Scope: Providing Awareness of Multiple Notifications at a Glance

Maarten van Dantzich, Daniel Robbins, Eric Horvitz, Mary Czerwinski
Microsoft Research
One Microsoft Way, Redmond WA 98052, USA
{maartenv, dcr, horvitz, marycz}@microsoft.com

ABSTRACT
We describe the design and functionality of the *Scope*, a glanceable notification summarizer. The Scope is an information visualization designed to unify notifications and minimize distractions. It allows users to remain aware of notifications from multiple sources of information, including e-mail, instant messaging, information alerts, and appointments. The design employs a circular radar-like screen divided into sectors that group different kinds of notifications. The more urgent a notification is, the more centrally it is placed. Visual emphasis and annotation is used to reveal important properties of notifications. Several natural gestures allow users to zoom in on particular regions and to selectively drill down on items. We present key aspects of the Scope design, review the results of an initial user study, and describe the motivation and outcome of an iteration on the visual design.

Categories and Subject Descriptors
H.5.2 [**INFORMATION INTERFACES AND PRESENTATION**]: User Interfaces--*Graphical User Interfaces*.

General Terms
Design, Human Factors.

Keywords
Information visualization, peripheral displays, awareness, notifications, interruptions, alerting and notification systems

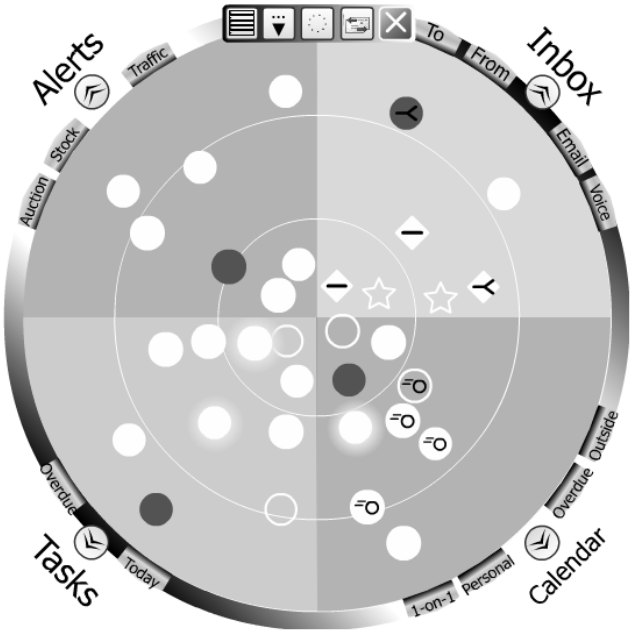


Figure 1 Scope (Figure 2 shows same in color).

Please see color plates for this paper.

Scope: Providing Awareness of Multiple Notifications at a Glance

Maarten van Dantzich, Daniel Robbins, Eric Horvitz, Mary Czerwinski
Microsoft Research
One Microsoft Way, Redmond WA 98052, USA
{maartenv, dcr, horvitz, marycz}@microsoft.com

ABSTRACT
We describe the design and functionality of the *Scope*, a glanceable notification summarizer. The Scope is an information visualization designed to unify notifications and minimize distractions. It allows users to remain aware of notifications from multiple sources of information, including e-mail, instant messaging, information alerts, and appointments. The design employs a circular radar-like screen divided into sectors that group different kinds of notifications. The more urgent a notification is, the more centrally it is placed. Visual emphasis and annotation is used to reveal important properties of notifications. Several natural gestures allow users to zoom in on particular regions and to selectively drill down on items. We present key aspects of the Scope design, review the results of an initial user study, and describe the motivation and outcome of an iteration on the visual design.

Categories and Subject Descriptors
H.5.2 [**INFORMATION INTERFACES AND PRESENTATION**]: User Interfaces--*Graphical User Interfaces*.

General Terms
Design, Human Factors.

Keywords
Information visualization, peripheral displays, awareness, notifications, interruptions, alerting and notification systems

1 INTRODUCTION
As personal computers have become connected to increasing numbers of information sources, users are challenged to manage higher rates of interruption by notifications. Today, many users handle alerts from a variety of sources, including newly arriving email, status changes in instant messenger "buddy lists," stock and traffic alerts, online auction progress, sports game scores, news headlines, special sales, and so on. Even more notifications are on the horizon; new development efforts such as the Microsoft .NET platform [21] promise to increase the types of services offered to users and thus provide even larger numbers of notifications. Users are now facing *notification overload*—the challenge of keeping up to date on incoming information alerts. Interruptions can be detrimental to productivity, especially when the user is deeply focused on a task [16]. Human attention has long been known to be a scarce resource [2, 6]. Thus, providing awareness of relevant information or events with minimal strain on cognitive resources promises to be increasingly valuable to users.

We have sought to develop visualizations that allow users to decide what to attend to, and when. This principle is central in the *Scope*, a graphical display that unifies information about email, pending work items, and other information in one central place.

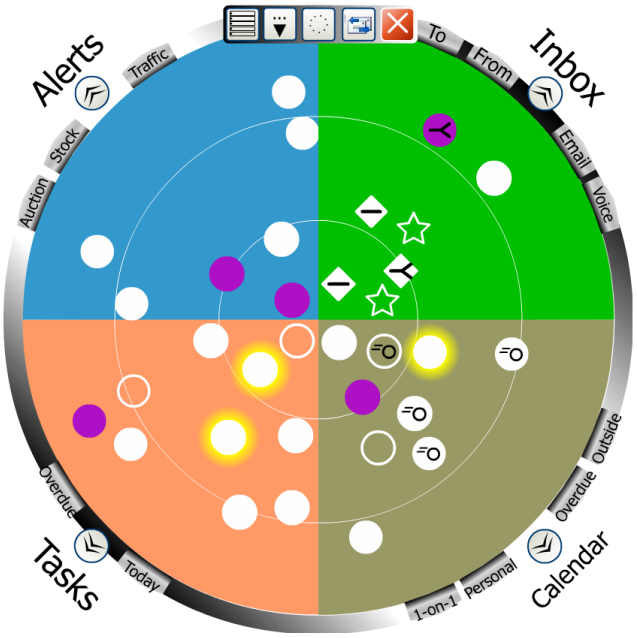


Figure 2 Scope after design iteration described below

2 WORK ON UNIFYING NOTIFICATION ARCHITECTURES
The Scope display was motivated by work over the last several years on systems that consider multiple sources of information and normalize different types of informational items into a unified notification and awareness framework. Existing work in this arena includes the *Priorities* system [12] and its descendant, the *Notification Platform*.

