# Developing Calendar Visualizers for the Information Visualizer

Jock D. Mackinlay, George G. Robertson, and Robert DeLine
Xerox Palo Alto Research Center
3333 Coyote Hill Rd; Palo Alto, CA 94304
<mackinlay@parc.xerox.com>

**ABSTRACT**

Increasing masses of information confronting a business or an individual have created a demand for information management applications. Time-based information, in particular, is an important part of many information access tasks. This paper explores how to use 3D graphics and interactive animation to design visualizers that improve access to large masses of time-based information. Two new visualizers have been developed for the Information Visualizer: 1) the Spiral Calendar was designed for rapid access to an individual's daily schedule, and 2) the Time Lattice was designed for analyzing the time relationships among the schedules of groups of people. The Spiral Calendar embodies a new 3D graphics technique for integrating detail and context by placing objects in a 3D spiral. It demonstrates that advanced graphic techniques can enhance routine office information tasks. The Spiral Calendar development process involved three major phases: 1) progressive design based on our experience using advanced graphics for user interfaces, 2) an implementation/evaluation cycle based on a new method for characterizing information access from dynamic displays, and 3) reuse of the spiral technique for a decision support visualization where the spiral helps the user keep track of decisions while designing the layout of aircraft cockpit controls. The Time Lattice, on the other hand, has only been informally evaluated. However, it illustrates how different tasks require different advanced graphics technology. In particular, the Time Lattice uses of 2D transparent shadows to provide interactive access to a complex 3D object. Our experience developing these visualizations should prove useful to others developing user interface that use advanced graphics.

**KEYWORDS:** Information Visualization, Graphical Representations, Information Retrieval, Detail+Context Technique, Interactive Animation, 3D Graphics, Calendars, Translucent Shadows

**INTRODUCTION**

A vast amount of useful information is rapidly becoming available to businesses and individuals via technology such as CD-ROM disks and high-speed networks. These increasing masses of information have created a demand for information management applications. Time-based information, in particular, is an important part of many information access tasks. It captures the patterns of work and provides a spanning property for accessing information [7]. This paper explores how to use 3D graphics and interactive animation to improve access to time-based information. It is part of a larger effort to develop a research prototype called the Information Visualizer (IV), which uses emerging graphics technology to develop visualizers for various types of information [9]. The goal of the IV research is to tap human perceptual abilities to increase both the volume and rate of information work. One IV research strategy has been to design visualizers for a variety of different types of information structures. We focus here on calendar information because it is a ubiquitous but routine part of most people's information work. We have developed two new visualizers for IV: 1) the Spiral Calendar for rapid access to an individual's daily schedule and 2) the Time Lattice for analyzing the time relationships among the schedules of groups of people. Our experience developing these visualizations should prove useful to others developing user interface that use advanced graphics.

Although a routine part of information work, time-based information can involve tasks that are facilitated by visualization techniques. For example, a group of people often have complex time relationships. Even a simple task such as scheduling a group meeting can involve many events and constraints. Tasks involving a single daily schedule can also benefit from visualization techniques. For example, a person might need to access the previous couple of years to generate a bill, justify a raise, or plan future activities, which could easily involve more than 1000 days and many thousands of events. Timelines of such information have wide aspect ratios that do not easily fit on paper or computer displays. A more common presentation technique is to break the linear nature of time into a hierarchy of calendars such as year, month, and day-at-a-glance. This hierarchy captures daily and seasonal cycles that can be quite useful for scheduling events. For example, a 1985 study by Kincaid et al of office worker use of paper and electronic calendars indicated that workers typically used at least two different types of paper calendars to support their scheduling tasks [6]. A person might use a day-at-a-glance calendar for daily events and a monthly calendar for larger events such as trips out of town. Since this hierarchy of calendars is useful and familiar, it represents a challenge for designing visualizers of time-based information because designs that depart radically from the hierarchy will probably be confusing.

Electronic calendars are slowly becoming more desirable than paper calendars. In 1985, the Kincaid study determined that office workers preferred their paper calendars because electronic calendars had limited flexibility. The study listed functionality that would make electronic calendars more desirable. A 1993 MacWorld product review article indicates that current electronic calendars provide this functionality, including multiday events, recurring events, priorities, to-dos, and automatic reminders [11]. Current electronic calendars should become even more desirable as portable computers let them be carried where paper calendars are carried. Previous research suggests other ways electronic calendars might become more desirable. One possibility is to use natural language to access electronic calendars [2]. Another possibility is to use visualization techniques. For example, Beard et al developed a research prototype designed to support the scheduling of group meetings by using transparency to blend together the calendars of a group so that lightly scheduled times would be visible [1].

Although electronic calendars have been improving, they use the same hierarchy of calendars that were developed for paper without particularly exploiting the new medium for visualization. They slightly augment the hierarchy of calendars by providing hypertext interactivity. For example, a user can click on a month in a year calendar and switch to the appropriate month calendar. However, the result is a maze that must be navigated by the user. A key question when designing calendar visualizers is how to use the new medium to design visualization techniques that support the user's navigation through the calendar hierarchy.

Previous research on IV suggests that advancing graphics technology can be used in a number of ways to design calendar visualizers [9]. 3D graphics can be used to increase the density of information that can be placed in a given area of a computer screen. Interactive animation can be used to increase the real-time interaction between a user and an electronic calendar. A "highly coupled" system can provide the user with immediate feedback and animate transitions for continuity. Animated transitions tap into human perceptual ability so that the user does not have to cognitively reassimilate the scene when it changes. Finally, visual abstraction can shift information to the perceptual system to speed information assimilation and retrieval. In particular, details can be viewed within their context, making them more meaningful.

This paper describes how we used IV approach to develop a new visualizer to support tasks associated with time-based information. The Spiral Calendar was designed for a common task associated with time-based information, which is the manipulation of an individual's daily schedule. The Spiral Calendar contributes a new 3D graphics technique for integrating detail and context by placing objects in a 3D spiral.

