UNIT-IV

Quality of Service: Introduction, Models of Response-Time impacts, Expectations and attitudes, User Productivity, Variability in Response Time, Frustrating Experiences

Balancing Function and Fashion: Introduction, Error Messages, Nonanthropomorphic Design, Display Design, Web Page Design Window Design, Color

41 Introduction:

In the 1960s, user perception of computer speed was determined by response time for mathematical computations, program compilations, or database searches. Then, as time-shared systems emerged, contention for the scarce computational resources led to more complex reasons for delays. With the emergence of the World Wide Web, user expectations for expanded services grew, along with still more complex explanations of delays. Users now have to understand the size differences between text and graphics pages to appreciate the huge variations in server loads, and to tolerate network congestion. They also have to understand the multiple sources of dropped connections, unavailable web sites, and network outages. This complex set of concerns is usually discussed under the term *Quality of Service.* Concern over Quality of Service stems from a basic human value: Time is precious. When externally imposed delays impede progress on a task, many people become frustrated, annoyed, and eventually angry. Lengthy or unexpected system response times and slow display rates produce these reactions in computer users, leading to frequent errors and low satisfaction. Some users accept the situation with a shrug of their shoulders, but most users prefer to work more quickly than the computer allows. Discussions of Quality of Service must also take into account a second basic human value: Harmful mistakes should be avoided. However, balancing rapid performance with low error rates sometimes means that the pace of work must slow. If users work too quickly, they may learn less, read with lower comprehension, commit more data-entry errors, and make more incorrect decisions.

Athird Quality of Service value is to reduce user frustration. With long delays, users may become frustrated enough to make mistakes or give up working. Delays are often a cause of frustration, but there are others, such as crashes that destroy data, software bugs that produce incorrect results, and poor designs that lead to user confusion. Networked environments generate further frustrations: unreliable service providers, dropped lines, e-mail spam, and malicious viruses.

4.2 Models of Response-Time impacts:

- Response time
  - The number of seconds it takes from the moment users initiate an activity until the computer presents results on the display
- User think time
  - The number of seconds the user thinks before entering the next action
  - Designers of response times and display rates in HCI must consider:
  - complex interaction of technical feasibility
  - cost
  - task complexity
  - user expectations
  - speed of task performance
  - error rates
  - error handling procedures
  - Overall majority of users prefer rapid interactions
  - Lengthy response times (15 seconds) are detrimental to productivity
  - Rapid response times (1 second or less) are preferable, but can increase errors for complex tasks
  - Display Rate
  - Alphanumeric displays: The speed in characters per second at which characters appear for the user to read
  - World Wide Web Applications: Display rate may be limited by network transmission speed or server performance

- Reading textual information from a screen is a challenging cognitive and perceptual task
- Users relax when the screen fills instantly- beyond a speed where someone may feel compelled to keep up
- Cognitive human performance would be useful for:
- making predictions
- designing systems
- formulating management policies

Designers can optimize web pages to reduce byte counts and numbers of files or provide previews of materials available in digital libraries or archives to help reduce the number of queries and accesses to the network

Models of response-time impacts: Limitations of short-term and working memory

- Any cognitive model must emerge from an understanding of human problem-solving abilities
- Magic number seven - plus or minus two
  - The average person can rapidly recognize seven chunks of information at a time
  - This information can be held for 15 to 30 seconds in short-term memory
  - Size of the chunks depends on the person' s familiarity with the material
- Short-term memory and working memory are used in conjunction for processing information and problem solving
  - Short-term memory processes perceptual input
  - Working memory generates and implements solutions
- People learn to cope with complex problems by developing higher-level concepts using several lower-level concepts brought together into a single chunk
- Short term and working memory are highly volatile
  - Disruptions cause loss of memory
  - Delays require that memory be refreshed
  - Source of errors
  - Solutions to problems must be recorded to memory or implemented
  - Chance of error increases when solutions are recorded
  - When using an interactive computer system users may formulate plans and have to wait for execution time of each step
  - Long (1976) found unskilled and skilled typists worked more slowly and made more errors with longer response times
- For a given user and task, there is a preferred response time
  - Conditions for optimum problem solving
  - Longer response time causes uneasiness in the user because the penalty for error increases
  - Shorter response time may cause the user to fail to comprehend the presented materials
  - Progress indicators shorten perceived elapsed time and heighten satisfaction:
  - graphical indicators
  - blinking messages
  - numeric seconds left for completion

Conditions for optimum problem solving (cont.)