Priorities considers email messages, tasks, and appointments. The system learns models of prioritization of items from user

behaviors or from explicit user feedback, and uses these models to automatically assign an urgency score based on multiple facets of items, including, for the case of email, the sender, the nature and number of recipients, and the content of the header and body of messages. A descendant of Priorities, the Notification Platform, maintains a general inbox, called a Unibox, which includes such items as email, voicemail, appointments, instant messaging, news and financial information, and the output of a set of information gathering agents that have generated information for users. Users can control the weightings of the urgency scores assigned to different items.

The Scope is built to provide a single glanceable visualization onto the multisource, prioritized information contained in the Notification Platform's Unibox. That is, we assume in our interface design that some notion of an urgency score is available for items considered for display.

3 VISUALIZATION DESIGN

3.1 Design goals

A primary design goal for the Scope was to provide a tool for safeguarding a user's attention. We sought designs that could empower users to stay focused on their primary task, requiring minimal attention to stay aware of incoming notifications and pending tasks. Thus, we wanted the Scope to provide unobtrusive display modalities, leaving initiative primarily with the user. Second, we designed the Scope to be glanceable, that is, easy to read and understand in a minimal amount of time. The design should direct a user's attention to high urgency items. Finally, we designed the Scope to present notifications from many sources in a standardized fashion.

We designed for several modes of interacting with the Scope: monitoring incoming items while working on a primary task; deciding what to do next when the user is ready to switch tasks; catching up on newly arrived items after having been away from the computer; and use as an implicit to-do list.

We did not explore features for manually creating new items on the Scope or for accessing or searching the user's email archives. We believe these tasks are better addressed by existing personal information management applications already in use, such as Microsoft Outlook.

3.2 Visual metaphor

Figures 2 and 3 display two incarnations of the basic view of the Scope visualization. The Scope is a circular display that borrows its metaphor from a traditional radar view: we consider the user situated at the center of the display, and notifications are arranged so that higher urgency items are closer to the center. Concentric rings delineate areas for high, normal, and low urgency items. This arrangement has the advantage of a single point of visual attention: at any time, all the important items will be near the center of the display. As the radius of the focus of visual attention increases, more items are included in the glance, but items farther from the center are increasingly less urgent. This is the primary design aspect that makes the Scope glanceable, in a manner that aligns the triage of notifications with visual search

The circular space is divided into several wedges, or sectors. In our current design, each sector represents one canonical type of notification: personal communications (including email, voice mail, instant messages), calendar items, "to do" items, and other alerts (notifications from web services or personal agents). This partitioning reflects the organization of personal information management software already familiar to users. In the current implementation, the wedges have a fixed size, each occupying a quarter of the Scope. We would like to make the partitioning reconfigurable, so that more highly populated wedges can occupy a greater arc within the Scope.

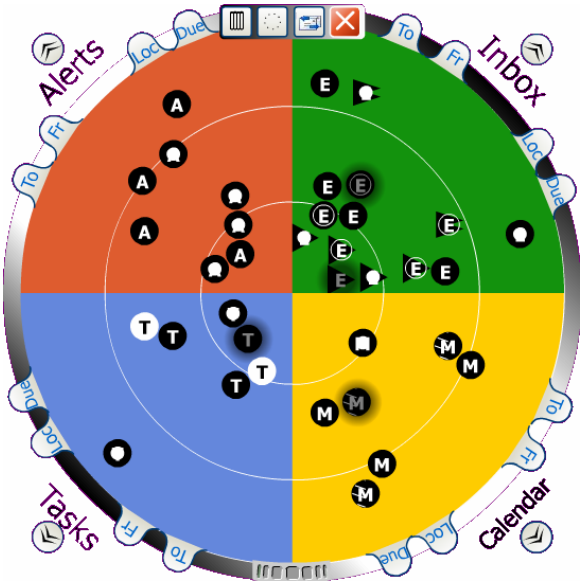


Figure 3 Original Scope design at large size and high LOD (approximately actual size).

Although we have focused on a set of familiar notification classes, we can imagine alternative arrangements of items into sectors. One alternative is to segment the Scope into subject-oriented sectors, providing such categories as "Work-related", "Home & Family", "Hobby & Interests", and "News". The work-related wedge could then be subdivided to group items related to the same project. The usefulness of such a formulation depends on the naturalness of the category and the ability of automated systems to appropriately classify the items. We foresee that later implementations could allow software plug-ins to populate newly created wedges on the Scope. Note that the Scope will always provide a single point of visual focus, independent of the number of wedges or services providing notifications. Thus, the glanceability of the design allows the interface to scale.

3.3 Visual annotations of metadata

In addition to the spatial layout of items, the Scope can assist users in deciding which items to focus on by providing visual annotations (iconography) to make items more distinct and identifiable. We identified numerous properties that the Notification Platform can provide that might be of interest to the user. We chose to highlight a subset of these properties in order to maintain visual clarity, focusing on email items. We chose a set of properties that appeared to be most important in helping users make decisions about where next to attend. These properties include newness, item type, and information about the addressing of recipients of the notification.

We will first describe our original design, to set the context for the user study discussion. A redesigned Scope with new visuals is described in later sections.

In our original visual design, displayed in Figure 3, every item on the Scope reveals its type by a letter in the body of the item. (e.g. "E" for email, "A" for agent alerts). This property is typically redundant with the wedge the item is placed in, though users can drag items to a different wedge when appropriate. (e.g., an email can be put in the Tasks wedge if it is a "to do" item in the user's mind.)

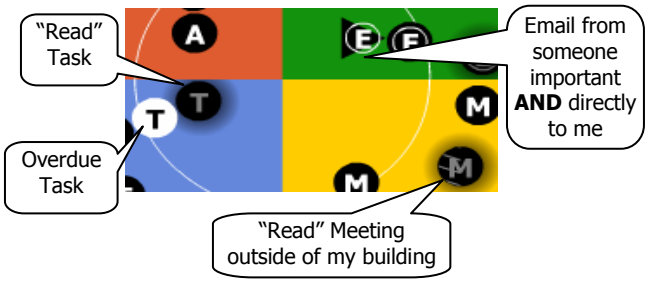
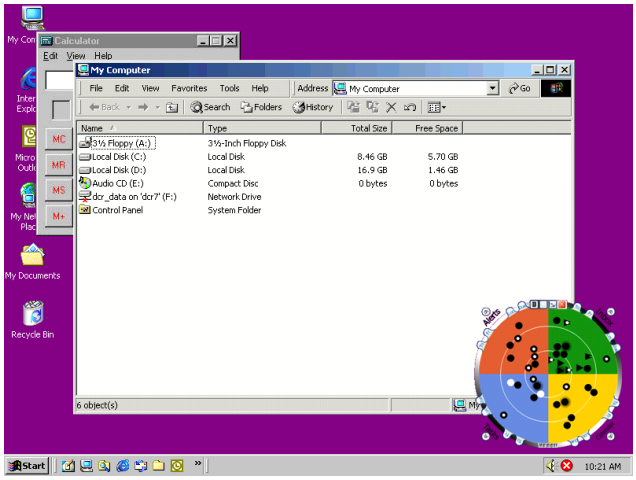


Figure 4 Visual annotations for various metadata.

