

|| 人机交互设计

刘钦

Outline

- What is HCI Design?
 - Objectives of HCI - Usability
- Three factors in HCI
 - Human
 - Computer
 - Interaction
- HCI Design Process
- GUI Design

Outline

- What is HCI Design?
 - Objectives of HCI - Usability
- Three factors in HCI
 - Human
 - Computer
 - Interaction
- HCI Design Process
- GUI Design



好



坏

Bad interfaces cause user frustration

Users & Poor Designed

- System users often judge a system **by its interface** rather than its functionality.
 - User interfaces should be designed to match the skills, experience and expectations of its anticipated users.
- A poorly designed interface can cause a user to make **catastrophic** errors.
 - A poor UI is often a reason why so many software systems are never used.

What is HCI Design

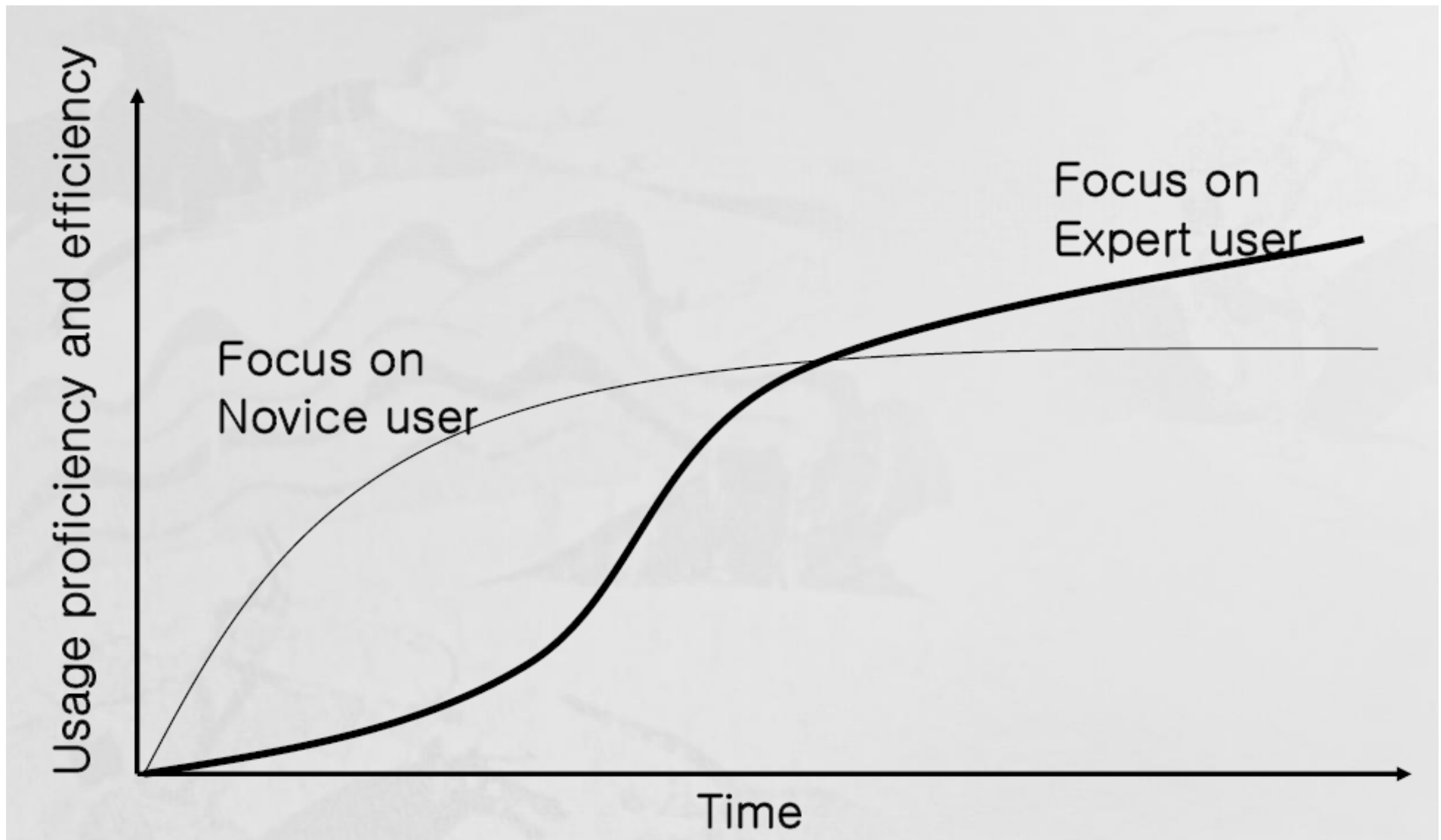
- Human-Computer Interaction Design seeks to discover the most efficient way to design **usable** and **effective** experiences between the human and the computer interface

What is a good HCI design?

- A good HCI interface design encourage an **easy**, **natural**, and **engaging** interaction between a user and a system.
- Joel Spolsky -- “A user interface is well-designed when the program behaves exactly how the user thought it would.”
 - It allows users to carry out their **required** tasks.
 - A good interface should be **transparent** .With a good user interface, users forget that they are using a computer.
- How to judge good or bad? -- Usability

Objectives of HCI --Usability

- Usability is not a single, one-dimensional property of a user interface
 - Learnability: learn easily, user can do things with it in little time
 - Efficiency: skilled users can use it with great efficiency
 - Memorability: interrupted users can continual use it ,do not need to starting from scratch
 - Errors: little errors, resume quickly from errors
 - Satisfaction: make users pleased



易学性和效率是存在冲突的

Learn Curve

Memory

- Involves encoding and recalling knowledge and acting appropriately
- We don't remember everything – involves filtering and processing
- Context is important in affecting our memory
- We recognize things much better than being able to recall things
- The rise of the GUI over command-based interfaces
- Better at remembering images than words
- The use of icons rather than names

Outline

- What is HCI Design?
 - Objectives of HCI - Usability
- Three factors in HCI
 - Human
 - Computer
 - Interaction
- HCI Design Process
- GUI Design

Human

Human

- What is the human in HCI?
 - The user—a person with a laptop, a group working together or remotely, a series of people working in sequence...
 - A person or persons trying to get a job done using computing technology
 - Users see only what they want to see!

人的特性

- Limited short-term memory
 - 7 ± 2 principles
 - Computer has knowledge
- People make mistakes
 - Avoid, Resume, Help.
- People are different
 - Novice users, Knowledgeable intermittent users, Expert frequent users
- People have different interaction preferences
 - Some like pictures, some like text.

考试成绩.xlsx - Microsoft Excel

文件 公式 数据 审阅 视图 加载项

字体 对齐方式 数字 样式 单元格

C	D	E	F	G	H	I	J	K	L	M	N
作业二	作业三	作业四	实验	考试	期末考试	作业成绩	平时成绩	最终成绩			
92	84	89	90	79	79	89	7	98			
83	85	85		55	55	83	0	79		>=90	47
85	83	83		55	55	83	0	79		80~89	131
85	85	85		60	60	84	0	83		70~79	42
85	85	87	90	69	69	86	7	97		60~69	11
85	85	85		54	54	84	0	79		<60	27
88	87	87		72	72	87	0	92		<57	27
94	84	85		75	75	88	0	95			
88	85	85		63	63	86	0	86			
86	85	85		55	55	85	0	80			
85	85	85		60	60	84	0	83			
88	87	87		69	69	86	0	90			
90	82	80		58	58	84	0	81			

What do you see?

