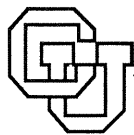PARTICIPATORY DESIGN
in the DEVELOPMENT of a
SCIENTIFIC SOFTWARE SYSTEM

Kristina D. Mickus-Miceli, Gitta O. Domik

CU-CS-561-91 December 1991

University of Colorado at Boulder

DEPARTMENT OF COMPUTER SCIENCE

PARTICIPATORY DESIGN

in the DEVELOPMENT of a

SCIENTIFIC SOFTWARE SYSTEM

Kristina D. Mickus-Miceli, Gitta O. Domik

CU-CS-561-91   December 1991

Department of Computer Science
University of Colorado at Boulder
Campus Box 430
Boulder, Colorado 80309-0430 USA

(303) 492-7514
(303) 492-2844 Fax

PARTICIPATORY DESIGN IN THE DEVELOPMENT OF A SCIENTIFIC SOFTWARE SYSTEM

Kristina D. Mickus-Miceli* and Gitta O. Domik

Department of Computer Science
University of Colorado
Boulder, CO 80309-0430
(303) 492-4062, domik@cs.colorado.edu

ABSTRACT
This paper describes a case study in the development of a scientific data analysis system performed at the Center for Astrophysics and Space Astronomy (CASA) at the University of Colorado, Boulder. The development of STAR, a Scientific Toolkit for Astrophysical Research, was performed in the environment of the user and provides insight into the design methodology known as participatory design. Experiences related to this design methodology are detailed in hopes of learning more about how effective software systems may be created. Techniques for evaluating the system throughout the development cycle and enhancing the participatory design process are also discussed so that future work may benefit from these experiences.

KEYWORDS
Participatory Design, User Interface Design, Data Analysis Software

INTRODUCTION
This case study describes the development of STAR, A Scientific Toolkit for Astrophysical Research, an interdisciplinary project between astrophysicists and computer scientists at the University of Colorado, Boulder. The goal of this project was to develop a successful software system for astrophysical data analysis that would bring together the available resources under one common, easy to use interface. This paper presents the basic concepts surrounding the notion of participatory design between astrophysicists and computer scientists (hereafter referred to as scientists and designers), how this project addressed these concepts and the lessons learned along the way. Techniques for designing the system from the onset of the project to its current status as well as methods for evaluating the results are presented. The early part of this paper includes a necessary description of the environment of the scientist so that the complex software environment constraining design and development issues may be understood. The project and its goals will then be detailed followed by the methods used to ensure a collaborative, iterative design. The final sections describe the required cost of participatory design as applied to our case and an outlook to future developments on the still ongoing project.

THE SCIENTIFIC ENVIRONMENT
The Center for Astrophysics and Space Astronomy (CASA) at the University of Colorado hosts fifteen scientists and approximately the same number of graduate students. The bond between their various research interests is established by large amounts of data acquired from space and ground based observations together with a network of hardware and software to manipulate these data. Images, spectra, tables and text, containing information from various wavelengths and sources, compose the greater portion of CASA's database. Scientists require access to all available databases to ensure that a complete inventory of information is presented to them from all available sources. Following identification and retrieval of data, scientists find that preprocessing is often necessary to deal with noisy data and to remove and correct instrumental effects. Subsequent numerical calculations, often in the form of statistical analysis, together with visual and interactive data processing, constitute the most significant aspects of scientific data analysis.

Most of the software used for the identification, retrieval and preprocessing of data items has been developed by CASA's scientists and staff. In addition, public domain software is available, supplying most of the numerical and visual analysis modules. Due to the various characteristics of space and ground sensor data from different wavelengths ranges (i.e. radio, infrared, visible or x-ray), different software packages exist and are used by CASA's scientists. Examining astrophysical objects from these different sources is termed multisensor/multispectral data analysis. This type of data analysis increases the scientist's insight into the physical properties of observed objects. However, the scientists have been hampered by the complex software environment created by the simultaneous use of various distinct software packages, each with its own learning curve.

