

# Looking Back

- Interaction styles
  - Giving instructions, conversing, manipulating and navigating (direct manipulation), exploring and browsing, proactive computing
- Activity-based vs. object-oriented design
- Interface metaphors
- Prototyping Methods
  - Low-fidelity vs. high-fidelity
  - Horizontal vs. vertical
  - Examples:
    - » paper prototyping
    - » Wizard-of-Oz technique
- Specifying interactive systems
  - Informal, semi-formal, informal methods

# 6 Implementing Interactive Systems

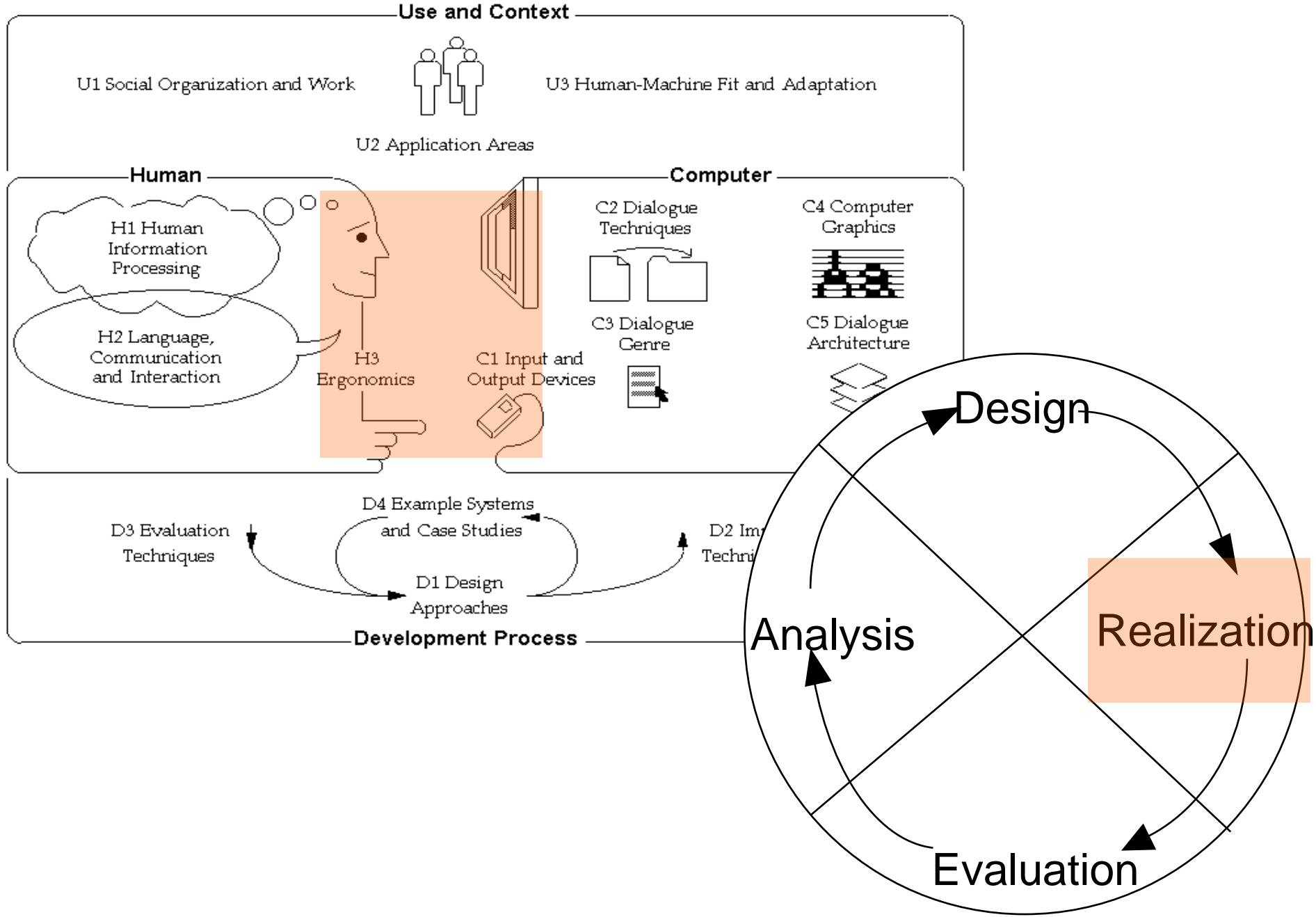
6.1 Designing Look-And-Feel

6.2 Constraints

6.3 Mapping

6.4 Implementation Technologies for Interactive Systems

6.5 Standards and Guidelines

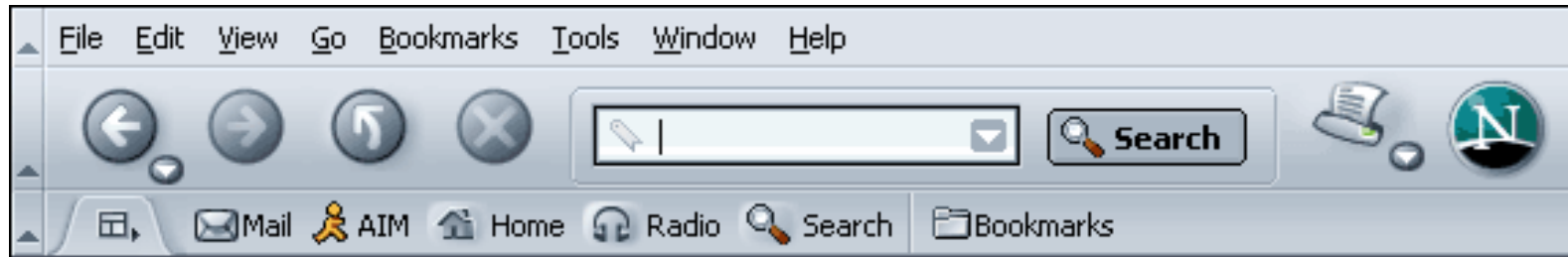


# Visual Design

- Visual Arts versus Visual Design
  - Goal of the artist: to create an observable artefact that provokes an aesthetic response (kind of self-expression)
  - Goal of the designer: to find the representation that is best suited to the communication of some specific information (oriented towards goals of other people)
- Graphic Design and Visual Interface Design
  - Aesthetic concerns placed within the constraints of a functional framework
  - Designers working on interfaces needs to understand
    - » colour, typography, form, composition, ...
    - » **and** interaction, behaviour
- Industrial Design and Interface Design
  - New relationship coming up as more physical artefacts become software-enabled

