Designing Interactive Systems I

Ten Golden Rules, Responsiveness

Prof. Dr. Jan Borchers
Media Computing Group
RWTH Aachen University

Winter Semester ’23/’24

https://hci.rwth-aachen.de/dis

RWTH AACHEN UNIVERSITY

Review

- GOMS and KLM
  - Execution times for routine tasks
- Information efficiency
- Character efficiency

Ten Golden Rules of Interface Design

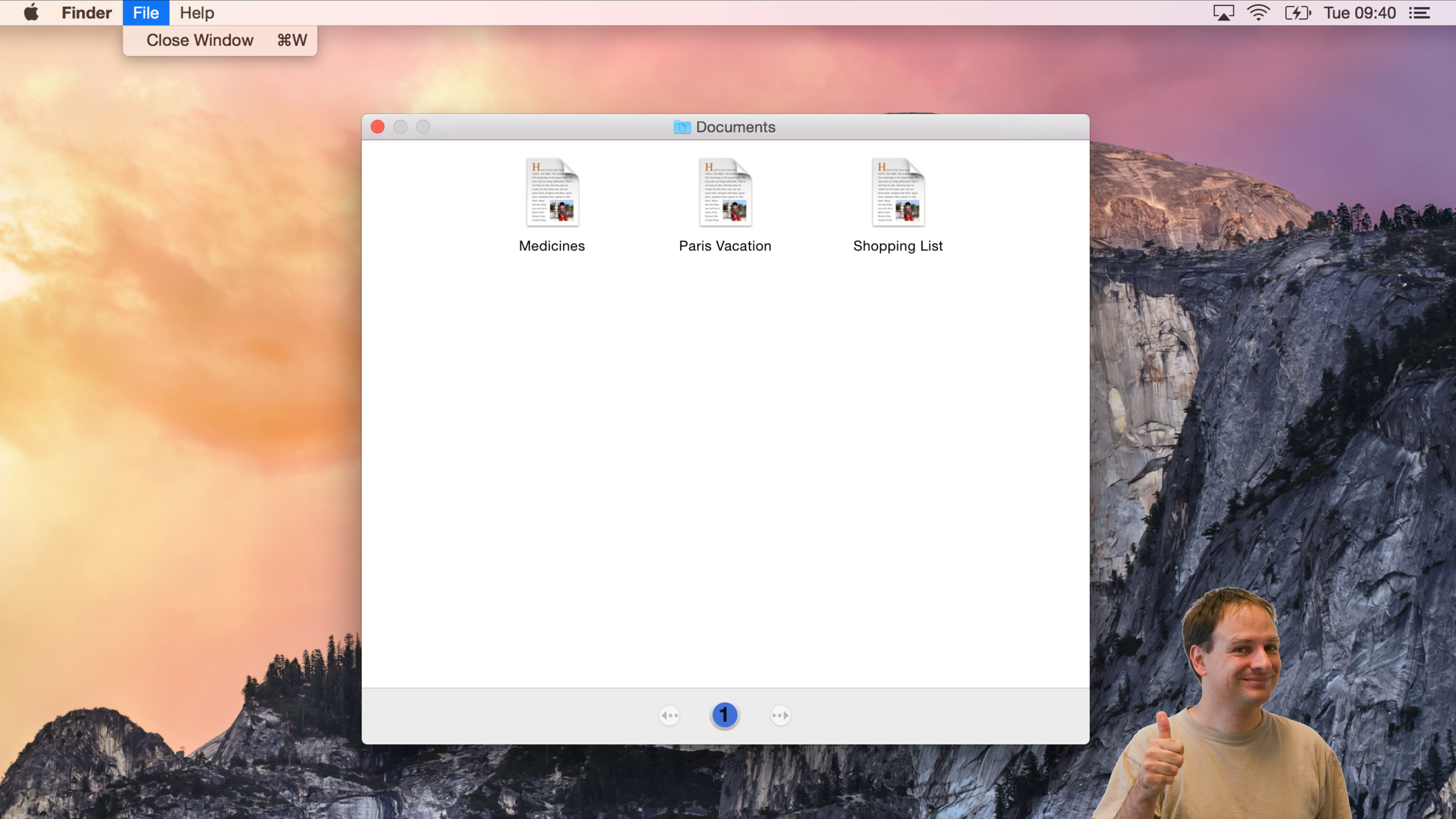
Ten Golden Rules of Interface Design

1. Keep the interface simple!
2. Speak the user's language!
3. Be consistent and predictable!
4. Provide feedback & be responsive!
5. Minimize memory load!
6. Avoid errors, help to recover, offer undo!
7. Design clear exits and closed dialogs!
8. Include help and documentation!
9. Address diverse user needs!
10. Hire a graphic designer!

1. Keep the Interface Simple!

Apple's Mac OS X Snow Leopard, 2009

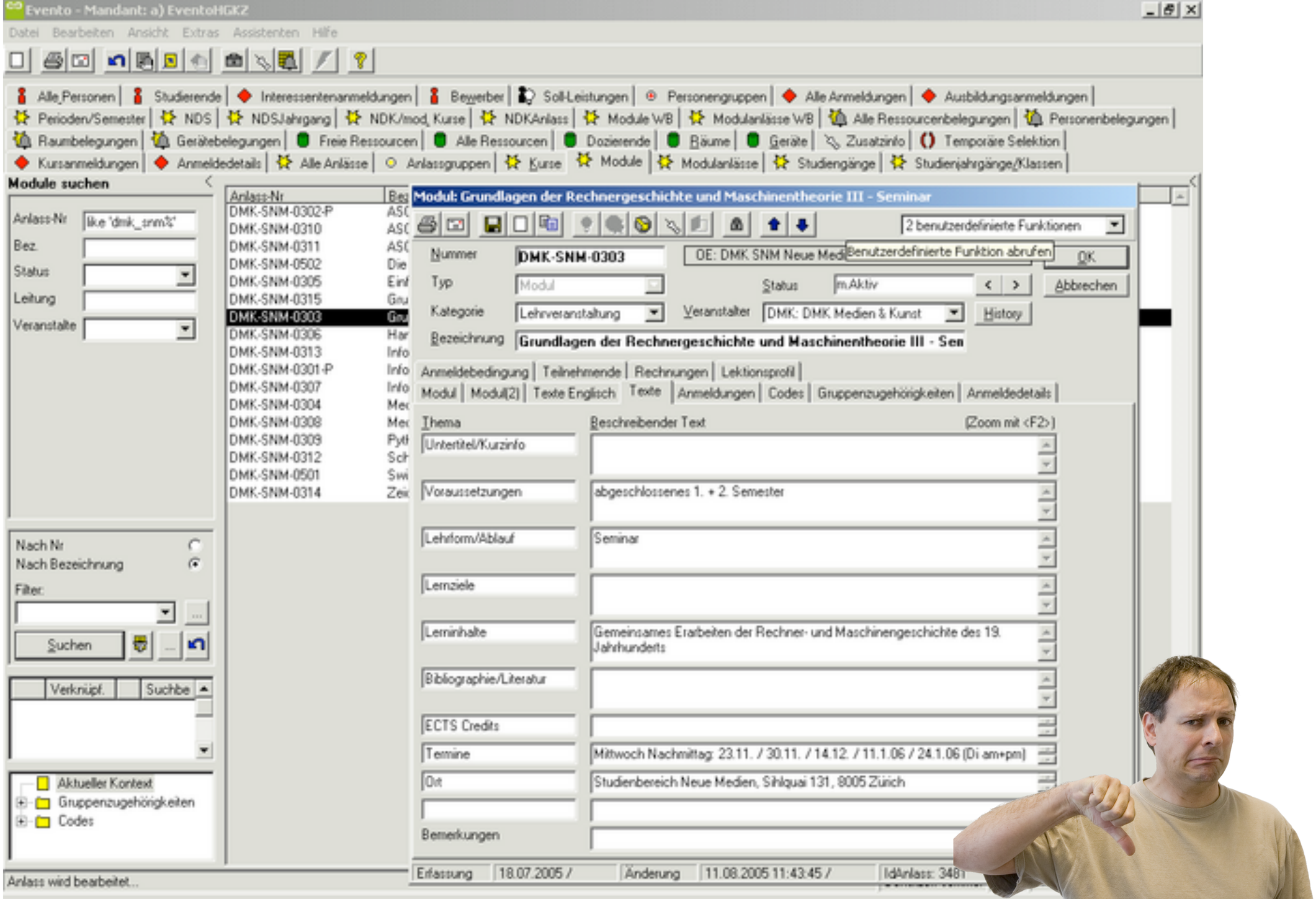
- Most important rule
- First design is often too complex & awkward
- Avoid feature creep
  - Some consumers will ask for more and more features
  - But usability must not suffer
  - Experience: 80% of users use only 20% of features (e.g., Word)
  - Honorable goal would be: Next version will have no new features, just be easier to use
  - If pressed, move feature sets out to sub-dialogs


Finder
File
Help
Close Window ⌘W
Tue 09:40
Documents
Medicines
Paris Vacation
Shopping List
1

Feature Creep Example: Blu-Ray Players

https://www.samsung.com/de/video-players/uhd-blu-ray-player/ultra-hd-blu-ray-player-black-ubd-m9500-zg/