- Rapid task performance, low error rates, and high satisfaction can come from:
  - Users have adequate knowledge of the objects and actions necessary for the problem-solving task
  - The solution plan can be carries out without delays
  - Distractions are eliminated

  - User anxiety is low
  - There is feedback about progress toward solution
  - Errors can be avoided or handled easily
- Other conjectures in choosing the optimum interaction speed
  - Novices may exhibit better performance with slower response time
  - Novices prefer to work at slower speeds
  - With little penalty for an error, users prefer to work more quickly
  - When the task is familiar and easily comprehended, users prefer more rapid action
  - If users have experienced rapid performance previously, they will expect in future situations

4.3 Expectations and attitudes:

- Related design issues may clarify the question of acceptable response time
  - E.g. how long before hearing a dial-tone
- Two-second limit (Miller, 1968) appropriate for many tasks
- But users have adapted a working style and expectation based on responses within a fraction of a second
- People can detect 8% changes in a 2-4 second response time

- A system is slowed down when the load is light and potential performance high
- Makes the response time more uniform over time and across users, avoiding expectations that can"t always be met
- Previous experiences
- The individual's tolerance for delays
- Task complexity

4.4 User productivity

- Repetitive tasks
  - Nature of the task has a strong influence on whether changes in response time alter user productivity
  - Shorter response time means users responds more quickly, but decisions may not be optimal
  - Goodman and Spence (1981) – reduced response time lead to more productivity
  - Teal and Rudnecky (1992) – slower response time lead to more accuracy

Problem solving tasks

  - Users will adapt their work style to the response time
  - Users will change their work habits as the response time changes
  - Grossberg, Wiesen, and Yntema (1976) – the time to solution was invariant with respect to response time

- Summary
  - Users pick up the pace of the system to work more quickly with shorter response time
  - Higher throughput of work demands more attention must be paid to minimizing the cost of delay of error recovery

4.5 Variability in Response Time

People are willing to pay substantial amounts of money to reduce the variability in their lives. The entire insurance industry is based on the reduction of present pleasures, through the payment of premiums, to reduce the severity of potential future losses. Most people appreciate predictable behavior that lessens the anxiety of contemplating unpleasant surprises. When using computers, users cannot see into the machines to gain reassurance that their actions are being executed properly, but the response time can provide a clue. If users come to expect a response time of 3 seconds for a common action, they may become apprehensive if this action takes 0.5 or 15 seconds. Such extreme variation is unsettling and should be prevented or acknowledged by the interface, with some indicator for an unusually fast response or a progress report for an unusually slow response. The more difficult issue is the effect of modest variations in response time. As discussed earlier, Miller (1968) raised this issue and reported that 75% of participants tested could perceive 8% variations in time for periods in the interval of 2 to 4 seconds. These results prompted some designers to suggest restrictive rules for variability of response time. Since it may not be technically feasible to provide a fixed short response time (such as one second) for all actions, several researchers have suggested that the time be fixed for classes of actions. Many actions could have a fixed response time of less than 1 second, other actions could take 4 seconds, and still other actions could take 12 seconds. Experimental results suggest that modest variations in response time do not severely affect performance. Users are apparently capable of adapting to varying situations, although some of them may become frustrated when performing certain tasks. Goodman and Spence (1982) measured performance changes as aresult of response-time variation in a problem-solving situation (a similar situation was used in their earlier experiment, described in Section 11.4). They found no significant performance changes as the variability was increased. The time to solution and the profile of command use were unchanged. As the variability increased, participants took advantage of fast responses by entering subsequent commands immediately, balancing the time lost in waiting for slower responses. Similar results were found by other researchers.

The physiological effect of response time is an important issue for stressful, long-duration tasks such as air-traffic control, but it is also a concern for office workers and sales personnel. While no dramatic differences have been found between constant and variable response-time treatments, statistically significantly higher error rates, higher systolic blood pressure, and more pronounced pain symptoms were found repeatedly with shorter response times (Kohlisch and Kuhmann, 1997). Similarly, a study of database queries compared an 8second constant response time to a variable response time ranging from 1 to 30 seconds (mean 8 seconds) (Emurian, 1991). Although diastolic blood pressure and masseter (jaw-muscle) tension did increase when compared to resting baseline values, there were no significant differences in these physiological measures between constant and variable treatments.