Source: A. Cooper

# Principles of Visual Design



- Avoid visual noise and clutter
  - No superfluous elements that distract the user
- Use contrast, similarity and layering to distinguish and organize elements (*visual patterns*)
  - Dimensional contrast (depth)
  - Layering
  - Figure and ground
- Provide visual structure and flow at each level of organization
- Use cohesive, consistent and contextually appropriate imagery
- Integrate style and function comprehensively and purposefully
  - Form and function, branding

Based on Mullet/Sano 1995

# Pattern: Deep Background (Tidwell)

## Deep Background

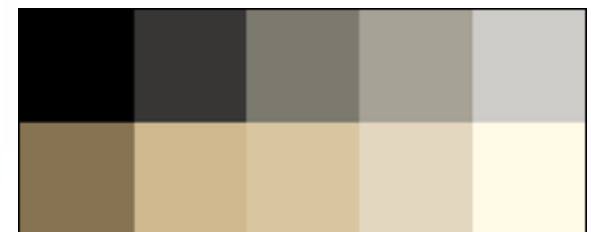


From Mac OS/X

**What:** Place an image or gradient into the page's background that visually recedes behind the foreground elements.

# Pattern: Few Hues, Many Values (Tidwell)

## Few Hues, Many Values



From <http://thebanmappingproject.org>

**What:** Choose one, two, or at most three major color hues to use in the interface. Create a color palette by selecting assorted values (brightnesses) from within those few hues.

# Example: Layering

The screenshot displays the Mercedes-Benz website homepage with a layered design. At the top left is the Mercedes-Benz logo and the text "Mercedes-Benz". A search bar is located at the top right. The main header features the text "4MATIC All-Wheel Drive. How to weather the weather." and a sub-header "► CONTROL. UNLIKE ANY OTHER." Below this is a row of five silver Mercedes-Benz cars. The left sidebar contains a "Models" section with links: "Select a Model", "Certified Pre-Owned", "Build Your Own", "Locate a Dealer", "Financial Solutions", "What's New", "Mercedes In Depth", and "Owners Online". Below the sidebar is a "Portfolio Log-in" section with a form for "Your e-mail" and a "submit" button. At the bottom, there are four promotional tiles: "Special Offer" with a Mercedes-Benz logo, "► Take Advantage of Special Lease Offers.", "► Grand Sports Touret Vision R.\*", "► The all-new 2005 SLK", and "► 4MATIC All-Wheel Drive".

Source: Tidwell



# Example: Visual Flow

Decide what to print (full document or just specific pages).

Select the printer you want to use (if you have a choice)

Decide how many copies you want to print.

The image shows a Windows 'Print' dialog box with several sections and controls. Annotations with lines pointing to specific elements are present:

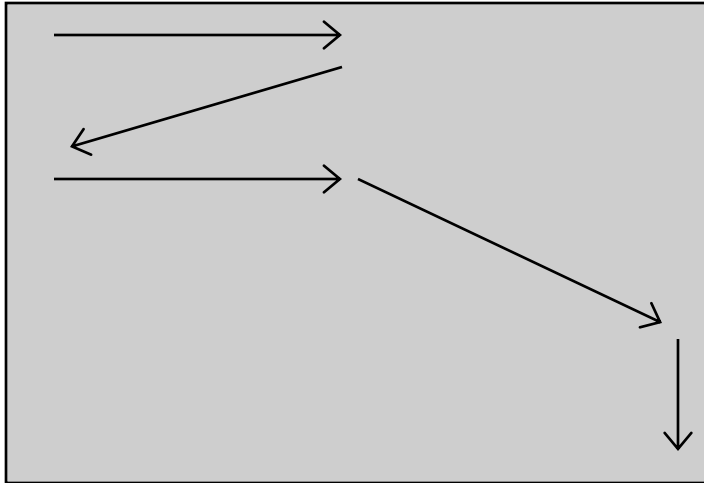
- Printer section:** A dropdown menu shows the selected printer as '\\Eastbanknt\HP\_LaserJet\_1200'. A 'Properties' button is to its right.
- Page range section:** Contains radio buttons for 'All', 'Current page', and 'Pages:'. The 'All' option is selected. Below is a text input field for page numbers and a note: 'Enter page numbers and/or page ranges separated by commas. For example, 1,3,5-12'.
- Copies section:** A 'Number of copies:' spinner control is set to 1. Below it are two icons representing collated pages (1, 2, 3) and a checked 'Collate' checkbox.
- Zoom section:** Contains 'Print what:' (set to 'Document'), 'Print:' (set to 'All pages in range'), 'Pages per sheet:' (set to '1 page'), and 'Scale to paper size:' (set to 'No Scaling').
- Buttons:** 'Options...', 'OK', and 'Cancel' buttons are at the bottom.

Annotations:

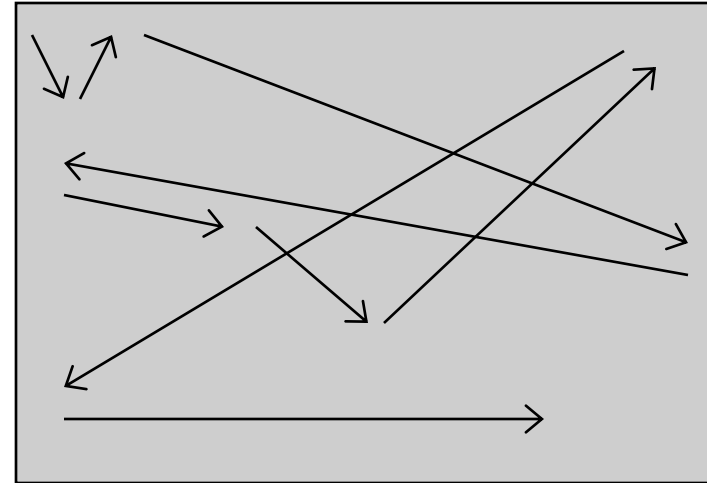
- A line from the text 'Decide what to print...' points to the 'Page range' section.
- A line from the text 'Select the printer you want to use...' points to the printer dropdown menu.
- A line from the text 'Decide how many copies you want to print...' points to the 'Number of copies' spinner.