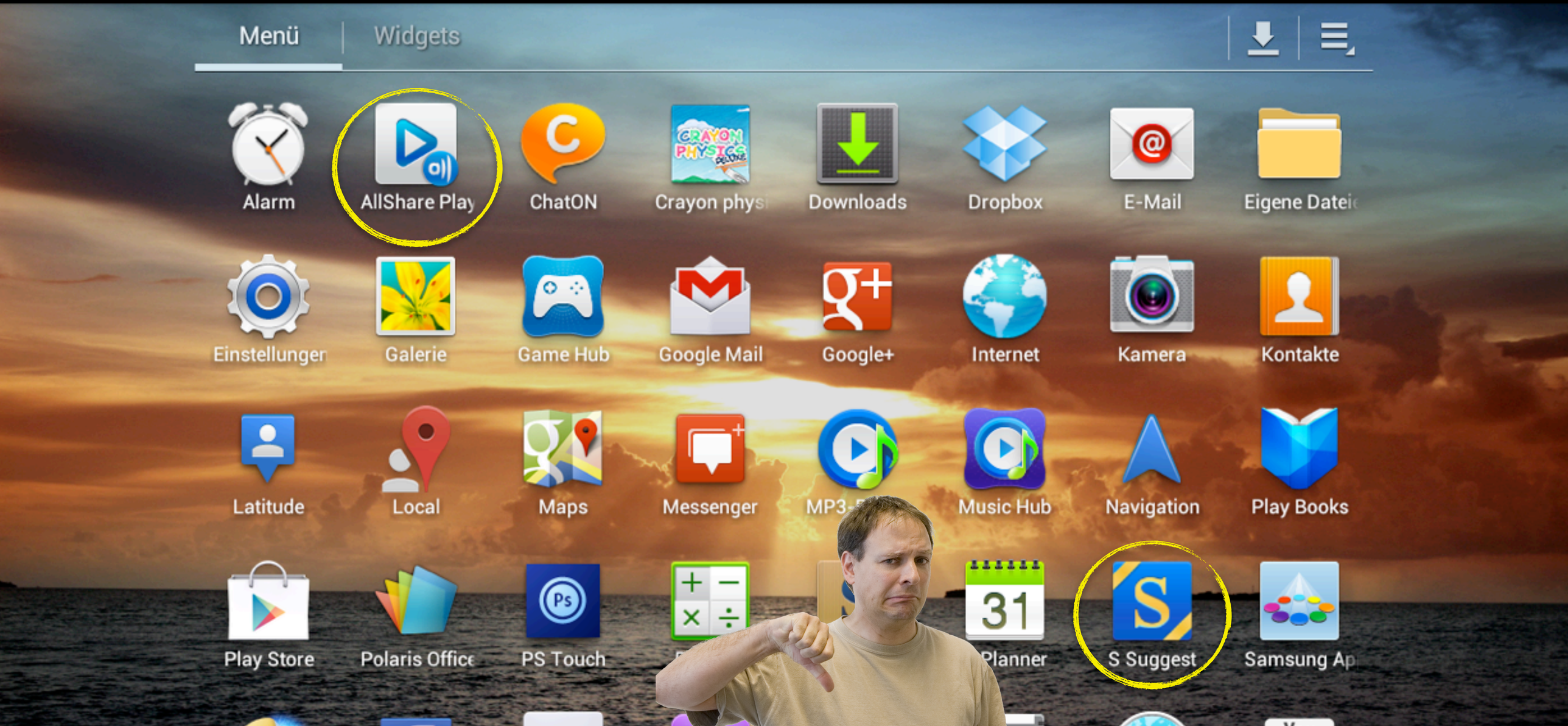
Example: Simple Alarm Clock


Evento - Mandant: a) EventoHGKZ
Datei Bearbeiten Ansicht Extras Assistenten Hilfe
Alle Personen | Studierende | Interessentenanmeldungen | Bewerber | Soll-Leistungen | Personengruppen | Alle Anmeldungen | Ausbildungsanmeldungen
Perioden/Semester | NDS | NDSJahrgang | NDK/mod. Kurse | NDKAnlass | Module WB | Modulanlässe WB | Alle Ressourcenbelegungen | Personenbelegungen
Raumbelegungen | Gerätebelegungen | Freie Ressourcen | Alle Ressourcen | Dozierende | Räume | Geräte | Zusatzinfo | Temporäre Selektion
Kursanmeldungen | Anmeldedetails | Alle Anlässe | Anlassgruppen | Kurse | Module | Modulanlässe | Studiengänge | Studienjahrgänge/Klassen
Module suchen
Anlass-Nr
Bez.
Status
Leitung
Veranstalte
Nach Nr
Nach Bezeichnung
Filter:
Suchen
Verknüpf.
Suchbe
Aktueller Kontext
Gruppenzugehörigkeiten
Codes
Anlass-Nr
DMK-SNM-0302-P
DMK-SNM-0310
DMK-SNM-0311
DMK-SNM-0502
DMK-SNM-0305
DMK-SNM-0315
DMK-SNM-0303
DMK-SNM-0306
DMK-SNM-0313
DMK-SNM-0301-P
DMK-SNM-0307
DMK-SNM-0304
DMK-SNM-0308
DMK-SNM-0309
DMK-SNM-0312
DMK-SNM-0501
DMK-SNM-0314
Modul: Grundlagen der Rechnergeschichte und Maschinentheorie III - Seminar
2 benutzerdefinierte Funktionen
Benutzerdefinierte Funktion abrufen
OK
Abbrechen
History
Nummer
DMK-SNM-0303
OE: DMK SNM Neue Medi
Typ
Modul
Status
m.Aktiv
Kategorie
Lehrveranstaltung
Veranstalter
DMK: DMK Medien & Kunst
Bezeichnung
Grundlagen der Rechnergeschichte und Maschinentheorie III - Sen
Anmeldebedingung | Teilnehmende | Rechnungen | Lektionsprofil
Modul | Modul(2) | Texte Englisch | Texte | Anmeldungen | Codes | Gruppenzugehörigkeiten | Anmeldedetails
Thema
Beschreibender Text
(Zoom mit <F2>)
Untertitel/Kurzinfo
Voraussetzungen
abgeschlossenes 1. + 2. Semester
Lehrform/Ablauf
Seminar
Lernziele
Lerninhalte
Gemeinsames Erarbeiten der Rechner- und Maschinengeschichte des 19. Jahrhunderts
Bibliographie/Literatur
ECTS Credits
Termine
Mittwoch Nachmittag: 23.11. / 30.11. / 14.12. / 11.1.06 / 24.1.06 (Di am+pm)
Ort
Studienbereich Neue Medien, Sihlquai 131, 8005 Zürich
Bemerkungen
Erfassung
18.07.2005 /
Änderung
11.08.2005 11:43:45 /
IdAnlass: 3481
Anlass wird bearbeitet...

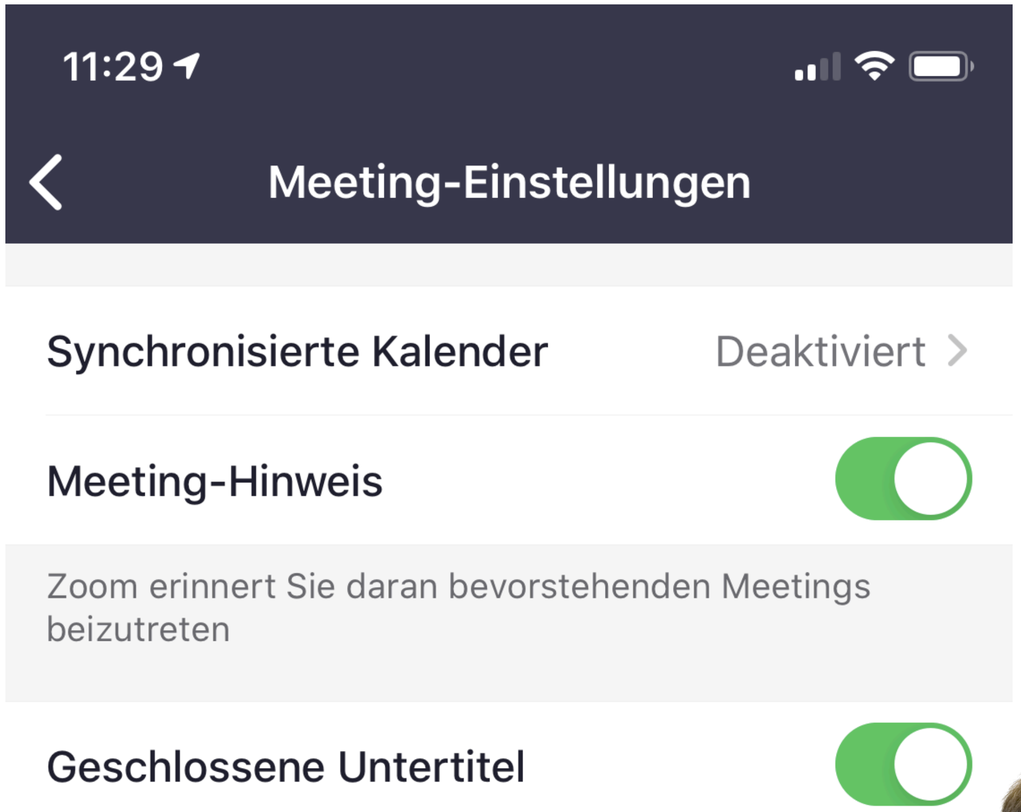
2. Speak the User's Language!

- Take words and concepts from the application domain, not technology
- Determine terminology during initial user interviews and task analysis
- Example:
  "File" means less to an architect who is new to computers than "drawing"
- Applies to words for objects, but also work processes and tasks (e.g., "order")

Example: Samsung Tablet



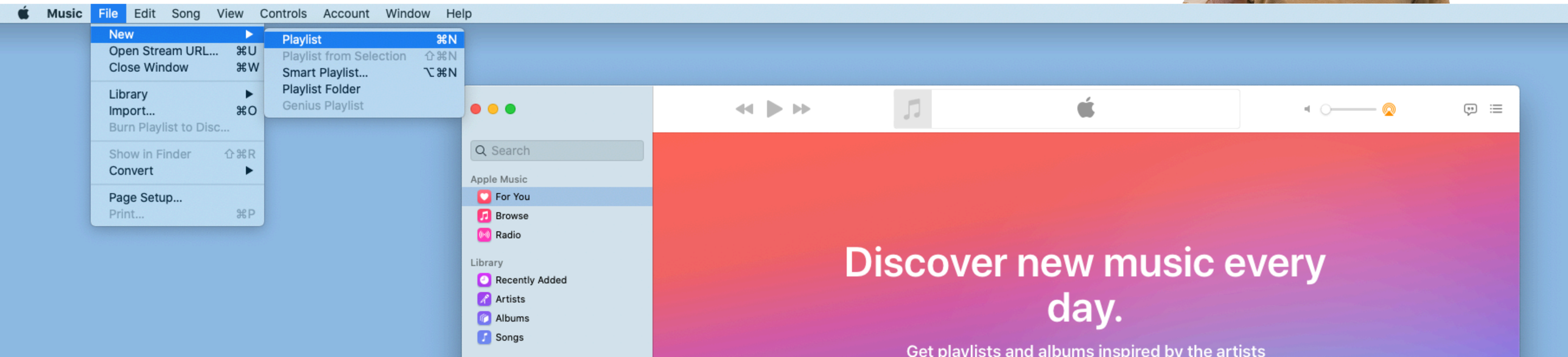
Example: Zoom Captions



Example: Apple Music

- Talks about “music”, “songs”, “video”, “movies”, “playlists”, not “files”
  - In menus, dialogs, and online help (⇒Rule 3: Consistency)