4.6 Frustrating experiences

- Quality of Service is usually defined in terms of network performance, but another perspective is to think about the quality of user experiences. Many technology promoters argue that the quality of user experiences with computers has been improving over the past four decades, pointing to the steadily increasing chip and network speeds and hard-drive capacities. However, critics believe frustra tion from interface complexity, network disruptions, and malicious interference has grown. Recent research has begun to document and understand the sources of user frustration with contemporary user interfaces.
- User surveys elicit strong responses that convey unsatisfactory experiences in the general population. A British study of 1,255 office workers by a major computer manufacturer found that nearly half of the respondents felt frustrated or stressed by the amount of time it takes to solve problems. In an American survey of *6,000* computer users, the average amount of wasted time was estimated as 5.1 hours per week.
- Replacing these possibly exaggerated impressions with more reliable data is a serious challenge. Self-reports and observations from more than 100 users doing their mvn work for an average of 2.5 hours each produced disturbing results: 46 to *53 Gl<* of the users' time was seen as being wasted (Ceaparu et aI., 2004). Frequent complaints included dropped network connections, application crashes,
- (Ceaparu et al., 2004) 46% to 53% of users‟ time was seen as being wasted
- Recommendations include improving the quality of service and changes by the user
- Poor quality of service is more difficult in emerging markets and developing nations
- User training can help
- Email a common application, but also a common source of frustration
- Viruses also a problem
- Since frustration, distractions, and interruptions can impede smooth progress, design strategies should enable users to maintain concentration.
- Three initial strategies can reduce user frustration:
- Reduce short-term and working memory load
- Provide information abundant interfaces
- Increase automaticity
- Automaticity in this context is the processing of information (in response to stimuli) in a way that is automatic and involuntary, occurring without conscious control.
    - An example is when a user performs a complex sequence of actions with only a light cognitive load, like a driver following a familiar route to work with little apparent effort.

4.7 Balancing Function and Fashion

4.7 Introduction:

User experiences play a critical role in influencing software acceptance

- Conversational messages have their limits; informative messages are better
- Design needs to be comprehensible, predictable, and controllable
- Information layout is important
- Multiwindow coordination can be difficult
- Large, fast, high-resolution color displays have potential

4.8 Error Messages:

User experience with computer-system prompts, explanations, error diagnostics, and warnings is crucial in influencing acceptance of SW systems

Lack of knowledge, incorrect understanding, inadequate slips

Users are likely to be confused, are anxious or feel inadequate

Make error messages as user-friendly as possible; this is especially important for novice users as they commonly have a lack of knowledge, confidence, and are sometimes easily frustrated or discouraged

Improving Error Messages

on these issues handling procedures, improve documentation and training manuals, change permissibleactions, etc.

peers, managers, should be tested empirically, and be included in user manuals Specificity, constructive guidance, positive tone, user-centered style, and appropriate physical format are recommended as the bases for preparing error messages (Box 12.1). These guidelines are especially important when the users are novices, but they can benefit experts as well. The phrasing and contents of error messages can significantly affect user performance and satisfaction.

4.8.1 Specificity:

Messages that are too general make it difficult for the novice to know what has gone wrong. Simple and condemning messages are frustrating because they provide neither enough information about what has gone wrong nor the knowledge to set things right. The right amount of specificity therefore is important.

Poor	**Better**
SYNTAX ERROR	Unmatched left parenthesis
ILLEGAL ENTRY	Type first letter: **S**end, **R**ead, or **D**rop
INVALID DATA	Days range from 1 to 31
BAD FILE NAME	File names must begin with a letter

Error-message guidelines for the end product and for the development process.These guidelines are derived from practical experience and empirical data.

Guidelines

- Product
- Be as specific and precise as possible
- Be constructive: Indicate what the user needs to do
- Use a positive tone: Avoid condemnation
- Choose user centered phrasing

4.8.2 Constructive guidance and positive tone:

Rather than condemning users for what they have done wrong, messages should, where possible, indicate what users need to do to set things right:

Poor	Better
Run-Time error „-2147469 (800405): Method „Private Profile String" of object „System" failed.	Virtual memory space consumed. Close some programs and retry.
Resource Conflict Bus: 00 Device: 03 Function: 01	Remove your compact flash card and restart
Network connection refused.	Your password was not recognized. Please retype.
Bad date.	Drop-off date must come after pickup date.