Grid  
Group boxes

# Good and Bad Logical Flow



Eye movements match  
the logical path through  
The interface

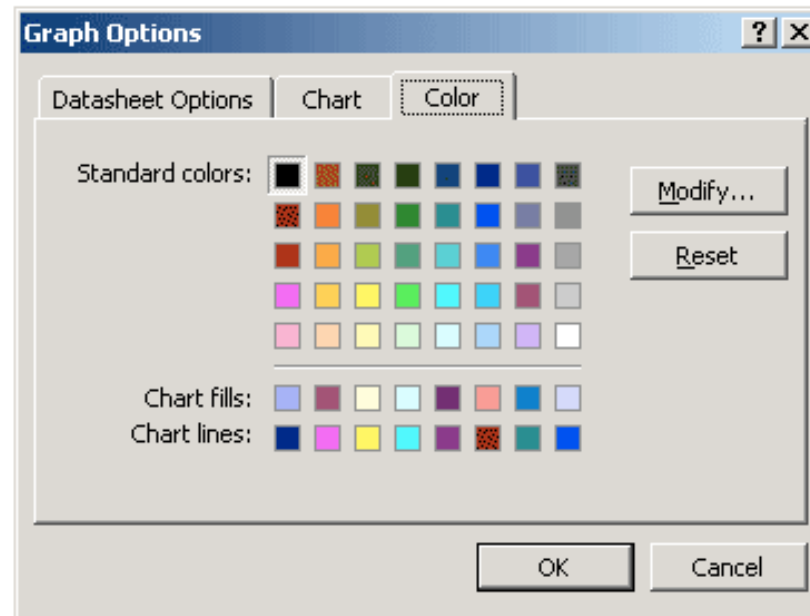


Everything is all over  
the place

# Symmetry and Balance

- Symmetry gives interfaces a solid, stable look
- Balance of visual weights in asymmetric design

## Diagonal Balance



Word's Graph Options dialog box

Tidwell:

**What:** Arrange page elements in an asymmetric fashion, but balance it by putting visual weight into both the upper-left and lower-right corners.

# 6 Implementing Interactive Systems

6.1 Designing Look-And-Feel

6.2 Constraints

6.3 Mapping

6.4 Implementation Technologies for Interactive Systems

6.5 Standards and Guidelines

# Constraints

- Physical constraints
  - Basic physical limitations
- Semantic constraints
  - Assumption to create something meaningful
- Cultural constraints
  - Borders and context provided by cultural conventions
- Logical constraints
  - Restrictions due to reasoning
- Applying constraints is a design decision!
  - Practical way to realise the principle “prevent errors”

## GUI Example

### Date unconstrained

The screenshot shows a flight booking interface with the following fields:

- von:** A dropdown menu with the text "bitte auswählen".
- nach:** An empty text input field.
- Hinflug am:** An empty date input field.
- Rückflug am:** An empty date input field.
- Erw.:** A spinner box containing the number "1".
- Kinder bis 11:** A spinner box containing the number "0".
- unter 2:** A spinner box containing the number "0".

### Date constrained

The screenshot shows a flight booking interface divided into two steps:

- 1. Schritt:**
  - Angebote suchen für:** A dropdown menu with "Alle Linien- & Charterflüge".
  - Hinreise am:** A date input field showing "Mi 12 Nov.2003" with a calendar icon.
  - Rückreise am:** A date input field showing "Mi 19 Nov.2003" with a calendar icon.
- 2. Schritt:**
  - Abflug von:** An empty text input field.
  - Reiseziel:** An empty text input field.
  - Klasse:** A dropdown menu with "Economy".

# Constraints & Redundancy



- Redundancy increases safety
  - E.g. labels and physical constraints
- Constraints can only work at their own level
- But: things can go wrong elsewhere

Defektes Narkosegerät

## Unfallopfer mit Lachgas beatmet - Tödliche Klinik-Panne

Dieser Artikel stellt eine am 25.03.04 um 13:59 veröffentlichte Nachricht dar.

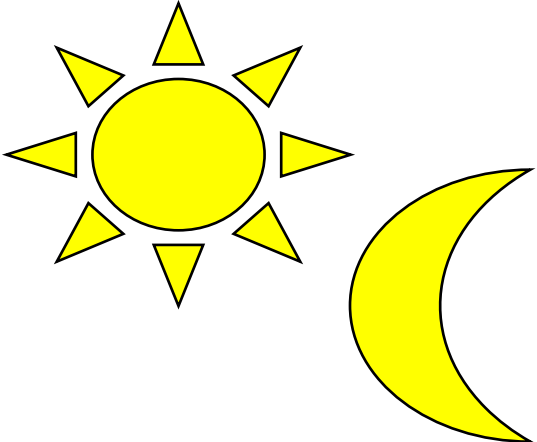
AKTUELLE NACHRICHTEN

**Traunstein (rpo). Lachgas statt Sauerstoff - in einer bayerischen Klinik musste diese Verwechslung ein 19-Jähriger mit dem Leben bezahlen.**

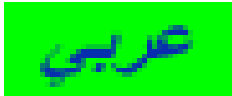
Durch ein falsch zusammengebautes Narkosegerät ist in einem bayerischen Krankenhaus ein Patient ums Leben gekommen. Der 19-Jährige war nach einem Verkehrsunfall in der Notaufnahme der Klinik in Trostbergen statt mit Sauerstoff mit Lachgas beatmet worden, wie die Staatsanwaltschaft Traunstein am Donnerstag sagte. Ermittelt werde gegen einen Mitarbeiter der Herstellerfirma, der das Gerät zuvor repariert hatte. Dabei seien die Anschlüsse für Lachgas und Sauerstoff vertauscht worden.

# Cultural Constraints

- Universal or culturally specific
- Arbitrary conventions that have been learned
- Users' expectations build on cultural constraints



: - )



: - (

; - )

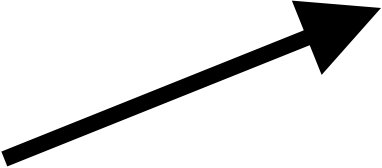
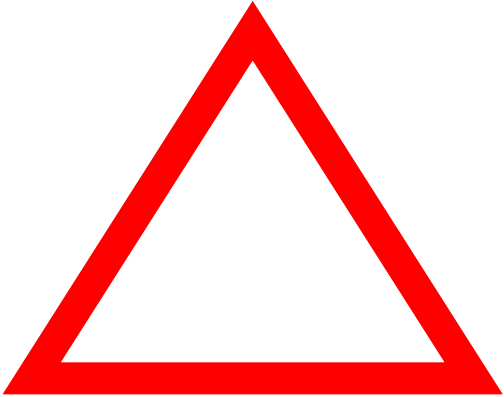
“Hi there!”