User's Answer

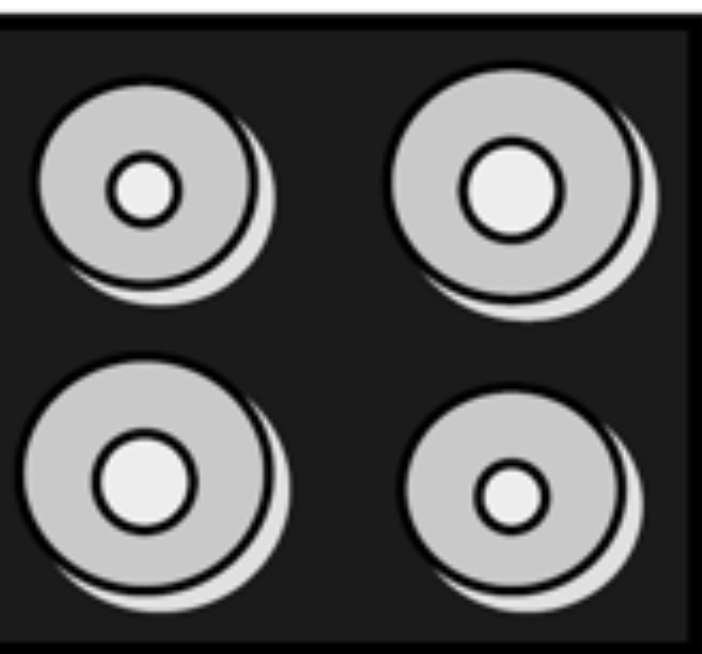
- Score
- Statistics

Technical Answer

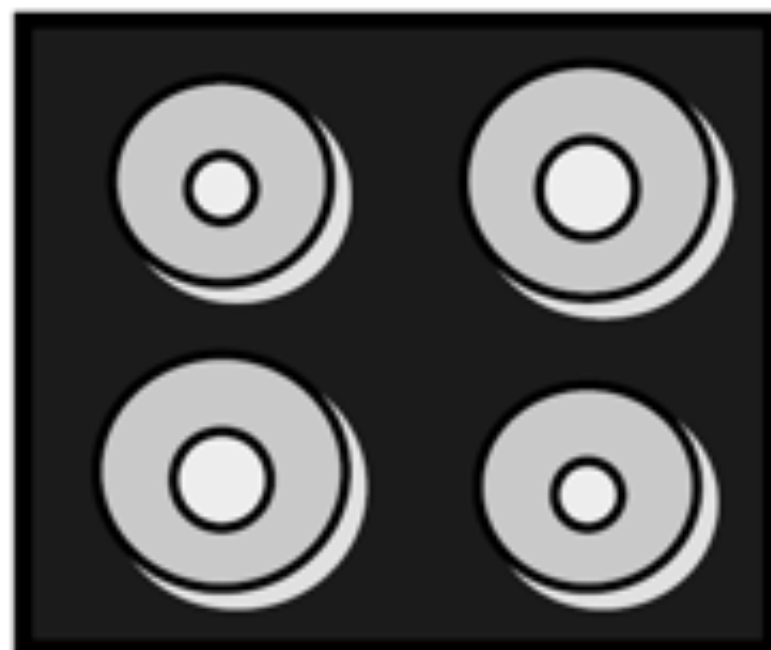
- 2-D surface
- Controllable Components
 - Text and figures
 - Icons and toolbars
 - Scroll bars and menus
 - Task bar and status information
 - Handles and a pointer
 - Colors sets
- Representations of documents
- Representations of “software function”

Mental Model

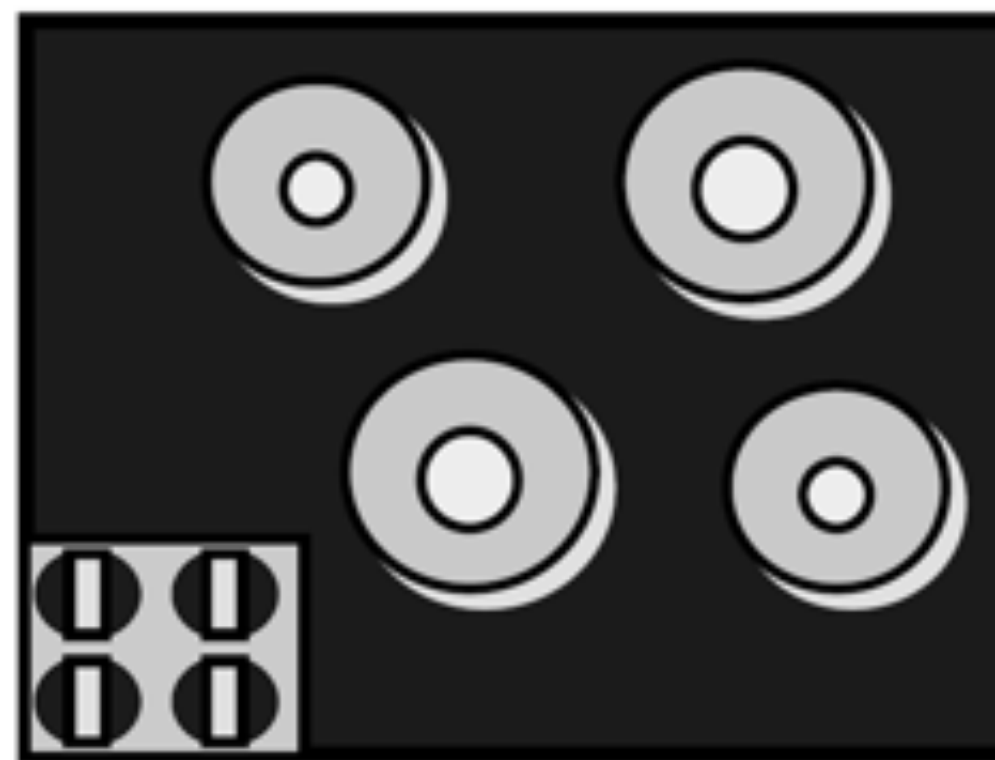
- Try to discover your users' mental model of the task your program helps them perform
- How?
 - Be aware of the model's inherent **metaphors**, which represent conceptual components of the task



Back Right Front Left Front Right Back Left



Back Front Back Front



隱喻

Discover the mental model

- Discover your users' mental model...
 - Find out what the user is trying to do! The Goal!
 - Needs of users, goals and resulting tasks
 - Functionality should only be added if identified to help solving tasks
 - Temptation: If additional functionality is cheap to include it is often done – this can seriously compromise the user interface concept!
 - Frequency of tasks related to user profiles