- Exceptions: E.g., “File” menu
  - Conflict with cross-application consistency



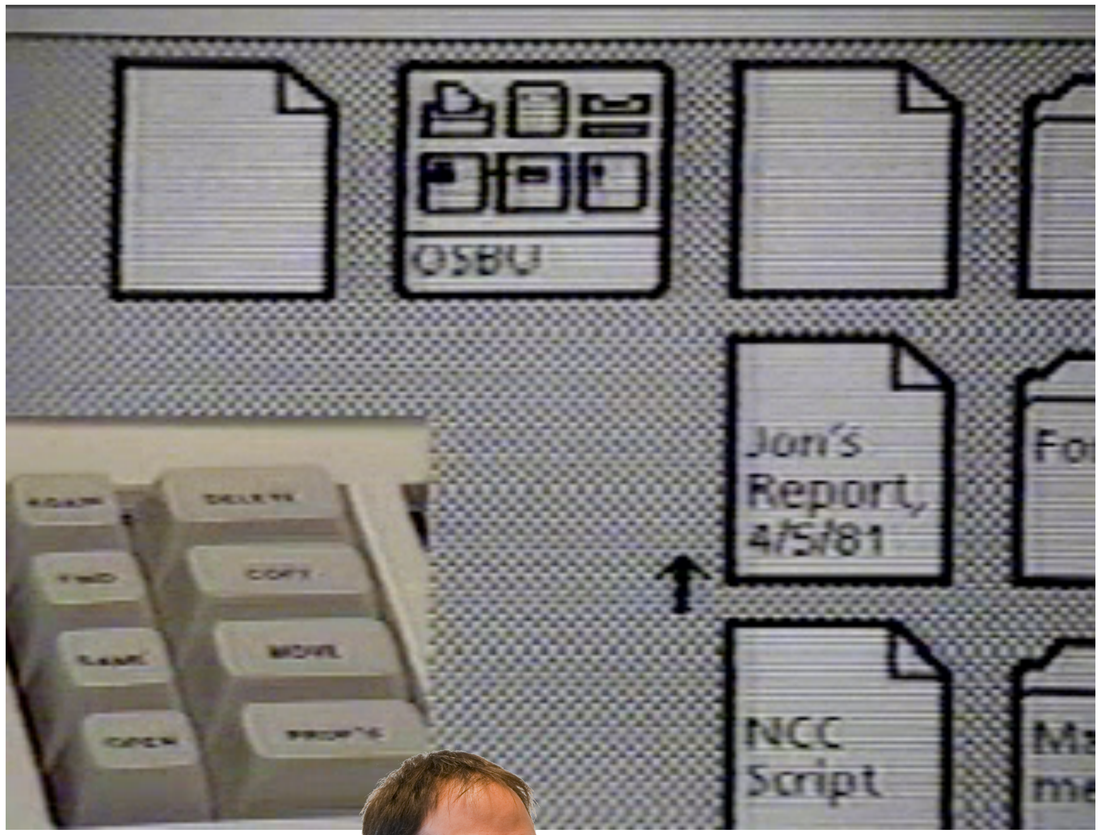
3. Be Consistent and Predictable!

- Consistency is needed across many levels:
  - Similar commands for similar situations
  - Consistent terminology in menus, dialogs, help pages, etc.
  - Consistent fonts, layout, color coding, upper/lower cases, etc. throughout the system
  - Only few obvious exceptions
    - No clear-text echo when entering passwords
    - Extra security check before erasing files, etc.

Example: Xerox Star Command Buttons

- Same (physical) buttons to copy a file, a word in a text editor, an object in a graphics program, etc.

- Still true today (Cut/Copy/Paste)

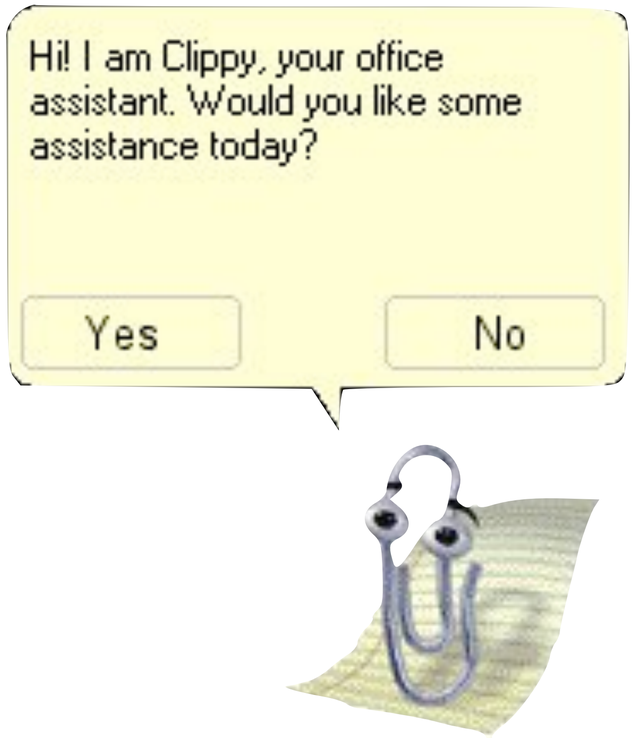


Consistency through Vertical Design

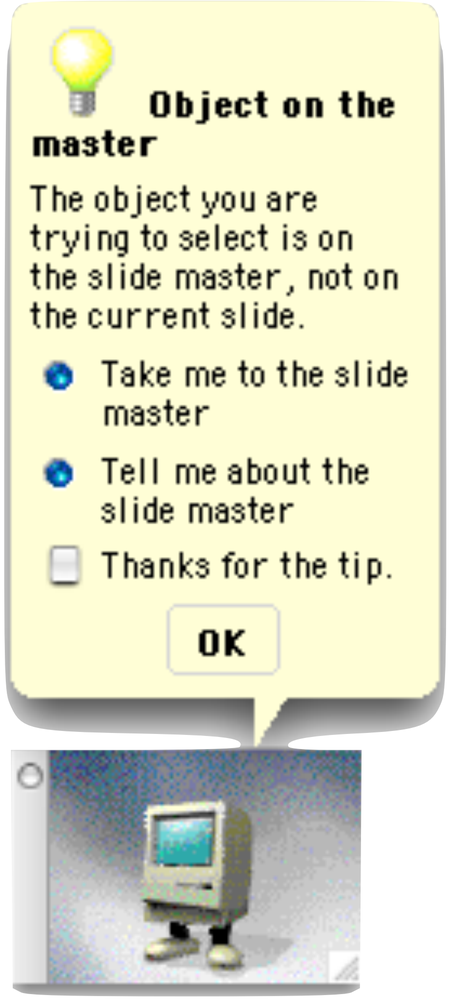
Predictability

- Follow the “Principle of Least Surprise”
  - System should always react so that it minimizes the user’s surprise (and therefore, confusion and irritation)
- Don’t do unexpected things
  - …and don’t make actions unexpectedly difficult (“…how do I print this in duplex?”)
- Users (especially experts) like to be “in control”
  - They initiate actions, the system responds

Principle of Least Surprise



PowerPoint Office Assistant



Timeouts are Evil!

RWTH AACHEN UNIVERSITY

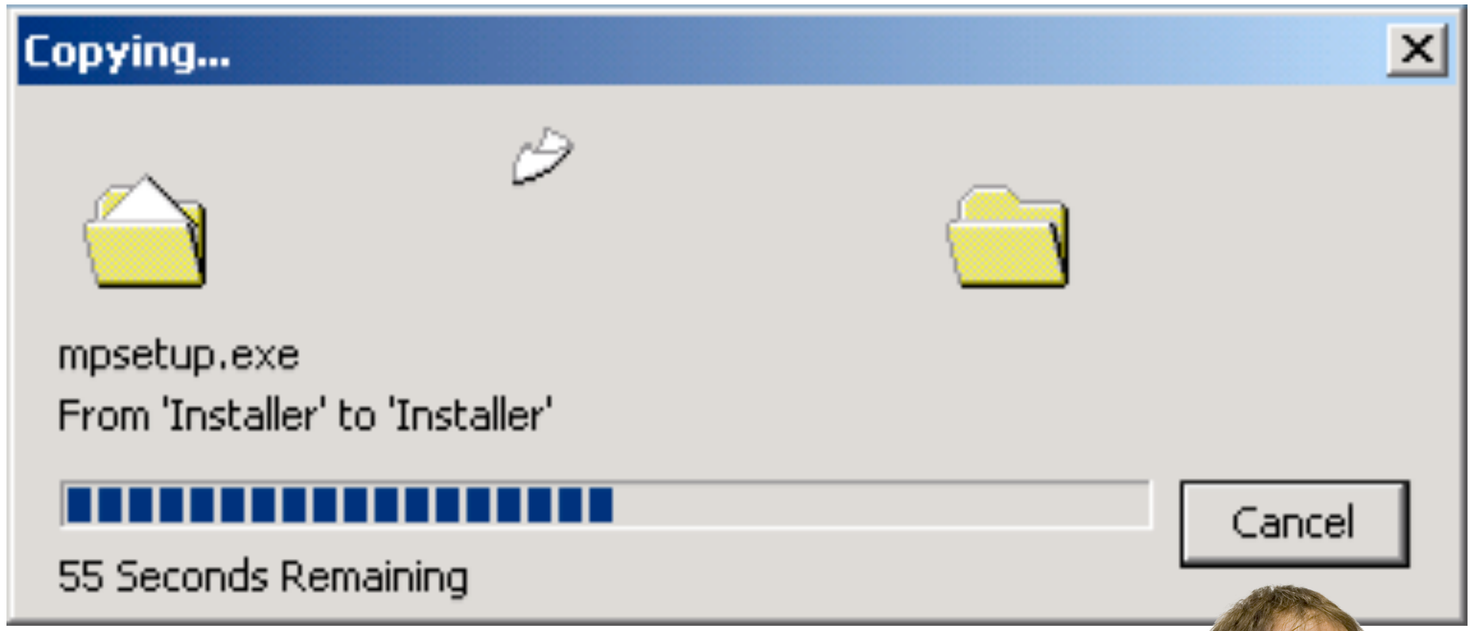

Emergency Exit
Press on bar for 3 seconds
Door lock will release in 15 seconds
Sortie de secours
Appuyer sur la barre pendant 3 secondes
Le dispositif de verrouillage se déclenchera dans 15 secondes
WARNING!

4. Provide Feedback & Be Responsive!

- Recall the Seven Stages of Action
  - Complete & continuous feedback bridges Gulf of Evaluation
- Each user action requires some feedback
  - Subtle for small/short/frequent actions (e.g., key press, menu selection)
  - More noticeable for main/long/infrequent actions (e.g., saving or deleting files)
  - Icons in GUIs simplify visualizing object state and actions: direct manipulation
- Nothing is more frustrating for the user than “Where am I?” or “What is it doing now?”