: D

: 0

8 - )

: ` - (



# Foreign Cultures: Example





# Physical Constraints & Affordances Examples

- USB Memory Stick vs. DVD vs. money
  - If there is more than one option (physically) cater for these cases
- Dials vs. Buttons vs. Sliders
  - Dials are turned
  - Buttons are pressed
  - Sliders are pushed



# 6 Implementing Interactive Systems

6.1 Designing Look-And-Feel

6.2 Constraints

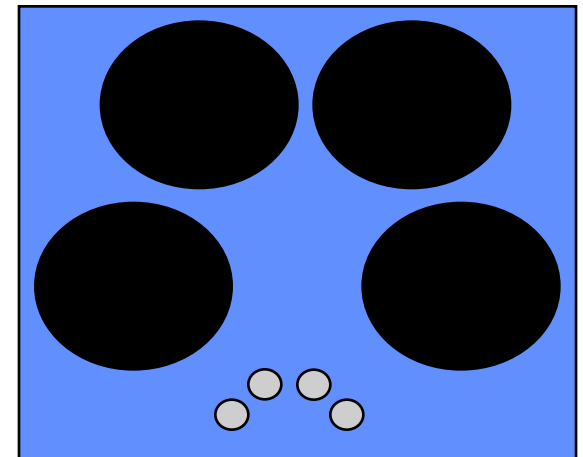
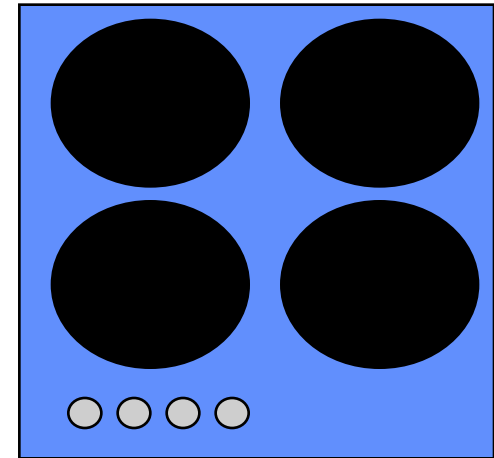
6.3 Mapping

6.4 Implementation Technologies for Interactive Systems

6.5 Standards and Guidelines

# Mapping

- Relationship between controls and action
- Mappings should be
  - Understandable  
(e.g. moving the mouse up move the slider up)
  - Consistent
  - Recognizable or at least quickly learnable and easy to recall
  - Natural, meaning to be consistent with knowledge the user already has
- Example: Cooker
- For these issues see also Gestalt theory!



# Mapping & Gulf of Execution

- Switch row on dashboard of a car:

ISO 2575



# Mapping and Usage Context

- Switch row on (cheap) travel alarm clock



# Mapping – Examples (1)

- Relationship between controls and action

**Please attach a Message to Your Order.**

Message Text:

Position to Print Message:

bottom

bottom-left

bottom-right

centre

left

right

top

top-left

top-right

# Mapping – Examples (2)

- Relationship between controls and action

**Please attach a Message to Your Order.**

Message Text:

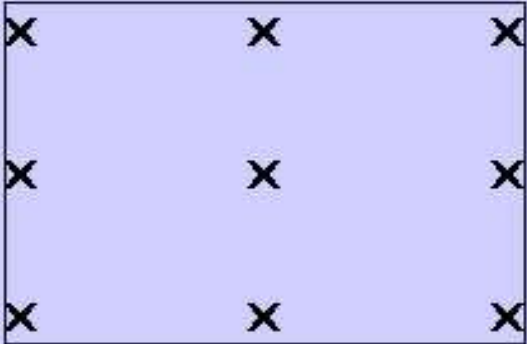
  
  

Position to Print Message:

- bottom
- bottom-left
- bottom-right
- centre
- left
- right
- top
- top-left
- top-right

submit reset

Possible Label Positions



The image shows a screenshot of a web form with a light blue background. At the top, it says 'Please attach a Message to Your Order.' Below this is a text input field labeled 'Message Text:'. Underneath the input field is a section titled 'Position to Print Message:' with a list of radio button options: 'bottom', 'bottom-left', 'bottom-right', 'centre', 'left', 'right' (which is selected), 'top', 'top-left', and 'top-right'. At the bottom of this section are two buttons: 'submit' and 'reset'. To the right of the radio button list is a section titled 'Possible Label Positions' which contains a rectangular frame with a 3x3 grid of 'X' marks, representing the potential locations for a label.

# Mapping – Examples (3)

- Relationship between controls and action

**Please attach a Message to Your Order.**

Message Text:

Position to Print Message

top-left       top       top-right  
 left       centre       right  
 bottom-left       bottom       bottom-right



# Mapping – Examples (4)

- Relationship between controls and action

**Please attach a Message to Your Order.**

Message Text:

Position to Print Message

<input type="radio"/> top-left	<input type="radio"/> top	<input type="radio"/> top-right
<input type="radio"/> left	<input type="radio"/> centre	<input checked="" type="radio"/> right
<input type="radio"/> bottom-left	<input type="radio"/> bottom	<input type="radio"/> bottom-right

# Mapping – Examples (5)

**Show Appointments**

next week   last week   tomorrow   yesterday   today

[Empty list box with scroll arrows]

Show orders received   today

- today
- this week
- this month
- last month

**Show Appointments**

last week   yesterday   today   tomorrow   next week

[Empty list box with scroll arrows]

Sort Data   Ascending

- Ascending
- Random
- Descending

- “Natural” mappings can be found in many areas
- It is not always obvious what the “natural” mapping is
- Correlation with cultural constraints

# 6 Implementing Interactive Systems

6.1 Designing Look-And-Feel

6.2 Constraints

6.3 Mapping

6.4 Implementation Technologies for Interactive Systems

6.5 Standards and Guidelines

# User Interface Toolkits

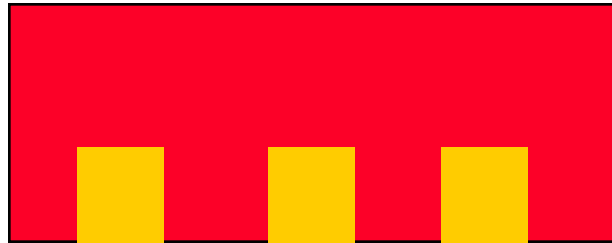
- Various forms:
  - Libraries
  - Frameworks
  - (Visual) components (*widgets*)
- Dependencies on
  - Programming language
  - Development tool (in particular for visual components)
  - Operating system
- Examples:
  - Java AWT & Swing
  - Microsoft MFC (C++, Windows)
  - Windows Forms (C#, Windows)
  - Qt (C++, Unix)
  - Cocoa (MacOS)





