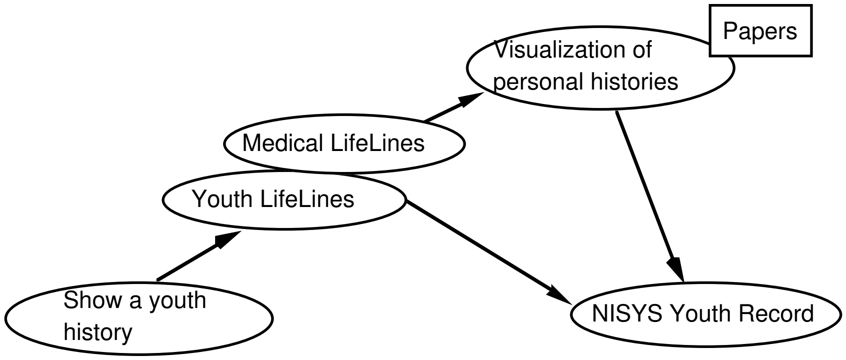

Figure 3. Life cycle of LifeLines

Problem:

With ISYS, it is very difficult and time consuming to get an overview of a youth's history with DJJ. Case workers have to use cryptic codes to navigate through dozens of screens. Important information is difficult to find, data entry problems are common, and processing delays often go undetected. During an ISYS training session we were told that "the trick is to find the magic case," i.e., you have to be lucky enough to find the case that contains the information you need.

Exploratory Prototypes:

LifeLines uses multiple timelines to present a youth record overview in one screen. The buttons on the top provide access to general information, such as aliases and significant others. Critical flags (e.g., suicide risk) are displayed above the buttons. All of the cases, placements, assigned workers, and reviews are plotted on timelines. Each facet can be expanded or collapsed (i.e., hide the labels and draw the lines closer together).

Figure 4. Exploratory prototype of LifeLines

The timelines provide a visual overview of the youth record. From Bart's record (Figure 4), we can see that he has been involved with DJJ for slightly over a year. During that period, he has been referred four times, twice for breaking and entering (B&E), once for auto theft, and most recently for attempted murder. Line thickness is used to indicate severity and color indicates the depth of penetration into the system. The two breaking and entering cases were handled informally by DJJ (no court intervention) and the auto theft case was sent to court where Bart was found delinquent (DELQ). As a result, Bart was committed to Cheltenham (a residential youth facility). The right edge of the timelines shows Bart's current status: he is detained at Waxter (a detention center) while case manager Brown decides whether to forward the attempted murder charge to court.

Detailed information can be retrieved with a click or two of the mouse. A click on a review icon brings up the text of the review. Similarly details about the cases or placements can be obtained by clicking on the lines or labels. Clicking on the worker's name brings up contact information. Relationships among the different timelines can also be highlighted. For example, one cannot tell by looking at the timelines if the drug abuse program was a result of the auto theft case or the breaking and entering case, but clicking on the drug abuse program highlights the related case (the B&E case in this instance) and the worker assigned to that case.

DJJ's reaction to LifeLines has been very positive. Users especially like how easy it is to access details and get a quick overview of a youth. The timelines also help users discover data entry problems, like cases that were never closed. This type of error normally often goes undetected in ISYS. A few users have expressed concerns about the possible bias associated with color and thickness coding (Shneiderman & Rose, 1996).

Generalization:

Following the exploratory prototype, the HCIL team explored other application domains and extended the LifeLines to a general technique for visualizing personal histories, such as medical, financial, legal, and education histories. For example, the facets of a patient record might include conditions, medications, and doctor's visits. The lines within each facet represent stories, like a three year history of heart problems. Each story may also have a series of episodes or periods, such as light versus acute back pain, which are shown using line thickness and color. Discrete events, like doctor visits or legal decisions, are shown with icons. LifeLines has been applied to a sample cardiology patient record (Plaisant & Rose, 1996).

Product:

Based on suggestions from DJJ and on the learning from the generalization phase, LifeLines is being refined and incorporated into the NISYS Youth Record (Figure 5). We believe that the successful completion of the generalization phase, followed by specialization of the LifeLines, was a very beneficial process which lead to a more complete and flexible interface.

The major difference is the addition of the event list. While LifeLines is very useful for showing the current status of a youth, highlighting relationships, and spotting anomalies, the event list is better for showing specific dates and more textual detail. The two displays are tightly coupled. Clicking on an event in the timelines highlights it in the event list, and vice versa. Double clicking on an event in either display brings more details. Users can also filter the list by type (e.g., reviews, placements).