Our experience with using the Spiral Calendar suggested that the spiral visualization technique might be useful for other information management tasks. It was generalized to display a path through a directed acyclic graph, and applied to the problem of keeping track of decisions made in a complex decision graph. The particular application was the design of aircraft cockpit control layout. In addition to describing the Spiral Calendar, we will describe the application of the same techniques to this decision support task.

The Time Lattice, on the other hand, was designed for the analysis of the large amounts of information found when multiple calendars are compared. Although not a common task for current paper and electronic calendars, we believe this application will become more important as portable computers encourage the accumulation of such rich calendar information. The Time Lattice uses 3D graphics to construct a 3D object by layering 2D calendars. Its key feature is the use of 2D transparent shadows to provide interactive access to this complex 3D object. Ultimately, we believe that information visualization will involve a multiplicity of visualizers for the same information, each designed for a different information task.

## PROGRESSIVE DESIGN OF THE SPIRAL CALENDAR

The design phase for developing the Spiral Calendar proceeded by asking how advanced graphics could be used to improve access to an individual's daily schedule by adding 3D interactive animation to conventional calendar presentations including tabular layouts, timelines, and the hierarchy of day, week, month, and year calendars. Our design strategy was to determine whether various advanced graphics techniques that had been used in other IV visualizers would be used to augment these presentations. We developed the following progression of designs during this analysis:

### Design 1: Timelines

We first considered timelines because they map directly onto the Perspective Wall visualizer, which was described at CHI'91 [7]. The Perspective Wall was designed to fold 2D presentations with wide aspect ratios, such as timelines, into 3D perspective. Color Plate 1 shows how we have extended the Perspective Wall to visualize a class schedule timeline. The vertical layers represent class instructors and color represent who is taking the class. However, timelines do not reveal the daily and seasonal patterns normally found in daily schedules.

### Design 2: Pan/Zoom Calendar

Since the hierarchy of calendars nests one inside the other, we reasoned that interactive animation might allow the user to quickly zoom and pan over these nested views in an interface similar by Perlin's PAD system [8]. Figure 1 shows a sketch of a zoomed view of this design. A prototype im-

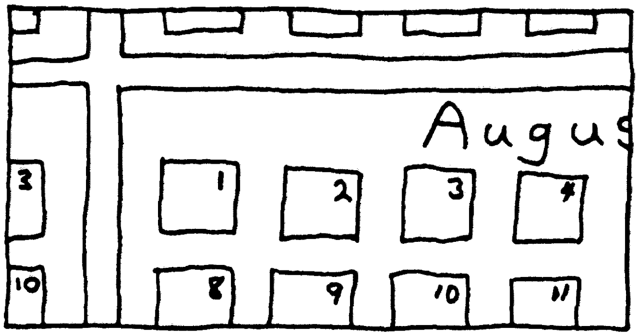



Figure 1: Sketch of a Pan/Zoom Calendar design. Useful global context is lost as the user zooms in to the homogeneous details.

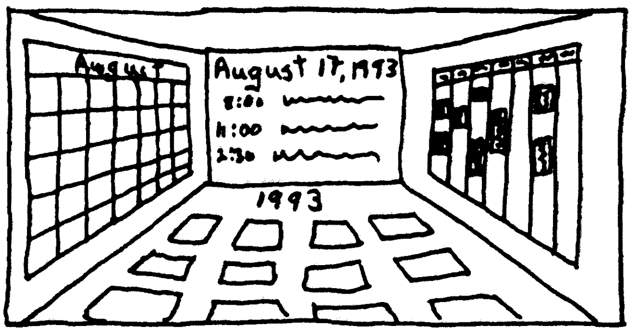



Figure 2: Sketch of a Calendar Room design. All calendars are visible but their relationships are not shown and the text on the walls and floor is foreshortened.

plementation revealed that zooming tends to lose important global context. Unlike PAD where details could be identified by their graphical appearance, we found that days tended to look alike, particularly with repeated events. When everything looks the same, pan/zoom navigation is a cognitive rather than a perceptual task.

**Design 3: The Calendar Room**

We next considered how 3D could be used to increase the density of the visualization to maximize the use of screen space. Figure 2 shows a sketch where the various calendars are placed into a room configuration. This design suffers from a number of problems. The side walls of a calendar room are in extreme perspective, which results in foreshortened text that can be hard to read. The back wall is the natural place to put important information but it is distant from the viewpoint, which makes the text smaller. Finally, the design does not show the connections between the various calendars, which makes navigation more difficult.

**Design 4: Tower Calendar**

Our next design, called the Tower Calendar, is sketched in Figure 3. Here we decided to use 3D to show the connections among calendars by stacking the calendars in a tower and connecting them using truncated pyramids. The calendars are tilted toward the user and visible through the

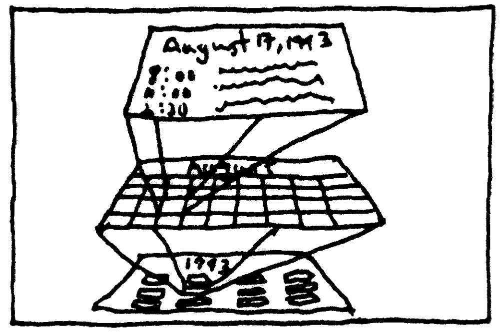



Figure 3: A sketch of a Tower Calendar design, which results in foreshortened text but does show the connections among calendars.

translucent pyramids. The major problem with this design is that the text on the calendars is very foreshortened and hard to read. We have found that text presentation is a critical part of effective design of 3D user interfaces. We applaud the trend towards improved texture mapping because it allows us to use high quality bitmap fonts rather than an ugly vector font. Improved 3D fonts are likely to open up new visualizations opportunities, particularly for routine office information texts, which involve a lot of text. However, even texture mapped fonts are hard to read when they are foreshortened. Furthermore, sampling artifacts become quite visible when tilting or zooming increase the size of texture mapped fonts.

**Design 5: Spiral Calendar**

Our final layout, called the Spiral Calendar, is shown in Color Plate 2. The Spiral Calendar uses 3D to place the various calendars in a receding spiral. The spiral lets the text face the user and also shows the translucent truncated pyramids connecting the calendars. Color Plate 2 extends the conventional hierarchy of calendars to include tables for decades, centuries and millenia. The design of the Spiral Calendar has a number of advantages. The 3D spiral makes efficient use of screen space because details are large and context is small but available for interaction. Perceptually, the user can interact with a familiar and engaging 3D object.