* Current Address is NASA Ames Research Center, MS T045-1, Moffett Field, CA 94035. (415) 604-3959. kmiceli@nas.nasa.gov

THE GOAL OF THE PROJECT

Starting January 1990, CASA received a two year funding award to enhance the capability of scientists to analyze data from different sources (space-borne and ground-based observations) through interactive visualization. Interactive visualization would link the scientists directly to their data for immediate decisions, reactions, manipulations and allow merging of data derived from different sources.

Criticism on the inflexibility of public domain software and incompleteness of in-house developed software led to the decision of attempting a participatory design to concentrate on the scientist's desires in developing the new software. The design team mentioned throughout this paper consists of the two authors. Mickus-Miceli, a computer science graduate student, developed the system to its current status. Domik managed and integrated all aspects of the project to make it a useful experience from the aspects of user interface design, computer graphics and astrophysical data analysis.

PARTICIPATORY DESIGN

The design of software systems is an active area of research [4] and novel software engineering strategies and user interface design techniques have been developed where the help of the user is a key ingredient. The Scandinavian approach, which is linked almost synonymously with participatory design, emphasizes active user participation throughout the design process as well as an emphasis on a "process-oriented" vs. "product-oriented" perspective.

> The product-oriented perspective regards software as a product standing on its own, consisting of a set of programs and related defining texts. The process-oriented perspective, on the other hand, views software in connection with human learning, work and communication taking place in an evolving world with changing needs [8].

The Scandinavian principles have been quite successful and their influence in other parts of the world has become widespread. Blomberg and Henderson define the participatory design approach as "advocating three tenets which influence the character of the interaction between designers and users: 1) the goal should be the improvement of the quality of work life of the users; 2) the orientation should be toward collaborative development; and 3) the process should be iterative [1]." The participatory design nature of STAR reinforced these tenets by working with scientists to develop a system that suits their needs. This involved establishing a relationship with the scientists, maintaining daily communication, listening to ideas and problems and reacting to them quickly. Finally, the iterative development of the system consisted of a prototype /evaluation cycle that involved continual refinement throughout the development process.

Improving the Quality of Work Life

In order to make computer aided data analysis less frustrating and more productive, scientists at CASA expressed their desire for a system that was easy to use and offered them functionality without hindering performance. This functionality included incorporating available resources under one common interface, such as software packages, existing software modules and new modules aimed at filling in some of the deficiencies (i.e. visual analysis tools). Scientists believed that if a coherent software package could accomplish this, a great deal of time would be saved and replaced by research.

In the initial stages of development, scientists expressed the problems they had encountered with previously developed software packages, especially the amount of learning that was required to use the system. They were also frustrated with the difficulties is accessing these software packages, transferring data between packages and by the lack of routines to display their data for visual interpretation. After discussing these problems with scientists, the designers implemented the following solutions.

To reduce the complex software environment, buttons and knobs reflected the availability of software packages. To travel between the various packages, a data conversion package was installed. A common data format recognized by the astronomical community, the Flexible Image Transport System (FITS) [7], helped to solve this problem. All of the software could be tied together if the ability to store and transfer data in FITS format was available. Developing this tool involved learning about the FITS format and creating software that could perform the data input and output. This capability has been commented on as one of the most useful features of STAR. It is an important function that improved the working environment. Figure 1 shows both of these features; the square buttons representing the software packages are located in the bottom right corner of the interface (i.e. IDL, AIPS, IRAF, SAOImage[1]) and the FITS data conversion routines are displayed in the menu boxes.

As far as visual data analysis, scientists suggested new tools that they felt were missing and would be a quality contribution to their interaction with data. Simple visual analysis tools such as a color table manipulator, an image scaling routine, profiling, contour maps, perspective projections and several measurement tools were added on request. Scientist's also requested the interface style to be one of direct manipulation. They were interested in being able to interact with their data directly. In addition, they felt this would be the most effective way to attract scientists to take advantage of the software. Figure 2 is an example of a simple tool that allows scientists to change the color tables interactively.