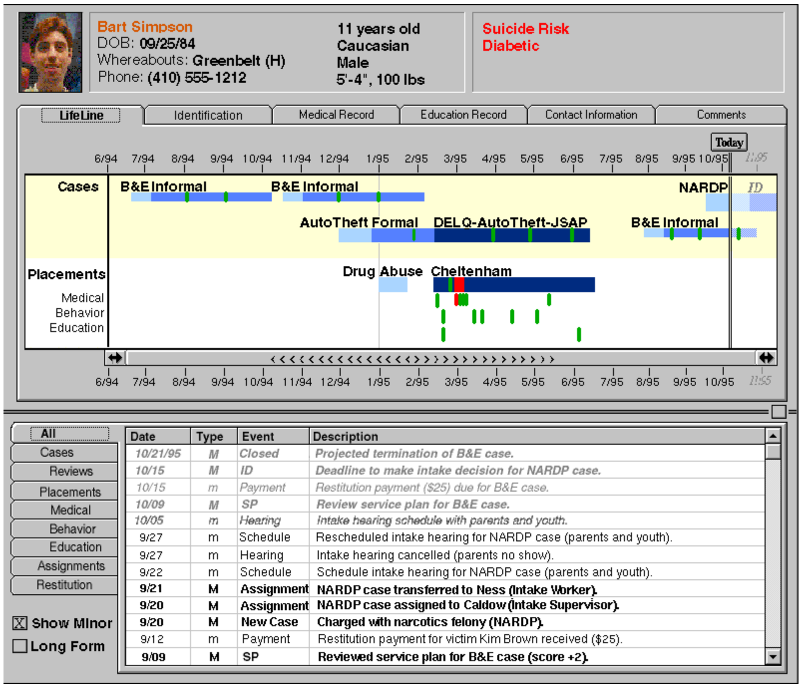

Figure 5. NISYS Youth Record

To help users focus their attention on important information, green is used to indicate normal events and red signals critical events. To facilitate workflow, future events are displayed and popup menus accessible from the line segments provide access to the documents associated with different processes. A click on a future review pops up a blank review form and a click on a case's intake period (light blue section) displays a menu of intake documents. The primary uses of the Youth Record will be to get overviews of new youths and to work on current assignments. The Today button allows users to quickly zoom in on the active cases.

Different types of users are interested in different information. For example, a nurse wants to see medical information while a teacher is more interested in the education history. Therefore, a popup control panel was added to allow users to customize the contents of the timelines.

Specific rules for label abbreviation and positioning have been designed. Other aspects specific to the application are still being worked on (e.g., how to display cases that have been merged or split by the courts).

DJJ Navigator

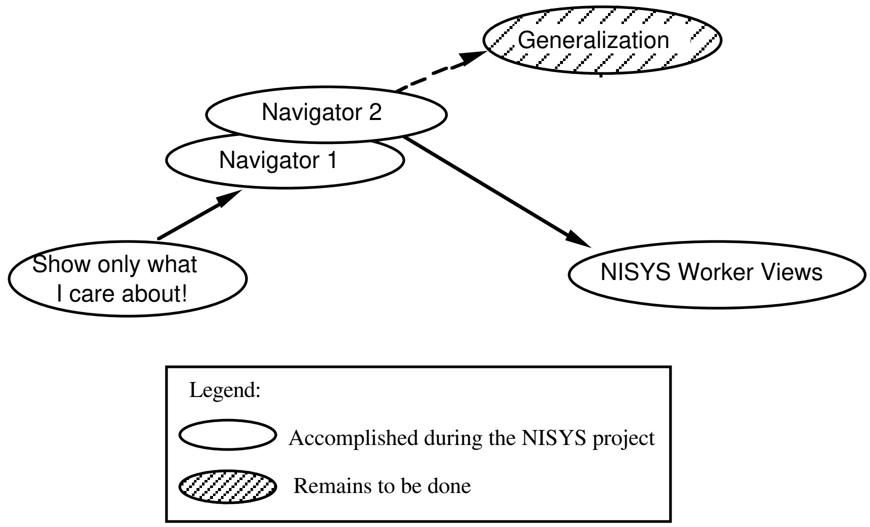

Figure 6. Life cycle of DJJ Navigator

Problem:

One of the key problems with ISYS is that users must search through thousands of records simply to find the information they are interested in. Supervisors, case managers and facility workers are all interested in different information. The information is there but is not presented in a manner which facilitates the users' work. Many workers see the system as a burden rather than an aid. They perceive ISYS as a sort of black hole that they must put data into while receiving little back for their effort.

Exploratory Prototypes:

The DJJ Navigator helps manage individual workloads by displaying different user views. For example, a facility worker needs to view the youths in their facility while a case manager would be interested in seeing only their caseload. In general, DJJ works with the youth as a whole or with an individual case involving a youth. Because of this distinction, the first Navigator prototype (Figures 7 and 8) used a multi-paned approach to show youth and case data. Additional data (e.g., victims) could be shown in another pane.

