MODULE 2

Chapter 1 Managing Design Processes

Introduction

In the first decades of computer-software development, technically oriented programmers designed text editors, programming languages, and applicationsfor themselves and their peers.

Now, the user population for mobile devices, instant messaging, e-business, and digital libraries is so vastly different from the original that programmers' intuitions may be inappropriate.

Designs should be based on careful observation of current users, refined by thoughtful analysis of task frequencies and sequences, and validated through early usability and thorough acceptance tests.

Designers seek direct interaction with users during the design phase, the development process, and throughout the system lifecycle.

Iterative design methods that allow early testing of low-fidelity prototypes, revisions based on feedback from users

Organizational Design to Support Usability

When competitive products provide similar functionality, usability engineering is vital for product acceptance.

Many organizations have created usability laboratories to provide expert reviews and to conduct usability tests of products during development

Organizational design is a step-by-step methodology which identifies dysfunctional aspects of work flow, procedures, structures and systems, realigns them to fit current business realities/goals and then develops plans to implement the new changes. The process focuses on improving both the technical and people side of the business.

The hallmark of the design process is a comprehensive and holistic approach to organizational improvement that touches all aspects of organizational life, so you can achieve:

- Excellent customer service
- Increased profitability
- Reduced operating costs
- Improved efficiency and cycle time
- A culture of committed and engaged employees
- A clear strategy for managing and growing your business

A well-designed organization ensures that the form of the organization matches its purpose or strategy, meets the challenges posed by business realities and significantly increases the likelihood that the collective efforts of people will be successful.

As companies grow and the challenges in the external environment become more complex, businesses processes, structures and systems that once worked become barriers to efficiency, customer service, employee morale and financial profitability. Organizations that don't periodically renew themselves suffer from such symptoms as:

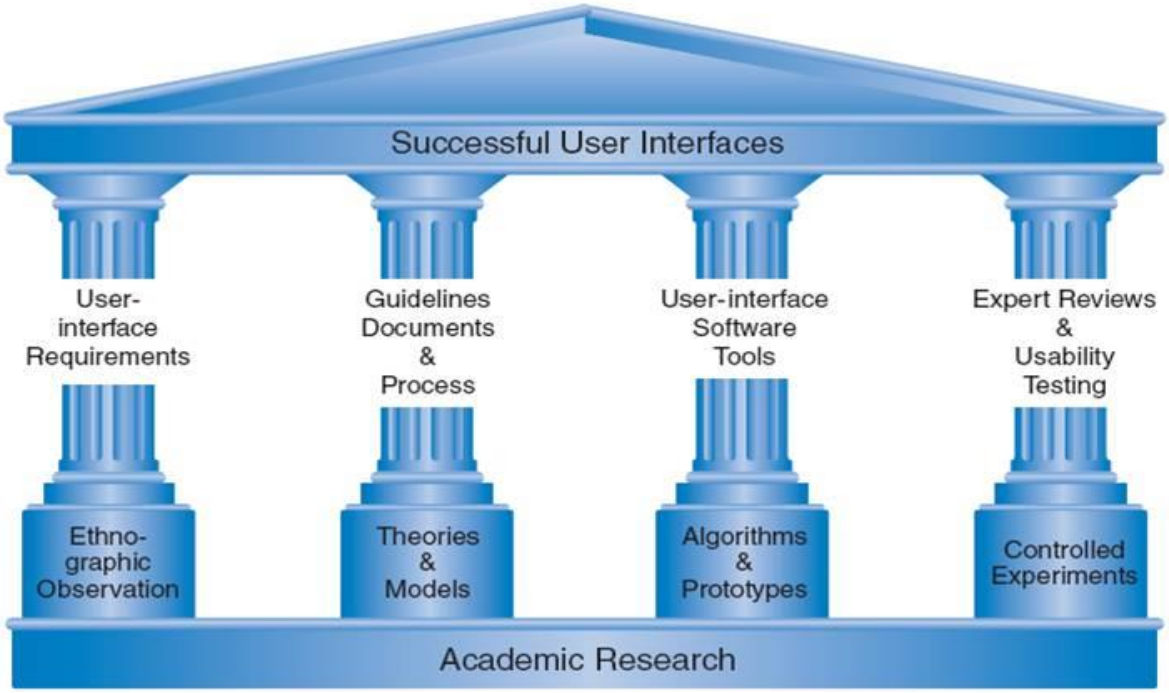
- Inefficient workflow with breakdowns and non value-added steps
- Redundancies in effort ("we don't have time to do things right, but do have time to do them over")
- Fragmented work with little regard for good of the whole (Production ships bad parts to meet their quotas)
- Lack of knowledge and focus on the customer
- Silo mentality and turf battles
- Lack of ownership ("It's not my job")
- Cover up and blame rather than identifying and solving problems
- Delays in decision-making
- People don't have information or authority to solve problems when and where they occur
- Management, rather than the front line, is responsible for solving problems when things go wrong
- It takes a long time to get something done
- Systems are ill-defined or reinforce wrong behaviors
- Mistrust between workers and management