Unnecessarily hostile messages using violent terminology can disturb nontechnical users. An interactive legal-citation-searching system uses this message: FATAL *ERROR*, RUN ABORTED. An early operating system threatened many users with CATASTROPHIC ERROR; LOGGED WITH OPERATOR. There is no excuse for these hostile messages; they can easily be rewritten to provide more information about what happened and what must be done to set things right. Where possible, be constructive and positive. Such negative words as ILLEGAL, ERROR, INVALID, or BAD should be eliminated or used infrequently.

4.8.3 User-centered phrasing:

- Suggests user controls the interface, initializing more than responding
- User should have control over amount of information system provides e.g. screen tips; a help button for context-sensitive help or an extensive online user manual

Telephone company, “We‟re sorry, but we are unable to complete your call as dialed. Please hang up, check your number, or c onsult the operator for assistance”, versus “Illegal telephone number. Call aborted. Error number 583-2R6.9. Consult your user manual for further information

4.8.4 Appropriate physical format:

– use uppercase-only messages for brief, serious warnings
– avoid code numbers; if required, include at end of message
– There is debate over best location of messages. E.g. Could be:
    • near where problem arose
    • placed in consistent position on bottom of screen
    • near to, but not obscuring relevant information
– audio signals useful, but can be embarrassing - place under user control

4.8.5 Development of effective messages:

– Messages should be evaluated by several people and tested with suitable participants
– Messages should appear in user manuals and be given high visibility
– Users may remember the one time when they had difficulties with a computer system rather than the 20 times when everything went well

Experimental studies support the contention that improving messages can upgrade performance and result in greater job satisfaction. They have led to the following recommendations for system developers

1. Increase attention to message design. The wording of messages should be considered carefully. Technical writers or copy editors can be consulted about the choice of words and phrasing to improve both clarity and consistency.

2. Establish quality control. Messages should be approved by an appropriate quality-control committee consisting of programmers, users, and humanfactors specialists. Changes or additions should be monitored and recorded.

3. Develop guidelines. Be as specific and precise as possible. Writing good messages- like writing good poems, essays, or advertisements-requires experience, practice, and sensitivity to how the reader will react. It is a skill that can be acquired and refined by programmers and designers who are intent on serving the user. However, perfection is impossible, and humility is the mark of the tme professional.

4. Carry out usability tests System messages should be subjected to a usability test with an appropriate user community to determine whether they are comprehensible. The test could range from a rigorous experiment with realistic situations (for life-critical or high-reliability systems) to an informal reading and review by interested users (for personal computing or noncritical applications).
Complex interactive systems that involve thousands of users are never really complete until they are obsolete. Under these conditions, the most effective designs emerge from iterative testing and evolutionary refinement.

5. Record the frequency of occurrence for each message. Frequency counts should be collected for each error condition whenever possible, particularly during usability tests. If possible, the users' actions should be captured for more detailed study. If you know where users run into difficulties, you can then revise the error messages, improve the training, modify the manual, or

change the interface. The error rate per 1,000 actions should be used as a metric of interface quality and a gauge of how improvements affect performance. An error-counting option is useful for internal systems and can be a marketing feature for software products.

4.9 Nonanthropomorphic Design

There is a great temptation to have computers "talk" as though they were people. It is a primitive urge that designers often follow, and that children and many adults accept without hesitation (Nass et a1., 1995; Reeves and Nass, 1996). Children accept human-like references and qualities for almost any object, from Humpty Dumpty to Tootle the Train. Adults reserve the *l1/lthropomorphic* references for objects of special attraction, such as cars, ships, or computers. The words and graphics in user interfaces can make important differences in people's perceptions, emotional reactions, and motivations. Attributions of intelligence, autonomy, free will, or knowledge to computers are appealing to some people, but to others such characterizations may be seen as deceptive, confusing, and misleading. The suggestion that computers can think, know, or understand may give users an erroneous model of hov\,' computers work and what the machines' capacities are. Ultimately, the deception becomes apparent, and users may feel poorly treated. Martin (1995/96) carefully traces the media impact of the 1946 ENIAC announcements: "Readers wer~ given hyperbole designed to raise their expectations about the use of the new electronic brains.... This engendered premature enthusiasm, which then led to disillusionment and distrust of computers on the part of the public when the new technology did not live up to these expectations."

A second reason for using nonanthropomorphic phrasing is to clarify the differences between people and computers. Relationships with people are different from relationships with computers. Users operate and control computers, but they respect the unique identity and autonomy of individuals. Furthermore, users and designers must accept responsibility for misuse of computers, rather than blaming the machines for errors. It is worrisome that, in one study, 24 of 29 computer-science students believed that computers can have intentions or be independent decision makers, and 6 consistently held computers morally responsible for errors (Friedman, 1995).