---

[1] IDL - Interactive Data Language [11]
AIPS - Astronomical Image Processing System
IRAF - Image Processing and Analysis Facility
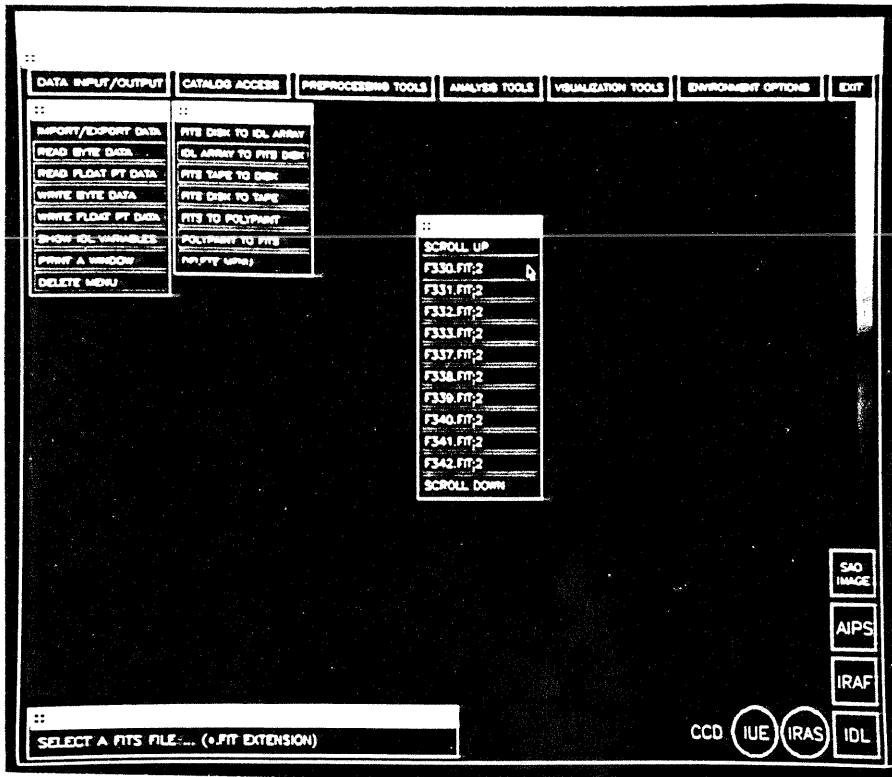SAOImage - Solar Astronomy Observatory Image Processing System

Figure 1: The available software packages in STAR are shown in the boxes in the lower right corner (i.e. IDL, IRAF, AIPS, SAOImage) Menus are currently displaying data transfer via FITS format.
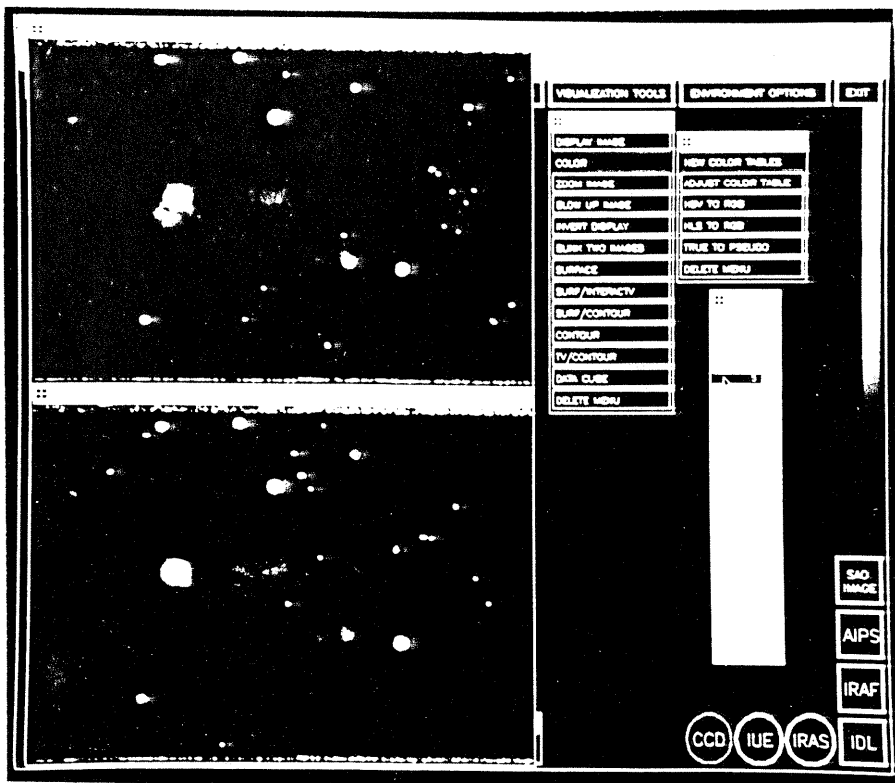