差异性

- 新手用户
 - 是对业务不熟悉的人，例如新员工或者新接触系统的人。为新手用户设计系统时要关注易学性，进行业务导航，尽量避免出错。如果一个系统的大多数用户都是新手用户，整个系统的人机交互设计都要侧重易学性。
- 专家用户
 - 是能够熟练操作计算机完成业务任务的人，一般都是长时间使用软件系统并且计算机操作技能熟练的人。为专家用户设计系统时，要关注效率。如果一个系统的大多数用户都是专家用户，整个系统的人机交互设计都要侧重效率。
- 熟练用户
 - 是介于新手用户和专家用户之间的人。为熟练用户设计人机交互系统要在易学性和效率之间进行折中。

-
- 好的人机交互应该为不同的用户群体提供差异化的交互机制。
 - 既为新手用户提供易学性高的人机交互机制（图形界面）
 - 又为专家用户提供效率高的人机交互机制（命令行、快捷方式、热键）

Computer

Computer devise

- 人机交互设备
 - 输入设备
 - 键盘
 - 鼠标
 - 手写板
 - 输出设备
 - 显示屏
 - 喇叭
 - 打印机

可视化设计

- 从可视化设计语言Visual Basic开始，对可视化构件的布局就成为可视化设计的主要工作。
- 常见的可视化构件包括：窗口、菜单、标签（Tab）、表单、按钮、列表、树形控件、组合框、输入框等等，[Cooper2007]对此有详细的描述。

可视化设计要点

- 按照任务模型设计界面隐喻，同时不要把软件系统的内部构造机制暴露给用户
- 可视化设计还应该基于界面隐喻，尽可能地把功能和任务细节表现出来

eZip Wizard - Evaluation Copy



What would you like to do?

- ☒ Unzip an existing ZIP file
- ☐ Create a new ZIP file
- ☐ Update an existing ZIP file

About...

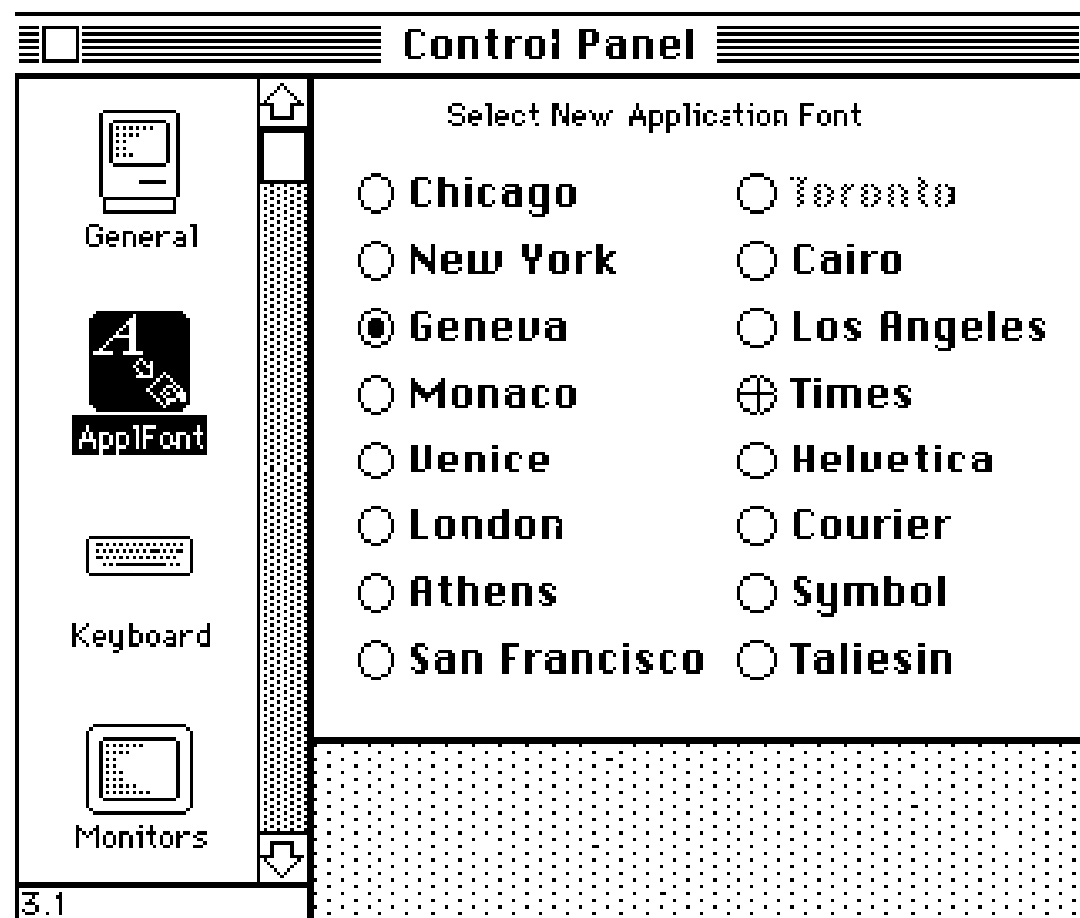
Register...