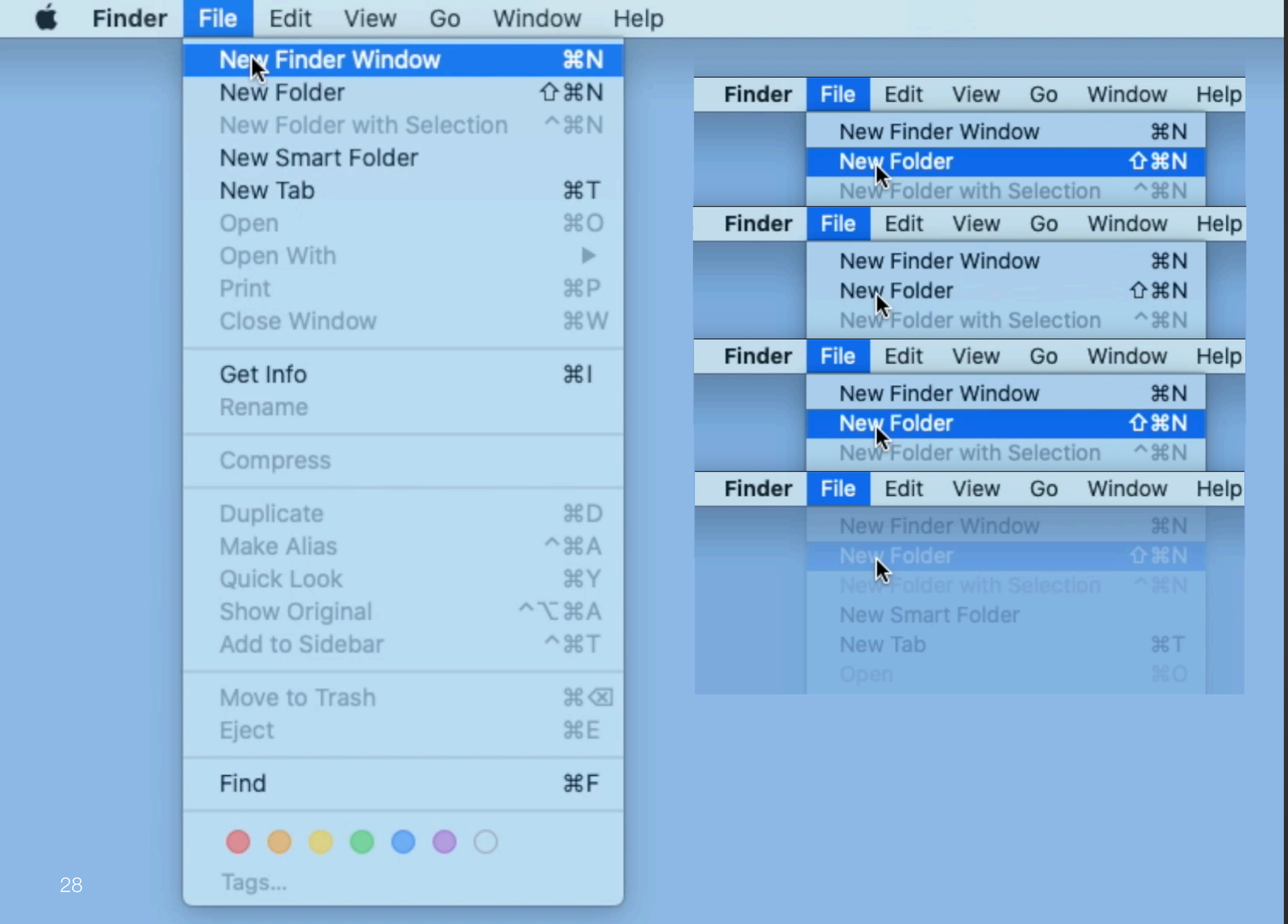
Example: Windows 2000 Progress Dialog for Copying Files

- What's wrong with this picture?



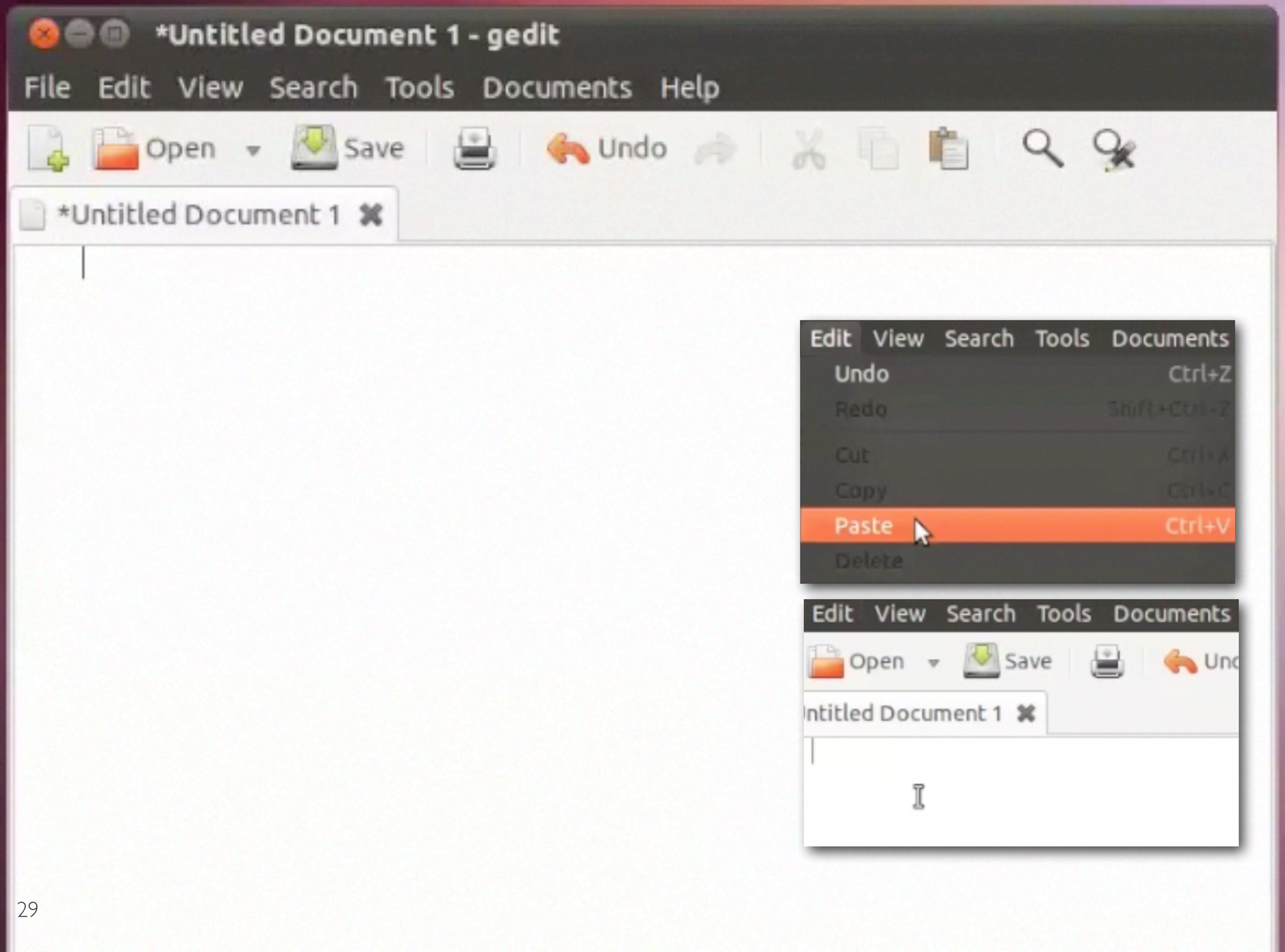
Example: Menu Selection

- What happens when you select a menu item?



macOS Catalina Menu

GNOME (CD Version)