**Design 6: Spiral Calendar Behavior**

Finally, we designed the behavior of the Spiral Calendar. The user can click on the calendars to select different calendars for focused viewing. After a click, a yellow rectangle is drawn to give immediate feedback of the selection, and an animation is started to transform the Spiral Calendar. Animation supports object constancy, which lets human perception track changes to the Spiral Calendar rather than requiring the user to figure out cognitively the new layout. For example, an animation can show a day calendar growing out of a week calendar while the week calendar recedes with the rest of the spiral to make room for the day calendar. Connections can also be removed when the user wants to move to a larger temporal grain-size. For example, the user can select a month calendar. The resulting complex animation includes the discarded day and week calendars discreetly shrinking out of the user's view while the new month calendar grows out of the year calendar and the spiral moves forward to focus on the month calendar. IV animations are typically designed to take less than a second, to support a smooth, non-confusing transition without taking so long that it gets in the way of working time.

## IMPLEMENTATION/EVALUATION CYCLE

The development of the Spiral Calendar has been through two implementation/evaluation cycles. The first cycle implemented and evaluated a Spiral Calendar Viewer, which did not support the editing of calendar information. After addressing the issue raised by the first evaluation, the second cycle resulted in a Spiral Calendar Editor, which is now in daily use.

### Implementing the Spiral Calendar Viewer

The implementation of the Spiral Calendar Viewer was based on IV's object oriented architecture, which allowed it to be done as a summer student project. The viewer has been implemented to read the database files of the CM electronic calendar, which is a Sun-based electronic calendar that is widely used at PARC. We found that the perception of the spiral was enhanced when calendars moved directly to their final resting place rather than through intermediate positions. Discarded objects skrink and move to the lower left corner of the display so that the user is not distracted from watching the new calendar grow out of the spiral.

The Spiral Calendar Viewer incorporates techniques from cartoon animation to enhance its animated transitions. For example, as a new calendar moves out from a location on an old calendar, it uses slow-in/slow-out, which is a cartoon animation technique that starts and ends slowly and goes quickly in the middle. The slow start lets the user see the origin of the new calendar, the quick middle makes for a crisp efficient transition, and the slow end lets the user see the new calendar come to rest. Chang and Ungar describe cartoon animation techniques for a 2D application at UIST'93 [4].

The animated transitions in the Spiral Calendar are specified in a local scripting language developed by Mark Shirley. This scripting language supports both sequential and parallel components. It proved quite useful in our implementation. For example, we decided to remove text from the calendars during animated transitions to increase animation rates because text is very costly to draw. However, the blinking that occurred as the text disappeared and reappeared turned out to be distracting. Cartoon animation suggested that we could soften this blinking by fading the text. The resulting script fragment to accomplish this fade down and up is shown in Figure 4.

Our experience incorporating cartoon animation techniques into the Spiral Calendar indicates that scripting animations in an interactive system is complex. When we implemented text fading, we had to modify the script to accommodate the faster animation rate that occurred when the text disappeared. This faster rate caused one more frame to be drawn during "fade up"and during "fade down", which was visibly asymetric. We modified the durations of these fades in Figure 4 to make them appear symmetric.

```
(parallel-script
    (step-point (position calendar)
       start-point end-point
       1.0 slow-in-slow-out)
    (step-point (scale calendar)
       start-scale end-scale
       1.0 slow-in-slow-out)
    (sequence-script
       (step-value text-fade 0 255
          0.35 linear)
       (nop 0.50)
       (step-value text-fade 255 0
          0.15 linear)))
```

Figure 4: A script fragment used to fade text, which is a cartoon animation technique for smoothing transitions.

### Evaluating the Spiral Calendar Viewer

The evaluation of the Spiral Calendar Viewer included a new method for characterizing information access from dynamic displays [3]. The study compared the Spiral Calendar viewer with the Sun CM calendar on the direct-walk task of navigating from one date to another. Analysis of the data resulted in the Cost-of-Knowledge Characteristic Function, which plots the number of elements accessible as a function of the cost in time (see Figure 5). The study and informal use experience indicated that the Spiral Calendar provided an intuitive interface to an individual's daily schedule. However, the implementation had performance problems and missing features that made it undesirable for daily use. In particular, users had to wait on the order of 3.5 seconds between picks. Assuming an average mouse point of around 1 second and an animation intended to last less than a second, the average picking time should have been around 2 seconds. The animations were clearly taking too long. A user of CM was able to get to nearby dates much more quickly that with the Spiral Calendar Viewer. Fortunately, the intuitive interface of the Spiral Calendar partially mitigated these problems. Furthermore, the Spiral Calendar Viewer turned out to be better than CM for navigating to distant dates. All movement is done with a single operation, clicking on calendars. For comparison, navigation in the Sun CM calendar involves a combination of menu selection and button clicking. Finally, the evaluation method allowed allowed us to replot the Cost-of-Knowledge Characteristic Function to determine that improved animation rates would bring the Spiral Calendar Viewer closer to CM in navigation performance.

The Spiral Calendar Viewer also required the user to always go through the week display to access a day display while CM let the user access day displays directly for nearby days. A replot of the Cost-of-Knowledge Characteristic Function showed that this also gave CM the advantage for navigation to nearby dates. Since daily calendar use typically focuses on nearby future dates, we resolved to address this issue in the next implementation/evaluation cycle.

**Implementing and Evaluating the Spiral Calendar Editor**

The problems identified in the study of the Spiral Calendar Viewer were all addressed during the development of the Spiral Calendar Editor, which has now been in daily use for several months. Substantial functionality was added, including editing, searches, and automatic notification of appointments. We also integrated the Spiral Calendar Editor with CM so that users could use both editors at the same time.

