

CISC 836

Design of Usable Computing Systems

T.C. Nicholas Graham

<http://stl.cs.queensu.ca/~graham/cisc836>

User Interface Design Guidelines



- ⌘ Capture design knowledge
- ⌘ General guidelines capture high-level attributes that make systems more usable
- ⌘ Specific guidelines
 - ☒ Enforce corporate or platform style
 - ☒ Ensure consistency within and across applications
- ⌘ Guidelines may be
 - ☒ Validated by psychological research
 - ☒ Based on experience of expert designers

General UI Design Guidelines

[Baecker et al., Nielsen]



- ⌘ Know the User
- ⌘ Minimize Memorization
- ⌘ Optimize Operations
- ⌘ Engineer for Errors
- ⌘ Avoid Modes
- ⌘ Be Consistent

Minimize Memorization



- ⌘ Don't require users to memorize complex commands

 - ☑ Make commands visible through menus, icons, palettes

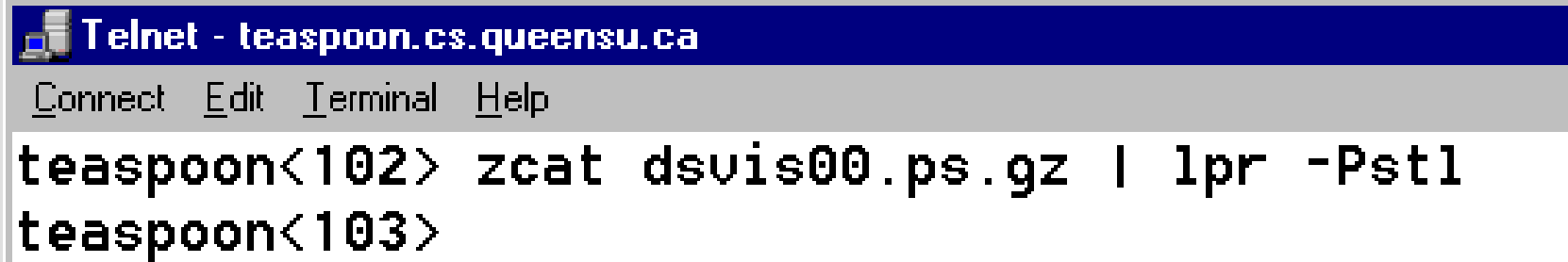
- ⌘ Don't require users to memorize information from one screen to the next

- ⌘ Short term memory holds 7 ± 2 chunks of information

[George Miller, *The Psychology of Communication*. Basic Books, 1967]