A third motivation is that, although an anthropomorphic interface may be attractive to some people, it can be distracting or produce anxiety for others. Some people express anxiety about using computers and believe that computers "make you feel dumb." Presenting the computer through the specific functions it offers may be a stronger stimulus to user acceptance than is promoting the fantasy that the computer is a friend, parent, or partner. As users become engaged, the computer becomes transparent, and they can concentrate on their writing, problem solving, or exploration. At the end, they have the experience of accomplishment and mastery, rather than the feeling that some magical machine has done their job for them. Anthropomorphic interfaces may distract users from their tasks and waste their time as they consider how to please or be socially appropriate to the onscreen character.

Historical precedents of failed anthropomorphic bank tellers such as Tillie the Teller, Harvey Wallbanker, and BOB (Bank of Baltimore) and of abandoned talking automobiles and soda machines do not seem to register on some designers. The bigger-than-life-sized Postal Buddy was supposed to be cute and friendly while providing several useful automated services, but users rejected this pseudo-postal clerk after the project had incurred costs of over $1 billion. The web-based news reader Ananova was heralded as the future of computing, but it has fallen into disuse (Fig. 12.1). Advocates of anthropomorphic interfaces suggest that they may be most useful as teachers, salespeople, therapists, or entertainment figures.

- Advocates of anthropomorphic interfaces suggest that they may be most useful as teachers, salespeople, therapists, or entertainment figures
- An alternative design is to present a human author of a package through prerecorded audio or video
- Guidelines
  - Be cautious in presenting computers as people.
  - Design comprehensible, predictable, and controllable interfaces.
  - Use appropriate humans for introductions or guides.
  - Use cartoon characters in games or children‟s software, but usually not elsewhere
  - Provide user-centered overviews for orientation and closure.
  - Do <u>not</u> use 'I' pronouns when the computer responds to human actions.
    - Eudora frequently did this
  - Use "you" to guide users, or just state facts.

4.10 Display Design

Deals with layout of information on the display

Samples of the 162 data-display guidelines from Smith and Mosier (1986).

• Ensure that any data that a user needs, at any step in a transaction sequence, are *available* for display.

• Display data to users in directly usable forms; do not require that users convert displayed data.

• Maintain a consistent format, for any particular type of data display, from one display to another.

• Use short, simple sentences.

• Use affirmative statements, rather than negative statements.

• Adopt a logical principle by which to order lists; where no other principle applies, order lists alphabetically.

• Ensure that labels are sufficiently close to their data fields to indicate association, yet are separated from their data fields by at least one space.

• Left-justify columns of alphabetic data to permit rapid scanning.

• Label each page in multipaged displays to show its relation to the others.

• Begin every display with a title or header, describing briefly the contents or purpose of the display; *leave* at least one blank line between the title and the body ofthe display.

• For size coding, make larger symbols be at least 1.5 times the height of the next-smaller symbol.

• Consider color coding for applications in which users must distinguish rapidly among *several* categories of data, particularly when the data items are dispersed on the display.

• When you use blink coding, make the blink rate 2 to 5 Hz, with a minimum duty cycle (ON interval) of 50%.

• For a large table that exceeds the capacity of one display frame, ensure that users can see column headings and row labels in all displayed sections of the table.

• Provide a means for users (or a system administrator) to make necessary changes to display functions, if data-display requirements may change (as is often the case).

systems. They propose six categories of principles that reveal the complexity of the designer's task:

1. Elegallce lIlId simplicity: Unity, refinement, and fitness

2. Scale. cOIltrast. mId proportioll: Clarity, harmony, activity, and restraint

3. Orgmli:::atioll alld l'islIal structure: Grouping, hierarchy, relationship, and balance *Module tJnd program:* Focus, flexibility, and consistent application

5. Image tJnd representation: Immediacy, generality, cohesiveness, and characterization

6. Style: Distinctiveness, integrity, comprehensiveness, and appropriateness This section deals with some of these issues, offering empirical support for concepts where available.