Figure 2: A color table routine allows scientists to interactively switch color tables.

Performance was not compromised with the addition of these new features. The connections between modules and interface was simple and without a great amount of overhead.

Collaborative Development

The central focus in the development of the STAR system was a constant working relationship between the scientists and designers. This relationship was essential to ensure that the system was designed to fit the needs of the scientist. Keeping the communication channel between scientists and designers open was another important aspect of the collaborative project. The development process involved informal communication such as daily interaction regarding the functionality of the system as well as the look and feel of the design. Formal communication was also a part of the development process and it included events such as user interviews, evaluation sessions and demos.

One of the most important steps in developing a collaborative system was to establish a relationship with the scientist. In order to establish this relationship, the designers felt it was necessary to introduce themselves and the concept of STAR to the group. This was performed by offering the scientists a new way to analyze their data in a system that would take into account their input. These interviews involved four scientists with different computer expertise. At the time of the interviews, specific tools and interface styles were discussed to focus on concrete user situations rather than general ideas. The designers showed the scientists some sketches they created and expressed some ideas about the design. After this initial exposure, the scientists gave suggestions and comments. Both low and high level suggestions were made. For example, low level suggestions included how the interface might look, such as the placement of buttons, text windows, and menu boxes. High level suggestions gave designers a clear understanding of what scientists want and expect from a software system. These issues are listed below and were held in high regarding throughout the development process:

*   Easy to learn and use software, unlike previous software packages which involved a slow learning curve;
*   Appealing functionality, which lets scientists look at their data in new ways;
*   Visual analysis functions, including tools to enhance multisensor/multispectral data analysis;
*   Rapid results, including the ability to interact with data in real time;
*   Capability to integrate new software (extensibility), which will allow scientists to integrate their own personal code into the interface;
*   Flexible, "customizeable" software that will allow scientists to adjust the system to their needs;
*   Online help that eliminates the need for outside sources such as manuals.

Becoming part of the team at CASA was also important for the collaboration between scientists and designers to be effective. This meant learning the environment of the scientists as well as each individual's work interests. The workstation dedicated to the development of the STAR system was located in the main computing lab at CASA and, as a result, interaction between scientists and designers was on a daily basis. Scientists were always willing to talk about their work, especially if it included the discussion of different ideas and the development of new tools to perform their data analysis. This interaction was comfortable since each participant felt that they were part of the team effort to develop a better software environment.

To enhance communication between scientists and designers, meetings were held throughout the development process. Formal meetings, like the initial user interviews mentioned in the previous paragraphs, were arranged with scientists to discuss data analysis needs. Informal meetings were frequently held in the main computing facility, where designers attempted to make themselves available as much as possible. These meetings often involved showing the scientist a new function and asking for input. This allowed designers to get impressions right away and implement any necessary changes. Ideas and suggestions were also voiced by the many different scientists that came to the computing site.