< Back

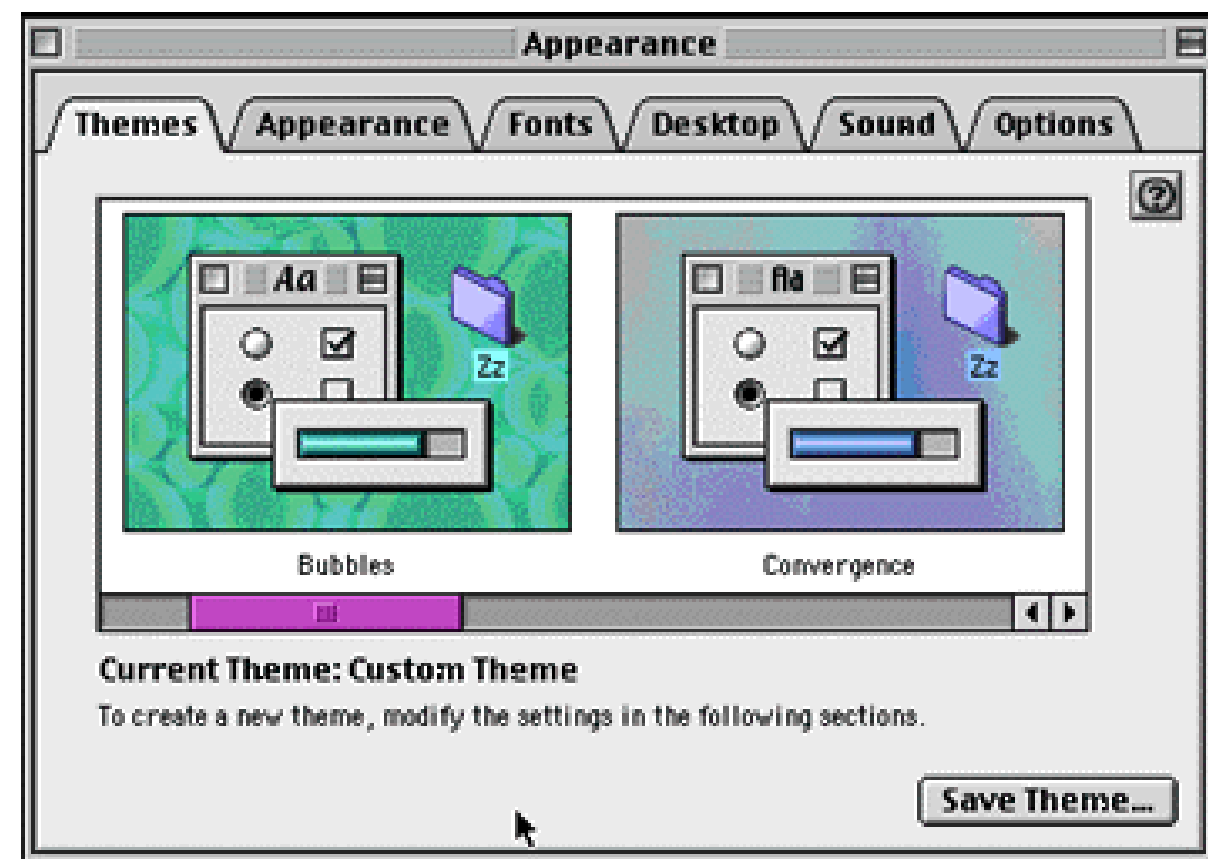
Next >

Cancel

不要暴露内部结构



可视化不佳的设计



可视化更好的设计

展示细节

常见界面类型

名称	特征	人机交互控件	适合场景
批处理	所有的命令预先设置好后，由机器直接一次性执行完毕，中途用户没法进行任何干涉。	不接受输入 无交互 行文本显示（一维输出）	不需要人机交互
命令行	命令模式是一维的交互方式。用户输入一条命令，机器执行一次。	键盘输入（命令） 命令行交互 行文本显示（一维输出）	执行固定任务的熟练用户。
全屏	全屏模式下，通过表单、菜单、导航键和用户进行交互	键盘输入（命令、热键、信息） 菜单、表单交互 可视化输出（菜单、表单，二维全屏输出）	有限任务下熟练用户
图形化	用户利用窗口、菜单、按钮、定位设备（如鼠标等）和机器交互。而且交互操作的方式也是面向对象的，多是可以直观感受到的实物的隐喻。	键盘、鼠标输入 直接操纵（所见即所得）交互 可视化输出（各类可视化控件，多窗口重叠产生二维半输出）	操作技能不熟练的用户
多维交互	正在发展和逐渐出现的交互方式。其中可能会用到声音、视频等机制。更易用和更个性化。	多维输入 多维输出	

Interaction

- “交互”是双向的：
 - 一方面用户主动向软件系统提出请求（输入信息），软件系统给予用户响应（输出信息）；
 - 另一方面，软件系统也应该主动告知用户相应的信息，并等待用户的响应。

Interaction Style

- Direct manipulation (Graphic)
 - Window-Icon
- Menu selection
- Form fill-in
- Command language
- Natural language

Title	JSD. example	<input type="checkbox"/>	Grid	Busy
Method	JSD			
Type	Network	Units	cm ▶	QUIT
Selection	Process	Reduce	Full ▶	PRINT
NODE LINKS FONT LABEL EDIT				

Example of Direct manipulation

Direct manipulation

- Advantages:
 - Users feel in control of the computer and are not intimidated by it.
 - User learning time is relatively short.
 - User get immediate feedback on their actions. Mistakes can often be detected and corrected quickly.
- Problems:
 - How can an appropriate information model and **metaphors** be derived?
 - Given that users have a large information space, how can they **navigate** around that space and always aware of their current position?
 - **Interfaces** usually are complicated.

Menu Systems

- pull-down menus:
 - predictable, but need more screen space
 - display the menu title.
 - User can select commands through this menu.
- pop-up menus:
 - flexible, tailorable, may cause user surprise
 - They are associated with entities (such as a field).

Menu System

- Advantages:
 - Users do not need to know the command names.
 - Typing efforts is minimal.
 - Some user errors can be avoid.
 - Context-dependent help can be provided.
- Problems:
 - Actions which involve logical conjunction (and)
 - or disjunction (or) are awkward to represent
 - Take care the structure of large menus.
 - Solutions: a) scrolling menus, b) hierarchical menus c) walking menus, d) associated control panels

NEW BOOK

Title

ISBN

Author

Price

Publisher

Publication
date

Edition

Number of
copies

Classification

Loan
status

Date of
purchase

Order
status

Form-based interface

Form-based Interface

- Advantage
 - For structured data representation and filling in
- Problems
 - Just for structured data representation and filling in

Command-line interfaces

- Advantages:
 - Implementation is easy and simple due to the language processing.
 - It can support very complex systems with a lot of commands.
 - User interface needs a little effort.
 - Typing efforts is minimal.
 - Some user errors can be avoid.
 - Context-dependent help can be provided.
- Problems:
 - Users must learn and remember all commands.
 - Hard to learn the system and not easy for operations.
 - Users inevitably make errors

Natural language interface

- The user types a command in a natural language. Generally, the vocabulary is limited and these systems are confined to specific application domains (e.g. timetable enquiries)
- NL processing technology is now good enough to make these interfaces effective for casual users but experienced users find that they require too much typing

Summary

Interaction style	Main advantages	Main disadvantages	Application examples
Direct manipulation	Fast and intuitive interaction Easy to learn	May be hard to implement. Only suitable where there is a visual metaphor for tasks and objects.	Video games CAD systems
Menu selection	Avoids user error Little typing required	Slow for experienced users. Can become complex if many menu options.	Most general-purpose systems
Form fill-in	Simple data entry Easy to learn Checkable	Takes up a lot of screen space. Causes problems where user options do not match the form fields.	Stock control, Personal loan processing
Command language	Powerful and flexible	Hard to learn. Poor error management.	Operating systems, Command and control systems
Natural language	Accessible to casual users Easily extended	Requires more typing. Natural language understanding systems are unreliable.	Information retrieval systems

Interaction

Navigation

- 好的人机交互设计就像一个服务周到的推销员，能够主动将自己的产品和服务简明扼要地告诉用户，这个就是导航。
- 好的导航就像一个好的餐厅菜单，餐厅菜单能够帮助顾客快速地找到喜欢的食物，软件系统导航也要能帮助用户找到任务的入口。
- 导航的目的就是为用户提供一个很好的完成任务的入口，好的导航会让这个入口非常符合人的精神模型。

导航

- 全局结构按照任务模型将软件产品的功能组织起来，并区分不同的重要性和主题提供给不同的用户。
 - 全局结构常用的导航控件包括窗口、菜单、列表、快捷方式、热键等等。
 - 全局结构的设计主要以功能分层和任务交互过程为主要依据。
- 局部结构通过安排界面布局细节，制造视觉上的线索来给用户提供导航。
 - 局部结构常用的导航控件包括可视化控件布局与组合、按钮设置、文本颜色或字体大小等等。
 - 局部结构的设计主要以用户关注的任务细节为主要依据。