Design is inherently creative and unpredictable. Interactive system designers must blend a thorough knowledge of technical feasibility with a mystical esthetic sense of what attracts users.

- Design is a *process;* it is not a state and it cannot be adequately representedstatically.
- The design process is *nonhierarchical;* it is neither strictly bottom-up norstrictly top-down.
- The process is *radically transformational;* it involves the development of partial and interim solutions that may ultimately play no role in the final design.
- Design intrinsically involves the *discovery of new goals.*

THE FOUR PILLARS OF DESIGN

The four pillars described in this section can help user-interface architects to turn good ideas into successful systems



User Interface Requirements

- Soliciting and clearly specifying user requirements is a major key to success in any development activity
- Laying out the user-interface requirements is part of the overall requirements development and management process
- The success or failure of software projects often depends on the precision and completeness of the understanding among all the users and implementers.
- User interface requirements describe system behaviour

- Performance requirements:

  Examples:

  1)The website shall give users the ability to update their user profiles examples name, address, phone number etc

  2)The mobile device shall be able to save draft text messages when out the service area

- Functional Requirements:

  Examples:

  1)The system shall ensure that the pin entered matches the one on file.

  2) The credit card transaction must be approved prior to displaying confirmation number.

- Interface Requirements:

  Examples:

  1) The web site shall permit ordering stamps online
  2) The mobile device shall permit downloading of ring tones.

The successful method for determining user-interface requirements is to use ethnographic observation, monitoring the context and environment of real users in action.

2. Guidelines documents and processes

Early in the design process, the user-interface architect should generate, or require other people to generate, a set of working guidelines.

Each project has different needs, but guidelines should be considered for:

- Words, icons, and graphics
  - Terminology (objects and actions), abbreviations, and capitalization
  - Character set, fonts, font sizes, and styles (bold, italic, underline)
  - Icons, buttons, graphics, and line thickness
  - Use of color, backgrounds, highlighting, and blinking
- Screen-layout issues
  - Menu selection, form filling, and dialog-box formats
  - Wording of prompts, feedback, and error messages
  - Justification, whitespace, and margins
  - Data entry and display formats for items and lists
  - Use and contents of headers and footers
  - Strategies for adapting to small and large displays
- Input and output devices
  - Keyboard, display, cursor control, and pointing devices
  - Audible sounds, voice feedback, touch input, and other special inputmodes or devices
  - Response times for a variety of tasks
  - Alternatives for users with disabilities
- Action sequences
  - Direct-manipulation clicking, dragging, dropping, and gestures
  - Command syntax, semantics, and sequences
  - Shortcuts and programmed function keys
  - Error handling and recovery procedures
- Training
  - Online help and tutorials
  - Training and reference rnaterials

- Guidelines creation should be a social process within an organization to gain visibility and build support.
- Guidelines documents must be living texts that are adapted to changing needs and refined through experience.
- The creation of a guidelines document at the beginning of an implementation project focuses attention on the interface design and provides an opportunity for discussion of controversial issues.
- The four Es provide a basis for creating a living document and a lively process:

- *EducatiOiI.* Users need training and a chance to discuss the guidelines.
- *Ellforcemellt.* A timely and clear process is necessary to verify that an interface adheres to the guidelines.
- *Exemption.* When creative ideas or new technologies are used, a rapid processfor gaining exemption is needed.
- *Enhancement.* A predictable process for review, possibly annually, will helpkeep the guidelines up-to-date.