Visual C++

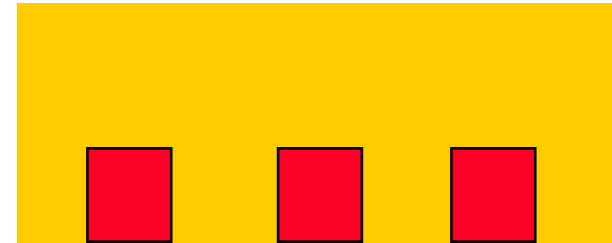
# Class Library vs. Framework

Class library



-  Application-specific parts
-  Pre-fabricated parts

Framework



*"Don't call us,  
we call you"*

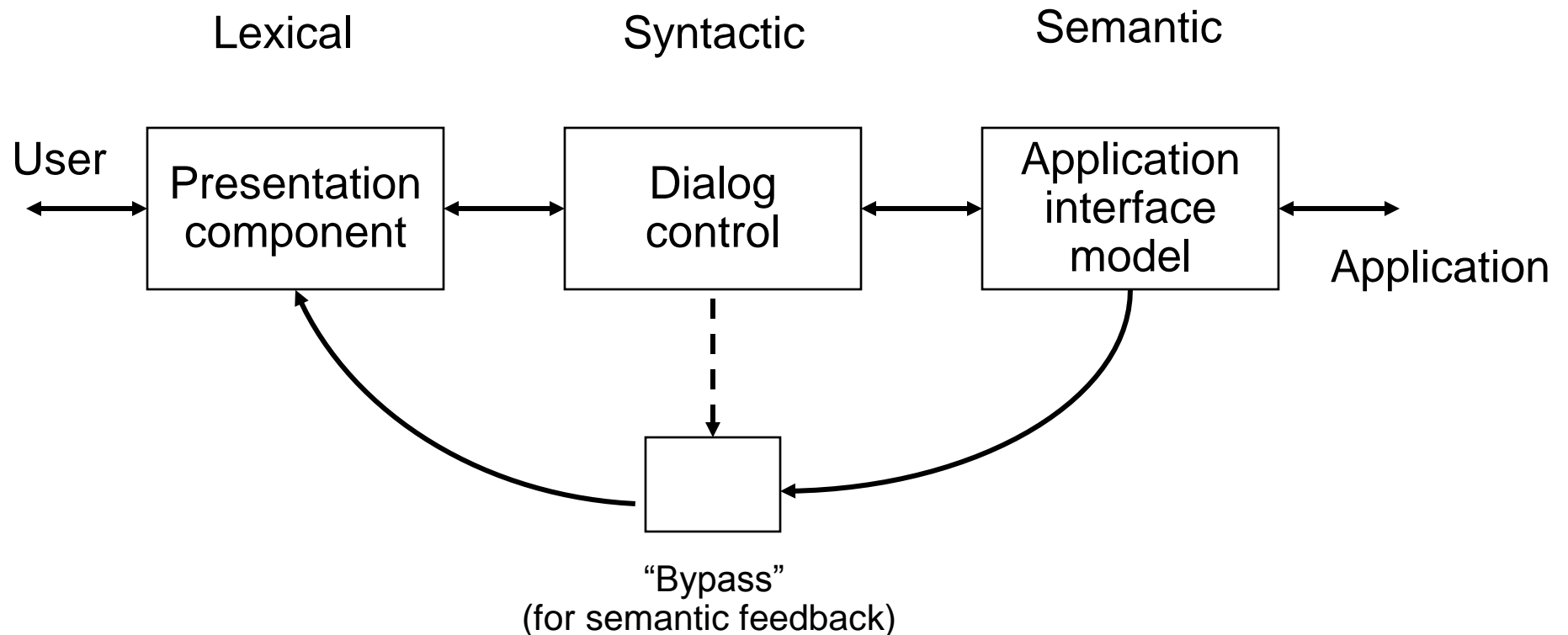
- A framework defines a stand-alone, executable basis for a class of applications.
- Framework:  
Application-specific code *is called from pre-fabricated code.*
- Class library:  
Application-specific code *calls pre-fabricated code.*

# User Interface Management System (UIMS)

- UIMS is a term used with a wide range of meanings:
  - Conceptual architecture for the structure of an interactive system
    - » Separating application logic and interface
  - Techniques for implementing application and presentation
    - » Providing the separation but preserving the intended connection
  - Support techniques for managing a run-time interactive environment
- In the following:
  - Focus on software architecture
- Advantages of presentation/application separation:
  - Portability
  - Reusability
  - Multiple interfaces
  - Customization of interface

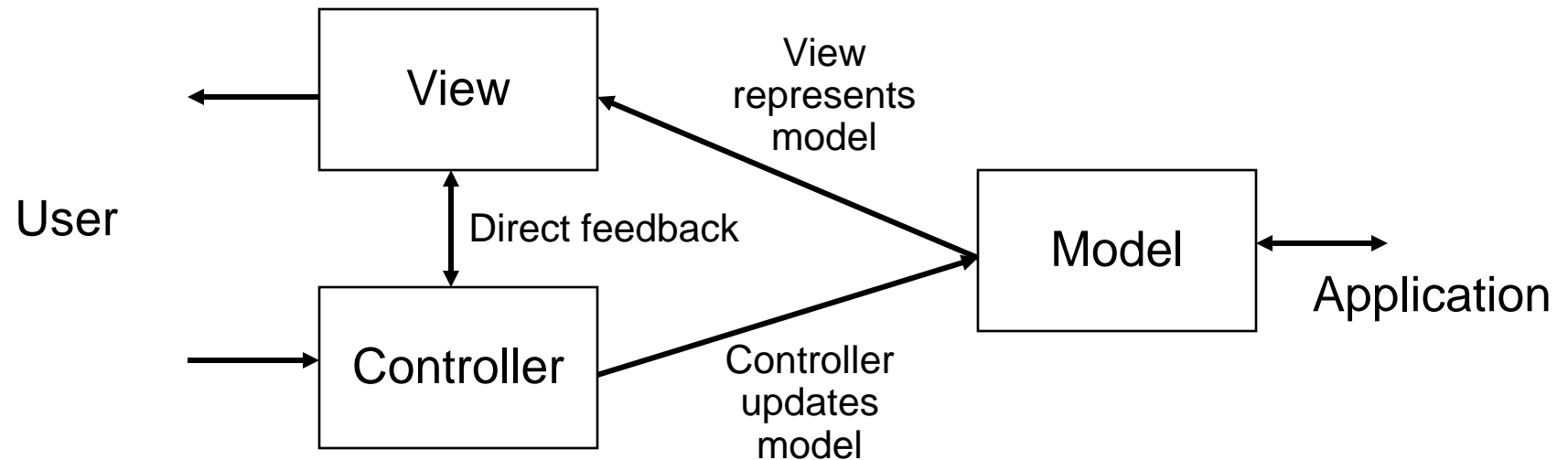
# Seeheim Model

- Developed at a workshop in Seeheim, Germany in 1985



# Model-View-Controller (MVC)

- Paradigm for application structure developed within the Smalltalk programming environment (T.Reenskaug 1979)