快捷方式导航

菜单导航



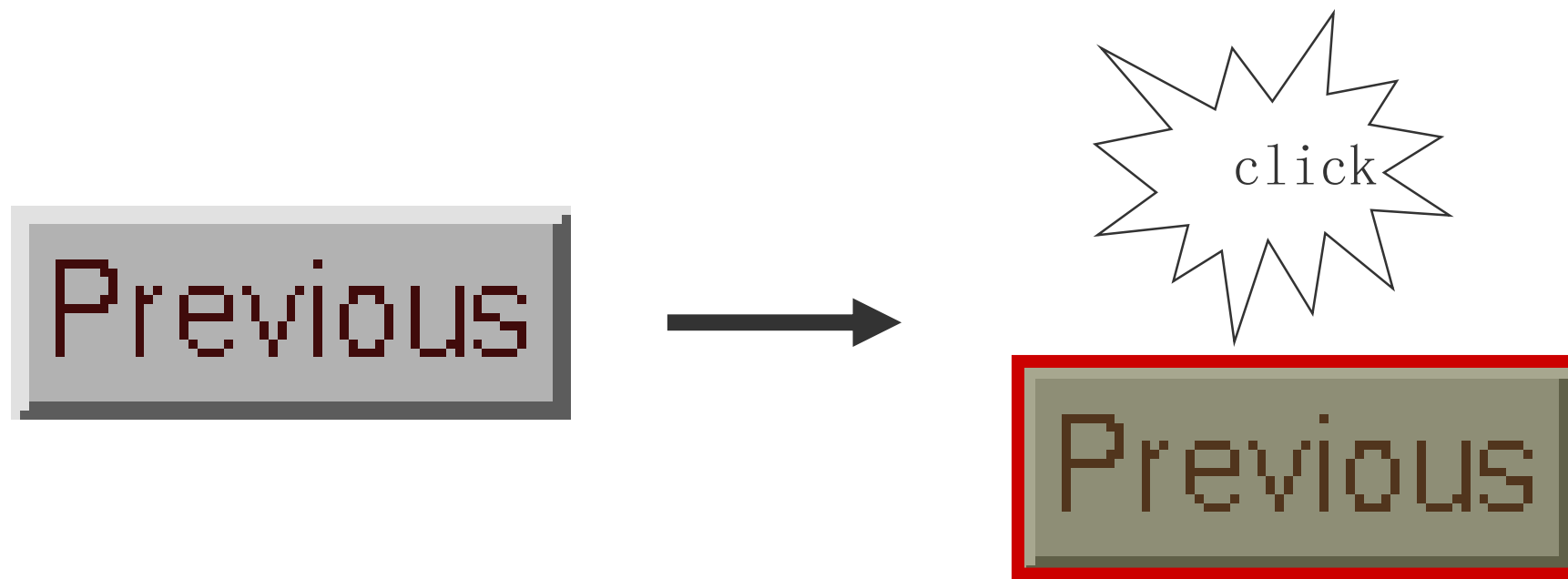
布局与组合导航

列表导航

状态栏导航

按钮导航

导航



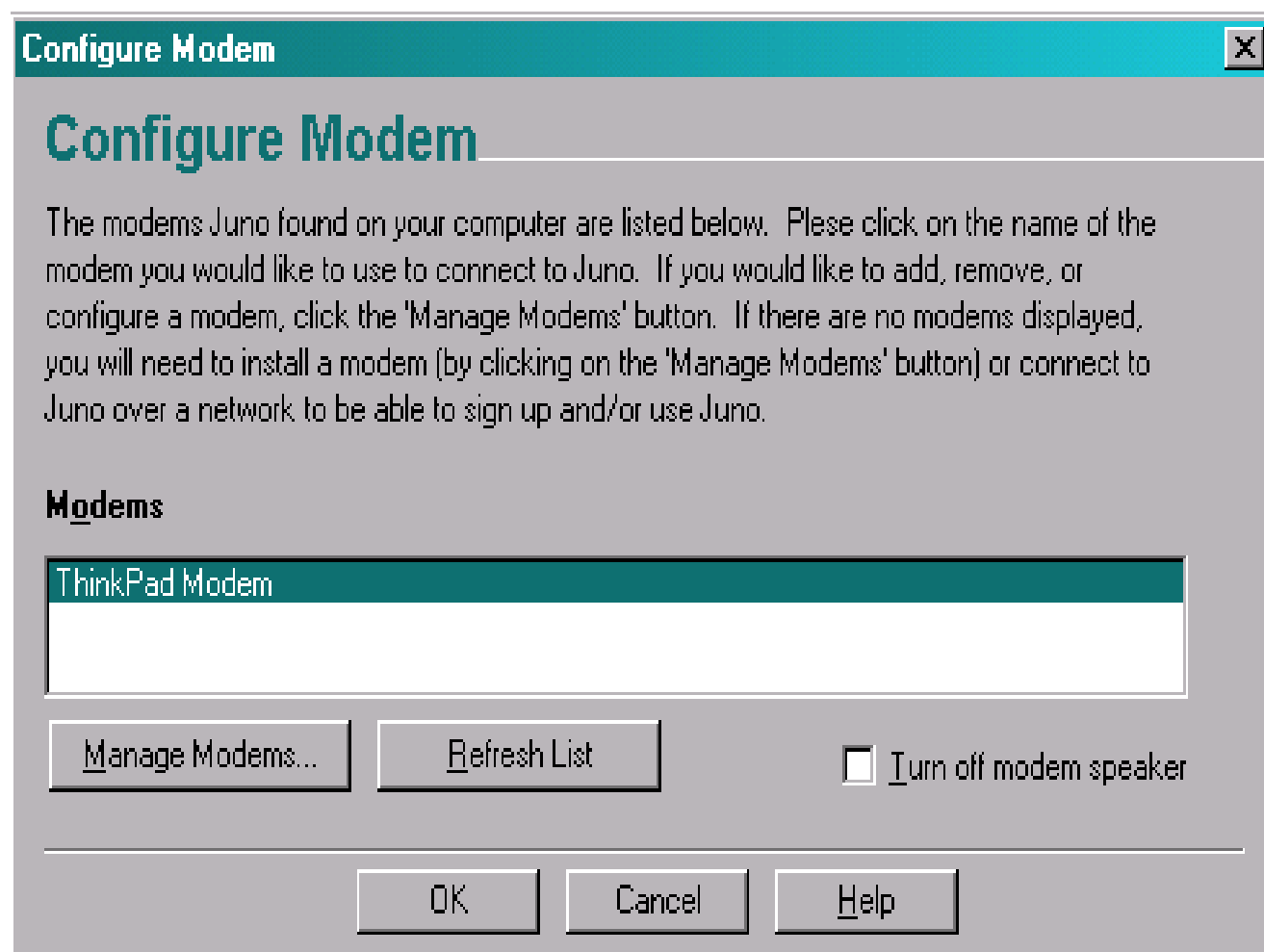
反馈

[Shneiderman2003]总结的部分经验

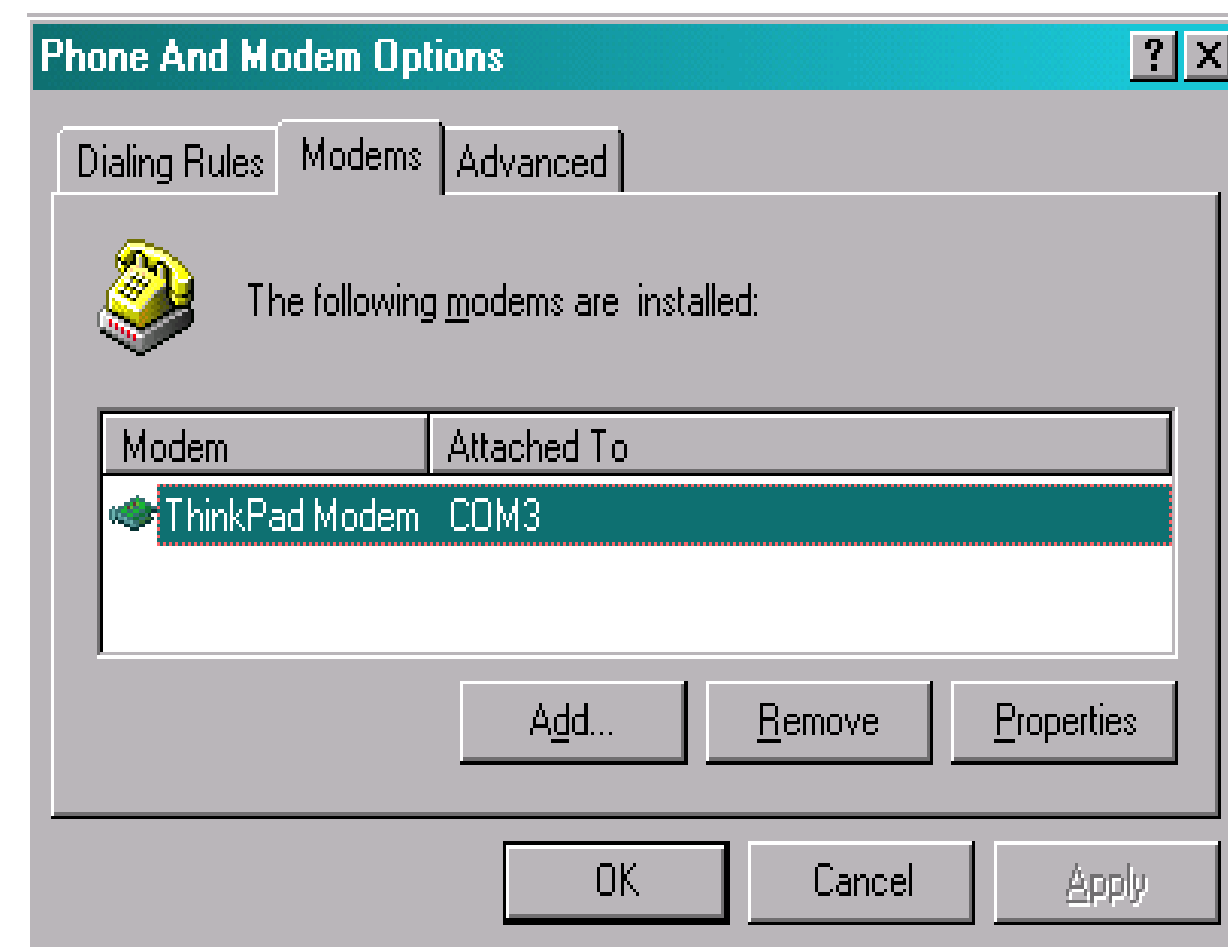
- 用户喜欢较短的响应时间；
- 较长的响应时间（>15秒）具有破坏性；
- 用户会根据响应时间的变化调整自己的工作方式；
- 较短的响应时间导致了较短的用户思考时间；
- 较快的节奏可能会提高效率，但也会增加出错率；
- 根据任务选择适当的响应时间：
 - 打字、光标移动、鼠标定位：50～150毫秒
 - 简单频繁的任务：1秒
 - 普通的任务：2～4秒
 - 复杂的任务：8～12秒
- 响应时间适度的变化是可接受的；
- 意外延迟可能具有破坏性；
- 经验测试有助于设置适当的响应时间。

协作式设计

- 人和计算机是人机交互的两方，其中人的因素是比较固定的，一定时期内不会发生大的变化，所以要让二者交互顺畅，就需要让计算机更多地适应人的因素，这也是人机交互设计以用户为中心的根本原因。
- 这种调整计算机因素以更好地适应并帮助用户的设计方式被称为协作式设计