3. User-Interface Software Tools

One difficulty in designing interactive systems is that customers and users may not have a clear idea of what the system will look like when it is done. Since interactive systems are novel in many situation, user may not realize the implications of design decisions. unfortunately, it is difficult, costly, and time-consuming to make major changes to systems once those systems have been implemented.

Even though this problem has no complete solution, some of the more serious difficulties can be avoided if, at an early stage, the customers and users can be given a realistic impression of what the final system will look like.

A printed version of the proposed displays is helpful for pilot tests, but an onscreen display with an active keyboard and mouse is more realistic.

For a form-fillin system, the prototype may simply show the fields but not actually process them. Prototypes have been developed with simple drawing or word-processing tools, but graphical design environments such as Macromedia's Director and Flash are widely used.

4. Expert Reviews and Usability Testing

In addition to a variety of expert review methods, test with the intended users, surveys, and automated analysis tools are proving to be valuable.

DEVELOPMENT METHODOLOGIES

- Many software development projects fail to achieve their goals. Some estimates of the failure rate put it as high as 60%.
- Successful developers work carefully to understand the business' needs and refine their skills in eliciting accurate requirements from nontechnical business managers
- Successful developers also know that careful attention to user-centered design issues at the early stages of software development dramatically reduces both development time and cost.
- User-centered design leads to systems that generate fewer problems during development and have lower maintenance costs over their lifetime.
- Software developers have learned that consistently following established development methodologies can help them meet budgets and schedules.
- These business-oriented approaches specify detailed deliverables for the various stages of design and incorporate cost/benefit and return-on-investment (ROl) analyses to facilitate decision making.

1. *Contextual inquiry*. Plan for, prepare, and then conduct field interviews to observe and understand the work tasks being performed. Review business practices.

2. *Interpretation sessions and work modeling*. Hold team discussions to draw conclusions based on the contextual inquiry, including gaining an understanding of the workflow processes in the organization as well as cultural and policy impacts on work performed. Capture key points (affinity notes).

3. *Model consolidation and affinity diagram building*. Present the data gathered to date from users and the interpretation and work modeling to a larger, targeted population to gain insight and concurrence. Consolidate the work models to illustrate common work patterns and processes and create *affinity diagrams* (hierarchical representations of the issues to address user needs).

4. *Persona development*. Develop personas (fictitious characters) to represent the different user types within a targeted demographic that might use a site or product (Cooper, 2004). This aids the team in communicating the needs of the users and bringing those user needs to fruition. Examples of personas, at a high level, are: 1) 22-year-old male with 5+ years of video game playing experience, or 2) 70-year-old female using computer only for e-mail and digital photo sharing.

5. *Visioning*. Review and "walk" the consolidated data, sharing the personas created. The visioning session helps define how the system will streamline and transform the work of the users. Capture key issues and ideas using flipcharts or any media that will facilitate expressing the vision of the revised business processes.

6. *Storyboarding*. The vision guides the detailed redesign of user tasks using pictures and graphs to describe the initial user-interface concepts, business rules, and automation assumptions. Storyboarding defines and illustrates the "to be built" assumptions.

7. *User environment design*. The single, coherent representation of the users and the work to be performed is expressed in the user environment design (UED). The UED is built from the storyboards.

8. *Interviews and evaluations with paper prototypes and mock-ups*. Conduct interviews and tests with actual users, beginning with paper prototypes and then moving on to higher-fidelity prototypes. Capturing the results of the interviews aids in ensuring that the systems will meet end-user requirements.

ETHNOGRAPHIC OSERVATION

- The early stages of most methodologies include observation of users. Since interface users form a unique culture, ethnographic methods for observing them in the workplace are becoming increasingly important.
- As ethnographers, user-interface designers gain insight into individual behavior and the organizational context. User-interface designers differ from traditional ethnographers in that in addition to understanding their subjects, user-interface designers focus on interfaces for the purpose of changing and improving those interfaces.
- The goal of an observation is to obtain the necessary data to influence interface redesign.
- Guidelines for preparing for the evaluation, performing the field study, analyzing the data, and reporting the findings might include the following
    - Preparation
        - Understand organization policies and work culture.
        - Familiarize yourself with the system and its history.
        - Set initial goals and prepare questions.
        - Gain access and permission to observe or interview.
    - Field Study
        - Establish rapport with managers and users.
        - Observe or interview users in their workplace, and collect subjective and objective quantitative and qualitative data.
        - Follow any leads that emerge from the visits.
        - Record your visits.
    - Analysis
        - Compile the collected data in numerical, textual, and multimedia databases.
        - Quantify data and compile statistics.
        - Reduce and interpret the data.
        - Refine the goals and the process used.
    - Reporting
        - Consider multiple audiences and goals.
        - Prepare a report and present the findings.