4.10.1 Field layout

Exploration with a variety of layouts can be a helpful process. These design alternatives should be developed directly on a display screen. An employee record with information about a spouse and children could be displayed crudely as follows: Poor: TAYLOR, SUSAN034787331 WILLIAM TAYLOR

THOMAS 10291974ANN08211977ALEXANDRA09081972

This record may contain the necessary information for a task, but extracting the information will be slow and error prone. As a first step at improving the format, blanks and separate lines can distinguish fields:

Better: TAYLOR, SUSAN 034787331WILLIAM TAYLOR

THOMAS 10291974

ANN 08211977

ALEXANDRA. 09081972

The children's names can be listed in chronological order, with alignment of the dates. Familiar separators for the dates and the employee'S social security number also aid recognition:

Better: TAYLOR, SUSAN 034-78-7331

ALEXANDRA. 09-08-1972

THOMAS 10-29-1974

ANN 08-21-1977

WILLIAM TAYL

The reversed order of "last name, first name" for the employee may be desired to highlight the lexicographic ordering in a long file. Ho\.vever, the "first name, last name" order for the spouse is usually more readable. Consistency seems important, 5(1 a compromise might be made:

Better: SUSAN TAYLOR 034-78-7331 WILLIAM TAYLOR

ALEXANDRA 09-08-1972

THOMAS 10-29-1974

ANN 08-21-1977

For frequent users, this format may be acceptable, since labels have a cluttering effect. For most users, however, labels will be helpful:

Better: Employee: SUSAN TAYLOR

Social Security Number: 034-78-7331

Spouse: WILLIAM TAYLOR

Children: Names Birthdates

ALEXANDRA 09-08-1972

THOMAS10-29-1974

ANN 08-21-1977

4.10.2 Empirical results:

Structured form superior to narrative form

- Improving data labels, clustering related information, using appropriate indentation and underlining, aligning numeric values, and eliminating extraneous characters improves performance
- Performance times improve with fewer, denser displays for expert users
- screen contents should contain only task-relevant information
- Consistent location, structure, and terminology across displays important
  - Sequences of displays should be similar throughout the system for similar tasks

4.10.3 Display-complexity metrics:

Although knowledge of the users' tasks and abilities is the key to designing effective screen displays, objective and automatable metrics of screen complexity are attractive aids. After a thorough review of the literature, Tullis (1997) developed four task-independent metrics for alphanumeric displays:

1. Overall density. The number of filled character spaces as a percentage of total spaces available.

2. Local density. The average number of filled character spaces in a five-degree visual angle around each character, expressed as a percentage of available spaces in the circle and weighted by distance from the character.

3. Grouping. (1) The number of groups of "connected" characters, where a connection is any pair of characters separated by less than twice the mean of the distances between each character and its nearest neighbor; and (2) the average visual angle subtended by groups and weighted by the number of characters in the group.

4. Layout complexity. The complexity, as defined in information theory. of the distribution of horizontal and vertical distances of each label and data item from a standard point on the display.

4.11 Window Design

Most computer users must consult documents, forms, e-mails, web pages, and more to complete their tasks. For example, a travel agent jumps from reviewing a client's e-mail request to viewing the proposed itinerary to reviewing calendars and flight schedules to choosing seat assignments and to selecting hotels. ven with large desktop displays, there is a limit to how many documents can

be displayed simultaneously. Designers have long struggled with strategies to offer users sufficient information and flexibility to accomplish their tasks while reducing window-housekeeping actions and minimizing distracting clutter. If tlsers' tasks are well understood and regular, then there is a good chance that an effective *l11ultiple-zcilzdmc display* strategy can be developed. The travel agent might start a client-itinerary window, review flight segments from a schedule window, and drag selected flight segments to the itinerary window. Windows labeled "Calendar," "Seat Selection," "Food Preferences," and "Hotels" might appear as needed, and then the charge-card information ,,,,indo\\' \vould appear to complete the transaction.

4.11.1 Coordinating multiple windows

Coordination is a task concept that describes how information objects change based on user actions. A careful study of user tasks can lead to taSk-specific coordinations based on sequences of actions. The especially interesting case of work with large images such as maps, circuit diagrams, or magazine layouts is covered in the next section. Other important coordinations that might be supported by interface developers include:

• *Synchronized scrolling.* A simple coordination is synchronized scrolling, in which the scroll bar of one window is coupled to another scroll bar, and action on one scroll bar causes the other to scroll the associated window contents in parallel. This technique is useful for comparing two versions of a program or document. Synchronization might be on a line-for-line basis, on a proportional basis, or keyed to matching tokens in the two windows.

• *Hierarchical browsing.* Coordinated windows can be used to support hierarchical browsing. For example, if one window contains a book's table of contents, selection of a chapter title by a pointing device should lead to the display, in an adjoining window, of the chapter contents. Hierarchical browsing was nicely integrated in Windows Explorer to allow users to browse hierarchical directories, in Outlook to browse folders of e-mails, and in many other applications.