The slow animation rate of the Spiral Calendar Viewer was addressed in a number of ways. We performance tuned the scripting system and the summer student's implementation to make the animations run is less than a second, which was our original goal. We also corrected a bug that kept the Spiral Calendar Viewer from responding to user clicks during animations. Videotape analysis indicated that slow-in/slow-out animation made it easy for users to attempt to speed the transition to the next level of the calendar hierarchy by clicking before calendar reached its final resting point. The bug appeared to have a noticeable effect on the Viewer' cost structure. Finally, we gave users the option of turning off the animation altogether. We reasoned that experienced users would see the spiral twist as a gestalt action and would not need the animation to maintain object constancy.

We also improved access to nearby dates, a common navigation task in a calendar editor. When the user is looking at the daily calendar, the weekly and monthly calendars can be used to select nearby day calendars without having to reselect the week calendar.

Finally, we repeated the evaluation study for the Spiral Calendar Editor. The repeated study only used one subject because the first study indicated there was almost no variation among subjects. We collected data for the Spiral Calendar Editor with and without slow-in/slow-out animation. The resulting Cost-of-Knowledge Characteristic Functions are also plotted in Figure 5. The Spiral Calendar Editor is clearly better than the Viewer for a direct walk task. Furthermore, it compares favorable with CM even though it much younger. CM's major advantage in the 10 to 100 range is that its daily calendar presentation includes three monthly calendars for selecting nearby dates. The Spiral Calendar Editor only provides one month calendar for nearby selecting. Finally, it is also worth noting that the animated transitions of the Spiral Calendar directly increase access cost. Our experience is that new users like the animated transitions because it helps them understand how the calendar is changing. Experienced users prefer lower access costs because they know how the calendar will respond to their selections. This suggests that animated transitions will be most effective when changes are not predictable by the user.

## DECISION SUPPORT VISUALIZATION

During early use of the spiral visualization, it became clear that it had other potential uses. One proposal was for a decision support visualization. The spiral visualization has been adapted to that task, and applied to the problem of keeping track of decisions while designed the layout of aircraft cockpit controls.

An aircraft designer has an enormous space of alternative designs possible for each subsystem in an aircraft. Design decisions potentially can interact is undesirable ways. In order to ensure that the resulting design is safe and effective, and can be certified by the FAA, complex decision graphs are built that show all the possible combinations of designs that lead to FAA certifiable aircraft. When a customer works with an aircraft manufacturer, they want to understand the context of the various design decisions they have made, be able to see the consequences, and be able to easily change decisions.

We obtained the decision graph for one aircraft subsystem: the flight controls layout. This is a directed acyclic graph with an average branching factor of three and a maximum depth of thirty. Associated with each decision is a photograph of the partial layout, and codes for what parts and options to order. Our goal was to design a visualization that shows the path through the decision graph (provided the context of the decisions) and the photo resulting from the latest decision.

One possibility might be to design a variation of Cone Tree [10] with multiple parents represented by different kinds of links. This alternative was rejected early because of the depth of the decision graph, which is well beyond the limits of current Cone Trees.

Another possibility might be to design a Perspective Wall [7] with a graphic representation of the decision graph laid out on the wall. The problem here is in both dimensions. Because of the depth of the decision graph, the width of the wall would be inadequate. And because of the unknown width of the graph, the height of the wall may well be inadequate.

In general, the decision graph is too large to show in its entirity. The spiral visualization offers a viable alternative. If we only show the current path through the decision graph, then the spiral can provide the context of the decisions. Color Plate 3 illustrates the resulting design. The image in the upper left corner is the photo of the layout resulting from the latest decision. At any point, the user can request a parts list and get all of the parts and options that have been ordered as a consequence of the decisions. Also, at any point, the user can point to any earlier decision and take a different path. When that is done, the discarded path is retained. The "Prev" button will pop up a list of discarded paths and let the user resume any one of them.

This use of the spiral visualization does a good job of showing the relative context of a decision; that is, the list of decisions made to get to this point. However, because the user never sees the whole decision graph, it does not provide a very good absolute context. While this is a drawback, it appears to be the only viable way to handle graphs that are too large to display. Informal evaluation suggests that it is easy to understand the context of the decisions and their consequences, and easy to change decisions.

## TIME LATTICE: VISUALIZING GROUP CALENDARS

This section describes how we designed a visualizer called the Time Lattice for analyzing multiple calendars. The Spiral Calendar was designed for a common task associated with calendar information, which is the manipulation of an individual's daily schedule. However, another important but less common task is the analysis of calendar information as a record of people's activity. One of the potential advantages of electronic calendars is that they support easy comparison of multiple calendars, which can be used to schedule meetings or plan group activities. Ultimately, portable computers will make it possible to capture a wealth of calendar information that can be used in such analysis tasks.

Since most calendars have a 2D layout, a natural design for visualizing multiple calendars is to align these calendars in 3D for comparison. The visualizer in Color Plate 4, which we call the Time Lattice, combines weekly calendars to form such a virtual 3D object. The vertical axis represents hours, the horizontal axis represents days, and the depth axis represents individuals. The individuals have been given distinct but random colors. The boxes in the 3D object represent scheduled events. Color Plate 4 presents real calendar information; details have been obscured to protect individual's privacy. Sliders on the floor are used to adjust the range of dates included in the 3D object.

The 3D object in Color Plate 4 is a natural design for a visualizer of multiple calendars. However, it is a complex object that involves challenging interactions. The key feature of the Time Lattice is that it includes a shadow metaphor to provide more intuitive interactions. This shadow metaphor was inspired by the Brown University work on interactive shadows that was reported at UIST'92 [5]. The Brown work focused on 3D graphics tasks. The Time Lattice shadows, on the other hand, support more abstract information analysis tasks. In particular, they include components that are not actually cast as shadows by the 3D object and they support a range of information filtering operations that differ from the 3D graphics tasks explored by the Brown work.

The shadow on the left wall composes the individual calendars. This shadow is particularly useful for scheduling meetings and is essentially identical to the calendar user interface described by Beard et al at CSCW'90 [1]. When the Time Lattice is run on a machine that supports alpha blending, this shadow blends the schedules together using transparency, which makes it possible to see time slots that are relatively available. Beard et al point out that opacity can also be used to encode high priority events to provide even more cues about partially available times. Unfortunately, the CM calendar database that we are using does not include priority information. The Time Lattice provides the additional feature that it also shows the 3D object that casts the shadow, which can be used to interpret the blended view.