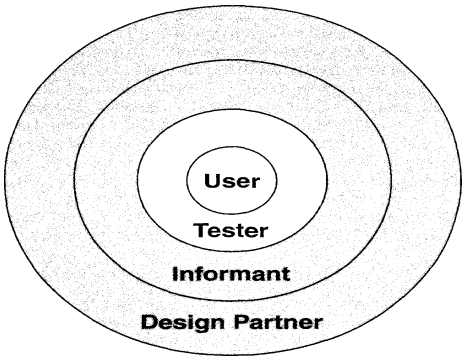
Data collection can include a wide range of subjective impressions that are qualitative or of subjective reactions that are quantitative, such as rating scales or rankings.

Participatory Design

- Participatory design is the direct involvement of people in the collaborative design of the things and technologies they use.
- The arguments in favor suggest that more user involvement brings more accurate information about tasks and an opportunity for users to influence design decisions.
- However, the sense of participation that builds users' ego investment in successful implementation may be the biggest influence on increased user acceptance of the

final system

- On the other hand, extensive user involvement may be costly and may lengthen the implementation period. It may also generate antagonism from people who are not involved or whose suggestions are rejected and even force designers to compromise their designs to satisfy incompetent participants .
- Participatory design experiences are usually positive, and advocates canpoint to many important contributions that would have been missed without user participation.
- Careful selection of users helps to build a successful participatory design experience. A competitive selection increases participants' sense of importance and emphasizes the seriousness of the project.
- The social and political environment surrounding the implementation of complex interfaces is not amenable to study by rigidly defined methods or controlled experimentation.
- The sensitive project leader must judge each case on its merits and must decide what is the right level of user involvement. The experienced user-interface architect knows that organizational politics
- and the preferences of individuals may be more important than technical issues in governing the success of an interactive system.
- Ideas about participatory design are being refined with diverse users, ranging from children to older adults. Arranging for participation is difficult for some users, such as those with cognitive disabilities or those whose time is precious.



Scenario Development

- When a current interface is being redesigned or a well-polished manual system is being automated, reliable data about the distribution of task frequencies and sequences is an enormous asset. If current data do not exist, then usage logs can quickly provide insight.

- A table with user communities listed across the top and tasks listed down the side is helpful. Each box can then be filled in with the relative frequency with which each user performs each task.

- Another representation tool is a table of task sequences, indicating which tasks follow other tasks.
- Often, a flowchart or transition diagram helps designers to record and convey the

sequences of possible actions; the thickness of the connecting lines indicates the frequency of the transitions.

- During the early design stages, data about current performance should be collected to provide a baseline. Information about similar systems helps, and interviews can be conducted with stakeholders, such as users and managers
- Some scenario writers take a further step and produce a videotape to convey their intentions.

Social Impact Statement For Early Design Review

- Interactive systems often have a dramatic impact on large numbers of users. To minimize risks, a thoughtful statement of anticipated impacts circulated among stakeholders can be a useful process for eliciting productive suggestions early in the development, when changes are easiest.
- Information systems are increasingly required to provide services by govern• ments, utilities, and publicly regulated industries.
- A *social impact statement,* similar to an environmental impact statement, might help to promote high-quality systems in government-related applications (reviews for private-sector corporate projects would be optional and self-administered).
- An outline for a social impact statement might include these sections
- Describe the new system and its benefits
    - Convey the high-level goals of the new system.
    - Identify the stakeholders.
    - Identify specific benefits.
- Address concerns and potential barriers
    - Anticipate changes in job functions and potential layoffs.
    - Address security and privacy issues.
    - Discuss accountability and responsibility for system misuse and failure.
    - Avoid potential biases.
    - Weigh individual rights versus societal benefits.
    - Assess tradeoffs between centralization and decentralization.
    - Preserve democratic principles.
    - Ensure diverse access.
    - Promote simplicity and preserve what works.
- Outline the development process
    - Present an estimated project schedule.
    - Propose a process for making decisions.
    - Discuss expectations of how stakeholders will be involved.
    - Recognize needs for more staff, training, and hardware.
    - Propose a plan for backups of data and equipment.