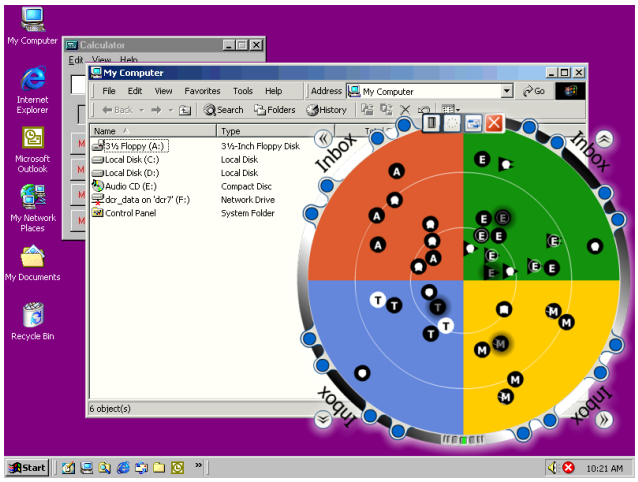
Second, each item reveals its degree of newness. Items can be "newly arrived" (as revealed by a pulsing of the item), "seen" (no longer blinking, denoting that the user has attended to but not fully opened the item), or "read" (shown as a blurry halo, after the user has opened the item). The distinction between "seen" and "read" allows the user to probe an item (yielding a tooltip, described below) and decide whether to open it or not; the item state will reflect the user's viewing behavior. We do not automatically remove "read" items from the Scope, as the user may still want to take further action; indeed users commonly leave read items in their email inbox for just this purpose [24].

Third, each item reveals some special properties if appropriate. We annotate email items if they were addressed explicitly to the user alone ("ToMeAlone" property), to just a few individuals, or if the sender is listed in the user's personal address book. The former two properties (collectively called the *ToType*) are displayed respectively by shaping the item as a triangle or double-triangle, suggesting single or multiple destinations. A significant sender is indicated with a circle inside the item; it can be thought of as an item from the user's "inner circle." For calendar items, we indicate whether a meeting may require travel time (e.g., if its location is known to be outside the user's building). This property was a popular end user suggestion. We indicate travel time with hash marks inside the item, reminiscent of "speed zips" drawn in cartoons to suggest movement. Finally, for calendar and "to do" items that have deadlines, we indicate items that are overdue by inverting their colors. We have explored only a portion of a larger design space of visual annotations. A future version of the Scope can allow a user to switch between different static and animated codings.

The same properties displayed in the visual annotations are also used by the Notification Platform to determine the urgency of the notification, and so could be viewed as redundant. However, the visual affordances help in several ways. They make items more distinct and assist users in deciding which items to attend to, helping the user feel in control. Earlier work suggests that the urgency scores are quite reliable [12], so seeing the correlation between properties and scores can only help to foster user trust in the system's intelligence.



(a)



(b)

Figure 5 Scope on the desktop at low LOD (a) and after mousing into the window at high LOD (b).

3.4 Levels of detail

We designed the Scope to live in a corner of the user's display, sized small enough to be unobtrusive—and optionally rendered in a translucent manner on top of other windows. As we experimented with alternate designs for the visual annotations, we quickly discovered that it is very difficult to make items distinct when they are very small: visual differences quickly become inscrutable when each item occupies only about 3x3 pixels. We solved this problem by implementing two levels of detail (LOD): a peripheral awareness mode (low LOD), and an active interaction mode (high LOD).

By default, the Scope positions itself in the lower right-hand corner of the display, fitting in a 180x180 pixel region as shown in Figure 5. This is the peripheral awareness or low-LOD mode: in this modality, items reveal only the most important properties, which are designed to be maximally distinct. These are *newness*, *ToType*, and *overdue status*. (Compare item visuals in Figure 6 with Figure 3.) Recall that urgency of items is still easily deduced from the item's location along the radius of the Scope. This allows

users to glance at the Scope and determine immediately whether new (pulsing) items have arrived, particularly in the high-urgency zone.

If the user moves the mouse pointer into the Scope window, it responds by changing into the active interaction mode: the Scope is popped to the top of the window stack, the window doubles in size, and the items are rendered in high-LOD mode, displaying all of the properties described above, and as displayed in Figure 3. Users can now gain additional information at a glance, or obtain more details from an on-hover tooltip. We believe that these two levels of detail provide a fluid way for the Scope to achieve both unobtrusiveness and useful richness of display.

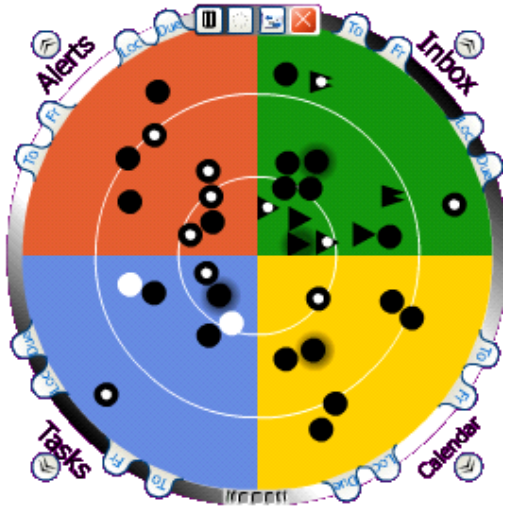


Figure 6 Scope at low level of detail, enlarged (cf Fig 3).