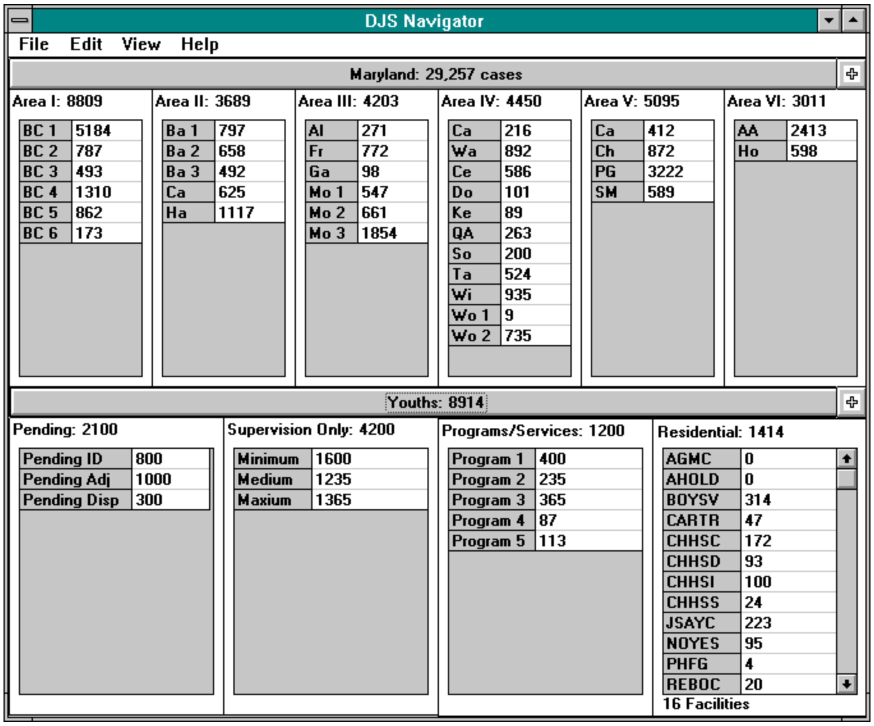

Figure 7. First Exploratory Prototype of Navigator showing Headquarters' View

The Headquarters' View provides an overview of all the cases and youths currently being handled by DJJ. The cases are divided by area of jurisdiction and the youths by type of supervision. Selecting different levels of information displays different views. For example, doubling clicking on PG in Area V would display the Prince George's Office View. The cases would be divided by units within the office (e.g., intake, investigation, case management) and only youths being supervised by PG workers would be displayed. Users can also use controls to focus on specific data (e.g., cases with pending intake decisions or youths who are currently AWOL).

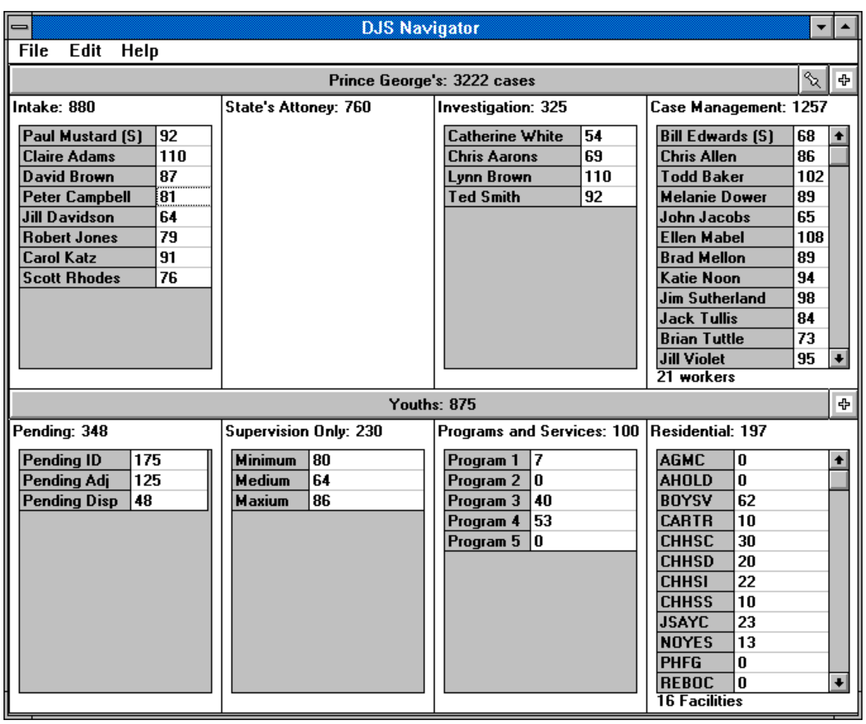

Figure 8. First Exploratory Prototype of Navigator showing Prince George's Office View

A second prototype was developed that focused more specifically on DJJ's workflow by providing a group of pre-defined user views to support different tasks (Figure 9). The emphasis moved from user manipulation of views to user selection of preset views and the multi-paned approach was replaced by a single spreadsheet-like display. A case worker would have a To Do view which would display a list of things they needed to do (e.g., schedule intake hearings, make intake decisions, etc.). Clicking on an item in the list would popup the paperwork partially filled out. More experienced users would be able to create their own custom views to help manage their workloads. They could display different type of elements, like cases, youths, victims, or events.