Minimize Memorization

⌘ Unix command to print a compressed file to the 'stl' printer

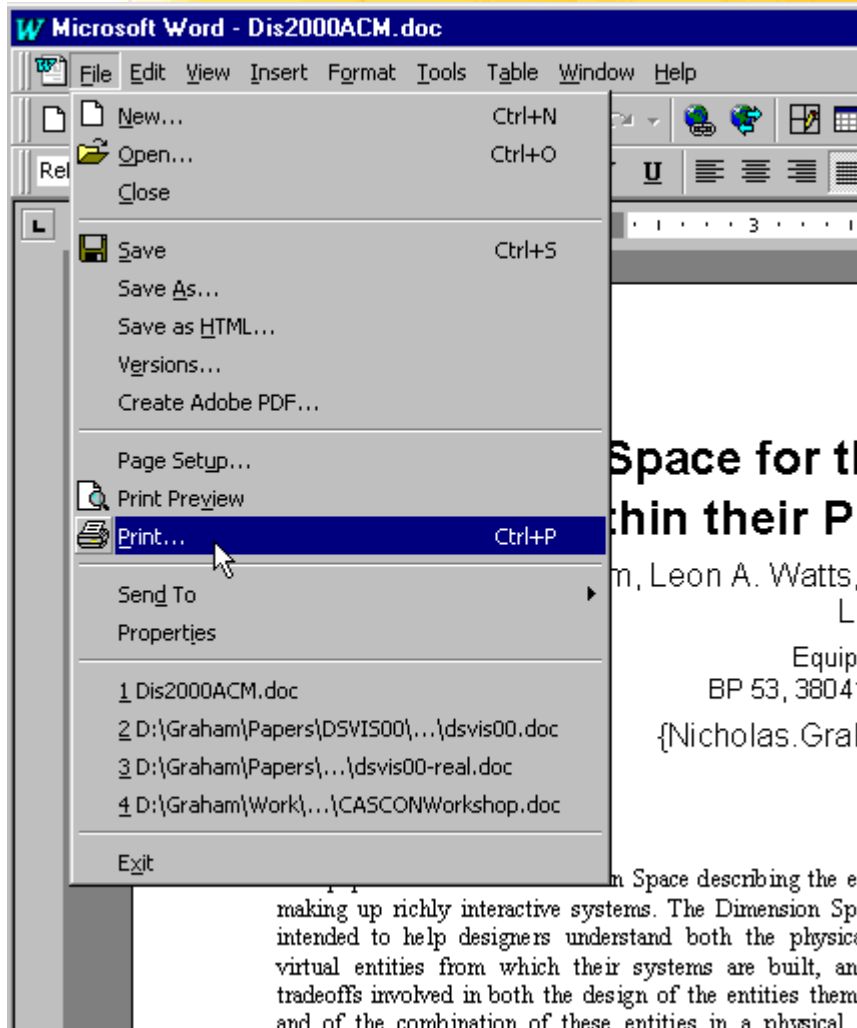
A screenshot of a Telnet terminal window. The title bar is dark blue with a small icon of a computer and the text "Telnet - teaspoon.cs.queensu.ca". Below the title bar is a light gray menu bar with the options "Connect", "Edit", "Terminal", and "Help". The main area of the terminal is white with black text. It shows a prompt "teaspoon<102>" followed by the command "zcat dsvis00.ps.gz | lpr -Pstl". The next line shows the prompt "teaspoon<103>" without any further text.

Telnet - teaspoon.cs.queensu.ca

Connect Edit Terminal Help

```
teaspoon<102> zcat dsvis00.ps.gz | lpr -Pstl
teaspoon<103>
```

Minimize Memorization



⌘ Microsoft Word
printing from menu

Minimize Memorization



- ⌘ Paying visa bill via online banking
 - ☑ Ensure user has available information required to carry out task





Back



Forward



Reload



Home



Search



Netscape



Print



Security



Shop



Stop



Bookmarks



Netsite:

ken=d7effaea72aeac8e8f1df51d1d1d1dz4AB7AAAA891AA8203AAAA8F5AA88z08A2A0A29Az00



What's Related

Account balances



To view an account balance:

Select the box to the left of the Account type, then select the View balances button.

You may select more than one account at a time.

Select	Account type	Transit #	Account #	ABM access
☒	Chequing	00542		Available
☒	VISA			Available

View balances



Convenience Banking menu



Sign off



Feedback



Help





Back



Forward



Reload



Home



Search



Netscape



Print



Security



Shop



Stop



Bookmarks



Netsite:

<https://www.pcbanking.cibc.com/retail.html?Ballnq>

What's Related

Account balances - Results



Select Help for Print instructions

Your bank account balance(s):

Account type	Transit #	Account #	Balance
Chequing	00542		\$3,992.51

Your VISA account balance(s):

Account #	Available credit	Last payment		Payment to be processed	Balance owing
		Date	Amount		
	\$4,130.25	Aug. 11, 2000	\$2,000.00	\$0.00	\$3,803.60

[Convenience Banking menu](#)[Sign off](#)[Feedback](#)[Help](#)



Back



Forward



Reload



Home



Search



Netscape



Print



Security



Shop



Stop



Bookmarks



Netsite:

https://www.pcbanking.cibc.com/retail.html?Menu



What's Related

Convenience Banking



[Account balances](#)



[Pay bills](#)