The visual coding of attributes went through several design iterations. Design criteria for the visual attributes are distinctiveness (each attribute having a unique visual representation) and discernability (items having a given property should be clearly different looking, even at small sizes). An early design overlayed multiple small icons on top of each item. Each icon represented the state of a particular property, such as *ToType*. Because we needed to overlay multiple icons for each item, each "sub-icon" ended up being so small as to be indistinguishable. Because of this, the design presented here instead uses more abstract visuals that affect the overall appearance of an item. For example, a stroked circle that outlines an item is much more discernable than a series of sub-icons that are blurry (due to their size). The design problem is made more difficult because we are representing Boolean states for many properties rather than one of multiple states for just a few properties. If we only had to deal with a few properties, we could easily cycle between several configurations of a basic geometric shape: for example, selecting from among a library of square, triangle, circle, and diamond shapes. Instead, we had to design visual modifications that do not conflict when overlaid with others. Although this design is quite abstract, we found that it does make for relatively efficient distinguishability, and that was supported by the user study.

3.5 User interaction techniques

Beyond serving as a visualization of pending notifications and tasks, the Scope also allows the user to inspect items further through low-effort interactions, in order to decide whether to open the item in its native application. The most common user action is to drill down on an item for more detail. When the user hovers the mouse pointer over a notification item, a tooltip appears with detail appropriate to the item type. This includes the title and a snippet of the body text, the sender's name if it is an email, the time and location if it is a meeting, and a deadline if one is set. (See Figure 10.)

If the item is newly arrived, inspecting the tooltip for more than a fraction of a second marks the item as "seen", removing the pulsing behavior. The user can click an item once to mark it as "read", or leave it "seen" as a reminder that the item requires further attention. A double click opens the item in its native application, e.g., mail messages are opened in an Outlook message window. This also marks the item as already read, changing its display state to have a fuzzy halo, causing a muted appearance with less salience.

Users are able to modify positions and groupings manually. The integration of this capability for the Scope was informed by the experience of Robertson et al. with the Data Mountain [22], where users created rich layouts of web favorites when presented with a free-form space to organize in. Users can drag and drop items on the Scope to produce implicit groupings, to change the urgency of an item, or to delete an item. A manual change in an item's urgency on the Scope visualization can send feedback to the Notification Platform, allowing refinement of its future prioritizations. This provides a novel and intuitive way for users to communicate with the underlying decision making system.

An item can be discarded by simply dragging it off the Scope.

The Scope provides several mechanisms that help combat overcrowding and make the display more scalable. First, users can focus on one wedge at a time by expanding that wedge with the expansion button. When the button is clicked, a very short animated transition occurs, shrinking the Scope while the chosen wedge expands. This yields space to show more low-urgency items below the normal urgency threshold. Thus, additional information is shown while retaining context. A second click on the button brings the wedge and the Scope back to their natural sizes.

Second, users can filter the items shown in any given wedge using the filter buttons at the rim of the Scope. When the user activates a filter, items that do not match the filter are hidden, and the appearance of the wedge background is changed to reflect its filtered state. As an example, emails can be filtered for the items that are "ToMeAlone"; tasks can be filtered for overdue times. Currently, filters are exclusive: clicking on a second filter turns off any currently active filter. Two more mechanisms are described in the section on our second design iteration: summarization of items below the urgency threshold, and inspection of detail on multiple items at a time.

3.6 Scope autonomous behavior

As the Scope was designed to be an unobtrusive application, it does not generally initiate interaction. When new items arrive, they are quietly faded onto the Scope. However, when a high urgency item arrives, a more salient behavior is triggered. As shown in Figure 7, a fly-in detail-view appears briefly to cover the low-LOD Scope with information similar to the item's tooltip. This is accompanied by a muted audio cue. The user has an

opportunity to glance at the item's content in the corner of the screen; if there is no reaction after a brief interval (3 seconds), the "leaflet" disappears again, and an item is shown arriving onto the Scope, indicating where on the Scope it was placed.

Figure 7 A newly arrived item displayed (low LOD Scope).

4 PILOT USABILITY STUDY

4.1 Study design

We performed a pilot usability study to evaluate our initial design and explore areas requiring improvement. We gathered six knowledge workers (1 male, primarily managers and administrative assistants at a large company) who were experienced users of PCs with Microsoft Windows. Participants were screened to be 25-50 years of age and to have trouble managing their email inbox, meetings, and to do lists.

Note that this user study was designed to identify major usability problems and to drive design iteration, rather than to formally validate specific claims. We believe that six participants is an appropriate sample of subjects for this purpose, as indicated in prior research by Nielsen [19, 20].

We wanted to present credible content on the Scope without accessing users' own mailboxes, so we asked our participants to imagine that they were Pat Maloney, a fictitious person, and that all data on the Scope was intended for Pat.

Participants completed a series of 11 tasks introducing progressively more of the Scope functionality. Some tasks were designed to observe merely what the user noticed and understood (or not) and then teach them the mapping of features and visual attributes; other tasks tested whether the participant had successfully learned to use the Scope.

Task times and verbal protocols were collected throughout the session, and a user satisfaction questionnaire was completed at the end. Sessions lasted approximately one hour and participants were run singly. A sampling of the tasks is provided below:

- See if you can determine which items are of high urgency and which are lower, using the email section.
- Find an unread email of high urgency that was sent only to you, from a known contact.
- How many meetings are not close to your office?
- Filter your view of the Scope so that you only see email messages that were sent specifically to you.
- Read a high urgency email sent only to you.

4.2 Results of the Pilot Study

We did find many usability issues with this initial version of the prototype, and have received good design feedback. For instance, many users did not like the Scope's auto-sizing behavior on mouse-over. In addition, it was clear that some of our mappings of visual features to "high urgency", "new", "overdue", and "from a known contact" were so abstract (*i.e.,* arbitrary) that they were hard for users to remember. The good news is that participants were able to learn these mappings within an hour and usually found them adequate by the end of the session.

Users' satisfaction ratings using a 7-point Likert scale (higher scores are better) reflected the fact that most participants found the Scope to be promising. Note in the discussion below that a score of 4.5 is a standard threshold for high satisfaction on a 7 point scale. The ratings were quite high for a first iteration of testing on a novel design, as can be seen from the average ratings of questions in Table 1.