When designers were not available, communication was also made possible by providing a menu item in the STAR interface that allowed the scientists to express their opinion or voice any problems or concerns. This menu item, labeled "PROBLEMS", allowed the scientist to write a short note to the designers about their concerns. The note was forwarded to designers who read it the next time they were in the computing lab.

A poster board, added to the computing site, helped gain attention for the development of STAR. This poster board, complete with color pictures and text, was placed in the entry to the computing site for everyone to see. Those not familiar with STAR could read about the system and become familiar with the project. This encouraged on the spot demos as well as quality interaction between scientists and designers about what features scientists desired in the system. This technique, similar to the 'storefront approach' [9], has proven to be very beneficial while involving only a small investment of time.

The STAR system was also presented at astrophysical meetings in the form of a poster paper (American Astronomical Society's (AAS) 1990 Summer Conference, Albuquerque, NM and AAS 1991 Winter Conference, Philadelphia, PA). This provided the designers with further insight into how such a system might be accepted in the astrophysical community. Effectively dealing with the complex software environment gained much more attention and positive reaction than the visualization tools themselves.

These aspects of the development of the system, together with the daily informal contact made possible by using the

same lab and equipment, emphasized a collaborative effort between the scientists and designers.

Iterative Design
The STAR design process was iterative in nature, involving system prototypes and user evaluations that led to continual improvement of the software. Frequent demos and user evaluations, both formal and informal, took place after the initial prototype was developed and have continued throughout the design process.

The first prototype was based on a typical research scenario prepared jointly by scientists and designers. The scenario consisted of several representative functions in each step of the data analysis cycle, namely data access, preprocessing, numerical/statistical analysis, and visualization. The scenario was effective in quickly implementing the initial prototype in that it gave the system a framework to build upon. In addition, the scenario supplied scientists with a working prototype of tools to support their data analysis. After this prototype was developed, a meeting was held to officially introduce the STAR system available for use to the scientists of CASA. Formal user evaluations, in the form of Thinking Aloud sessions [12], were performed after the initial introduction to several scientists.

The Thinking Aloud approach, attributed to psychologists Newell and Simon [16], is based on cognitive theory. This theory maintains that in order to develop an effective interface, the informational aspects of the system (i.e. menus, messages, manuals) must fit the thinking patterns of the user. In other words, users should be able to progress through a given series of tasks if the information presented to them was cognitively correct. The detailed observation of the user performing tasks looks at how problem solving is accomplished. The results of Thinking Aloud experiments lead to areas in the interface where confusion results. These problems can then be handled in the next version of the iterative design process.

A list of tasks that covered many of the functions in STAR was provided for scientists who volunteered to perform a Thinking Aloud session. The tasks included performing catalog searches for data, accessing data, preprocessing data, displaying data and performing various data manipulation functions. The scientists performed the scenario and provided criticism about the system, including comments on the appearance of the system as well as its usability. A sample Thinking Aloud session is outlined below.

The subject is an undergraduate physics major and has experience in processing satellite images obtained from NASA. She mainly preprocesses and prepares data for scientists. The subject had the following comments about STAR:

* In some instances the subject was unsure if she should use the keyboard or the mouse to enter data,

* The subject did not use the DATA button to determine what data she had read in,
* The subject was unsure if she performed a certain task successfully. The system did not give her feedback when a task was completed.

The results of this Thinking Aloud session prompted the designers to:

* Concentrate on being consistent when data input is required,
* Remove the DATA button from its original place (which went unnoticed by ALL those participating in the Thinking Aloud experiment) and put it in as a menu option,
* Provide a STATUS window to tell the user when and if an operation was completed successfully.