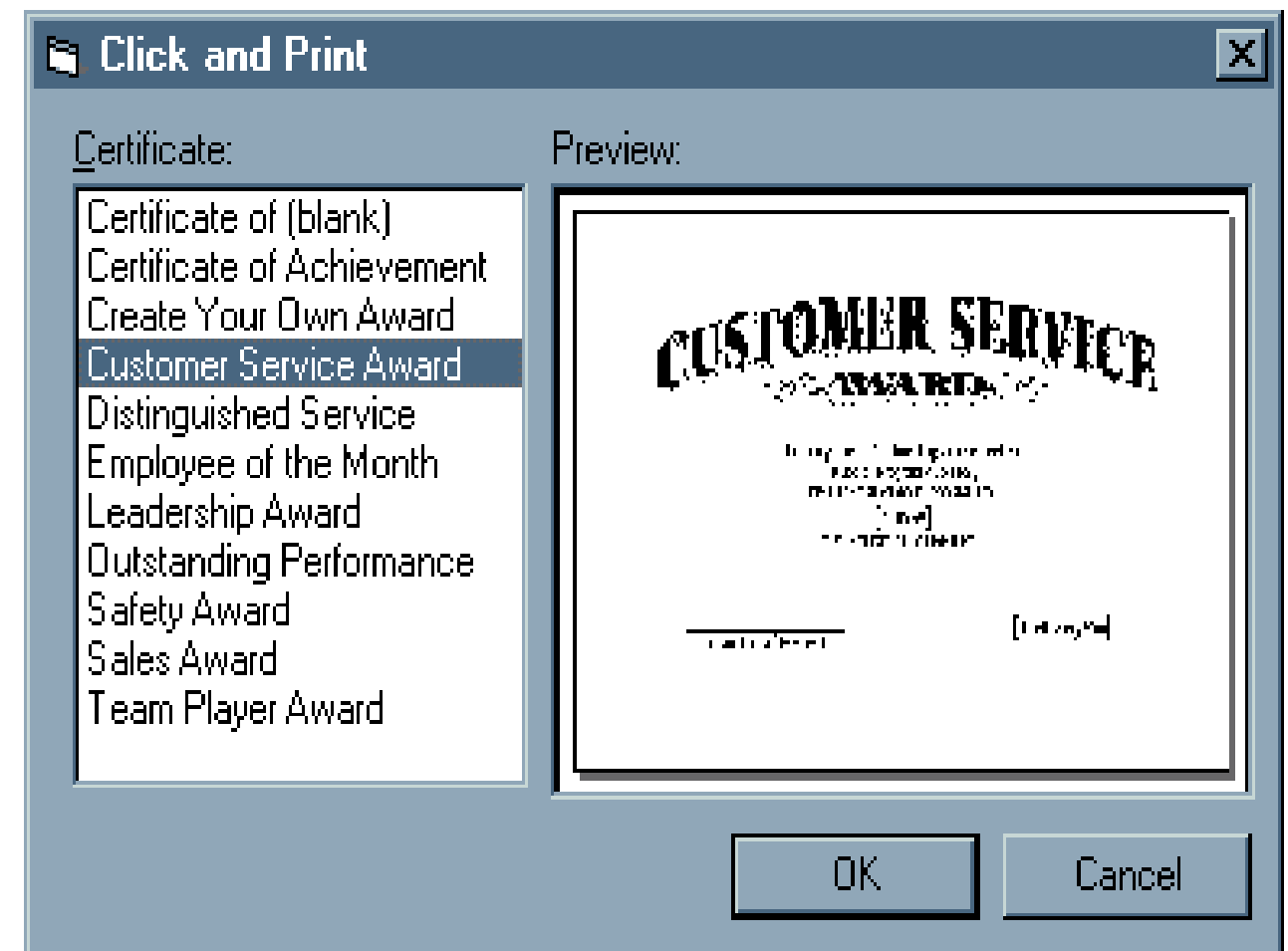


冗余的设计



简洁的设计

简洁的设计

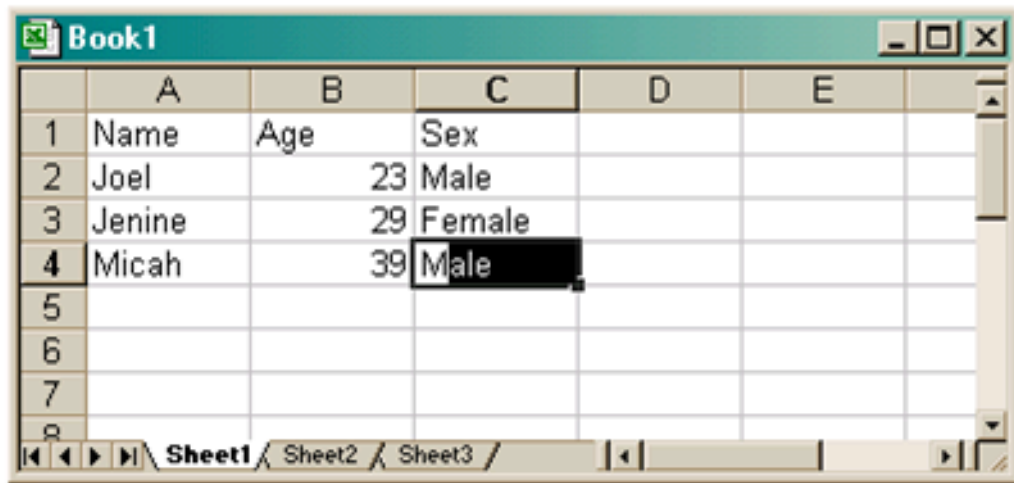


不一致的设计示例

低出错率设计

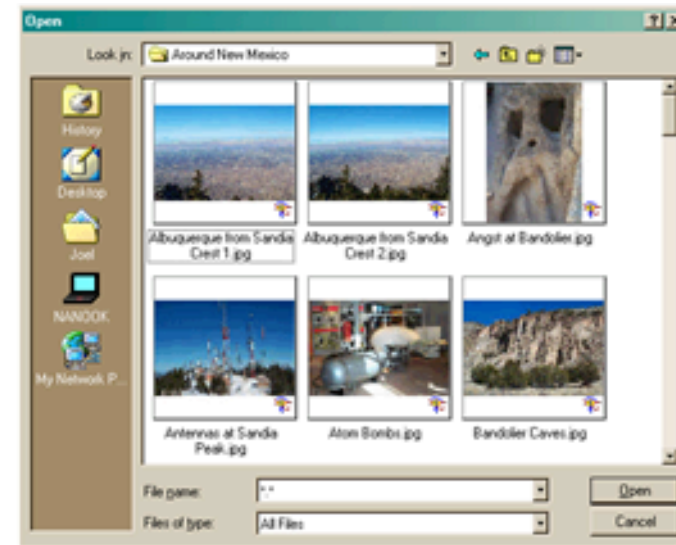
- 人机交互设计首先要帮助人们避免犯错，尽可能设计不让用户犯严重错误的系统
 - 具体措施包括将不适当的菜单选项功能以灰色显示屏蔽
 - 禁止在数值输入域中出现字母字符
- 当错误出现时，系统还要在人机交互中提供简洁、有建设性、具体的指导来帮助用户消除错误。
 - 填写表单时如果用户输入了无效的编码，那么系统应该引导他们对此进行修改，而不是要求用户重新填写整个表单。
- 出错信息应当遵循以下四个简单原则[Shneiderman 1982]:
 - ○1应当使用清晰的语言来表达，而不要使用难懂的代码；
 - ○2使用的语言应当精炼准确，而不是空泛、模糊的；
 - ○3应当对用户解决问题提供建设性的帮助；
 - ○4出错信息应当友好，不要威胁或责备用户。
- 系统还应该提供错误恢复和故障解决帮助手册。

A 减少短期记忆负担。



	A	B	C	D	E
1	Name	Age	Sex		
2	Joel	23	Male		
3	Jenine	29	Female		
4	Micah	39	Male		
5					
6					
7					

A



B

B 使用逐层递进的方式展示信息。在展现复杂信息时，可以将其分为不同层次，递进展现越来越丰富的信息。

C 使用直观的快捷方式。人的记忆具有短时记忆的特点，时间越长越容易忘记。



C

I think I may |

D

D 设置有意义的缺省值。有意义的缺省值（例如大多数用户会选择的输入、特定的场景条件等）可以帮助用户减少输入负担，也可以帮助用户减少记忆负担。

易记性设计

Outline

- What is HCI Design?
 - Objectives of HCI - Usability
- Three factors in HCI
 - Human
 - Computer
 - Interaction
- HCI Design Process
- GUI Design

需求开发

需求收集
场景分析

软件设计

导航设计

界面设计

界面原型化

界面评估与修正

构造、测试与维护

构造、测试与
维护

人机交互设计过程

菜单：销售处理；退货；注销		
会员信息显示区域		商品输入区域
商品列表		
赠品列表		
按钮区域（含热键）		
状态栏		

界面设计

收银员×××

销售处理 退货 注销

会员信息

姓名: ×××

积分: ×××

输入商品

ID: 输入ID

数量: 输入数量

商品列表

ID	名称	价格	特价	数量	总价
----	----	----	----	----	----

赠品列表

ID	名称	数量
----	----	----

标记会员 (L) 删除商品 (D) 结账 (P) 结束销售 (E) 取消销售 (C)