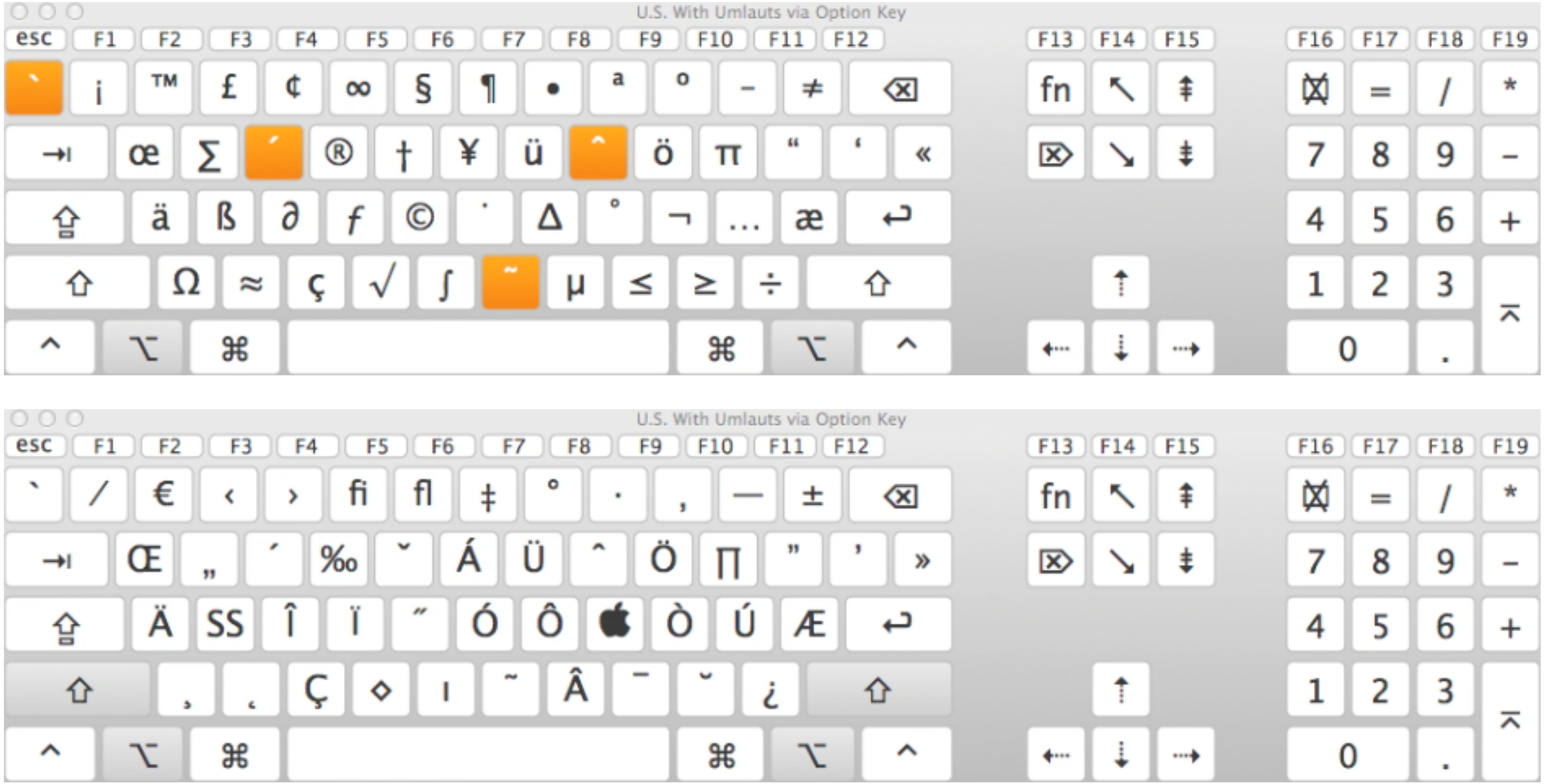


Haptic Feedback

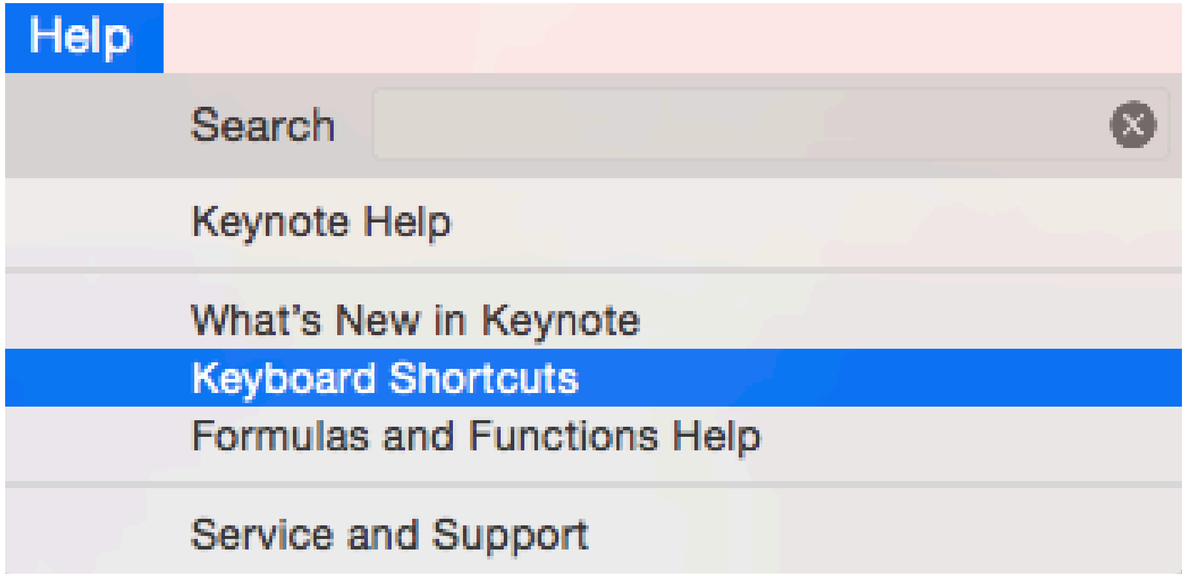
5. Minimize Memory Load!

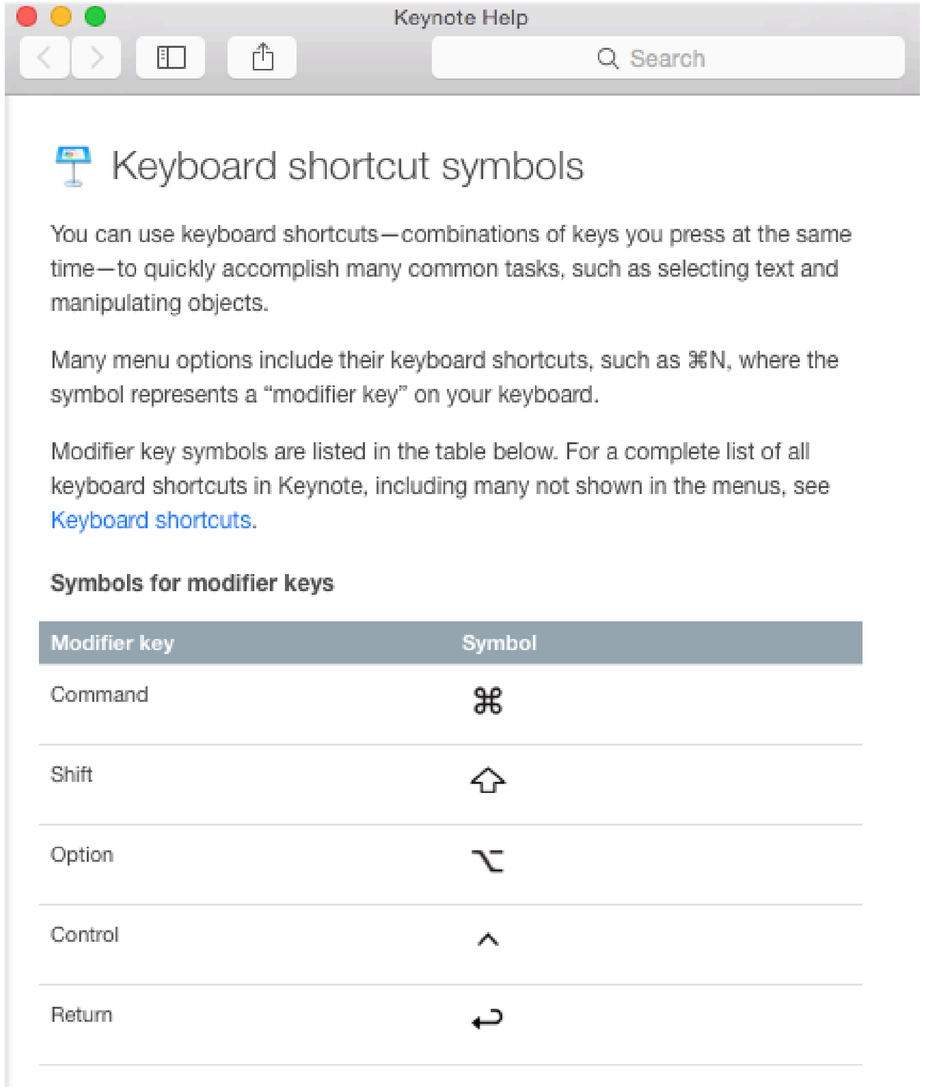
- Short-term memory: limited capacity (ca. 4 ± 1 chunks)
- Avoid situations where prior dialog information has to be reproduced from memory
  - E.g., user should not have to type anything in twice.
- Display information so it’s easy to parse (Gestalt laws)
- Provide obvious access to help pages for codes, abbreviations, etc.
- It’s easier to minimize memory load with GUIs than command line interfaces
  - “Read & Select” instead of “Remember & Type”