The first prototype system was evaluated using a formal method, namely the Thinking Aloud method. Throughout the rest of the design cycle, informal evaluation methods were used in a more "fine-grained" approach. The "fine-grained" approach involved presenting the user with a new small segment of the software as soon as it was developed. The evaluation was informal in nature, often involving only a short demo and corresponding evaluation. The scientists would give their impressions and suggestions for change or enhancement. This quick response time would allow the designers to fix any problems with a function immediately. This process avoided unnecessary work in developing functions that might have been encountered if the software was evaluated in larger segments.

OBSERVATIONS FROM THE PARTICIPATORY DESIGN PROCESS
During the development of STAR, insight has been gained into the benefits and disadvantages of participatory design. This section details how design techniques affected the development of STAR.

All stages of STAR's development emphasized interaction between designers and scientists on a personal level. Involvement in design allows the scientists to participate in important decisions that contribute to the compatibility of the system to their needs. However, the small scale development effort did not permit the designers to fulfill the needs of each scientist. Each time the scientists were presented with a new tool, they wanted something new added to it. They often become disappointed that the integration of their new feature could not be done easily or in a short amount of time. Designers were overwhelmed with requests from scientists who wanted new tools that would benefit their research. The small team of two designers, only one of whom performed most of the coding, was not enough to create the many and varied software tools that scientists were looking for. Rather, the designers had to concentrate on general tools that would benefit a large cross section of CASA's scientists. The participatory design process requires that development teams remain small, in order to remain familiar with the entire system and

to ensure that communication with the user is maintained. These frustrations are not uncommon for designers and users in a participatory design setting as Scandinavian participatory design projects have also expressed similar concerns [18].

The collaborative nature of the project has encouraged discussion on the effectiveness of the system. Discussions between designers and scientists were frequent, mostly because the development of STAR was performed in the environment of the user, CASA's main computing lab. However, the time spent on discussing the system with scientists often limited the time available to implement the software tools. Working in the environment of the user provided the designers with a great amount of user input. Although this would seem to be an ideal situation, it sometimes proved to be overwhelming to the small design group.

The research scenario approach, mentioned earlier, was an effective way to begin the development of STAR. However, as time progressed, the scenario limited the evaluation as it was useful to some, but did not interest the complete cross section of scientists. For future reference, the designers of STAR believe that the use of a more general research scenario or multiple research scenarios would be more advantageous in the prototype development stages of a software system.

Other problems with the computing environment at CASA jeopardized the participatory nature of the project. The development environment chosen for the project was a color workstation of type VAXstation 3100/38, based on the X-windowing system. Not very many scientists were exposed to STAR because of lack of hardware. The few X-window based workstations that are available to scientists are usually occupied and, as a result, scientists are unable to use the system. The limited development environment (Interactive Data Language [11]) was sufficient to develop prototype code, however, it prohibited the designers from reaching adequate solutions to integration problems. Software engineering issues related to the integration of existing software were hampered by the prototyping environment. The solution to these problems required a great deal of time and, as a result, other issues in the development of STAR, such as the visualization tools, suffered.

The implementation of the "PROBLEMS" menu item, as described in a previous section, was introduced through recommendations by designers and scientists, however, it was rarely used, because the designers were almost always around to personally answer questions and receive suggestions. This method of communication was preferred by scientists as compared to communicating through messages and meetings.

Finally, one last fundamental problem influenced the success of the STAR project. Tradition and momentum have made scientists reluctant to use any new systems that become available. Scientists are comfortable with their current method of data analysis and are very hesitant to switch. This has been one of the biggest barriers in the development cycle of STAR, specifically, having scientists use the system on a regular basis. It was evident that the designers must develop attractive software tools in order to convince scientists to try something new. In addition, they must develop these tools while under direct consultation with the scientist. With this in mind, more visualization tools must be created throughout the development of STAR to present data to scientists in a different, and hopefully, more advantageous, manner.