The other two shadows, on the floor and right wall, show how busy individual people are. Commercial electronic calendars described in a recent MacWorld review also show how busy people are [11]. Such information can be useful when deciding whether to schedule an event. The calendar on the floor shows how busy a person is on a particular day and the calendar on the right wall shows how busy a person is at a particular time across a date range. We have found these shadows useful for identifying who always schedules lunch meetings and seeing the pattern of activities that repeat from week to week.

The shadows in the Time Lattice also include components not cast by the 3D object. The shadow on the floor has a set of buttons for adding and removing an individual from the 3D object. Fine-grain selection is accomplished by clicking in shadows. For example, the right shadow can select events at a particular time.

The Time Lattice visualizer is similar to other visualizers in the IV system in that it can be used in conjunction with more traditional information access techniques. For example, Color Plate 5 shows the result of a search for events that are not repeated. Layers from the front of the 3D object have also been removed by clicking on the buttons on the floor. Repeated events are draw as rectangles rather than as 3D boxes. Visualizing this search reveals that the future is primarily repeated events that represent regularly scheduled meetings, while the past is primarily unique events that represent what actually happened. Color Plate 5 also shows colored shadows, which make it easier to identify who is contributing a shadow but harder to see the overlaps of events.

Informal evaluation indicates that the Time Lattice is very useful for exploring the time relationships among schedules of groups of people. For example, we were able to find meetings attended by members of the group even though individuals referred to the meetings with completely different text strings. Every time we use the visualizer it prompts us to ask and answer new questions about people's calendars.

## DISCUSSION

The Spiral Calendar and Time Lattice demonstrate that emerging graphics technology can be used to design visualizers for a routine task, working with calendars and time-based information. We used 3D graphics to increase the density of visualizations. A spiral was designed to combine detail and context. Animated transitions allow the human perceptual system to track transitions that otherwise would have to be analyzed cognitively. Cartoon animation techniques were used to enhance these animated transitions. Our experience designing visualizations for electronic calendars suggests that emerging graphics technology can also be used for other user interfaces that have been frozen by the established desktop user interface standard.

### Acknowledgements

We thank Stu Card and Pete Pirolli for their work on the user studies. We thank Stu Card, Polle Zellweger, and Pete Pirolli for reading a draft of this paper. We also thank the Information Sciences and Technology Lab. for supporting

Robert Deline's as a PARC summer student intern.

## REFERENCES

1. D. Beard, M. Palanlappan, A. Humm, D. Banks, r A. Nai, and Y. Shan. A visual calendar for scheduling group meetings. In *Proceedings of the Conference on Computer-Supported Cooperative Work*, pages 279–290. ACM, 1990.

2. A. W. Biermann, L. Fineman, and K. C. Gilbert. An imperative sentence processor for voice interactive office applications. *ACM Transactions on Office Information Systems*, 3(4):321–346, 1985.

3. S. K. Card, P. Pirolli, and J. Mackinlay. The cost-of-knowledge characteristic function: Display evaluations for direct-walk dynamic information visualizations. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*. ACM, April 1994.

4. B. Chang and D. Ungar. Animation: From cartoons to the user interface. In *Proceedings of the ACM Symposium on User Interface Software and Technology*. ACM Press, Nov 1993. In press.

5. K. P. Herndon, R. C. Zeleznik, D. C. Robbins, B. Conner, S. S. Snibbe, and A. van Dam. Interactive shadows. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 1–6. ACM Press, Nov 1992.

6. C. M. Kincaid, P. B. Dupont, and A. R. Kaye. Electronic calendars in the office: An assessment of user needs and current technology. *ACM Transactions on Office Information Systems*, 3(1):89–102, 1985.

7. J. D. Mackinlay, G. G. Robertson, and S. K. Card. The perspective wall: Detail and context smoothly integrated. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 173–179. ACM, April 1991.

8. K. Perlin and D. Fox. Pad: An alternaitve approach to the computer interface. In *Proceedings SIGGRAPH'93*, pages 57–64. ACM, 1993.

9. G. G. Robertson, S. K. Card, and J. D. Mackinlay. Information visualization using 3d interactive animation. *Communications of the ACM*, 36(4), 1993.

10. G. G. Robertson, J. D. Mackinlay, and S. K. Card. Cone trees: Animated 3d visualizations of hierarchical information. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 189–194. ACM, April 1991.