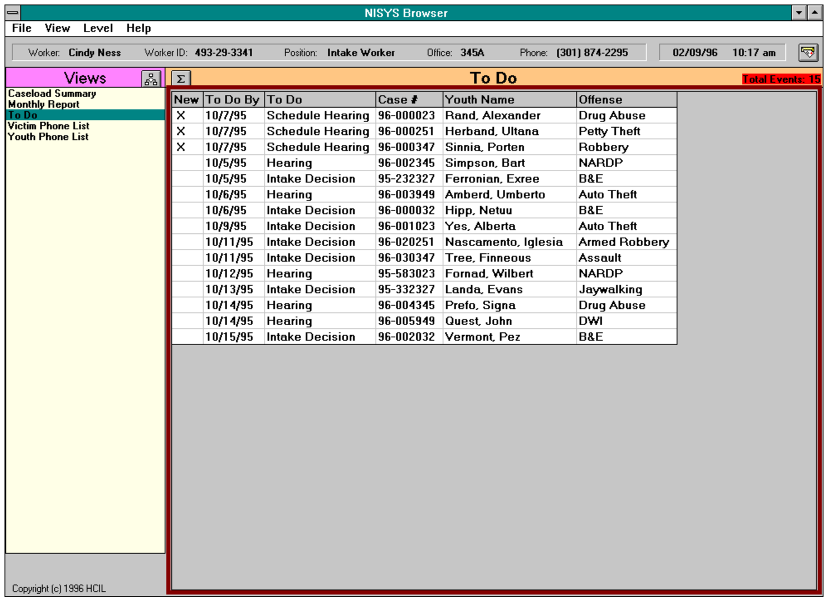

Figure 9. Second Exploratory Prototype of Navigator showing To Do view

Generalization:

We have not explicitly generalized the Navigator to other applications but we can imagine using it as a generic workflow or personal role management tool (Shneiderman & Plaisant, 1994; Landsdale, 1988). The first Navigator could be used to display and navigate hierarchical, relational datasets. For a school, cases and youths could become students, courses, and teachers.

To support generic datasets in the first Navigator (Figures 7 and 8), we proposed the Pane Definition Language (PDL), a formal language which allowed users to specify Navigator views and relationships. This language proved inadequate when there are dozens of views because describing each one independently is time consuming and leads to potential inconsistencies. Our second attempt was a rule-based scheme which endeavored to define a concise set of rules for generating the views. This also proved problematic as some views needed to be tailored to specific tasks and did not fit our rules. Finally, we started investigating a hybrid approach that would consist of a general rule set as well as some PDL-like screen descriptions. This approach would allow the user to address special cases while doing away with the need to describe every screen.

Product:

Without a good general model of the Navigator user interface we had to move toward very specific individual Navigator screens custom tailored for each type of user. These screens will be the first screen a user sees when they log on to NISYS. Current efforts are focused on the Case Worker screen and defining the tools needed in this domain (Figure 10). The worker's case assignments plus a set of preset views are still shown in a spreadsheet display. The contact and to do lists have been given their own areas since his information is used frequently. We have tried to propagate frequently used youth information to the Case Worker screen so the time spent swapping between screens is minimized. Hooks to other applications, like word processors and email, were also added.