THE COST OF PARTICIPATORY DESIGN
Originally, the STAR system was envisioned to be built by designers in a limited timeframe and then released to the users. The designers had planned for the system to be self-supporting, giving scientists the ability to add new software as it was developed. However, the participatory design experience has led designers to the conclusion that, even after many iterations in the development cycle, the system cannot be given to the scientists without additional support. Too many technical issues arise that scientists do not want to personally deal with. The iterative cycle never really has an end; scientists continually come up with new problems to solve and new types of data to display.

The participatory design process has proven to be a more expensive operation, as far as time is concerned, then designers had originally expected. The additional overhead required in communicating with users puts a limit on the time spent on developing the software. This has resulted in the delay of producing a "final" product.

A rough time estimate is given to judge the demands that participatory design puts on the design process. In the first stages of development of STAR, one-hundred percent of the time was devoted to user interaction and exploring what scientists wanted from a software system. The first two months of the project were spent interviewing users, coming up with initial designs and creating a prototype system. After this process was completed, the amount of user interaction dropped to about forty percent; approximately half of this time was spent with scientists discussing STAR and its functionality, the other half can be attributed to discussing new ideas and assisting scientists with their data analysis. The remaining sixty percent of the time was spent on programming. The complete design/development time discussed in this report spanned roughly one and a half years. This time division is represented in Figure 3.

CONCLUSIONS AND THE FUTURE OF STAR
STAR is still an active project and more development will be necessary to continue the iterative design cycle. As time goes on, the system will become more comfortable and tailored to fit the needs and capabilities of the scientist. New visualization tools will be added to create a more interesting and productive environment for the scientist. Increased functionality, the capability for expandability and
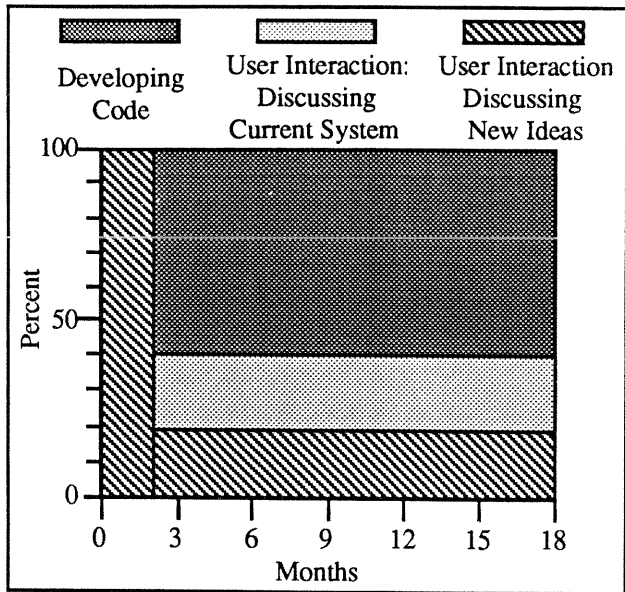
Figure 3: - Time Divisions in the Development of STAR

customization, and enhanced user interaction techniques will all be a part of STAR's future. Broadening STAR's user base will also be a priority as STAR will be ported to other systems (UNIX) for greater exposure.

The three tenets of participatory design were represented in the development of STAR by the three principles of collaboration, communication, and "fine-grained" iterative design. Establishing a collaborative relationship involved learning about the user and their work. Communication was essential throughout design stages, both formally in group meetings or informally with scientists at the computer. The "fine-grained" iterative design cycle encouraged designers to take advantage of the presence of the user and solicit as much input as possible. However, both benefits AND disadvantages of working in the environment of the user were uncovered during the development of STAR. The small scale prototypinng team often found themselves overwhelmed by the larger group of enthusiastic scientists. Finally, the tradition and momentum that has developed in the environment of the user was found to be a barrier. Scientific software must offer scientists a great deal of functionality in a system that is easy use if it is to be effective.

ACKNOWLEDGEMENTS

REFERENCES

1. Blomberg, Jeanette L. and Henderson, Austin. Reflections on Participatory Design: Lessons from the Trillium Experience. In Proceedings of ACM SIGCHI Conference on Computer-Human Interaction (Seattle, WA, April 1-5, 1990). ACM, New York, pp. 353-359.