Questionnaire Item:	**Avg. Rating (1=Disagree, 7=Agree)**
1 The ability to change priorities for an item by drag and drop was easy:	6.33
2 It was easy to move between low and high levels of detail:	6.167
3 The use of pulsing to show "new" items allowed for good detectability:	6
4 The ability to expand a wedge was useful:	6
5 I was able to see an overview of my high priority item from the Scope at a glance:	5.67
6 The different shapes of the items were easy to learn and useful:	4.83
7 The Scope Alerts prototype was easy to use:	4.5
8 The different textures/motions of items (pulsing, fuzzy or inverted) were easy to learn and useful:	4.33
9 The ability to delete items from the Scope view was easy:	4.33
10 It was clear what region of the Scope meant "high priority" and what region meant "lower priority":	4.167
Overall Average Subjective Rating:	**5.3**

Table 1 Subjective usability ratings

Users did not intuit without instruction how to identify high urgency items; the low score on the question 10 reflects this. We believe this partly reflects the fact that spatial location (other than strict sequence) is rarely used to code attributes in current graphical interfaces. We observed that once users had learned the interface, they could find new items and high priority items easily and quickly. This is reflected in the higher satisfaction score (5.67) for question 5 ("I was able to see an overview of my high priority items at a glance"). More subtle attributes, while taking a little more time, were taking about half as long on average as in earlier tasks. This is in keeping with our design goal: the most important attributes (urgency, newness) should be apparent at a

glance, while other important attributes are available through further inspection.

Task time analysis showed that although participants took some time to learn the mapping of the Scope features to their meanings, this could easily be accomplished within a single session of interaction. For instance, finding a high urgency email without guidance or training took 2:34 minutes as users initially explored the display arrangement, but by the end of the session it only took participants 34 seconds to find, read, and close a high urgency email in a more crowded version of the Scope.

Overall, participants liked the key visual metaphor and thrusts of the Scope design and the idea of a "one stop shop" approach to presenting a picture of the high urgency information demanding their attention. Participants especially liked annotations such as "ToMeAlone", and "requires travel time." Participants were most interested in easily finding email from their manager, and easily identifying the sender of a message. Several requested that urgent items be shown in red, to match the red exclamation mark on such items in their email client. Many people have commented that the pulsing animation of new items, while designed to be unobtrusive, is still distracting. Finally, participants stated they were very interested in the adaptive nature of the Scope, especially when they discovered that they could drag and drop Scope items to a higher or lower urgency rating in order to relay feedback to the system. In one participant's words: "What it's trying to do is create a prioritization scale that is applicable to all these things. It does happen, everyone does it naturally, but I don't think a lot of that is explicit in our lives, at least not to me. I do it on the fly."

5 DESIGN ITERATION

Following the user study, we redesigned the Scope item visuals. We describe the new design here to document how use of design constraints led us to a more effective design.

We addressed several concerns in the design iteration: First, users frequently remarked that they did not like the pulsing of new items. To address this, we decided to use color to code newness, which in turn pre-empted us from displaying high urgency items in red. Second, since the original wedge colors made it very hard to find item colors that contrasted well on all of the background colors, we redesigned the wedge color scheme. Third, we sought to soften the stark black rendering of items; this could enable color and internal details of items to be used to relay attributes of items. Finally, we felt we should forego the use of letters for item type annotation ("E", "M", etc.) as sectors generally only house a single item type; the type annotation added of visual clutter while providing only a small amount of informational value.

5.1 Design Constraints

The redesign process prompted us to explicitly consider multiple constraints on the Scope design. These include:

- items must have good contrast against each of the sector background colors
- the sector background colors must have good pairwise contrast
- if multiple item shapes are used, they must be easily distinguished, even at low level of detail
- any iconic or "glyph" annotations must be recognizable even when the user resizes the Scope window and must be simple to minimize visual clutter. Traditional icons do not resize gracefully and usually have a great deal of high-frequency visual detail. We have limited ourselves to simpler, more abstract designs.
- any properties coded as "glyph" annotations might not be shown in low-LOD mode, and thus should be used for less important properties
- design of item visuals should take into account impact of graphical complexity on the application's frame rate. Some rendering techniques like animation and transparency can slow down display updates substantially when applied extensively. Their use should be carefully considered.
- at most one (subtle) animation cue should be used, and employed in a synchronized manner across items so that multiple items can be highlighted with a unified visual "pop-out." We found that having items animate in sync (all starting the animation cycle at the same time) greatly added to a gestalt appearance that reduced the need to visually scan the display.

In addition, we ordered the properties by importance, so we could map them appropriately to the most salient cues:

- "high urgency" and "newly arrived" are the most important properties, and must be identifiable at a glance
- "overdue" and "pinned" items should be easily identifiable
- with slightly more effort, users should be able to read other properties, such as the sender and addressee types.

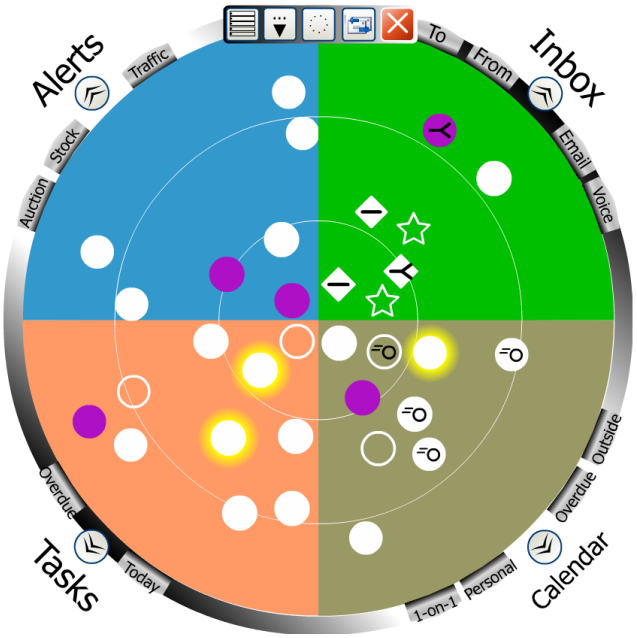


Figure 8 Redesigned Scope visuals

5.2 Resulting New Design

The resulting new design is illustrated in Figures 8 and 9. Basic items are rendered as white circles; overdue items have a yellow halo; new items no longer pulse but are colored (purple). We have combined the "seen" and "read" states into the notion of a "touched state," and allow the user (with a simple click) to mark items of special interest as being "pinned"; these are shown as a hollow outline, retaining both shape and color. Emails from known senders have special shapes: a diamond for "known" senders and a star if the sender is on a user-configurable list of "special senders." If an email is marked "OnlyToMe" or "ToMeAndAFew," it is annotated with a straight bar or forked bar, respectively. Meetings that require travel time are annotated with a "speeding wheel" icon.

These redesigned visual annotations seem to be quite legible even with all properties displayed at the low level-of-detail sizing of the Scope; thus, we now display the properties at all times, although we retain the automatic resizing behavior illustrated in Figure 5.