[Recent transactions](#)



[Transfer funds](#)



[Detailed account history](#)



[Session summary](#)



[Download transactions](#)



[Stop payments](#)



[Manage bank card](#)



Sign off



View Legal Agreement



Feedback



Help

CIBC PC Banking Convenience Banking Transfer funds - Netscape

File Edit View Go Communicator Help

BackForwardReloadHomeSearchNetscapePrintSecurityShopStop

BookmarksNetsite: en=d5ee7aaa52bea48e8f9d759d9d9d9dz4AB7AAAA891AA8203AAAA8F5AA88z08A2A0A29Az00What's Related

Transfer funds

Please note that for post-dated and recurring transfers, funds must be available in your account one business day before the scheduled transfer date.

From which account?

Chequing

To which account?

VISA

How much?

\$3900

When would you like this transfer to be made?

☒ Now

☐ Later - Date:

September

2000

☐ Later - Recurring how often?

select >>

Date: Beginning

September

2000

Ending

September

2000

Transfer funds

Clear all

View upcoming transfers

Convenience Banking menu

Sign off

Feedback

Help

Document: Done



Back



Forward



Reload



Home



Search



Netscape



Print



Security



Shop



Stop



Bookmarks



Netsite:

:&balance=\$92.51&reference=4337&date=Sep_26_2000&time=05:41_PM&createdby=1-2&standin=0



What's Related

Transfer funds - Results



Select Help for Print instructions

You have successfully transferred the following funds on Sep. 26, 2000 at 05:41 PM Eastern Standard Time.

From: Chequing

To: VISA

Amount transferred: \$3,900.00

Account balance: Chequing \$92.51

Reference number: 4337

Please record the reference number.

[Return to Transfer funds](#)

[Convenience Banking menu](#)[Sign off](#)[Feedback](#)[Help](#)

Optimize Operations



⌘ Frequently done operations should be simple, quick, error-free

- ☑ Interaction techniques: accelerators, macros

- ☑ Dialogues: streamline for frequent operations

- ☑ UI can adapt for expert users

 - ☒ Hard to design such adaptation so that it's not confusing

Optimize Operations

⌘ Vi text editor

- ⏏ Command to change all old Queen's phone numbers into new phone numbers with area code

- ⏏ E.g., 545-6526 -> (613) 533-6526

- ⏏ Make sure we only do old Queen's phone numbers

 - ⏏ 545-dddd

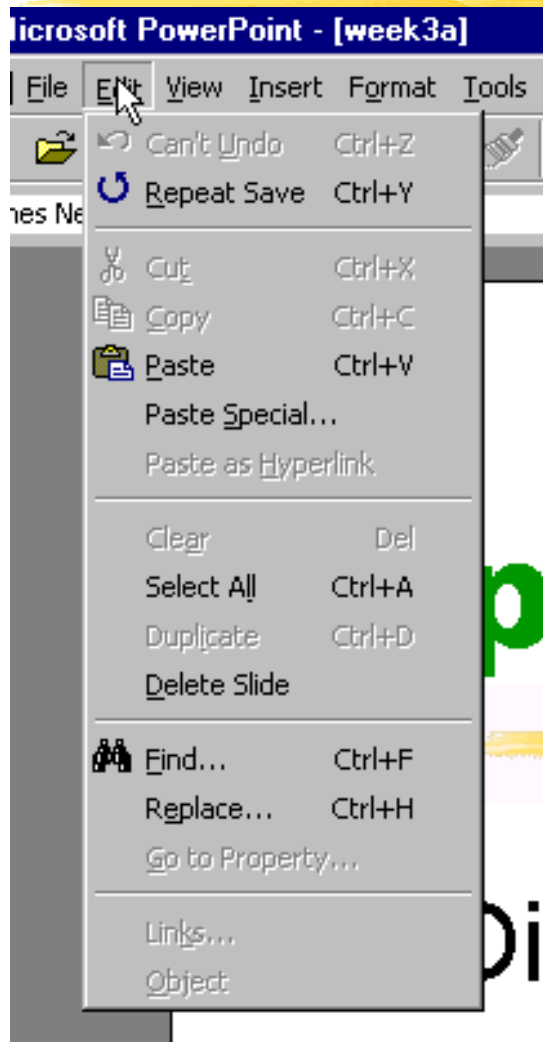
- ⏏ `g/545-\ ([0-9] [0-9] [0-9] [0-9]) /s// (613) 533-\1/g`

Optimize Operations



- ⌘ Dialogue for paying Visa could be optimized
- ⌘ How would you do this?