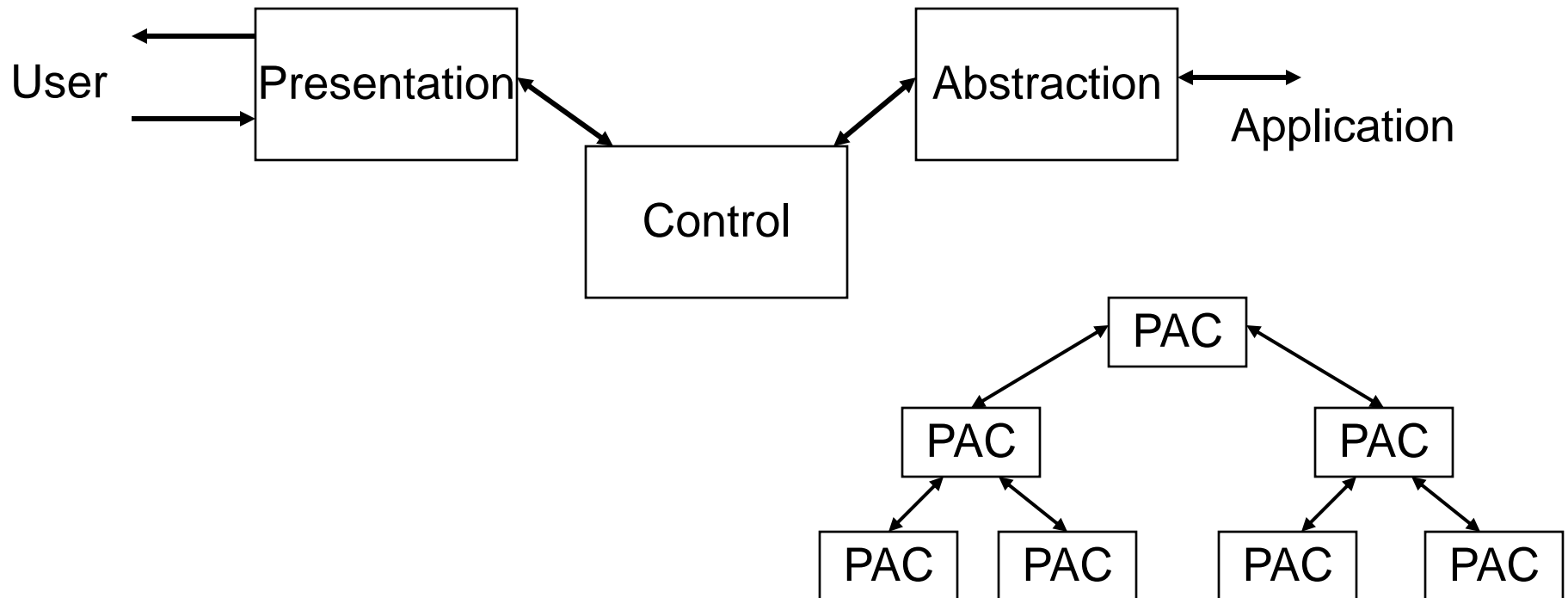


# Scaling MVC to Large Interfaces

- What happens if an interface consists of many parts, organized hierarchically?
- Different levels of updates:
  - Number of windows
  - Number of interface elements
  - Organization of interface elements: order, position
  - Status of interface elements: label, icon, dimming
- Hierarchical MVC-style interface
  - Need for a single point of control used for communication across hierarchy levels
  - “Seeheim-style” central dialog control, but applied to separated aspect of interface: PAC

# Presentation-Abstraction-Control (PAC)

- J. Coutaz 1987
  - Complete separation of presentation and application
  - Input and output handled by single component
  - Hierarchical structure of PAC components, multi-threaded



# 6 Implementing Interactive Systems

6.1 Designing Look-And-Feel

6.2 Constraints

6.3 Mapping

6.4 Implementation Technologies for Interactive Systems

6.5 Standards and Guidelines

# Standards (1)

- ISO 9241
  - Original title: *Ergonomic requirements for office work with visual display terminals (VDTs)*
  - New title: *Ergonomics of Human System Interaction*
  - Example: ISO 9241 Part 110 “Dialogue Principles”
    - » Suitability for the task
    - » Self-descriptiveness
    - » Controllability
    - » Conformity with user expectations
    - » Error tolerance
    - » Suitability for individualisation
    - » Suitability for learning

**Aufgabenangemessenheit**

**Selbstbeschreibungsfähigkeit**

**Steuerbarkeit**

**Erwartungskonformität**

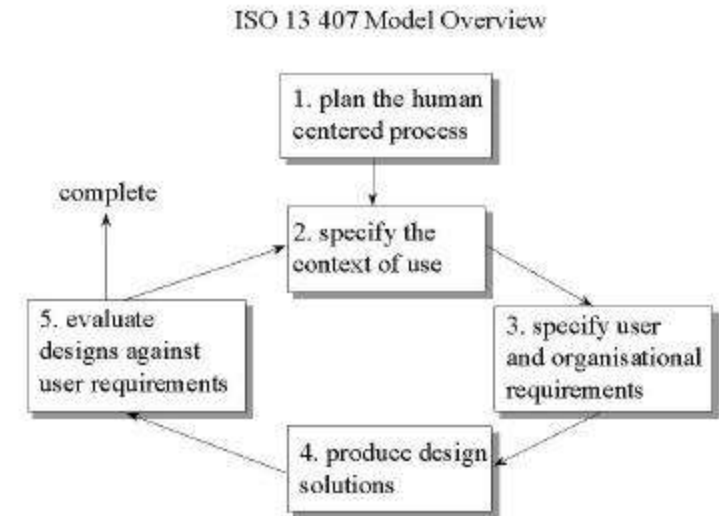
**Fehlertoleranz**

**Individualisierbarkeit**

**Lernförderlichkeit**

# Standards (2)

- ISO 13407
  - Human-centered development process
  - See chapter 4
- ISO 14915
  - Design principles for multimedia user interfaces
- ISO 16071
  - Accessibility of human-computer interfaces
- BITV
  - Barrierefreie Informationstechnik-Verordnung
- BildscharbV
  - Bildschirmarbeitsverordnung



# Hix and Hartson's Guidelines (1)

- User centered design
- Know the user
- Involve the user
- Prevent user errors
- Optimize user operation
- Keep control with the user
- Help the user to get started
- Give a task-based mental model
- Be consistent
- Keep it simple
- Design for memory limitations
- Use recognition rather recall
- Use cognitive directness
- Draw on real world analogies