Our redesign methodology was motivated by the goal of enhancing the glanceability of the Scope display: we generated a hierarchy of visual cues and matched it with an importance ordering of the properties, indicating which should be available at a glance. We mapped those properties to cues in separate visual domains: location, color, brightness, animation (subtle pulsing), and we designed the low LOD mode to hide the least important properties. While one might argue that the overall display is complex and encodes a lot of information, these strategies ensure that it is carefully crafted to make the most important properties available at a glance.

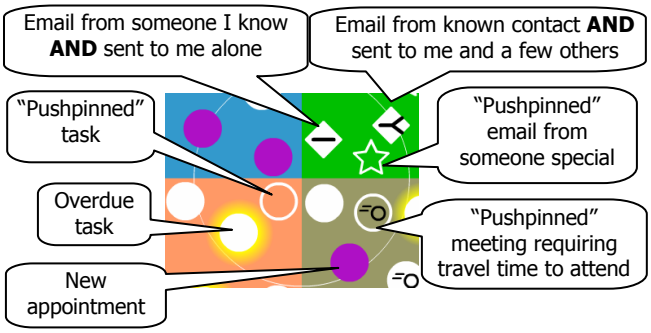


Figure 9 Redesigned visual annotations for metadata

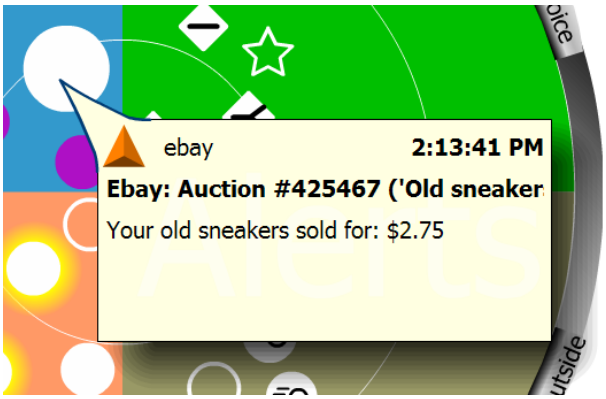


Figure 10 Tooltip appearing when mouse is on item

6 IMPLEMENTATION

The Scope is a Windows application and currently works together with Microsoft Outlook and personal information stored on a Microsoft Exchange Server. It is implemented in two major parts: the visual interface in Macromedia Flash and supporting system code in Microsoft Visual Basic. This was accomplished by placing the Flash control on the main form of a simple Visual Basic application.

This arrangement has several advantages. First, Flash "movies" are built from vector graphics and thus render well at multiple sizes. Second, Flash is designed to support animation. This makes it easy to provide animation in all visual changes and transitions, and gives the interface a very smooth and polished feel. Most importantly, the combination of these tools has enabled a close collaboration between a designer, concentrating on visuals and interaction, and a programmer, working on internals and data issues. We have been able to do rapid prototyping of a real working application, using real user data. Many interface design aspects are easily iterated: changing what is shown in a tooltip or changing the attributes used by a filter button is trivial; adding new attributes requires code changes in both parts of the prototype but is straightforward and still affects very small parts of the code.

Notifications are retrieved from the Exchange Server using the MAPI and CDO interfaces provided in Visual Basic. They are translated into XML data for transfer to the Flash control. Updates are similarly propagated back in XML form from the UI to the VB wrapper, so that item state and layout can be saved and restored in future sessions.

The Scope relies on the use of the existing Priorities system (described above) to rate incoming emails for urgency; the user can run Priorities in parallel to monitor incoming mail and rate messages. Future versions of the Scope will draw upon more sophisticated methods employed in the Notification Platform for assigning urgencies to multiple items. Currently the Scope prototype implements ad hoc rating schemes for several items (e.g., meetings are rated based on how soon they take place; task items are rated based on the urgency set by the user (high/normal/low).

One limitation of the Flash runtime for user interface work should be noted: its animation speed is strictly based on the current frame rate, not on a "real-world" clock. Thus animation slows down as display gets more cluttered. However, for UI prototyping we have found this limitation to be a manageable one.

7 RELATED WORK

7.1 Existing Desktop UI and user strategies for awareness

Currently users do not have sophisticated machinery for visualizing and exploring multiple incoming notifications. For the commonly used email channel, some email clients allow the user to author rules to sort email into multiple folders (e.g. all email from a given mailing list into a designated folder) or to customize what alerts are presented (e.g. special sounds for special senders, colorization of items in the inbox).

The Notification Area on the Windows Taskbar [17] is an example of a centralized mechanism provided by the operating system, allowing applications to display a notification icon or affordance. Its possible uses are quite limited, though: each

application only gets to display a 16x16 pixel icon in a peripheral area of the screen, and there is little uniformity in how users interact with the notifications. As more notification icons are displayed in the area, the value of each diminishes. Similarly, the Mac OS has long allowed applications to "blink" their (small) application icon in the system menu header to indicate that that application needs attention. Microsoft MSN Messenger uses small peripheral textual pop-ups to notify users of changes in buddy status and new mail arrival.

The Priorities system [12] displays unread email by urgency in a list view, constrained to user-specified periods of time. The client can be deployed as a transparent ambient display with user-configurable properties that define policies for audio alerting and the fading in and out of a translucent display, based on the urgency of incoming items and the sensed context of the user. New messages associated with a notification are highlighted within the client as it fades in.

In practice, many users promote their awareness of newly arriving information by either arranging windows so that a small but important region is always visible (e.g. the last few lines of the email inbox application), or by running one or more awareness applets.

In summary, it is important to note that in current systems notifications are delivered in many different fashions, items are rarely ranked by urgency, notification policy is spread out over many applications, and there is no single place for users to check where their attention is needed. We feel that adding these properties to a single, unified UI is one of the main contributions of the Scope notification design.

7.2 Research on Awareness and Notifications

There has been a long history of human factors and engineering efforts exploring the use of dashboards and heads up displays for notification and awareness. Many research projects have pursued peripheral awareness of other people on the desktop with video [7] or with abstracted graphics [10]. Much recent work has been done on peripheral information displays outside of the PC, in the user's physical environment (e.g. Weiser and Seely Brown [23]). This is generally called *ambient* information after the ambientROOM by Ishii and Ullmer [13]. InfoCanvas is a customizable display attached to a personal computer and positioned in the periphery [18].

Several research projects have attempted to provide awareness of mulitple sources of information. Sideshow [3] provides a desktop toolbar that can accommodate many information sources. The What's Happening? project [25] attempts to present information about communty events in a non-distracting fashion and is extensible with new sources. The Scope differs from these systems in several ways: it presents notifications on a uniform axis of urgency, it provides a single point of visual focus, and it abstracts information in an attempt to reduce distraction.