Keyboard Viewer



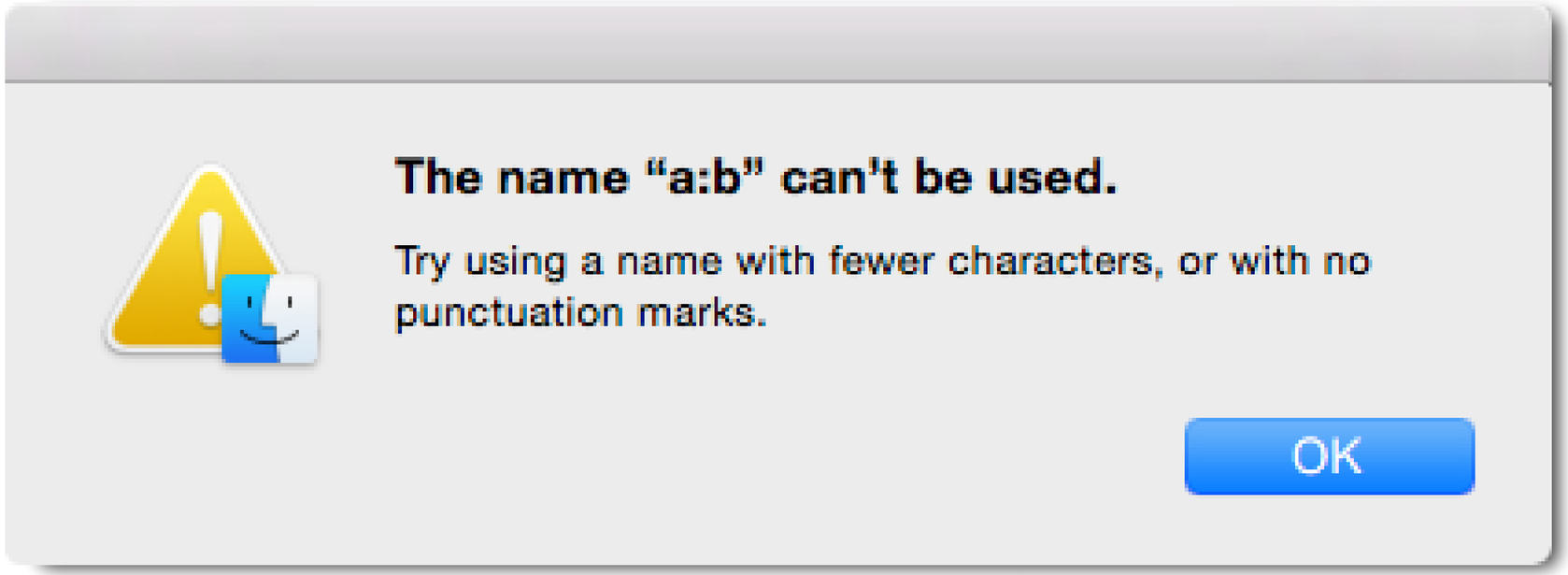
Keyboard Shortcut List





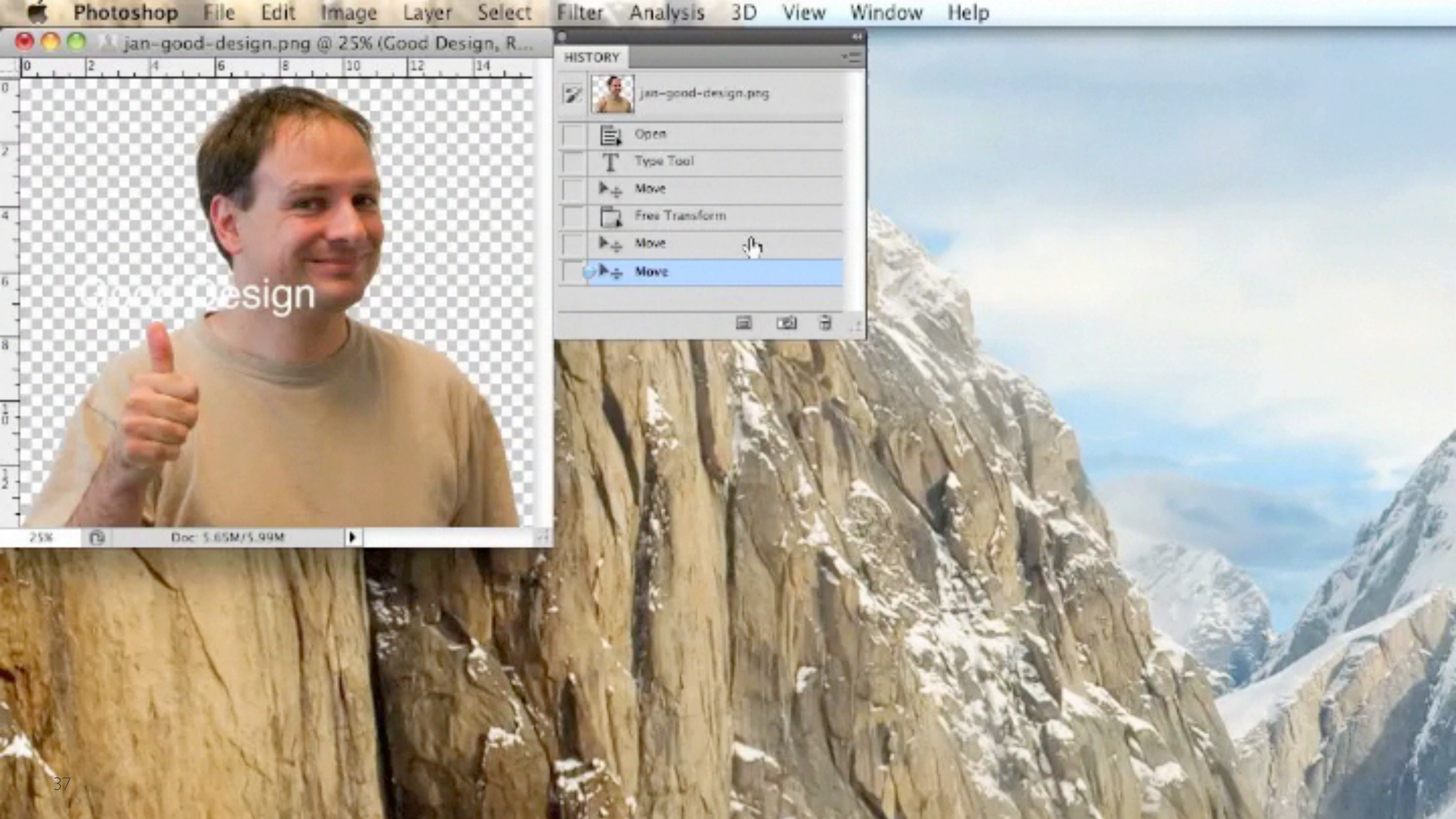
6. Avoid Errors, Help to Recover!

- Errors lead to stress
  - So offer simple, constructive, concrete, helpful, and comfortable instructions to recover
  - System state should not change through wrong input, or should be easy to restore
- Best: Design system so mistakes cannot be made in the first place. Examples:
  - Selection instead of (mis)typing
  - Cannot type letters in numerical data fields
  - Arcade game machines have virtually no error messages!
  - Automatic correction of illegal characters in file names



6. Avoid Errors, Help to Recover!

- Offer undo
  - As many actions as possible should be reversible
  - Lowers anxiety because users know errors are correctable
  - Encourages users to try out new functions
  - Ideal: multiple undo, and at multiple levels


Photoshop File Edit Image Layer Select Filter Analysis 3D View Window Help
jan-good-design.png @ 25% (Good Design, R...
HISTORY
jan-good-design.png
Open
Type Tool
Move
Free Transform
Move
Move
25%
Doc: 5.65M/5.99M

7. Design Clear Exits & Closed Dialogs!

- Three most common questions of users during a dialog:
  - Where am I?
  - What can I do here?
  - How do I get back to where I was?
- Clear exits (“Back”, “Quit”) help with Question 3
- Closed dialogs:
  - Provide feeling of having completed a step
  - Allows user to relax, “take a breath”, frees the mind for the next step

amazon.com

Hello, C WACHARAMANOTHAM. We have recommendations for you. (Not C?)

C's Amazon.com | Today's Deals | Gifts & Wish Lists | Gift Cards | Your Account | Help

Shop All Departments | Search | All Departments | GO | Cart | Wish List

Thank you, your order has been placed.

An e-mail confirmation has been sent to you.

Order Number: **104-1969352-5141057**

- 1 item will be shipped to Chatchavan Wacharamanotham by Amazon.com. Estimated delivery **January 18, 2011 - February 7, 2011**

› Review or edit your order

Next time use Express Checkout with PayPhrase

Buy on Amazon and across the web with a simple phrase.

Choose your PayPhrase: "C's Surprising Doors"

(Use this suggestion, see others, or enter your own)

Orders will ship to: Chatchavan Wacharamanotham, Lehrstuhl In...
Orders will be paid using: VISA ****-

Create your PayPhrase

A Payphrase is an easy-to-remember shortcut to shipping and payment information in your Amazon.com account. Use it on Amazon.com and across the web. (Learn more)

Recommendations Based on Your Order


Connection was lost
Would you want to reconnect?
Yes
OK
SAMSUNG
Fail.in.th

8. Include Help and Documentation!

- Hierarchy of help systems, with increasing breadth and decreasing ease-of-access:
  - Dynamic Descriptors, such as Tooltips (but let users disable them!)
  - Online tutorials and references
  - Printed documentation (but…)

Users don’t read manuals!

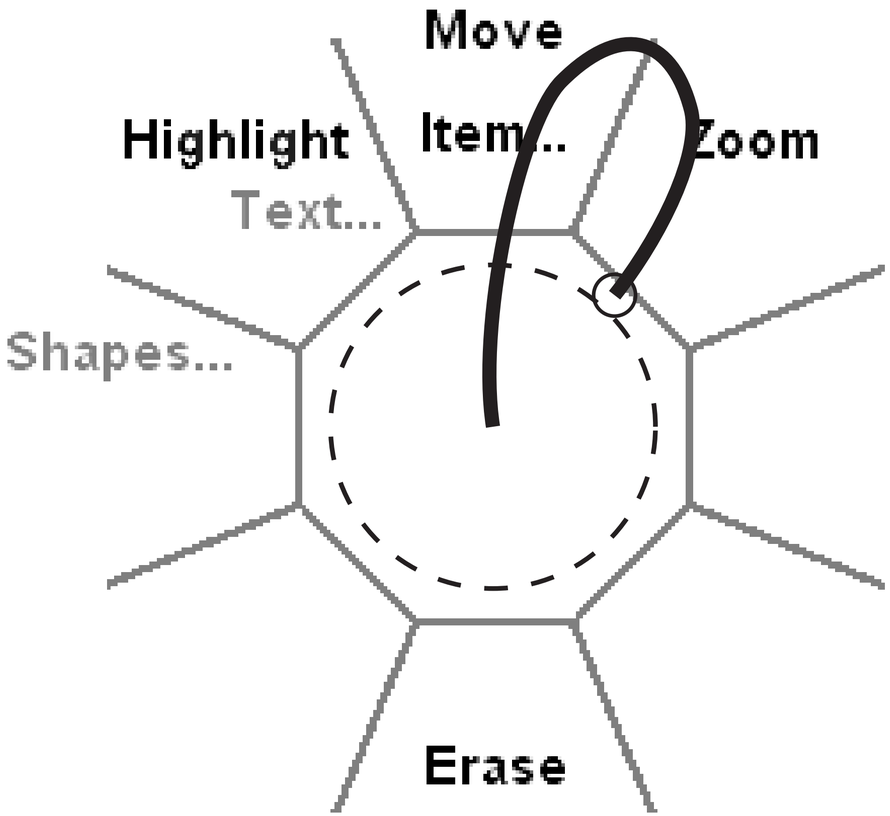
- More active help can be useful:
  - Assistants and Wizards
  - But danger: system takes over initiative, which breaks Rule 3 (predictability)

9. Address Diverse User Needs!

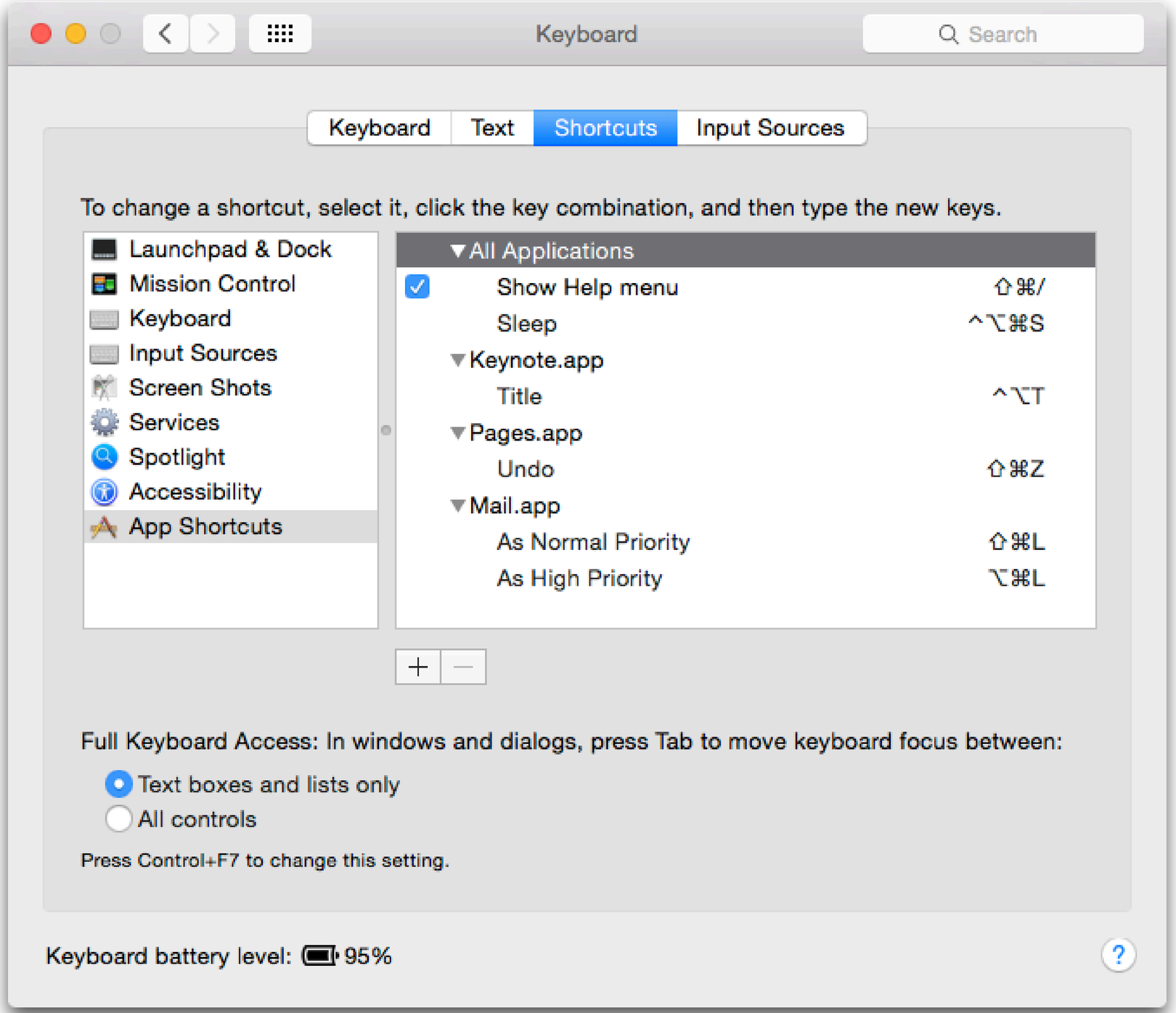
- Novices want more explanations
- Frequent users want less fussy and faster interaction
  - They value (configurable) keyboard shortcuts, macro recording, programmability, and quick responses without unnecessary feedback (for them)
- Different age ranges have different interface expectations
- Technology affinity (“enjoying to play with gadgets”) varies widely among people
- But conflict: If in doubt, Rule 1 (“Keep the interface simple”) is more important! May have to focus on a user group