- Outline a plan for migrating to the new system
- Describe a plan for measuring the success of the new system

- A social impact statement should be produced early enough in the devel• opment process to influence the project schedule, system requirements, and budget. It can be developed by the system design team, which might include end users, managers, internal or external software developers, and possibly clients.
- After the social impact statement is written, it should be evaluated *by* the appropriate review panel as well as *by* managers, other designers, end users, and anyone else who will be affected by the proposed *system.*
- A social impact statement documents the intentions for the new system, and the stakeholders need to see that those intentions are backed up by actions. Typically, the review panel is the proper authority for enforcement.

Legal Issues

As user interfaces have become prominent, serious legal issues have emerged. Every developer of software and information should review legal issues thatmay affect design, implementation, or marketing.

- Privacy is always a concern whenever computers are used to store data or to monitor activity. Medical, legal, financial, and other data often have to be protected to prevent unapproved access, illegal tampering, inadvertent loss, or malicious mischief.
- Effective protection provides a high degree of privacy with a minimum of confusion and intrusion into work. Web-site developers should provide easily accessible and understandable privacy policies.
- A second concern encompasses safety and reliability. User interfaces for aircraft, automobiles, medical equipment, military systems, or utility control rooms can affect life-or-death decisions.
- If air-traffic controllers are confused by the situation display, they can make fatal errors. If the user interface for such a system is demonstrated to be difficult to understand, it could leave the designer, developer, and operator open to a lawsuit alleging improper design.
- Designers should strive to make high-quality and well-tested interfaces that adhere to state-of-the-art design guidelines.
- A third issue is copyright or patent protection for software. Software developers who have spent time and money to develop a package are frustrated in their attempts to recover their costs and to make a profit if potential users make illegal copies of the package, rather than buying it.
- A vocal community of developers, led by the League for Programming Freedom, opposes software copyright and patents, believing that broad dissemination is the best policy.

- The Open Source Initiative describes their movement as follows: "When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs.
- A fourth concern is with copyright protection for online information, images, or music.
- If customers access an online resource, do they have the right to store the information electronically for later use? Can the customer send an electronic copy to a colleague or friend?
- Publishers seek to protect their intellectual assets, while librarians are torn between their desire to serve patrons and their obligations to publishers.
- A fifth issue is freedom of speech in electronic environments. Do users have a right to make controversial or potentially offensive statements through e-mail.

CHAPTER 2- EVAUATING INTERFACE DESIGN

Introduction

Designers can become so entranced with their creations that they may fail to evaluate them adequately. Experienced designers have attained the wisdom and humility to know that extensive testing is a necessity.

There are many factors that influence when and where evaluation is performed within the development cycle includes

- Stage of design (early, middle, late)
- Novelty of project (well defined versus exploratory)
- Number of expected users
- Criticality of the interface (for example, life-critical medical system versusmuseum-exhibit support)
- Costs of product and finances allocated for testing
- Time available
- Experience of the design and evaluation team

One troubling aspect of testing is the uncertainty that remains even after exhaustive testing by multiple methods.

- Perfection is not possible in complex human endeavors, so planning must include continuing methods to assess and repair problems during the lifecycle of an interface.
- Second, even though problems may continue to be found, at some point a decision has to be made about completing prototype testing and delivering the product.
- Third, most testing methods will account appropriately for normal usage, but performance in unpredictable situations with high levels of input, such as in nuclear reactor-control or air-traffic-control emergencies, is extremely difficult to test.
- Development of testing methods to deal with stressful situations and even with partial equipment failures will have to be undertaken as user interfaces are developed for an increasing number of life-critical applications.

Expert Reviews

Expert reviews can occur early or late in the design phase. The outcome can be a formal report with problems identified or recommendations for changes.Expert reviewers should be sensitive to the design team's ego involvement and professional skill, so suggestions should be made cautiously.