The Vista system [11] explored decision-theoretic approaches to the display of auxiliary views on information and the highlighting of important information for flight engineers at NASA Mission Control. Measures being tracked by engineers were selectively highlighted in place with color, depending on the inferred importance of the information in different situations.

There have been multiple studies of the disruptiveness of notifications. McFarland found that task performance is better when the user has control over the delivery of interruptions [16]. Gillie and Broadbent report [9] that cognitively taxing interruptions are harmful to task performance, suggesting to us that a glanceable awareness display can be valuable. Czerwinski et al. [3] specifically describe the cost of interruptions from instant messaging on task performance, varying the main task and time of interruption. Cutrell et al. [4] examine the influence of notifications on memory and explore factors in reorienting to a task following a notification.

A number of researchers have investigated the issues involved in peripheral display design. Lim et al. [14] examined the best screen locations for placing information that users may want to glance at. The results suggested the bottom right display corner as the best balance between noticeability and distraction, guiding us in the default placement of the Scope. Maglio and Campbell [15] and Entin [8] showed that continuously updating peripheral displays are more distracting than discrete updates (those that start and stop) and more disruptive to ongoing tasks. It is for this reason that the scope uses periodic updates that are visually subtle. Bartram et al. [1] studied the use of motion in peripheral displays. This work suggested that a slow blink motion for new items provides a good balance: it is not too distracting but still very detectable.

8 FUTURE WORK

Our user study suggested several immediate refinements, which guided the redesign of the visualization. The three major open issues in work on the Scope are discoverability, scalability, and easy identification of multiple items.

As described above, it is difficult to design visual annotations that are intuitively recognizable, simple looking, and combinable with other properties. Currently, users need an opportunity to familiarize themselves with visual attributes in a training session. To ease the familiarization process we have implemented easily accessible legend that we plan to test in future user studies. Further, we are exploring the value of adding a summary of active attributes in each item's tooltip, with text describing the semantics.

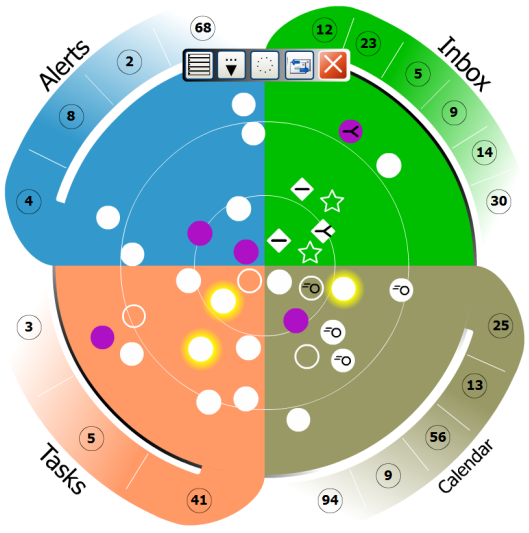


Figure 11 Scope design with "more items" summary

Next, we need to ensure that the Scope can scale to numbers of notifications that users receive. Currently, the Scope can accommodate at most 250 items. To increase this number, we have designed (but not yet implemented or tested) several features. First, we can drop lower-urgency items off the Scope, and adjust the high/normal/low zones as appropriate. Alternately, Figure 11 illustrates a design we are currently implementing, in which low-urgency items that were dropped from the main Scope are summarized. Each sector has a region that extends outward and then along the periphery of the sector. This "tail" contains several discrete visual indicators ("buckets") for how many items have been dropped and which urgency ranges the items are in. A tail with more "buckets" indicates that more of the urgency scale has dropped off the main Scope display. (*e.g.*, items with scores 0-30 in the Tasks sector versus 0-50 in the Calendar sector.) Alternatively, the wedge expansion interaction described earlier can be used to enlarge one specific wedge and gain space to see more items beyond the current urgency threshold.

Second, we are revising the wedge-specific filter buttons to make them maximally useful based on user feedback. The current design adds filters for email versus voicemail, one-on-one meetings, tasks that are due today, and for common categories of alerts. Third, we can provide a distortion mode in which the center (high urgency) region is given more space while outer regions are compressed—in the manner of fish-eye displays. Finally, if we cluster items based on content, we can collapse related low-urgency items into group objects, revealing individual items upon inspection.

Several people have commented that the high-urgency items are currently located in the smallest zone, and that most space is dedicated to low urgency items—and have suggested that the layout might be inverted. Although we continue to entertain this idea, two issues speak against it. First, we expect (and observe) that users receive far fewer urgent items than normal or low-urgency items. Second, having all urgent items at the center of the Scope makes them glanceable without requiring a larger visual search.

Currently, proximity to the center encodes urgency, but we do not make effective use of the radial placement within a wedge. There is an opportunity to group items within a wedge according to a semantic clustering. For example, items on similar topics, from the same sender, or about related projects would be placed together, allowing the user to learn a spatial map of the layout over time, and anticipate what an item might relate to without further inspecting it.

An important challenge that is not handled well by the current Scope design is the efficient display of specific details about multiple items. Users get metadata information about items at a glance, and this helps them decide which items to look at, but to explore the details of a specific item, they have to interrogate each item in turn to raise a tooltip. We are pursuing the creation of a mechanism that could provide easy access to detailed information on the *n* most urgent items, and a sweep-to-reveal gesture which would allow users to access tooltips on several colocated items. Figure 12 shows a user controllable lens that reveals details on the four most urgent items. A similar technique is useful to quickly find items one has seen before but now misplaced.

Finally, as the Scope prototype is implemented entirely using scalable vector graphics, it can be transferred to displays smaller than a desktop monitor. We intend to port the Scope application to the Pocket PC PDA platform, and even envision the possibility of a Scope display rendered on a wristwatch, for which the circular design is ideally suited.