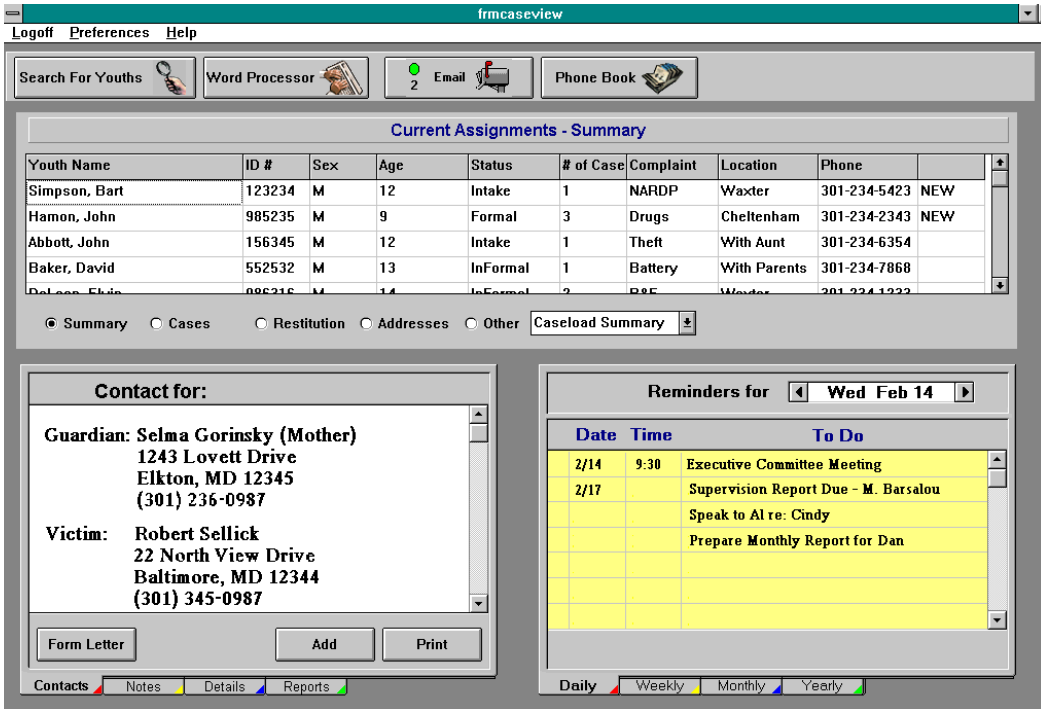

Figure 10. NISYS Version of Case Worker View

We plan to continue refining the Case Worker View and delve further into defining other user views such as clerical and facility workers. Here the lack of generalization principles limits our ability to rapidly prepare screens for all the different users. On the other hand, it is likely that the continuing design of these screens will sharpen our understanding of the interface challenges and lead us toward general principles for the design of case management user interfaces.

ProgramFinder

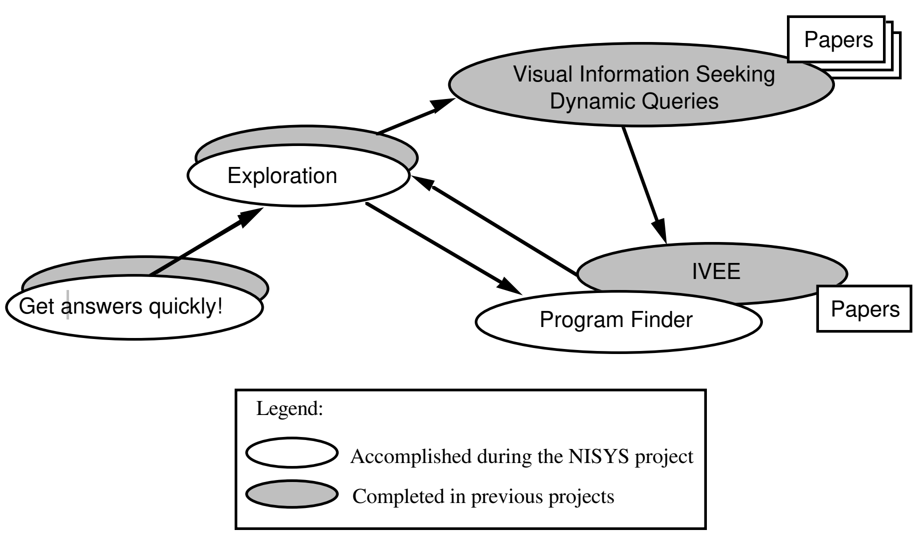

Figure 11. Life cycle of the ProgramFinder

Problem:

DJJ personnel, newspaper reporters and other social service agencies often need answers to questions like “How many auto thefts were committed last month?”, “What is the best program for this youth?”, or “Which offenses are typically committed by 11 to 13 year olds?” Answering these questions may involve writing a computer program to extract data from ISYS or manually searching through hundreds of screens or paper files. Both of these techniques are very time consuming and often lead to incorrect answers. NISYS needs to allow users to explore the data and make queries.

Exploratory Prototypes:

Database query languages with tabular output or spreadsheets could satisfy many of DJJ’s standard needs but we suggested the use of data visualization tools as an appealing and powerful alternative. By visualizing data and allowing fast and easy exploration, dynamic query (DQ) applications help users identify outliers, explore trends, and support routine and ad-hoc queries. Dynamic queries continuously update search results in a visual display (e.g., scatterplot, map, etc.) as users make queries by adjusting sliders or selecting buttons (Williamson & Shneiderman, 1992; Ahlberg & Shneiderman, 1994).