There are a variety of expert-review methods from which to choose:

- **Heuristic evaluation.**
    - The expert reviewers critique an interface to determine conformance with a short list of design heuristics(encouraging to learn something through doing and discovering things themselves) , such as the eight golden rules (seen in module 1).

  - It makes an enormous difference if the experts are familiar with the rules and are able to interpret and apply them.
  - This spilts the heuristics into three categories : game usability, mobility heuristics and gameplay heuristics.
- **Guidelines review:** The interface is checked for conformance with the organizational or other guidelines document. Because guidelines documents may contain a thousand items, it may take the expert reviewers some time to master the guidelines and days or weeks to review a large interface.
- **Consistency inspection**. The experts verify consistency across a family of inter• faces, checking for consistency of terminology, fonts, color schemes, layout, input and output formats, and so on within the interface as well as in the training materials and online help. Software tools can help automate the process, as well as produce concordances of words and abbreviations.

- **Cognitive walkthrough**
  - The experts simulate users walking through the interface to carry out typical tasks. High-frequency tasks are a starting point, but rare critical tasks, such as error recovery, also should be walked through.
  - Cognitive walkthroughs were developed for interfaces that can be learned by exploratory browsing, but they are useful even for interfaces that require substantial training.
  - An expert might try the walkthrough privately and explore the system, but there also should be a group meeting with designers, users, or managers to conduct the walkthrough and provoke discussion.
- **Metaphors of human thinking(MOT):** The experts conduct an inspection that focuses on how users think when interacting with an interface.
- **Formal usability inspection:**. The experts hold a courtroom-style meeting, witha moderator or judge, to present the interface and to discuss its merits and weaknesses. Design-team members may rebut the evidence about problemsin an adversarial format. Formal usability inspections can be educational experiences for novice designers and managers, but they may take longer to prepare and more personnel to carry out than do other types of review.

An expert-review report should aspire to comprehensiveness, rather than making opportunistic comments about specific features or presenting a random collection of suggested improvements.

Comparative evaluation of expert-review methods and usability-testing methods is difficult because of the many uncontrollable variables. However, the studies that have been conducted provide evidence for the benefits of expert reviews

The expert reviewers should take training courses, read manuals, take tutorials, and try the interface in as close as possible to a realistic work environment, complete with noise and distractions. However, expert reviewers may also retreat to a quieter environment for detailed reviewof each screen.

Usability Testing and Laboratories

- Traditional managers and developers resisted at first, saying that usability testing seemed like a nice idea, but that time pressures or limited resources prevented them from trying it.
- As experience grew and successful projects gave credit to the testing process, demand swelled and design teams began to compete for the scarce resource of the usability-laboratory staff.
- Managers came to realize that having a usability test on the schedule was a powerful incentive to complete a design phase. The usability-test report provided supportive confirmation of progress and specific recommendations for changes.
- Usability-laboratory advocates split from their academic roots as these practitioners developed innovative approaches that were influenced by advertising and market research.
- The movement towards usability testing stimulated the construction of usability laboratories.
- A typical modest usability laboratory would have two 10- by 10-foot areas, divided by a half-silvered mirror: one for the participants to do their work and another for the testers and observers.
- Usability laboratories are typically staffed by one or more people with expertise in testing and user-interface design, who may serve 10 to 15 projects per year throughout an organization.
- Professional practice is to ask all participants to read and sign a statement like this one:
    - I have freely volunteered to participate in this experiment.
    - I have been informed in advance what my task(s) will be and what procedures will be followed.
    - I have been given the opportunity to ask questions and have had my ques•tions answered to my satisfaction.
    - I am aware that I have the right to withdraw consent and to discontinue participation at any time, without prejudice to my future treatment.
    - My signature below may be taken as affirmation of all the above statements; it was given prior to my participation in this study.
- *Videotaping* participants performing tasks is often valuable for later review and for showing designers or managers the problems that users encounter.
- At each design stage, the interface can be refined iteratively and the improved version can be tested. It is important to fix quickly even small flaws, such as spelling errors or inconsistent layout, since they influence user expectations.
- **The** spectrum of usability testing are: Many variant forms of usability testing have been tried:
    - **Paper mackups and prototyping:.** Early usability studies can be conducted using paper mockups of screen displays to assess user reactions to wording, layout, and sequencing. A test administrator plays the role of the computer by flipping the pages while asking a participant user to carry out typical tasks.