Example: PostBrainstorm

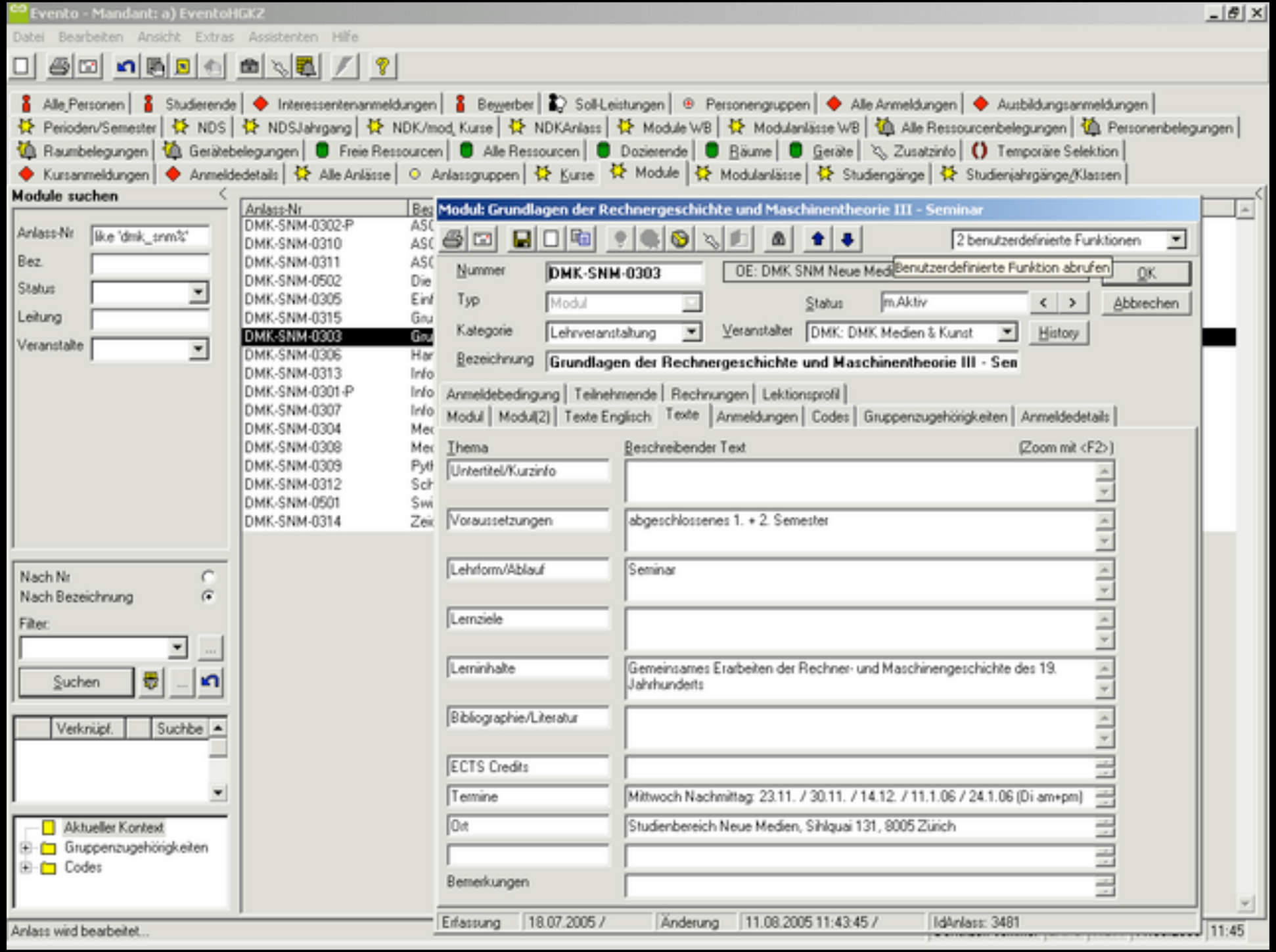
- New users get popup menu
- Experienced users remember the gestures to select frequent commands from the menu
- The menu does not even pop up when the gesture is done rapidly
- But: If you ever forget the gesture, just wait for a fraction of a second, and you can revert to using the popup menu
- The result: Fluid and reversible transition from menu selection to gesture commands



[F. Guimbretière, Stanford, UIST 2000]


Keyboard
Search
Keyboard
Text
Shortcuts
Input Sources
To change a shortcut, select it, click the key combination, and then type the new keys.
Launchpad & Dock
Mission Control
Keyboard
Input Sources
Screen Shots
Services
Spotlight
Accessibility
App Shortcuts
All Applications
Show Help menu ⇧⌘/
Sleep ^⌥⌘S
Keynote.app
Title ^⌥T
Pages.app
Undo ⇧⌘Z
Mail.app
As Normal Priority ⇧⌘L
As High Priority ⌥⌘L
Full Keyboard Access: In windows and dialogs, press Tab to move keyboard focus between:
Text boxes and lists only
All controls
Press Control+F7 to change this setting.
Keyboard battery level: 95%

10. Hire a Graphic Designer!


Evento - Mandant: a) EventoHGKZ
Datei Bearbeiten Ansicht Extras Assistenten Hilfe
Alle Personen Studierende Interessentenanmeldungen Bewerber Soll-Leistungen Personengruppen Alle Anmeldungen Ausbildungsanmeldungen
Perioden/Semester NDS NDSJahrgang NDK/mod. Kurse NDKAnlass Module WB Modulanlässe WB Alle Ressourcenbelegungen Personenbelegungen
Raumbelegungen Gerätebelegungen Freie Ressourcen Alle Ressourcen Dozierende Räume Geräte Zusatzinfo Temporäre Selektion
Kursanmeldungen Anmeldedetails Alle Anlässe Anlassgruppen Kurse Module Modulanlässe Studiengänge Studienjahrgänge/Klassen
Module suchen
Anlass-Nr like 'dmk_snm%'
Bez.
Status
Leitung
Veranstalte
Nach Nr
Nach Bezeichnung
Filter:
Suchen
Verknüpf.
Suchbe
Aktueller Kontext
Gruppenzugehörigkeiten
Codes
Anlass-Nr
DMK-SNM-0302-P
DMK-SNM-0310
DMK-SNM-0311
DMK-SNM-0502
DMK-SNM-0305
DMK-SNM-0315
DMK-SNM-0303
DMK-SNM-0306
DMK-SNM-0313
DMK-SNM-0301-P
DMK-SNM-0307
DMK-SNM-0304
DMK-SNM-0308
DMK-SNM-0309
DMK-SNM-0312
DMK-SNM-0501
DMK-SNM-0314
Modul: Grundlagen der Rechnergeschichte und Maschinentheorie III - Seminar
2 benutzerdefinierte Funktionen
Benutzerdefinierte Funktion abrufen
Nummer DMK-SNM-0303
OE: DMK SNM Neue Medi
OK
Typ Modul
Status mAktiv
Abbrechen
Kategorie Lehrveranstaltung
Veranstalter DMK: DMK Medien & Kunst
History
Bezeichnung Grundlagen der Rechnergeschichte und Maschinentheorie III - Sen
Anmeldebedingung Teilnehmende Rechnungen Lektionsprofil
Modul Modul(2) Texte Englisch Texte Anmeldungen Codes Gruppenzugehörigkeiten Anmeldedetails
Thema
Beschreibender Text
(Zoom mit <F2>)
Untertitel/Kurzinfo
Voraussetzungen
abgeschlossenes 1. + 2. Semester
Lehrform/Ablauf
Seminar
Lernziele
Lerninhalte
Gemeinsames Erarbeiten der Rechner- und Maschinengeschichte des 19. Jahrhunderts
Bibliographie/Literatur
ECTS Credits
Termine
Mittwoch Nachmittag: 23.11. / 30.11. / 14.12. / 11.1.06 / 24.1.06 (Di am+pm)
Ort
Studienbereich Neue Medien, Sihlquai 131, 8005 Zürich
Bemerkungen
Erfassung 18.07.2005 /
Änderung 11.08.2005 11:43:45 /
IdAnlass: 3481
Anlass wird bearbeitet...
11:45


PENNY
MARKT
HOHE QUALITÄT ZUM NIEDRIGSTEN PREIS!
Seit 11.10.
PREISSTURZ!