Optimize Operations



- ⌘ *Keyboard accelerators* allow frequent operations to be performed quickly
- ⌘ Showing them on the menu minimizes memorization

Engineer for Errors



- ⌘ We tend to assume that correct usage is the norm
- ⌘ Should treat erroneous use as the norm
 - ☑ People exploring, learning make mistakes
- ⌘ Application should be designed so that mistakes are not costly
 - ☑ Decreases insecurity
 - ☑ Encourages exploration of system
- ⌘ Errors may be one-time or recurring

Engineer for Errors



⌘ Unix command

```
⌘ rm *.class
```

⌘ removes all files ending with “.class” from directory

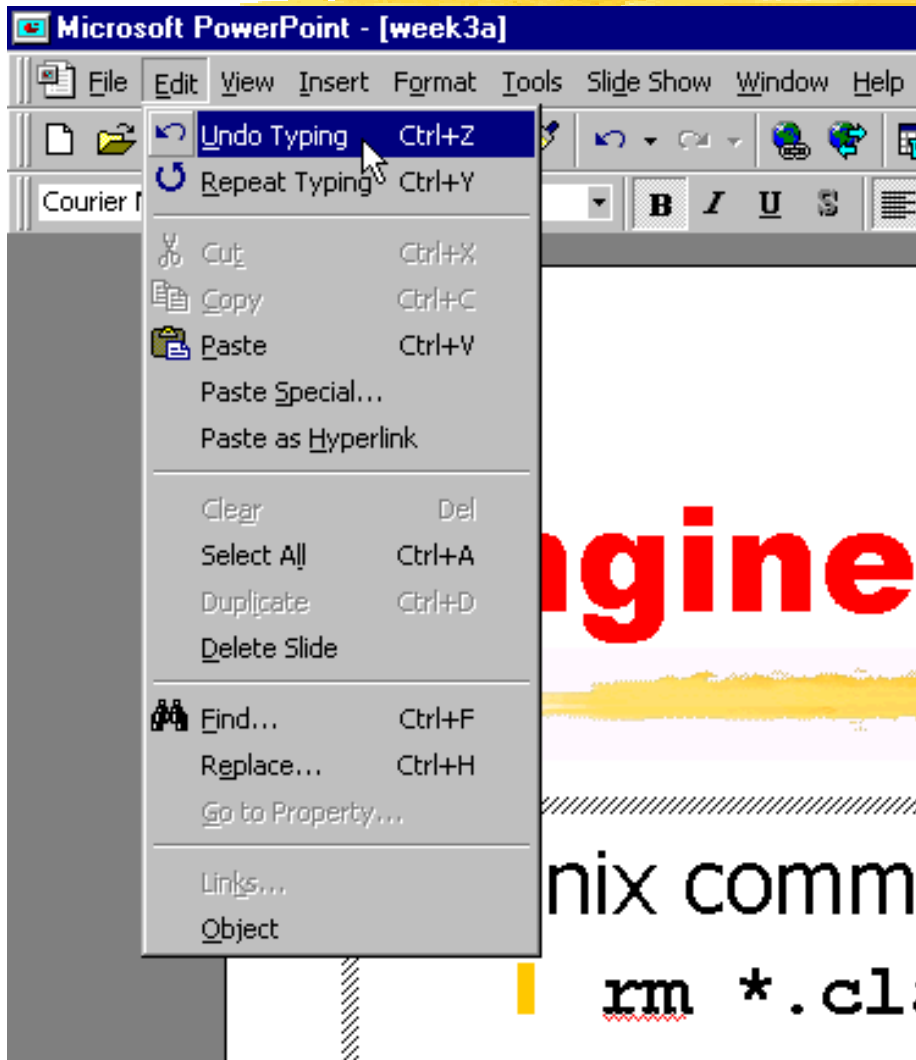
⌘ However, Unix command

```
⌘ rm * .class
```