We used IVEE to demonstrate DQ applications to DJJ. The Information Visualization and Exploration Environment (IVEE)[1] is a product based on earlier dynamic query research. IVEE is a generic tool that automatically builds DQ applications for a given dataset by creating query controls (alpha sliders, range sliders, or toggle boxes) for each attribute and visualizing the elements in a starfield display (xy scatterplot) or on a map (Ahlberg & Wistrand, 1995).

First, a sample dataset of 4800 case referrals was visualized and explored (Figure 12). Two interesting trends were discovered immediately: the ages for several youths were incorrect (e.g., 95 years old, 35 years old, less than one year old) and many intake decisions required more than the 25 day maximum. Upon further exploration, we discovered several patterns of referrals, including one that indicated a drug bust (Rose & Vanniamparampil, 1995).

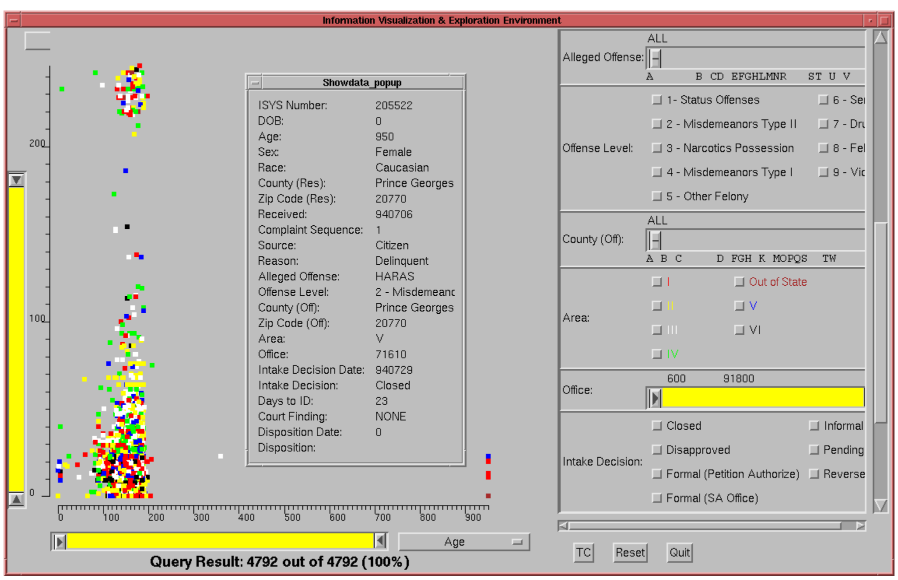

Figure 12: Complaints received during July 1994 colored by area of jurisdiction

[1]IVEE is being developed by Christopher Ahlberg and Erik Wistrand of Chalmers University, Sweden. It is based on earlier research by HCIL. URL: http://www.ivee.com

DJJ found it very appealing to be able to explore trends in their data and make queries so easily so we started exploring other potential applications. Using IVEE, we demonstrated the Dynamic ProgramFinder (Figure 13), a tool for selecting the best programs for a youth based on criteria such as security, location, and services provided.

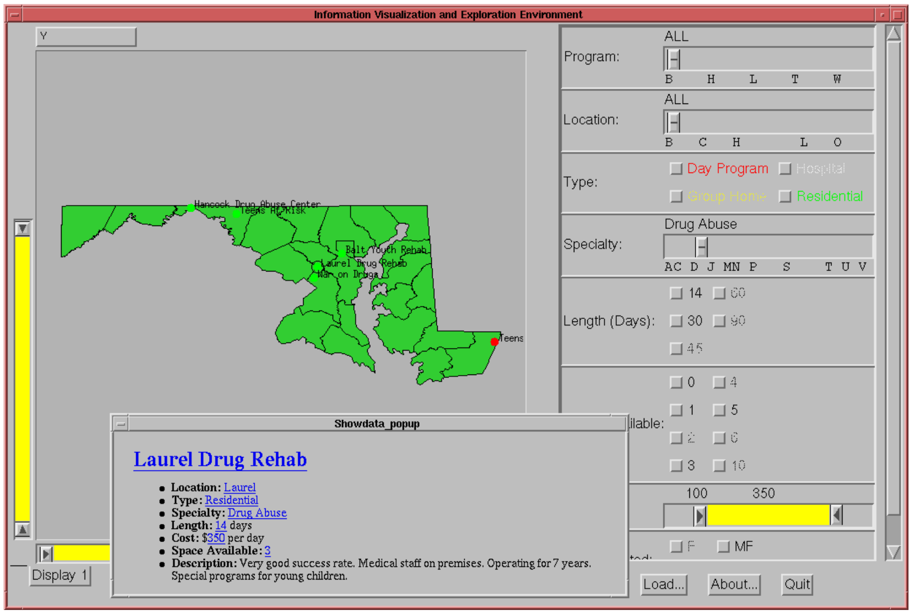

Figure 13: Exploratory Prototype of the Dynamic ProgramFinder

Generalization:

In this case, the generalization phase took place before the DJJ project even started. We could follow visual information seeking principles (Ahlberg & Shneiderman, 1994) and use a product outcome of the research to generate the next round of exploratory prototype.