2. Brittain, Donald L., Aller, Josh, Wilson, Michael, and Wang, Sue-Ling C. Design of an End-User Data Visualization System. In Proceedings of IEEE Visualization '90 Conference (San Francisco, CA, October 23-26, 1990). IEEE, Los Angeles, pp. 323-327.

3. Brugel, Edward W., Domik, Gitta, and Ayres, Tom. 1989 NASA Proposal: Visualization Techniques to Aid in the Analysis of Multi-Spectral Astrophysical Data Sets. University of Colorado, Center for Astrophysics and Space Astronomy, Boulder, CO, 80309-0391.

4. Chew, Jane Carrasco and Whiteside, John. Proceedings of ACM SIGCHI Conference on Computer-Human Interaction (Seattle, WA, April 1-5, 1990). ACM, New York.

5. DeFanti, Thomas A., Brown, Maxine D., and McCormick, Bruce H. Visualization: Expanding Scientific and Engineering Research Opportunities. IEEE Computer, August 1989, pp. 12-25.

6. Domik, Gitta and Mickus, Kristina D. Visualization in the Analysis Cycle of Observational Data. In Proceedings of Gesellschaft fur Klassifikation (Salzburg, Austria, February 25-27, 1991).

7. Flexible Image Transport System, Draft Standard, December 1990 (NSDSSO 100-0.1). NASA Science Data Systems Standards Office, Code 933, NASA Goddard Space Flight Center, Greenbelt, MD, 20771.

8. Floyd, Christiane. Outline of a Paradigm Change in Software Engineering. In Computers and Democracy. Bjerknes, G., Ehn, P. and Kyng, M, editors. Avebury Press, England, 1987, pp. 193-202.

9. Gould, John D. How to Design Useable Systems. Technical Report. IBM Research Center-Hawthorne, Yorktown Heights, NY, 10598, 1987.

10. Gould, John D. and Lewis, Clayton. Designing for Usability: Key Principles and What Designers

Think. In Readings in Human Computer Interaction: A Multidisciplinary Approach. Baecker, Roland M. and Buxton, A.S., editors. Morgan Kaufmann Publishers, Inc., San Mateo, CA, 1987, pp. 528-539.

11. IDL (Interactive Data Language) Version 2.0. Research Systems, Incorporated. Boulder, Colorado, March 1990.

12. Lewis, Clayton. Using the "Thinking Aloud" Method in Cognitive Interface Design. Technical Report. IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 10598, 1982.

13. Lewis, Clayton. A Research Agenda for the Nineties in Human-Computer Interaction. Paper representing a talk given at the Winter Conference of Human-Computer Interaction Consortium, Vail, Colorado, January 22-24, 1989.

14. McCormick, Bruce H., DeFanti Thomas A., and Brown, Maxine D. Visualization in Scientific Computing. Computer Graphics (special issue), vol. 21, no. 6, November 1987.

15. Mickus, Kristina D. and Domik, Gitta. Designing User Interfaces for Applications in Scientific Computing-STAR: A Scientific Toolkit for Astrophysical Research. Center for Astrophysics and Space Astronomy, University of Colorado, Boulder, CO, 80309-0391, 1990.

16. Newell, A. and Simon, H.A. Human Problem Solving. Englewood Cliffs, NJ, Prentice Hall, 1972.

17. Nygaard, Kristen. Program Development as a Social Activity. In Proceedings from Conference on Participatory Design (Seattle, WA, March 31-April 1). Computer Professionals for Social Responsibility, Palo Alto, CA, pp. 4-13.

18. Thoresen, Kari. Prototyping Organizational Changes. In Proceedings from Conference on Participatory Design (Seattle, WA, March 31-April 1). Computer Professionals for Social Responsibility, Palo Alto, CA, pp. 22-33.