The specification is on the left. As users click on components (DoubleAttrWebAdapter), the detail view in a Nassi-Sh neiderman Chart appears on the right. *Opening/closing of dependent windows.* An option on opening a window might be to simultaneously open dependent windows in a nearby and convenient location. For example, when users are browsing a program, if they open a main procedure, the dependent set of procedures could open up automatically. Conversely, in filling in a form, users might get a dialog box with a choice of preferences. That dialog box might lead the user to activate a pop-up or error-message window, which in turn might lead to an invocation of the help window. After the user indicates the desired choice in the dialog box, it would be convenient to have automatic closing of all the windows.

• *Saving/opening of window state.* A natural extension of saving a document or a set of preferences is to save the current state of the display, with all the windows and their contents. This feature might be implemented by the simple addition of a "Save screen as..." menu item to the "File" menu of actions. This action would create a new icon representing the current state; clicking on the icon would reproduce that state. This feature is a simple version of the rooms approach (Henderson and Card, 1986).

4.11.2 Image browsing

A two-dimensional cousin of hierarchical browsing

- Work with large images
- Overview in one window (context), detail in another (focus)
- Field of view box in the overview
- Panning in the detail view, changes the field of view box
- Matched aspect ratios between field of view box and the detail view.

• *Image generation.* Paint or construct a large image or diagram.

- • *Open-ended exploration.* Browse to gain an understanding of the map or image.
- • *Diagnostics.* Scan for fla\vs in an entire circuit diagram, medical image, or newspaper layout.
- • *Navigation.* Have knowledge of the overview, but need to pursue details along a highway or vein.
- • *Monitoring.* Watch the overview and, when a problem occurs. zoom in on details.

- Zoom factors: 5-30
  - Larger suggests an intermediate view is needed

- Semantic zooming
- Side by side placement, versus fisheye view

4.11.3 Personal role management:

The personal role manager could simplify and accelerate the performance of common coordination tasks, in the same way that graphical user interfaces simplify file-management tasks. The requirements for a personal role manager

Support a *unified framework* for information organization according to users' roles.

• Provide a *visual, spatial layout* that matches tasks.

• Support *1Hultiwindo'W actions* for fast arrangement of information.

Support *information access* with partial knowledge of an information item's nominal, spatial, temporal, and visual attributes and its relationships to otherpieces of information.

• Allow *fast switching* and *resumption* of roles.

• Free users' *cognitive resources* to work on *task-domain actions*, rather than making users concentrate on interface-domain actions.

• Use *screen space* efficiently and productively for tasks.

4.12: Color

Color displays are attractive to users and can often improve task performance, but the danger of misuse is high. Color can

• Soothe or strike the eye

• Add accents to an uninteresting display

• Facilitate subtle discriminations in complex displays

• Emphasize the logical organization of information

• Draw attention to warnings

• Evoke strong emotional reactions of joy, excitement, fear, or anger

The principles developed by graphic artists for using color in books, magazines, highway signs, and other print media have been adapted for user interfaces (Thorell and Smith, 1990; Travis, 1991; Marcus, 1992; Shubin, Falck, and Johansen, 1996). Programmers and interactive-systems designers are learning how to create effective computer displays and to avoid pitfalls (Salomon, *1990;* Galitz,2003; Brewer, Hatchard, and Harrower, 2003). There is no doubt that color makes video games more attractive to users, conveys more information on power-plant or process-control diagrams, and is necessary for realistic images of people, scenery, or three-dimensional objects (Foley et al., 1997; Weinman, *2002).* These applications require color. However, greater controversy exists about the benefits of color for alphanumeric displays, spreadsheets, graphs, and user-interface components. No simple set of rules governs use of color, but these guidelines are a starting point for designers:

• ***Use color consermtively***. Many programmers and novice designers are eager to use color to brighten up their displays, but the results are often counterproductive. One home-information system displayed the seven letters of its name in large letters, each 'with a different color. At a distance, the display appeared indting and eye catching; up close, hovvever, it was difficult to read.

Limit the number of colors. Many design guides suggest limiting the number of colors in a single display to four, with a limit of seven colors in the entire sequence of displays. Experienced users may be able to benefit from a larger number of color codes, but for novice users too many color codes can cause confusion.