# Hix and Hartson's Guidelines (2)

- Use informative feedback
- Give status indicators
- Use user-centred wording
- Use non-threatening wording
- Use specific constructive advice
- Make the system take the blame
- Do not anthropomorphise
- Use modes cautiously
- Make user action reversible
- Get attention judiciously
- Maintain display inertia
- Organize screen to manage complexity
- Accommodate individual difference

(Hix and Hartson, Developing User Interfaces, Wiley, 1993)

# GNOME Guideline

- 1. Usability Principles
  - Design for People
  - Don't Limit Your User Base
  - Accessibility
  - Internationalization and Localization
  - Create a Match Between Your Application and the Real World
  - Make Your Application Consistent
  - Keep the User Informed
  - Keep It Simple and Pretty
  - Put the User in Control
  - Forgive the User
  - Provide Direct Manipulation
- 2. Desktop Integration
  - Placing Entries in the Applications Menu
  - Menu Item Names
  - ...
- 3. Windows
  - Titles
  - ...
  - Layout
  - Common Dialogs
- 4. Menus
  - The Menubar
  - Types of Menu
  - Drop-down Menus
  - ...
  - Help
- 5. Toolbars
  - Appearance and Content
  - ...
- 6. Controls
  - ...
  - Sliders
  - Buttons
  - Check Boxes
  - ...



# Drag and Drop Semantics

---

Your application must determine whether to move or copy a dragged item after it is dropped on a destination. The appropriate behavior depends on the context of the drag-and-drop operation, as described in this section.

## Move Versus Copy

---

If the source and destination are in the same container (for example, a window or a volume), a drag-and-drop operation is interpreted as a move (that is, cut and paste). Dragging an item from one container to another initiates a copy (copy and paste). The user can perform a copy operation within the same container by pressing the Option key while dragging. When performing a copy operation, indicate a copy operation to the user by using the copy cursor. (See [“Standard Cursors”](#) (page 67).)

**Table 3-1** Common drag-and-drop operations and results

Dragged item	Destination	Result
Data in a document	The same document	Move
Data in a document	Another document	Copy
Data in a document	The Finder	Copy (creates a clipping)
Finder icon	An open document window	Copy
Finder icon	The same volume	Move
Finder icon	Another volume	Copy

**Example 1:**  
**Apple Human Interface Guidelines**  
**(page 42)**

# Icon Genres and Families

**Icon genres** help communicate what you can do with an application before you open it. Applications are classified by role—user applications, software utilities, and so on—and each category, or genre, has its own icon style. This differentiation is very important for helping users easily distinguish between types of icons in the Dock.

**Figure 5-1** Application icons of different genres—user applications and utilities—shown as they might appear in the Dock



For example, the icons for user applications are colorful and inviting, while utilities have a more serious appearance. Figure 5-2 shows user application icons in the top row and utility icons in the bottom row. These genres are further described in *“User Application Icons”* (page 57) and *“Utility Icons”* (page 58).

**Figure 5-2** Two icon genres: User application icons in top row; utility icons in bottom row



## Example 2: Apple Human Interface Guidelines (page 55)

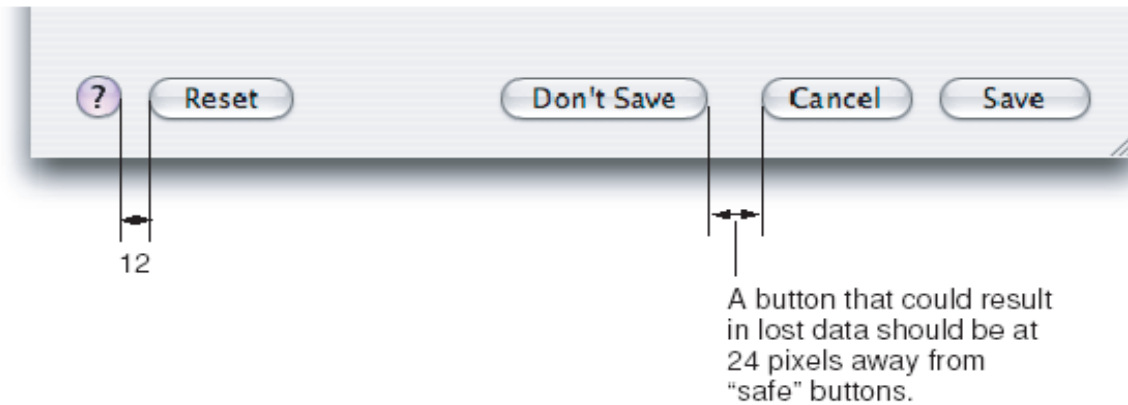


**Figure 9-2** A standard alert



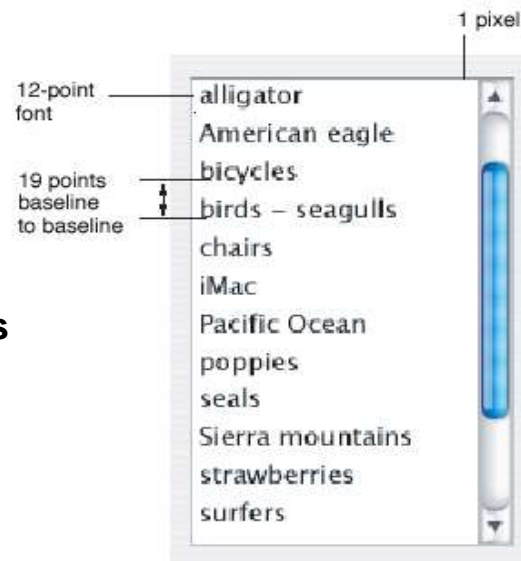
**Example 3:  
Apple Human Interface Guidelines  
(page 126 & 134)**