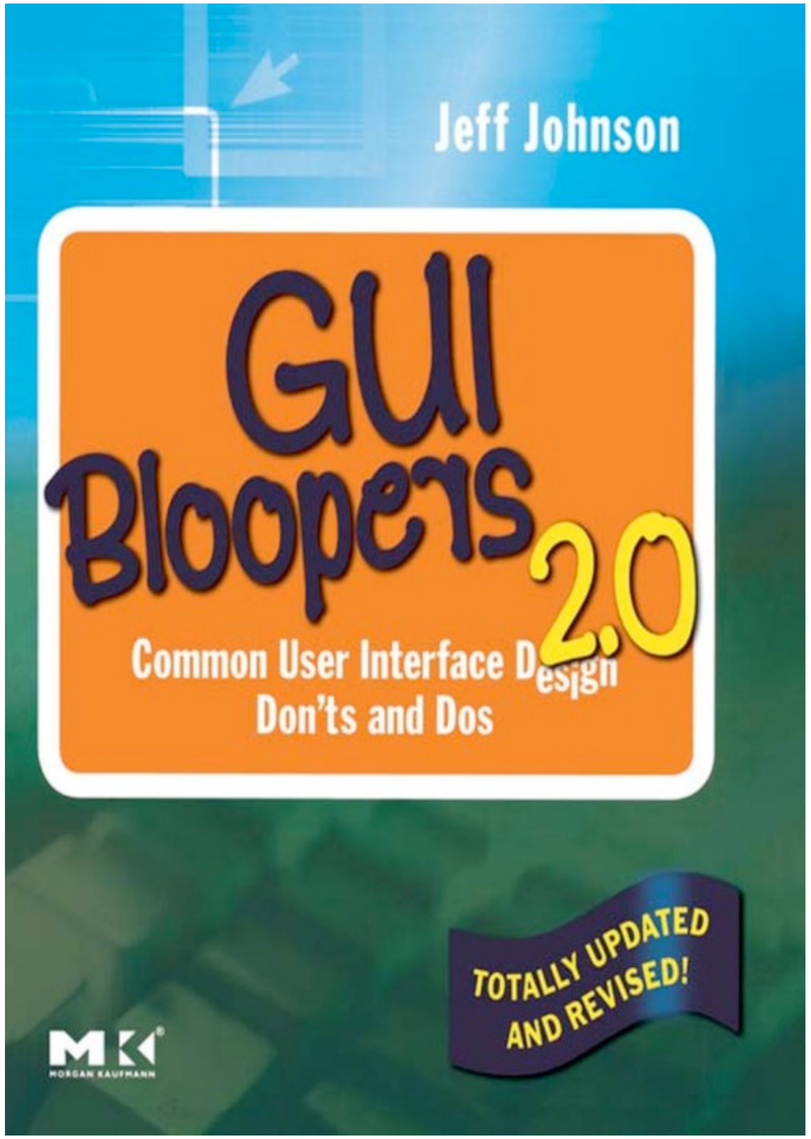
Ten Golden Rules of Interface Design

1. Keep the interface simple!
2. Speak the user's language!
3. Be consistent and predictable!
4. Provide feedback & be responsive!
5. Minimize memory load!
6. Avoid errors, help to recover, offer undo!
7. Design clear exits and closed dialogs!
8. Include help and documentation!
9. Address diverse user needs!
10. Hire a graphic designer!

Responsiveness and Performance

Responsiveness

- See also: Jeff Johnson, *GUI Bloopers 2.0*
- Key usability problem of interactive systems
  - Bad responsiveness opens Gulf of Evaluation
- Examples for bad responsiveness:
  - A screen pointer that doesn't keep up
  - Delayed response to button-clicks
  - Sliders and scrollbars that lag
  - Applications that go "dead" during disk operations
  - Multiple screen repaints



Reasons for Poor Responsiveness

- Importance not widely known
  - UI designers think of other things first
  - UI designers rarely specify responsiveness
  - Programmers tend to equate it with performance
- This kind of tuning is always difficult
  - “We’ll get it in the next release,” and so on
- Developers treat human input like machine input
- Simple, naïve implementations
- GUI tools and platforms are inadequate
  - Limitations of online apps (which everybody knows about)

Example: Scrollbar

- Does text move as you scroll (good) or after you let go (bad)?
- If designer doesn’t specify, developer will make a decision
- That will usually be the technically simplest
  - Since developers are not trained in user interface theory and concepts
  - Just as UI designers are generally not trained in implementing large software products in C++

Some Eternal Facts

- Responsiveness ≠ performance!
- Processing resources will always be limited
  - We still look at hourglass as much as 15 years ago
  - UIs are real-time systems with deadlines based on human cognition
  - Software does not need to do everything instantly, or in a given order, or even at all

Three Human Deadlines

- **0.1 seconds**
  - Perception of cause and effect (recall CMN model)
  - E.g., delay between moving mouse and pointer following, or between mouse click and inverting button
- **1 second**
  - Turn-taking in conversation, minimum reaction time for unexpected events
  - E.g., you have 1s max to show progress indicator, open window, or finish system-initiated operations (like auto-save)
- **10 seconds**
  - Typical human attention span
  - Max. time for one step of a task
    - E.g., entering a check into a banking program, or completing one step of a wizard
  - Max. time to finish giving input for a task
    - E.g., from selecting “Print” menu entry to sending off the print job

Design Techniques for Responsiveness

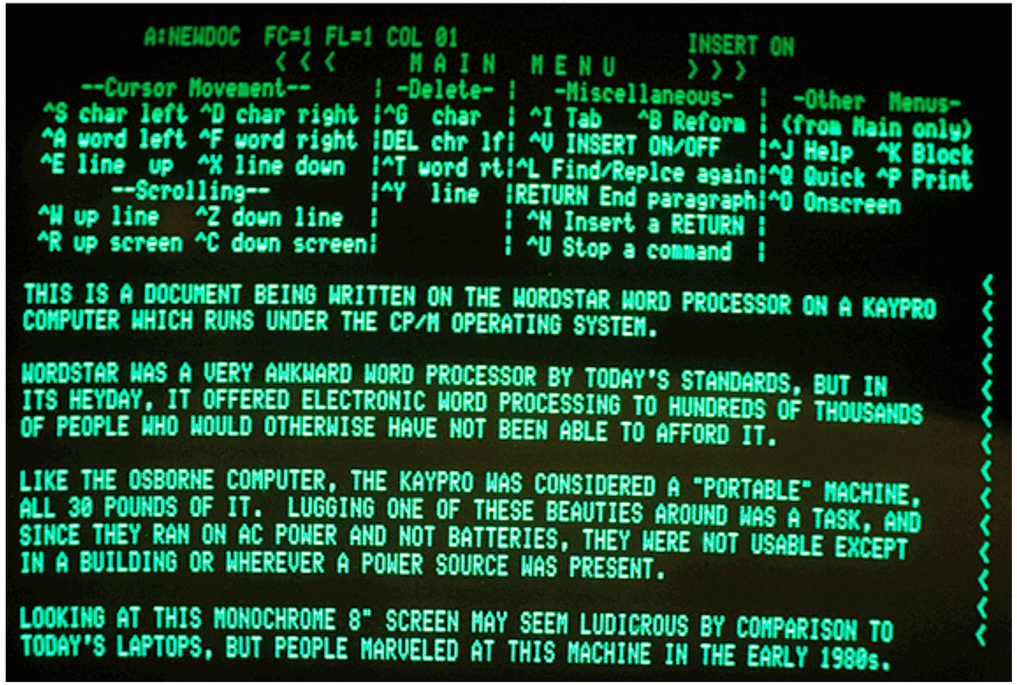
- Meet human-time deadlines
  - Rely on the three deadlines and recognize the differences
  - Acknowledge user input immediately, and display busy and progress indicators
  - Use them as frequently as you can, you never know when it will take longer
  - Example “Progress bar”:
    - Make it real, show total items remaining, overall progress, and estimated total time remaining
    - Only useful if it advances roughly linearly! (no hanging at 99% please)
    - Estimated time should always go down, never up
    - “Less than a minute” is better than “47 seconds” (why?)

Design Techniques for Responsiveness

- Display important information first
  - Example: How to draw a clock
- Work in parallel
  - Delegate work that isn't time-critical to background processes
  - Work ahead by preparing likely requests
- Optimize Queueing
  - Create a logical order by looking at all pieces first, then prioritize

Design Techniques for Responsiveness

- Manage time dynamically
  - Adjust the strategy if not keeping up
  - Decrease quality or quantity to keep up
- Example: WordStar (1978)
  - Ran on a 1 MHz computer, killed by IBM PC
  - Written by an amateur, but he accommodated by making the system responsive
  - WordStar never dropped characters typed
  - Characters typed were always on screen instantly
  - Instead stopped updating other areas of the screen



Design Techniques for Responsiveness

- Test under different conditions
  - Test under heavy loads
  - Test on slower systems, like your customers have
  - Test over slower net connections

Latency

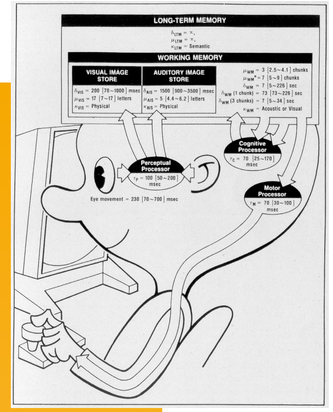
- Latency is crucial in every interactive system
- Stay well below 100 ms from input to display
- Throughput ≠ Latency
- Experimenters, watch your end-to-end latency (test with oscilloscope and LDR)
  - LCDs add dozens of ms of constant lag, USB can add 0..125 ms of lag *with jitter(!)*
- CRTs have near-zero lag, embedded systems and FPGA emulators can read inputs with zero lag
- See http://hci.rwth-aachen.de/latency

Summary

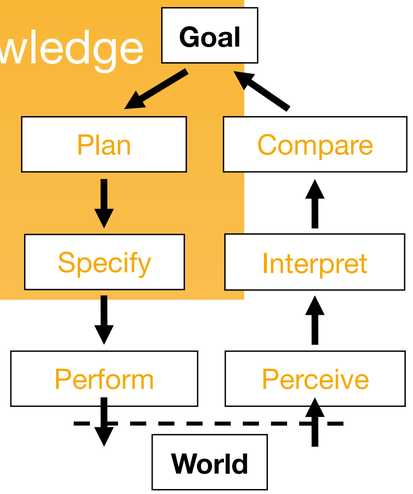
- **10 Golden Rules of Interface Design**
  - Keep the interface simple!
  - Speak the user's language!
  - Be consistent and predictable!
  - Provide feedback & be responsive!
  - Minimize memory load!
  - Avoid errors, help to recover, offer undo!
  - Design clear exits and closed dialogs!
  - Include help and documentation!
  - Address diverse user needs!
  - Hire a graphic designer!
- **Responsiveness and Performance**

Roadmap

Human



- Performance
- Models of interaction
  - Affordances
  - Mappings
  - Constraints
  - Types of knowledge
  - Errors
- Visual Design



Case Studies

- History of HCI
- Visions
- Technology Phases

Development Process

- Iterative design
- User observation
- Ideation
- Prototyping
- User studies and evaluation
- Interaction design notation