• ***Recognize the power of color as a coding technique***. Color speeds recognition for many tasks. For example, in an accounting application, if data lines with accounts overdue more than 30 days are coded in red, they will be readily visible among the nonoverdue accounts coded in green. In air-traffic control, high-flying planes might be coded differently from low-flying planes to facilitate recognition. In programming workstations, keywords are color-coded differently from variables.

• ***Ensure that color coding supports the task***. Be aware that using color as a coding technique can inhibit performance of tasks that go against the grain of the coding scheme. If, in the above accounting application with color coding by days overdue, the task is now to locate accounts with balances of more than $55, the existing coding scheme may inhibit performance on the second task. Also, in the programming application, the color coding of recent additions may make it more difficult to read the entire program. Designers should attempt to make a close linkage between the users' tasks and the color coding, and offer users control where possible.

• ***Have color coding appear with minimal user effort.*** In general, the color coding should not have to be assigned by users each time they perform a task, but rather should appear automatically. For example, when users click on a \veb link to start the task of checking for overdue accounts, the color coding is set automatically. When the users click on the task of locating accounts with balances of more than $55, the new color-coding scheme should appear automaticallv.

• ***Place color coding IInder user control.*** When appropriate, the users should be able to tum off color coding. For example, if a spelling checker highlights possibly misspelled \\'ords in red, then the user should be able to accept or change the spelling and to turn off the coding. The presence of the highly visible red coding is a distraction from reading the text for comprehension.

• ***Design for monochrome first***. The primary goal of a display designer should be to layout the contents in a logical pattern. Related fields can be shown by contiguity or by similar structural patterns; for example, successive employee records may have the same indentation pattern. Related fields can also be grouped by a box drawn around the group. Unrelated fields can be kept separate by blank space-at least one blank line vertically or three blank characters horizontally. It may be advantageous to design for monochrome rather than relying on color, because color displays may not be universally available.

• ***Consider the needs of color-deficient users.*** One important aspect to consider is readability of colors by users with color blindness (either red/green confusion, the most common case, or total color blindness). Color impairment is a very common condition that should not be overlooked (Rosenthal and Phillips, 1997; Olson and Brewer, 1997). Approximately eight percent of males and less than one percent of females in North America and Europe have some permanent color deficiency in their vision. Many others have temporary deficiencies due to illness or medications.

• ***Use color to help* in *formatting***. In densely packed displays where space is at a premium, similar colors can be used to group related items. For example, in a police dispatcher's tabular display of assignments, the police cars on emergency calls might be coded in red, and the police cars on routine calls might be coded in green. Then, when a new emergency arises, it would be relatively easy to identify the cars on routine calls and to assign one to the emergency.

Be consistent in color coding. Use the same color-coding rules throughout the system. If some error messages are displayed in red, then make sure that every error message appears in red; a change to yellow may be interpreted as a change in the importance of the message. If colors are used differently by several designers of the same system, users will hesitate as they attempt to assign meaning to the color changes.

• ***Be alert to common expectations about color codes***. Designers need to speak to users to determine what color codes are applied in the task domain. From automobile-driving experience, red is commonly considered to indicate stop or danger, yellow is a warning, and green is all clear or go. In investment circles, red is a financial loss and black is a gain. For chemical engineers, red is hot and blue is cold. For map makers, blue means water, green means forests, and yellow means deserts.

Be alert to problems with color pairings. If saturated (pure) red and blue appear on a display at the same time, it may be difficult for users to absorb the information. Red and blue are on the opposite ends of the spectrum, and the muscles surrounding the human eye will be strained by attempts to produce a sharp focus for both colors simultaneously. The blue will appear to recede and the red will appear to come forward. Blue text on a red background would present an especially difficult challenge for users to read.

Use color changes to indicate status changes. For automobile speedometers with digital readouts and a wireless receiver of the driving speed limits, it might be helpful to change from green numbers below the speed limit to red above the speed limit to act as a "'laming. Similarly, in an oil refinery. pressure indicatorsmight change color as the value went above or below acceptable limits. In this way, color acts as an attention-getting method. This technique is potentially valuable when there are hundreds of values displayed continuously.

• ***Use color in graphic displays for greater information density***. In graphs with multiple plots, color can be helpful in showing which line segments form the full graph. The usual strategies for differentiating lines in black-on-white graphssuch as dotted lines, thicker lines, and dashed lines-are not as effective as using separate colors for each line. Architectural plans benefit from color coding of electrical, telephone, hot-water, cold-water, and natural-gas lines. Similarly, maps can have greater information density when color coding is used.