状态栏

原型化界面

Outline

- What is HCI Design?
 - Objectives of HCI - Usability
- Three factors in HCI
 - Human
 - Computer
 - Interaction
- HCI Design Process
- GUI Design

GUI Design

- Key points
 - Common GUI elements
 - Planning your GUI
 - Building your GUI
 - Ensuring usability
 - Ensuring accessibility
-
- <http://www.jiscdigitalmedia.ac.uk/guide/graphical-user-interface-design-developing-usable-and-accessible-collection>

Key points in developing effective GUI

- Involve your users at all stages
- Look at what others are doing
- Know and work within your limitations
- Ensure your GUI is both usable and accessible

Common GUI Elements

- Introductory and supporting information
- Facilities for finding content
- Main collection content
- Extended features

Introductory and supporting information

- Welcome.
 - Often a colourful 'splash' page reflecting something of the collection's content
- Information about the collection.
 - Who produced it, its scope, and, sometimes, how it was made
- Help information.
 - How to use the resource
- Contact information.
 - How to get in touch with those responsible for the resource
- Rights information or Terms of use.
 - Information about ownership, copyright and the user's obligations

Facilities for finding content

- Way(s) of browsing the collection.
 - Where the user can **navigate** through the collection by clicking on hyperlinks
- Way(s) of searching the collection.
 - Where the user can enter a **search** query and view the results

Main collection content

- Summary pages.
 - Provide an overview of browse or search results, typically with thumbnails
- Detailed pages.
 - With screen-sized images, video or audio players, and full metadata

Extended features

- Ability to view digital images at larger sizes.
 - Often through zooming
- Option to watch video streams at different sizes.
 - Depending on network bandwidth
- Option to download files.
 - To local hard drive for later use
- Option to embed a file.
 - User embeds a digital video clip into another online context
- Ability to 'bookmark' the files for later reference.
 - Often called 'albums' or 'favourites'
- Ability to annotate or comment on an image, video or audio file.
 - Tagging for personal use or for public display

Planning your GUI

- Your user GUI is one of the first things to think about - not the last!
- Get your users involved in the design of their GUI
- Find out what others have done or are doing

Building your GUI

- Using an 'off the shelf' solution
- Do-it-yourself GUI
- Combined solutions

Checklist

- **Data entry interface.** Can you customise the data entry interface to improve your cataloguing workflow (e.g. rename or re-order fields, set up templates or drop-down lists, remove from view fields that you don't use)? Are there other features to aid productivity or accuracy (e.g. keyboard short-cuts, tabbing, automatic word completion, spell-checking, data validation)? Can you enlarge the font or change the colour combinations?
- **Customisable visual design.** Can you add your own graphics and text to make it look and feel like the rest of your Web site? Does it work well and look good across different Web browsers and operating systems?
- **Search and retrieval interface.** Can you customise the search or browse interface to better suit your users and your collection? Are there advanced search options? Are search results displayed well?
- **Additional user functionality.** Some systems offer extra features, such as enabling users to collect and annotate their own selections, the ability to group items etc.
- **Control over user access.** Can you limit parts of the collection to specific users or groups of users? Can you deliver different features or levels of functionality or detail depending on user? Are there any digital rights management (DRM) features such as automatic watermarking?
- **Interoperability with other collections.** Is the linking content and metadata supported? Can a user search across collections?

Ensuring usability

- ISO 9241: Ergonomic requirements for office work with visual display terminals (VDTs) , especially Part 11: Guidance on usability (1998)
- ISO 13407: Human-centred design processes for interactive systems (1999)

Evaluating your GUI

- **Usability testing with real users.** Giving typical users some tasks to perform and recording what they do and what they think of the resource. This might involve observing a user in action and noting down how they navigate.
- **Usability evaluations by you or experts.** They might make use of formal guidelines, checklists or questions (e.g. 'usability inspections' or 'heuristic evaluation') or might involve role-playing particular users types ('cognitive walkthroughs').
- **Gathering user feedback.** These approaches involve seeking feedback from users after they've used the resource. A common and easy approach is to survey users, but interviews and focus groups are likely to deliver much richer information.
- **Usage logging.** A lot of useful information is recorded automatically by the server or software used to deliver your resource. By analysing which pages users have visited and how successful their searches have been, it is possible to build up a picture of the usability of a resource.

Designing for usability

- **Looking at competitors' designs.** Getting some typical end users to evaluate resources that are similar to the one you're developing; and tell you what they like or dislike about them.
- **Card sorting and labelling.** Asking your users to organise and label your content and interface in the way that makes most sense to them. Different users will come up with different solutions, but it is likely that this approach will provide some useful and surprising ideas.
- **Prototypes.** Asking users to evaluate models and mock-ups of your resource. These could be created on paper or on screen, and might include real content or dummy content.

Processes involved in developing a high-quality and usable interface(ISO 13407)

- **Specify the context of use.** Understand who will be using the resource and how they will be using it (e.g. which browser will they be using to access the resource?).
- **Specify user and organisational requirements.** Set out the tasks that must be supported (e.g. searching for video) along with some measurable targets for how effectively, efficiently and satisfyingly these tasks should be performed (e.g. "find video in 70% of searches, within 2 clicks, with 90% satisfaction").
- **Produce design solutions.** Design the resource in order to meet the requirements identified in Step 2.
- **Evaluate designs against user requirements.** Check that the development does in fact meet the requirements and targets you have identified in earlier steps.

Ensuring accessibility

- The legal obligation to consider accessibility
- Accessibility is not just an issue for those with disabilities
- Try to be flexible
- Some things to think about when developing online collections

Some things to think about when developing online collections

- **Interface.** Is the interface accessible across a range of devices and browsers? Is it possible for the user to alter the colours and font sizes? Does it use style-sheets to format the resource (rather than frames, tables or graphics)? Are all instructions clearly written? Is textual information presented as text (rather than just pictures of text)?
- **Navigation.** Is it always clear where you are and where you can go to next? Are hyperlinks clearly identified and available as text (rather than just graphics)? Can the user fully control their navigation and use of the resource (e.g. pause any video or sound elements)?
- **Search and retrieval.** Can the user access search forms via the keyboard (or does it depend upon the mouse)? Are the search and results presented as HTML or XML (or embedded in another application, such as Flash or Java)?
- **Dimensions and size of files.** Are there a range of image/video sizes (dimensions and file sizes) to suit different needs? Is the user made aware when they are about to access a very large file? Is it possible to look at an image in detail (either a large image, enlarged sections of the image, or a zooming feature)?
- **Description of files.** Are there textual descriptions for all visual resources? Are there transcriptions for audio and video files? Are there searchable text equivalents where pages of writing have been digitised?

Golden Rules

- Place the user in control
- Reduce the user's memory load
- Be consistent

Ten Principles

- Compact and natural dialogue
- Design graph and color
- Few is much
- Using user language
- Least memory needed for user
- Consistent
- Feedback

-
- Identify exit clear
 - Shortcut
 - Good error message
 - Avoid mistake
 - Help and document