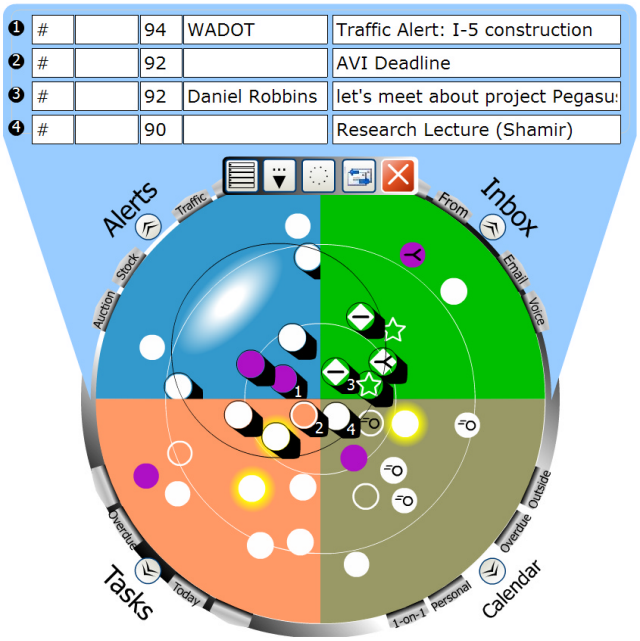


Figure 12 Multi-item detail inspection

9 CONCLUSIONS

We have described a novel visualization and application that helps people manage the increasing numbers of notifications from multiple applications. We believe that the Scope represents a new direction of research on information visualization for awareness. The Scope unifies many notification sources, providing a single interface for users to "stay on top" of their work and communications. By coupling visual design and information visualization techniques, we have created an application that is highly glanceable, reducing the mental effort users have to spend to decide what to attend to. Initial usability results suggest that, while the Scope is not immediately intuitive to grasp, information workers come to understand it in a brief session, and to appreciate its functionality. We believe that, as users increasingly experience notification overload, designs like the Scope will become increasingly valuable.

10 ACKNOWLEDGMENTS

We thank Ken Hinckley, Gina Venolia, Ed Cutrell, George Robertson and our usability participants for valuable comments and feedback.

11 REFERENCES

1. Bartram, L., Ware, C., and Calvert, T. Moving Icons, Detection and Distraction, *Interact 2001*, Tokyo, July.

2. Broadbent, D.E. (1958). *Perception and communications*. London: Pergamon Press.

3. Cadiz, J.J., Gupta, A., Jancke, G., and Venolia, G.D. Sideshow: Providing Peripheral Awareness of Important Information. Microsoft Research Tech Report MSR-TR-2001-83 (2001).

4. Cutrell, E., Czerwinski, M. and Horvitz, E.. Notification, Disruption, and Memory: Effects of Messaging Interruptions on Memory and Performance, *Proceedings of Interact 2001: IFIP Conference on Human-Computer Interaction*, Tokyo, Japan, July 2001.

5. Czerwinski, M., Cutrell, E. & Horvitz, E. (2000). Instant Messaging and Interruption: Influence of Task Type on Performance, In *Paris, C., Ozkan, N., Howard, S. and Lu, S. (Ed's.), OZCHI 2000 Conference Proceedings*, Sydney, Australia, Dec. 4-8, pp. 356-361.

6. Deutsch, J., & Deutsch, D. (1963). Attention: Some theoretical considerations. *Psychological Review*, 70, 80-90.

7. Dourish, P. and Bly, S., Portholes: supporting awareness in a distributed work group, in *CHI'92 Conference Proceedings on Human Factors in Computing Systems*, 1992, pp541 – 547.

8. Entin, E. (2000). Representing and visualizing a dynamically changing tactical situation. In *Proceedings of the IEA 2000/HFES 2000 Congress*. San Diego, CA: Human Factors and Ergonomics Society, pp. 427-430.

9. Gillie, T. and Broadbent, D. What makes interruptions disruptive? A study of length, similarity and complexity *Psychological Research*, 50, 243-250. (1989).

10. Greenberg, S.. Peepholes: Low cost awareness of one's community. In *ACM CHI '96 Companion*, pp206--207, Vancouver, BC, Canada, April 1996.

11. Horvitz, E. and Barry, M. Display of Information for Time-Critical Decision Making. In: *Proceedings of Eleventh Conference on Uncertainty in Artificial Intelligence*, Montreal, August 1995, pages 296-305. Morgan Kaufmann: San Francisco.

12. Horvitz, E., Jacobs, A., and Hovel, D. Attention-Sensitive Alerting. In: *Proceedings of UAI '99, Conference on Uncertainty and Artificial Intelligence*, Stockholm, Sweden, July 1999. Morgan Kaufmann: San Francisco. pp. 305-313.

13. Ishii, H. and Ullmer, B. Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, in *Proc. of CHI'97*, New York: ACM Press, pp 234-241.

14. Lim, R.W., & Wogalter, M.S. (2000). The position of static and on-off banners in WWW displays on subsequent recognition. In *Proceedings of the IEA 2000/HFES 2000 Congress*. San Diego, CA: Human Factors and Ergonomics Society, pp. 420-423.

15. Maglio, P.P. & Campbell, C.S. Tradeoffs in displaying peripheral information. In *Proc. of ACM CHI 2000 Human Factors in Computing Systems,* 241-248.

16. McFarland, D. (1999). Coordinating the interruptions of people in human-computer interaction. In: *Human-Computer Interaction—Interact '99*, Sasse, M.A. & Johnson, C. Eds., IOS Press, Inc., IFIP TC 13, 295-303.

17. Microsoft Corporation, *Microsoft Windows User Experience*. Microsoft Press, 1999.

18. Miller, T. and Stasko, J. "InfoCanvas: Information Conveyance through Personalized, Expressive Art", in *Proceedings of CHI 2001*, ACM, 2001, Seattle, USA.

19. Nielsen, J. Why You Only Need to Test With 5 Users. In *Alertbox*, March 19, 2000 at http://www.useit.com/alertbox/20000319.html

20. Nielsen, J. and Landauer, T. K. A Mathematical Model of the Finding of Usability Problems. In *CHI'93 Conference Proceedings on Human Factors in Computing Systems*. ACM Press 1993, Amsterdam, The Netherlands.

21. Parthasarathy, S. The Simplest Way to Define .NET. http://www.microsoft.com/net/define_net.asp

22. Robertson, G. , Czerwinski, M., Larson, K., Robbins, D., Thiel, D. & van Dantzich, M. Data Mountain: Using Spatial Memory for Document Management, In: *Proceedings of UIST '98, 11th Annual Symposium on User Interface Software and Technology*, pp. 153-162. ACM Press, 1998.

23. Weiser, M., Brown, J. S., Designing Calm Technology. http://ubiq.com/hypertext/weiser/calmtech/calmtech.htm

24. Whittaker, S., and Sidner, C. 1996. Email Overloading: Exploring Personal Information Management of Email. In *Proceedings of Conference on Human Factors in Computing Systems (CHI 96),* 276--283.

25. Zhao, Q. A. and Stasko, J. T. What's Happening? The Community Awareness Application. In *ACM CHI 2000 Extended Abstracts*, pp 253-254. ACM Press, 2000.