⌘ removes *a//* files from directory, and reports message

```
⌘ .class: not found
```

Engineer for Errors



⌘ **Undo** operation supports exploration, allows errors to be undone

⌘ Two variants -- assume UI starts in state s_1 , and user performs actions $\langle a_1 \rangle$ then $\langle a_2 \rangle$, then selects undo twice

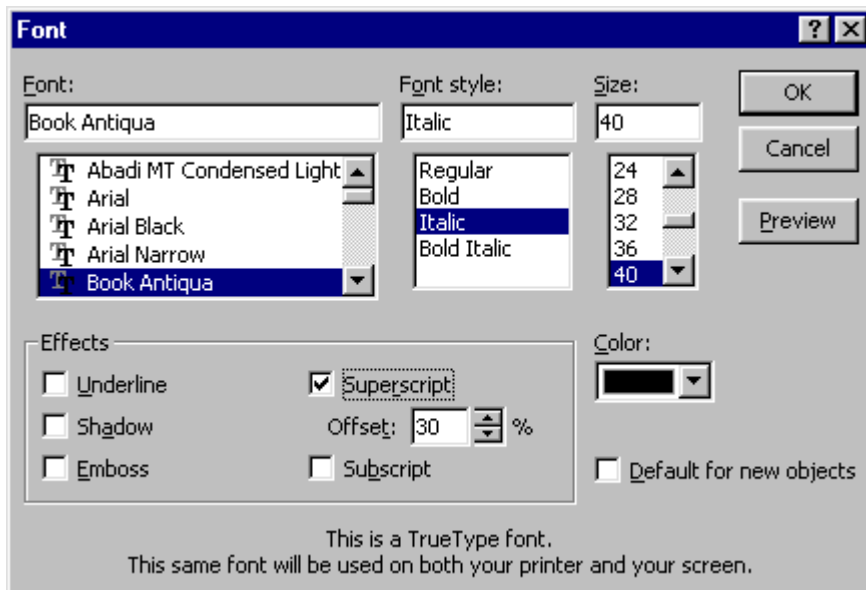
⏏ Undo-undo: second undo undoes first undo

⊗ $s_1 \langle a_1 \rangle s_2 \langle a_2 \rangle s_3$
 $\langle \text{undo} \rangle s_2 \langle \text{undo} \rangle s_3$

⏏ Undo-redo: continues undoing through history

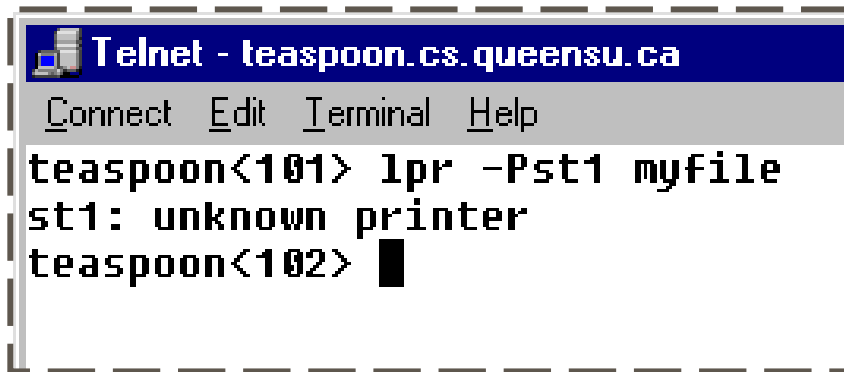
⊗ $s_1 \langle a_1 \rangle s_2 \langle a_2 \rangle s_3$
 $\langle \text{undo} \rangle s_2 \langle \text{undo} \rangle s_1$