11. F. N. Tessler. Desktop calendars. *MacWorld*, 10(7):104–109, 1993.

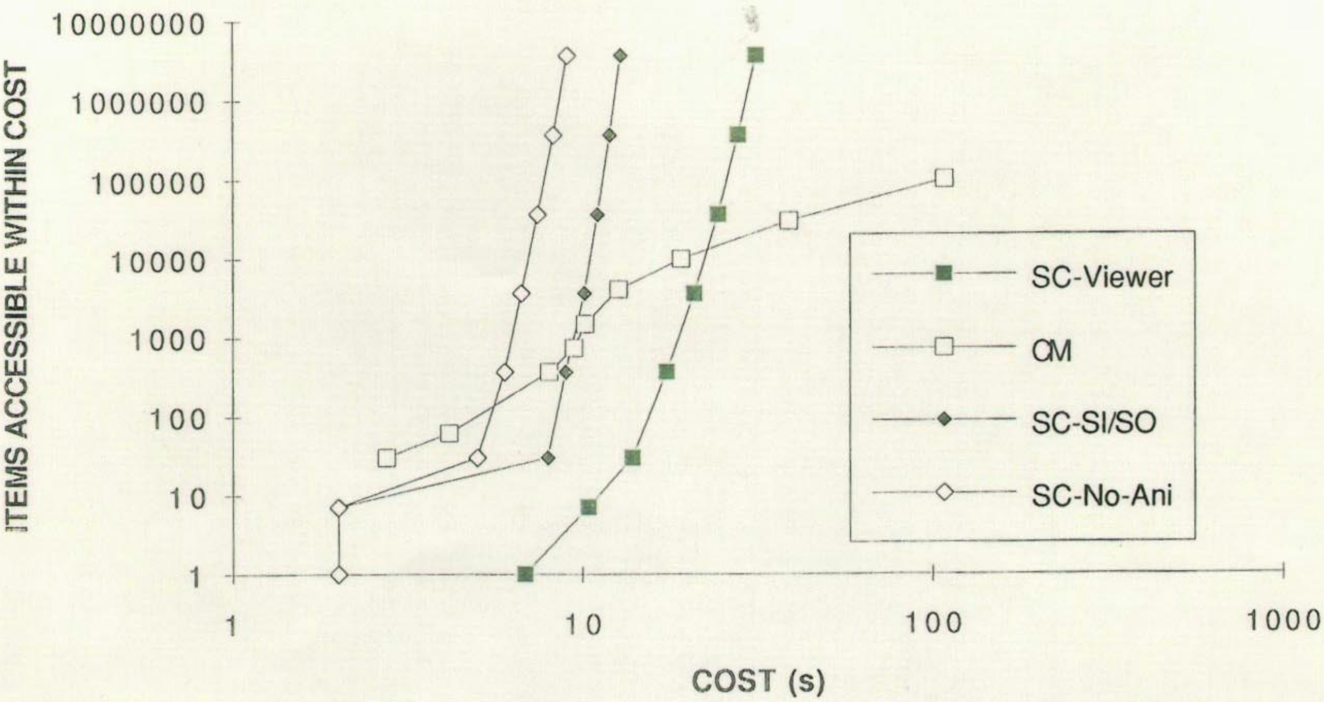
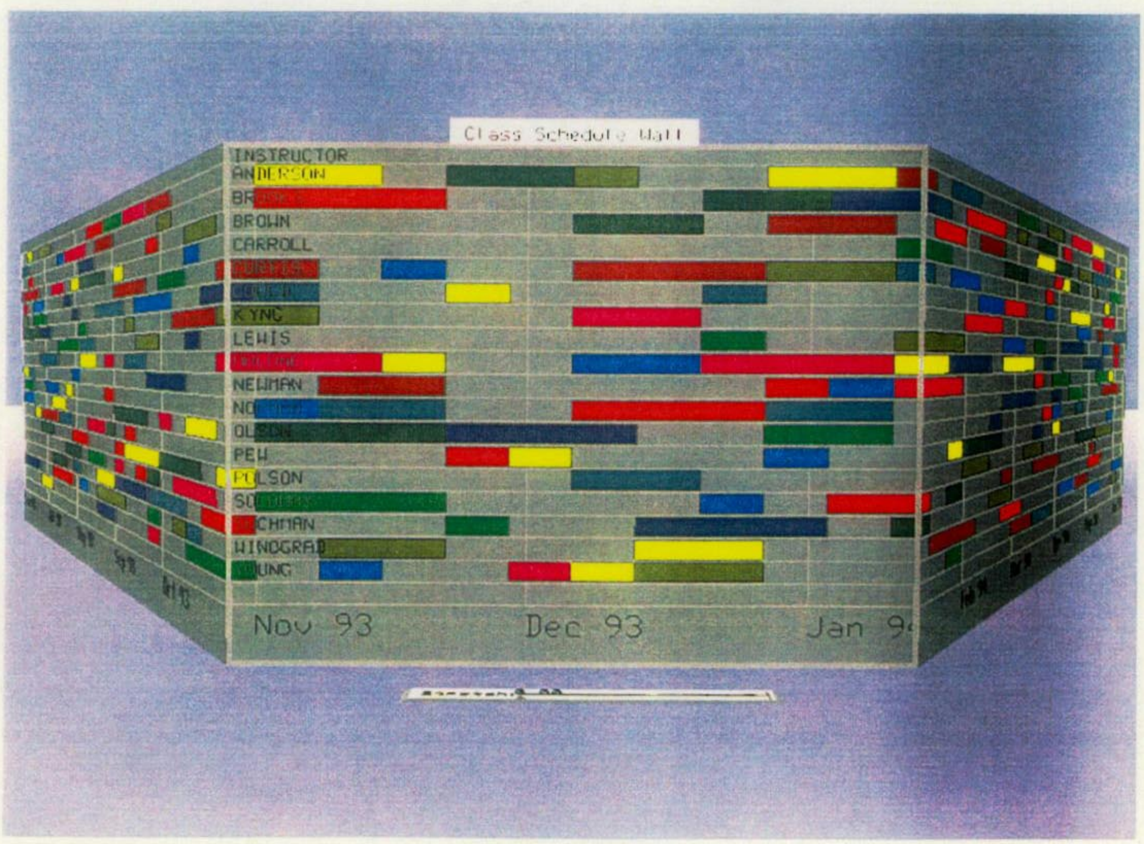
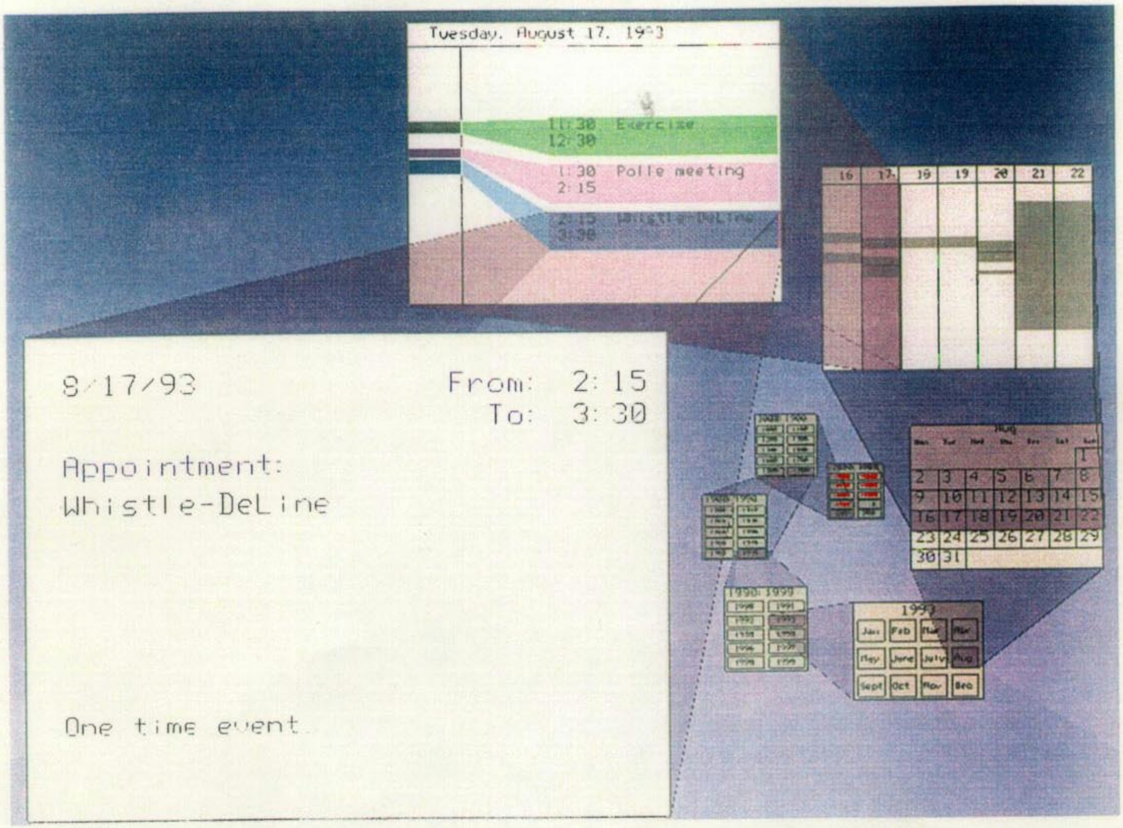