This infor• mal testing is inexpensive, rapid, and usually productive.

- ***Discount usability testing.*** This quick-and-dirty approach to task analysis, prototype development, and testing has been widely influential because it lowered the barriers to newcomers. One resolution to the controversy is that discount usability testing be used as a formative evaluation (while designs are changing substantially) and more extensive usability testing be used as a summative evaluation (near the end of the design process).
- ***Competitive usability testing.*** Competitive testing compares a new interface to previous versions or to similar products from competitors. This approach is close to a controlled experimental study, and staff must be careful to construct parallel sets of tasks and to counterbalance the order of presentation of the interfaces. Within-subjects designs seem the most powerful, because participants can make comparisons between the competing interfaces-fewer participants are needed, although each is needed for a longer time period.
- **Universal *usability testing.*** This approach tests interfaces with highly diverse users, hardware, software platforms, and networks. When a wide range of international users is anticipated, such as for consumer electronics products, web-based information services, or e-government services, ambitious testing is necessary to clean up problems and thereby help ensure success. Trials with small and large displays, slow and fast networks, and a range of operating systems or Internet browsers will do much to raise the rate of customer success.
- **Fieldset and portable labs:** This testing method puts new interfaces to work in realistic environments for a fixed trial period. Field tests can be made more fruitful if logging software is used to capture error, command, and help frequencies, as well as productivity measures. Portable usability laboratories with videotaping and logging facilities have been developed to support more thorough field testing. A different kind of field testing is to supply users with test versions of new software or consumer products; tens or even thousands of users might receive beta versions and be asked to comment.
- ***Remote usability testing.*** Since web-based applications are available internationally, it is tempting to conduct usability tests online, without incurring the complexity and cost of bringing participants to a lab. This makes it possible to have larger numbers of participants with more diverse backgrounds, and may add to the realism since participants do their tests in their own environments, using their own equipment. Participants can be recruited bye-mail from customer lists or through online communities. The downside is that there is less control over user behavior and less chance to observe their reactions, although usage logs and phone interviews are useful supplements.
- ***Can-you-break-this tests.*** Game designers pioneered the *can-you-break-this* approach to usability testing by providing energetic teenagers with the challenge of trying to beat new games. This destructive testing approach, in which the users try to find fatal flaws in the system or otherwise destroy it, has been used in other projects and should be considered seriously. Software

purchasers have little patience with flawed products, and the cost of sending out thousands of replacement disks is high. Furthermore, the loss of goodwill when customers have to download and install revised versions is one that few companies can bear.

Survey Instruments

- Written user surveys are a familiar, inexpensive, and generally acceptable companion for usability tests and expert reviews.
- Managers and users can easily grasp the notion of surveys, and the typically large numbers of respondents (hundreds to thousands of users) offer a sense of authority compared to the potentially biased and highly variable results from small numbers of usability test participants or expert reviewers
- The keys to successful surveys are clear goals in advance and development of focused items that help to attain those goals.
- A survey form should be prepared, reviewed by colleagues, and tested with a small sample of users before a large-scale survey is conducted. Methods of statistical analysis (beyond means and standard deviations) and presentation (histograms, scatterplots, and so on) should also be developed before the final survey is distributed.
- Survey goals can be tied to the components of the OAI model of interface design That is, users can be asked for their subjective impressions about specific aspects of the interface, such as the representation of
  - Task domain objects and actions
  - Interface domain metaphors and action handles
  - Syntax of inputs and design of

  displays Other goals would be to

  ascertain the user's

  - Background (age, gender, origins, education, income)
  - Experience with computers (specific applications or software packages,length of time, depth of knowledge)
  - Job responsibilities (decision-making influence, managerial roles, motivation)
  - Personality style (introvert versus extrovert, risk taking versus risk averse,early versus late adopter, systematic versus opportunistic)
  - Reasons for not using an interface (inadequate services, too complex, tooslow)
  - Familiarity with features (printing, macros, shortcuts, tutorials)
  - Feelings after using an interface (confused versus clear, frustrated versus in control, bored versus excited)