Product:

DJJ can use IVEE to support their research and general analysis needs but some parts of NISYS were found critical enough to require the design of a customized DQ application (e.g., a ProgramFinder to find the best programs for the youths). By including the ProgramFinder in NISYS, it may also be possible to automatically set some of the query criteria based on prior youth assessments. The NISYS version of the ProgramFinder (Figure 14) provides access to individual Youth Records and allows users to select a set of programs and then send a referral packet to each. A referral packet is the set of documents sent to a program requesting that they consider admitting a youth.

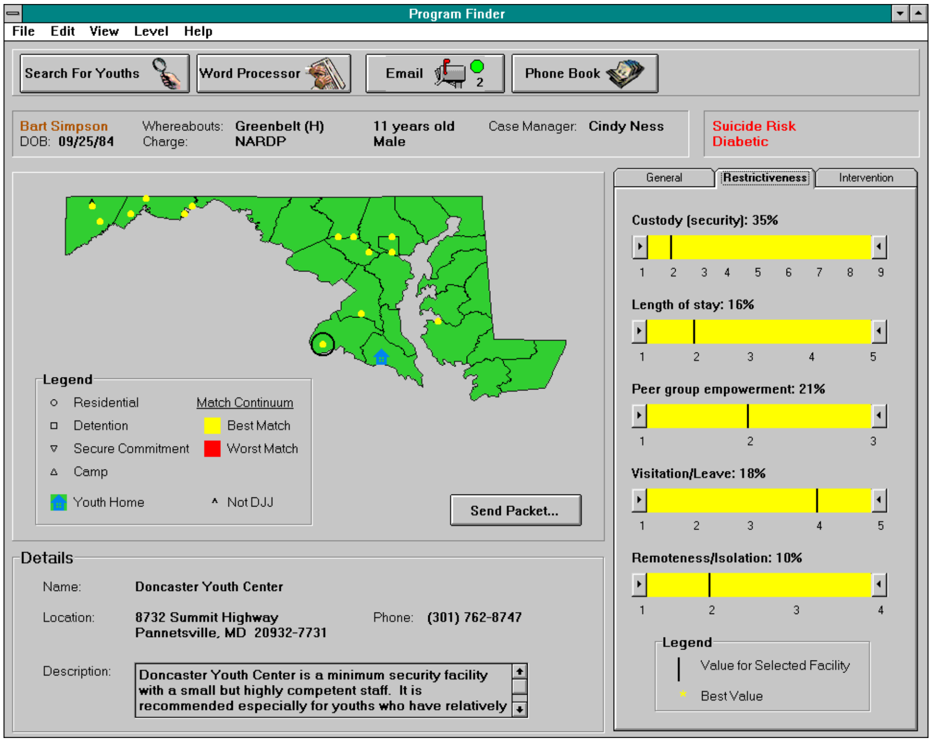

Figure 14: NISYS version of Dynamic ProgramFinder

The custom ProgramFinder can display the programs on a map or in a spreadsheet. The spreadsheet has the advantage of being able to show more program attributes at once while the map is better for spatial information. In the map display, shape indicates the type of program (e.g., residential or non-residential) and color shows how well a program matches the settings. The house icon shows where the youth lives so transportation and visitation issues can be considered. The program details were included in the side window.

To avoid scrolling lists of controls, tab panels were used to divide the controls into smaller more manageable groups: general, restrictiveness, and intervention. The restrictiveness controls presented a challenge. In addition to specifying an acceptable range of values, DJJ also wanted to be able to specify the ideal value. The ideal value along with an assigned weighting determines how well a program matches. The range sliders were modified to allow users to specify an ideal value by clicking on it. Yellow stars mark the ideal values and black vertical lines mark the values of the selected program.

The criteria that should be used when choosing an appropriate program are still an area of debate. DJJ is also interested in extending the ProgramFinder to monitor program acceptances and rejections. IVEE could be used to analyze the data and determine the kinds of programs that are needed.

CONCLUSION

The NISYS project is an example of a successful collaboration among two state agencies—the University of Maryland working with the Maryland Department of Juvenile Justice. The University has been able to offer DJJ objective solutions to their user interface problems, unbiased by commercial agendas. By demonstrating our exploratory prototypes early on, we were able to help motivate DJJ to design a new system. We also acted as instructors and facilitators - teaching DJJ the basics about graphical user interfaces, networking, and database design, in order to put them in a better position to manage the implementors of NISYS.