Engineer for Errors



⌘ "Cancel" operation in dialogue returns state of system to state before dialogue started

Engineer for Errors



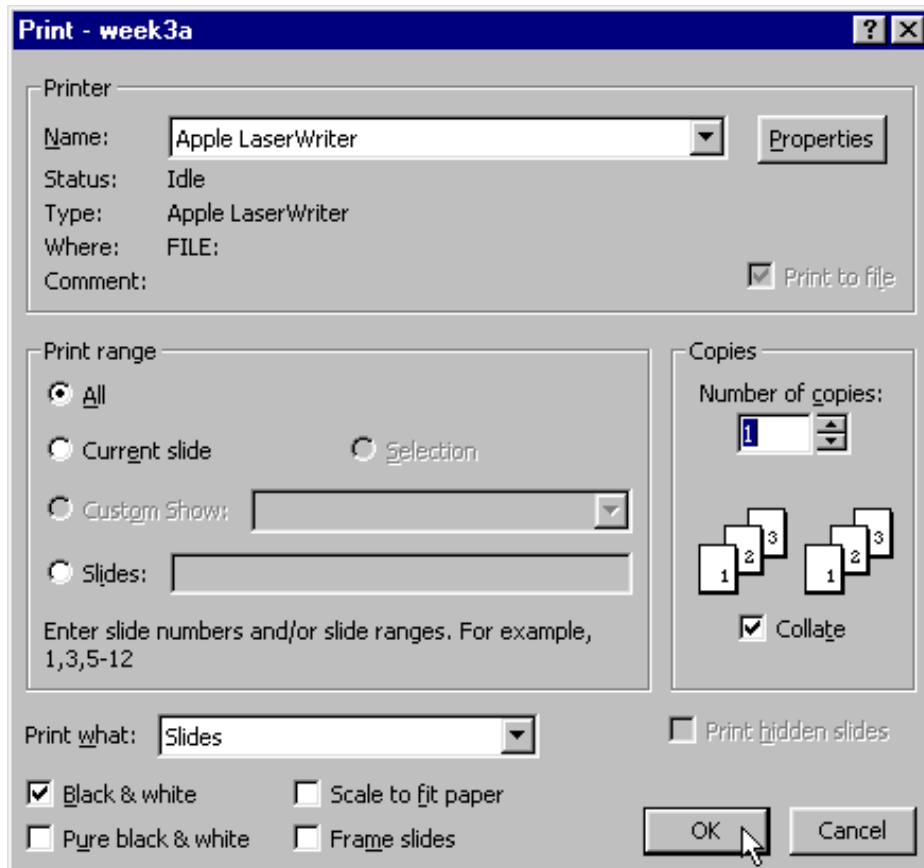
A screenshot of a Telnet terminal window. The title bar reads "Telnet - teaspoon.cs.queensu.ca". Below the title bar is a menu bar with "Connect", "Edit", "Terminal", and "Help". The terminal text shows a user prompt "teaspoon<101>" followed by the command "lpr -Pst1 myfile". The response is "st1: unknown printer". The user then enters "teaspoon<102>" followed by a cursor.

```
Telnet - teaspoon.cs.queensu.ca
Connect Edit Terminal Help
teaspoon<101> lpr -Pst1 myfile
st1: unknown printer
teaspoon<102> █
```

⌘ "lpr" command fails if:

- ⏏ "lpr" misspelled
- ⏏ name of printer unknown or misspelled
- ⏏ name of file misspelled

Engineer for Errors



⌘ MS Windows print dialogue box

☑ Names of printers chosen from menu, therefore no error possible

⌘ Errors still possible

☑ Select page range is free form

☑ If you don't want to print black and white, easy to miss check box that has to be reset