Acceptance Tests

- For large implementation projects, the customer or manager usually sets objective and measurable goals for hardware and software performance.
- These notions can be neatly extended to the human interface. Explicit acceptance criteria should be established when the requirements document is written or when a contract is offered. Rather than using the vague and misleading criterion of user friendly," measurable criteria for the user interface can be established for the following:
    - Time for users to learn specific functions
    - Speed of task performance
    - Rate of errors by users
    - User retention of commands over time
    - Subjective user satisfaction
- In a large interface, there may be 8 or 10 such tests to carry out on different components of the interface and with different user communities. Other criteria, such as subjective satisfaction, output comprehensibility, system response time, installation procedures, printed documentation, or graphics appeal, may also be considered in acceptance tests of complete commercial products.
- After successful acceptance testing, there may be a period of field testing before national or international distribution. In addition to further refining the user interface, field tests can improve training methods, tutorial materials, telephone-help procedures, marketing methods, and publicity strategies.
- The goal of early expert reviews, usability testing, surveys, acceptance testing, and field testing is to force as much as possible of the evolutionary development into the prerelease phase, when change is relatively easy and inexpensive to accomplish.

Evaluation During Active Use

A carefully designed and thoroughly tested interface is a wonderful asset, but successful active use requires constant attention from dedicated managers, user• service personnel, and maintenance staff.

Everyone involved in supporting the user community can contribute to interface refinements that provide ever higher levels of service.

Interviews and focus-group discussions

- Interviews with individual users can be productive because the interviewer can pursue specific issues of concern.
- After a series of individual discussions, focus• group discussions are valuable to ascertain the universality of comments
- Interviewing can be costly and time-consuming, so usually only a small fraction of the user community is involved. On the other hand, direct contact with users often leads to specific, constructive suggestions.
- Professionally led focus groups can elicit surprising patterns of usage or hidden

problems, which can be quickly explored and confirmed by participants.

- As a result of this interview project, a set of 42 enhancements to the interface was proposed and implemented. The designers of the interface had earlier proposed an alternate set of enhancements, but the results of the interviews led to a changed set of priorities that more closely reflected the users' needs.

Continuous user-performance data logging

- The software architecture should make it easy for system managers to collect data about the patterns of interface usage, speed of user performance, rate of errors, or frequency of requests for online assistance.
- Logging data provide guidance in the acquisition of new hardware, changes in operating procedures, improvements to training, plans for system expansion, and so on.
- For example, if the frequency of each error message is recorded, the highest frequency error is a candidate for attention. The message could be rewritten, training materials could be revised,
- If logging data are available for each command, each help screen, and each database record, changes to the human-computer interface can be made to simplify access to frequently used features.
- A major benefit of usage-frequency data is the guidance that they provide to system maintainers in optimizing performance and in reducing costs for all participants.

Online or telephone consultants, Online suggestion box or e-mail trouble reporting

- Online or telephone consultants can provide extremely effective and personal assistance to users who are experiencing difficulties.
- Many organizations offer toll-free numbers via which the users can reach a knowledgeable consultant; others charge for consultation by the minute. On some network systems, the consultants can monitor the user's computer and see the same displays that the user sees while maintaining telephone voice contact.
- Electronic mail can be employed to allow users to send messages to the maintainers or designers.
- Such an *online suggestion box* encourages some users to make pro• ductive comments, since writing a letter may be seen as requiring too much effort.

Discussion groups and newsgroups

- Some users may have questions about the suitability of a software package for their application, or may be seeking someone who has had experience using an interface feature.
- Many interface designers and web-site managers offer users discussion groups or newsgroups to permit posting of open messages and questions.
- Discussion groups usually offer lists of item headlines, allowing users to scan items

for relevant topics.

- Personal relationships established by face-to-face meetings also increase the sense of community among users. Ultimately, it is the people who matter, and human needs for social interaction should be satisfied.
- By soliciting user feedback in any of these ways, managers can gauge user attitudes and elicit useful suggestions.

Tools for automated evaluation

- Software tools can be effective in evaluating user interfaces for desktop applications, web sites, and mobile devices.
- Even straightforward tools to check spelling or concordance of terms benefit interface designer's.
- Simple metrics that report numbers of displays, widgets or links between displays capture the size of user interface project.