**Figure 9-7** Position of buttons at the bottom of a dialog



### Scrolling List Specifications

**Figure 10-44** Scrolling list dimensions

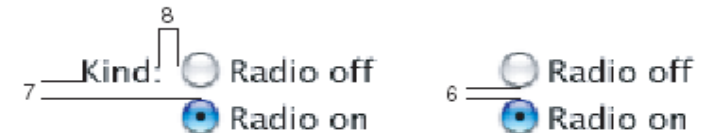


**Example 4:**  
**Apple Human Interface Guidelines**  
 (page 138, 163 & 190)

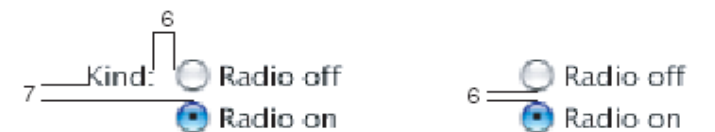
### Radio Button Specifications

**Figure 10-14** Radio button spacing

Full-size radio button



Small radio button



Mini radio button

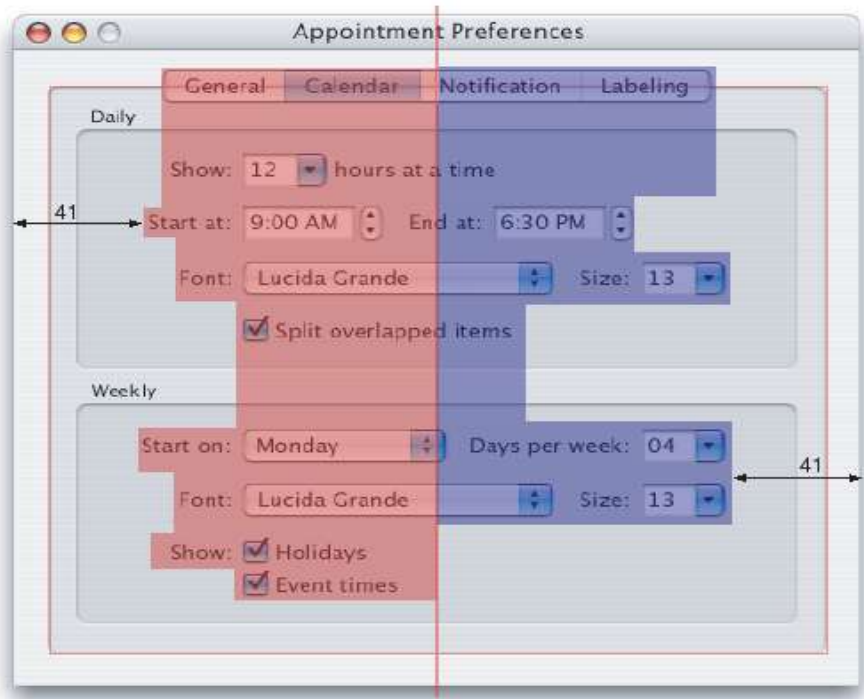


Align the baselines of the label and the first button's text.

Figure 11-10 Layout dimensions for a standard alert

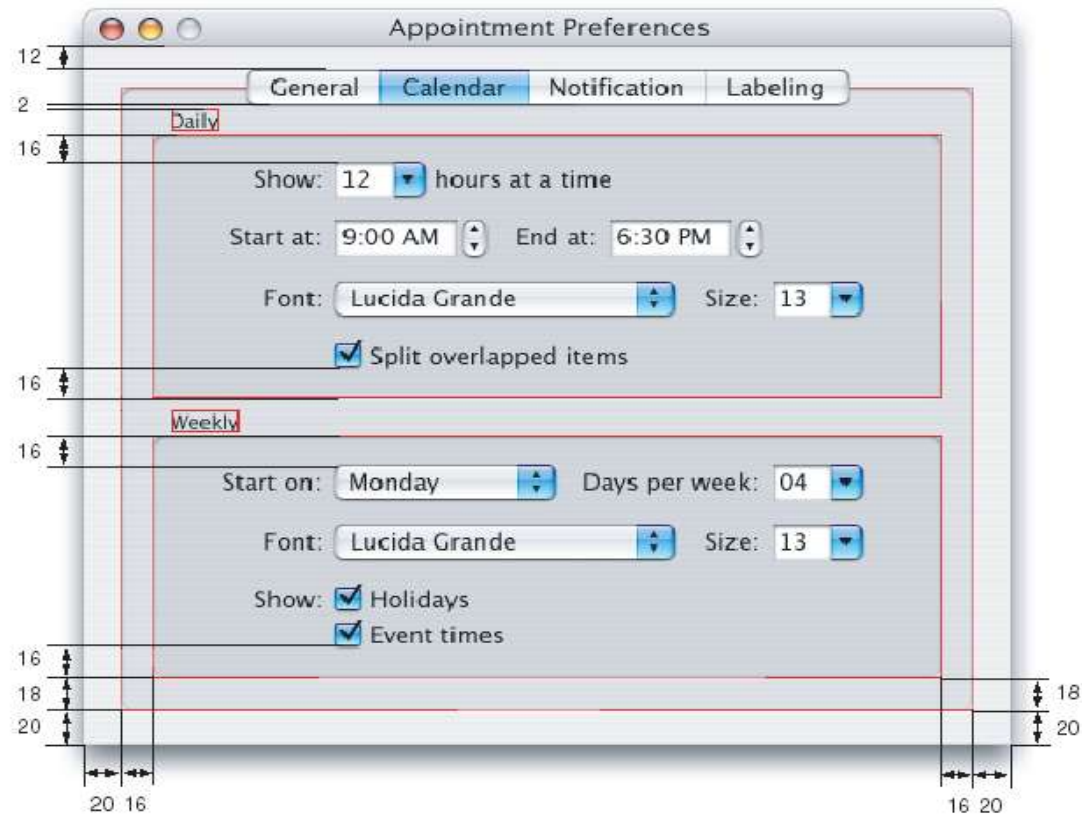


Figure 11-6 Center-equalization in a changeable pane dialog



### Example 5: Apple Human Interface Guidelines (page 207, 209 & 210)

Figure 11-8 Layout dimensions for a changeable pane dialog



# Specific Guidelines for Operating Systems, Window Managers, and the WWW

- Introduction to the Apple Human Interface Guidelines  
<http://developer.apple.com/documentation/UserExperience/index.html>
- KDE User Interface Guidelines  
<http://developer.kde.org/documentation/design/ui/>  
<http://developer.kde.org/documentation/standards/kde/style/basics/>
- Palm OS® User Interface Guidelines  
<http://www.accessdevnet.com/docs/ui/UIGuidelinesTOC.html>
- MSDN - User Interface Design and Development  
<http://msdn.microsoft.com>
- GNOME Human Interface Guidelines (V2.3)  
[http://developer.gnome.org/projects/gup/hig/draft\\_hig\\_new/](http://developer.gnome.org/projects/gup/hig/draft_hig_new/)
- Web Guidelines  
<http://www.webstyleguide.com/wsg3/index.html>
- ... and many others!

# References

- B. Shneiderman: Designing the User Interface: Strategies for Effective Human-Computer Interaction, Third Edition. 1997.
- A. Cooper: About Face 2.0. Chapter 1 and 19 ff.
- Alan Dix, Janet Finlay, Gregory Abowd and Russell Beale: Human-Computer Interaction (third edition), Prentice Hall 2003
- D. A. Norman. The Design of Everyday Things. Basic Books 2002. Chapter 4.
- Jennifer Tidwell: Designing Interfaces - Patterns for Effective Interaction Design, O'Reilly 2005  
<http://designinginterfaces.com/>