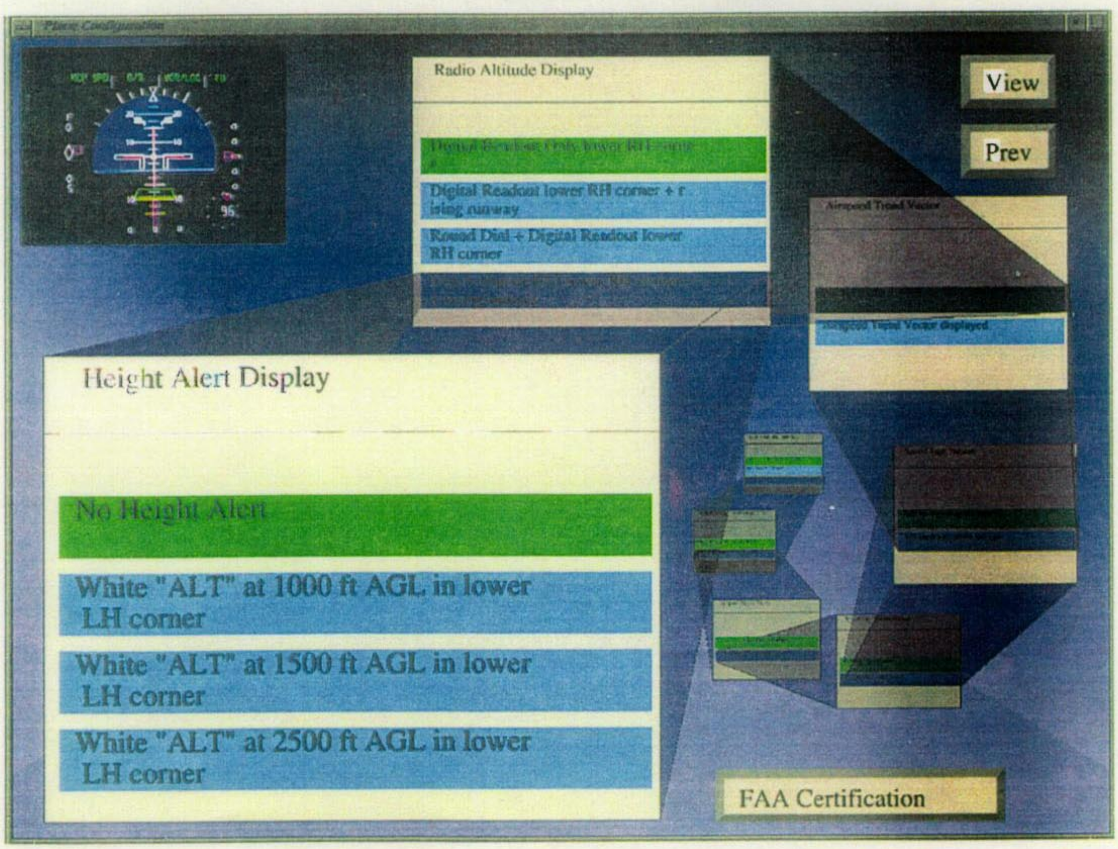

Figure 5: Cost-of-Knowledge Characteristic Function for to several variants of the Spiral Calendar and the CM calendar program.