The academic-industry partnership between HCIL and Cognetics is the other contributing factor to this project's success. There are two main paths in the life cycle of a user interface technique: one that is more research oriented and one that is more commercial. While Cognetics follows the product path and works on the overall product design, HCIL can focus on the research issues and explore key functional areas, like LifeLines. Working on real problems generates ideas and participation in the commercialization of these ideas helps refine the principles and produce more general designs. Three of HCIL's research prototypes have been refined and integrated into the NISYS design.

By including the user interface design in the RFP, DJJ hopes to better convey their needs to potential bidders plus the design can be used to hold implementors accountable for what they deliver. Designing the system up front will also help avoid implementation biases and ensure that the resulting system suits DJJ's needs. Many detailed decisions will still be left to the implementors so basic guidelines for making additions and modifications to the design will be included in the RFP.

ACKNOWLEDGMENTS

We thank Chris Cassatt (DJJ) and the Cognetics team, Charles Kreitzberg, Scott Gilkeson, and Jim Kauffman, for their contributions to the NISYS project. We also thank Ben Shneiderman for his early contributions to the project. The preparation of this report was supported by funding from the Maryland Department of Juvenile Justice.

REFERENCES

Ahlberg, C., Shneiderman, B., (1994), "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays," *Proceedings of Human Factors in Computer Systems '94 (Boston, Mass., April 1994)*, ACM, New York, 313-317.

Ahlberg, C., Wistrand, E., (1995), "IVEE: An Information and Visualization & Exploration Environment," *Proceedings of IEEE Visualization '95 (Atlanta, Georgia, October 1995)*, IEEE Computer Society, Los Alamitos, CA, 66-73.

Greene, S., (in preparation), "Information and Process Integration From User Requirements Elicitation: A Case Study of a Social Services Agency," HCIL Technical Report, University of Maryland, College Park, MD.

Kreitzberg, C., (1995), "Managing for Usability: The Cognetics Design Methodology," Cognetics, Corp., Princeton Junction, NJ.

Landsdale, M., (1988), "The psychology of personal information management," *Applied Ergonomics*, 19(1), Butterworth & Co. Ltd., 55-66.

Milash, B., Plaisant, C., Rose, A., (1995), "LifeLines: Visualizing Personal Histories," *HCIL 1995 Video Reports*, University of Maryland, College Park, MD.

Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B., (1995), "LifeLines: Visualizing Personal Histories," *Proceedings of CHI '96 (Vancouver, BC, April 13-18 1996)*, ACM, NY, 221-227.

Plaisant, C., Rose, A., "Exploring LifeLines to Visualize Patient Records," *HCIL Technical Report*, CAR-TR-819, University of Maryland, College Park, MD.

Rose, A., Shneiderman, B., Plaisant, C., (1995), “An Applied Ethnographic Method for Redesigning User Interfaces,” *Proceedings of DIS ‘95 (Ann Arbor, Michigan, August 1995)*, ACM, NY, 115-122.

Rose, A., Vanniamparampil, A., (1995), “Using Dynamic Queries for Youth Services Information,” *HCIL 1995 Video Reports*, University of Maryland, College Park, MD.

Sawyer, P., Mariani, J., (1995), “Database systems: challenges and opportunities for graphical HCI,” *Interacting with Computers*, 7(3), Butterworth-Heinemann, 273-303.

Shneiderman, B., Plaisant, C. (1994), “The Future of Graphic User Interfaces: Personal Role Managers,” *People and Computers IX, British Computer Society’s HCI ‘94 (Glasgow, Scotland, August 1995)*, 3-8.

Shneiderman, B., Rose, A. (1996), “Social Impact Statements: Engaging Public Participation in Information Technology Design,” *Proceedings of Computers and the Quality of Life ‘96 (Philadelphia, PA, February 14-15, 1996)*, ACM, NY, 90-96.

Slaughter, L., Norman, K., Shneiderman, B., (1995), “Assessing Users’ Subjective Satisfaction with the Information System for Youth Services (ISYS),” *Proceedings of Third Annual Mid-Atlantic Human Facotrs Conference (Blacksburg, VA, March 26-28, 1995)*, 164-170.

Williamson, C., Shneiderman, B., (1992), “The Dynamic HomeFinder: Evaluating Dynamic Queries in a Real-Estate Information Exploration System,” *Proceedings of ACM SIGIR ‘92 (Copenhagen, June 21-24, 1992)*, ACM, New York, 338-346.

Vanniamparampil, A., Shneiderman, B., Plaisant, C., Rose, A., (1995), “User Interface Reengineering: A Diagnostic Approach,” *HCIL Technical Report*, CAR-TR-3459, University of Maryland, College Park, MD.