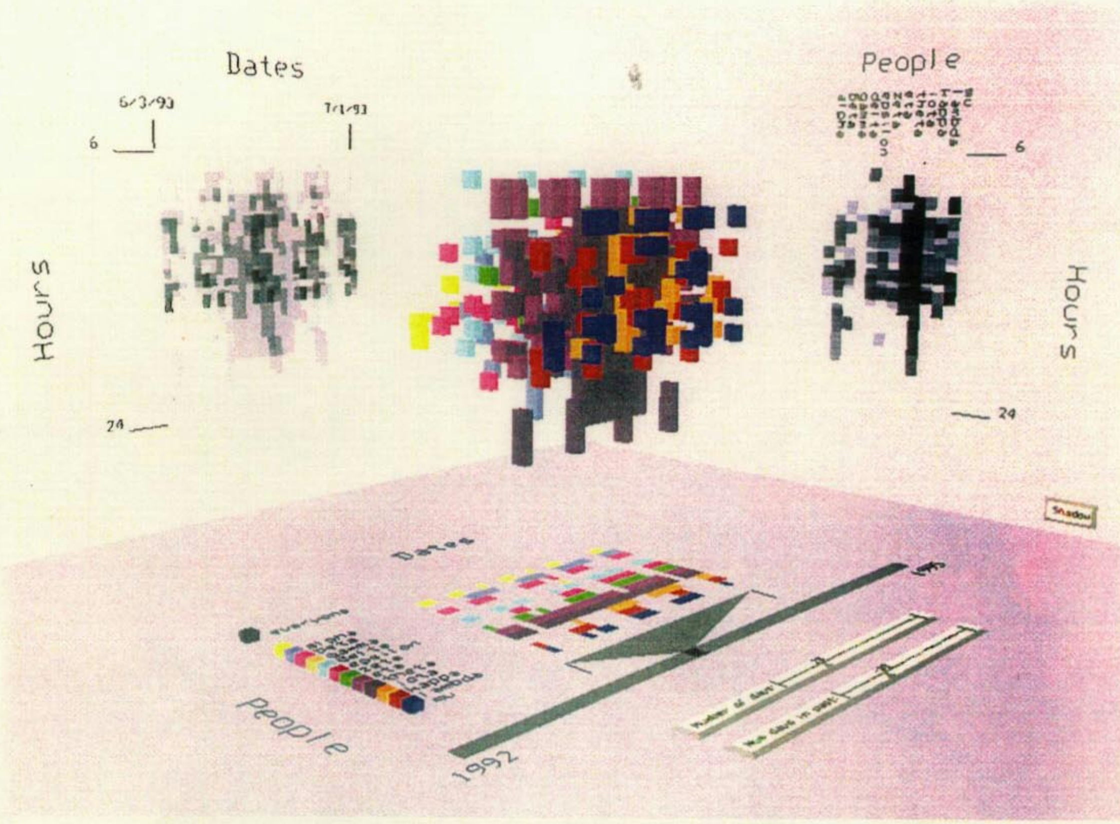

Mackinlay et al, Color Plate 1: A Timeline of a class schedule on a Perspective Wall. This visualizer does not reveal daily and seasonal cycles.



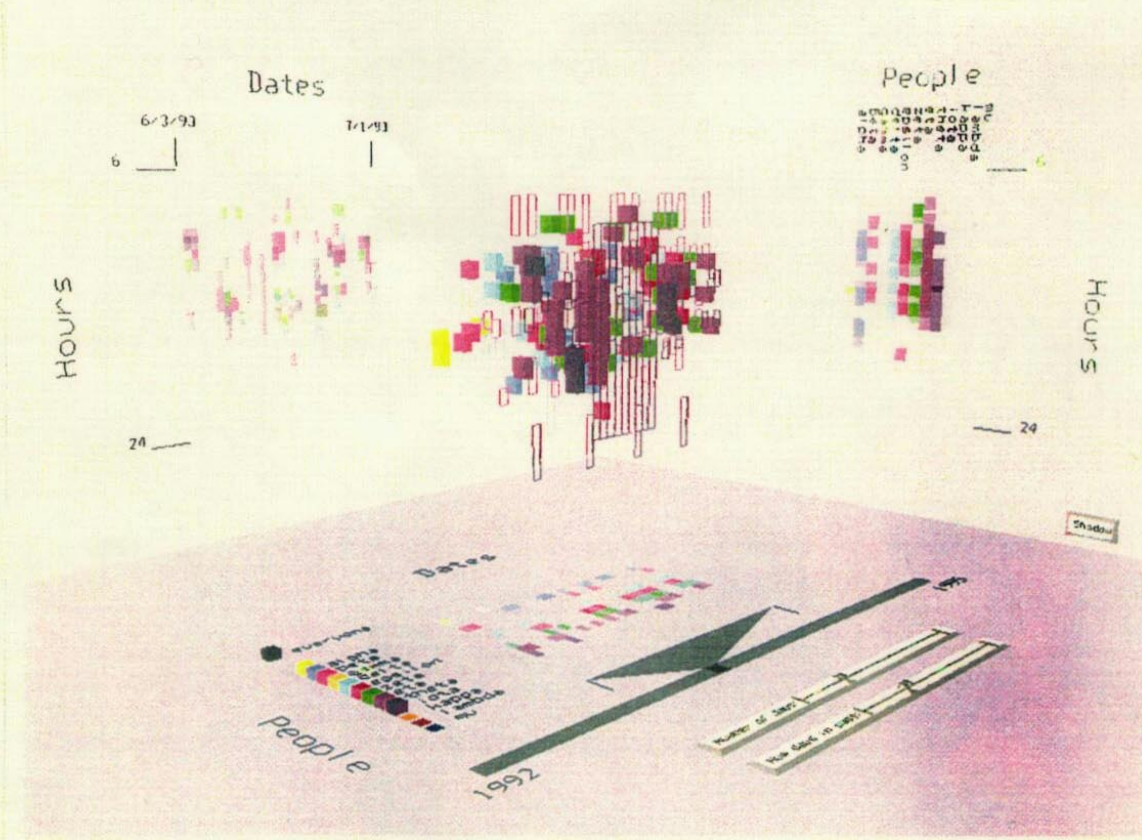

Mackinlay et al, Color Plate 2: The Spiral Calendar visualizer, which was designed for rapid access to an individual's daily schedule. A spiral layout combines detail and context in an intuitive 3D layout that allows the connection among calendars to be visible.




Mackinlay at al, Color Plate 3: The Plane Configuration visualizer, which was designed for keeping track of decisions made in a decision graph for designing the layout of controls in a aircraft cockpit. The spiral shows all decisions that have been made, along with a photo of the current layout.




Mackinlay et al, Color Plate 4: The Time Lattice visualizer, which was designed for analyzing the time relationships among the schedules of groups of people. A 3D object compares the weekly calendars of a group of people. Translucent shadows are used to support understanding and interaction with this complex 3D object.




Mackinlay et al, Color Plate 5: This Time Lattice draws repeating events as wire frame rectangles. The buttons on the floor have also been used to remove individual calendars from the 3D object. Color shadows are used to reveal individual contributions.