Avoid Modes



- ⌘ Modes cause an interpretation on input depending on context
- ⌘ Danger is
 - ☒ user won't understand what context they are in
 - ☒ user will forget what context they are in
 - ☒ imposes cognitive burden of thinking about "what does 'k' key mean now?"
- ⌘ Modes can be acceptable if they are visible and intuitive

Avoid Modes

⌘ In *vi* editor, there are

✗ *Text mode*: typing enters text into document

✗ e.g., typing "xxx" enters "xxx" at current insertion point

✗ *Command mode*: characters interpreted as commands

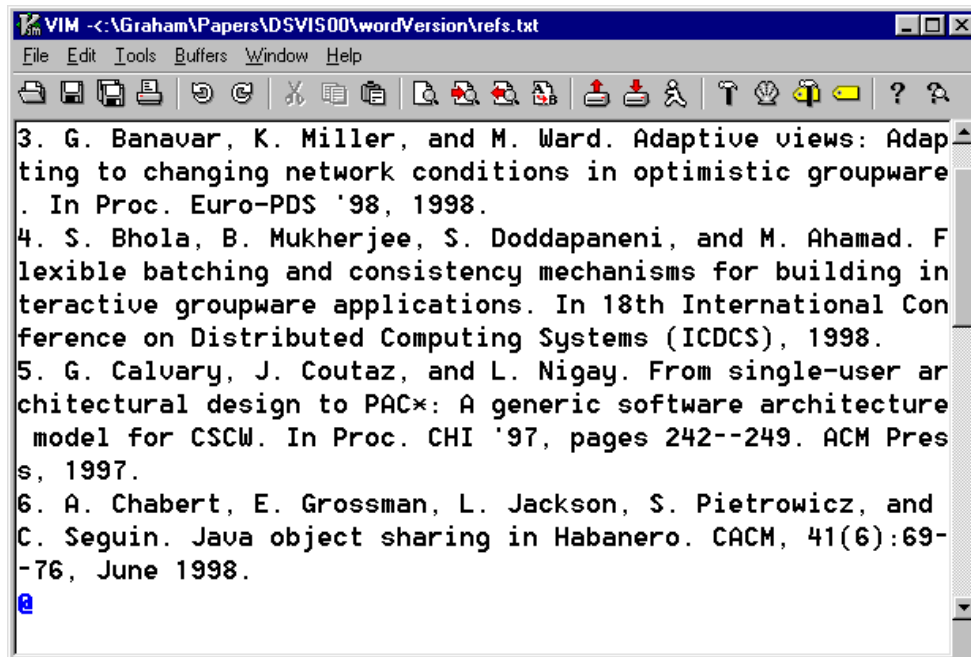
✗ e.g., typing "xxx" deletes three characters forward from insertion point

✗ Default is command mode

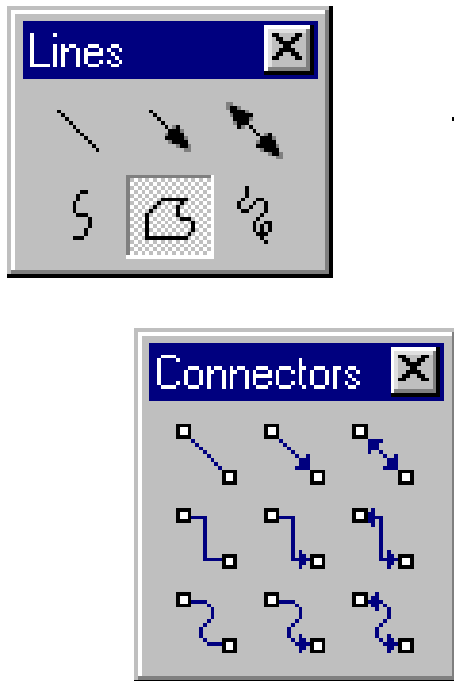
✗ Commands such as "i" (insert), "o" (add line) enter text mode

✗ <Esc> key enters command mode

✗ No way of seeing what mode you are in



Avoid Modes



⌘ Modes can be made visible

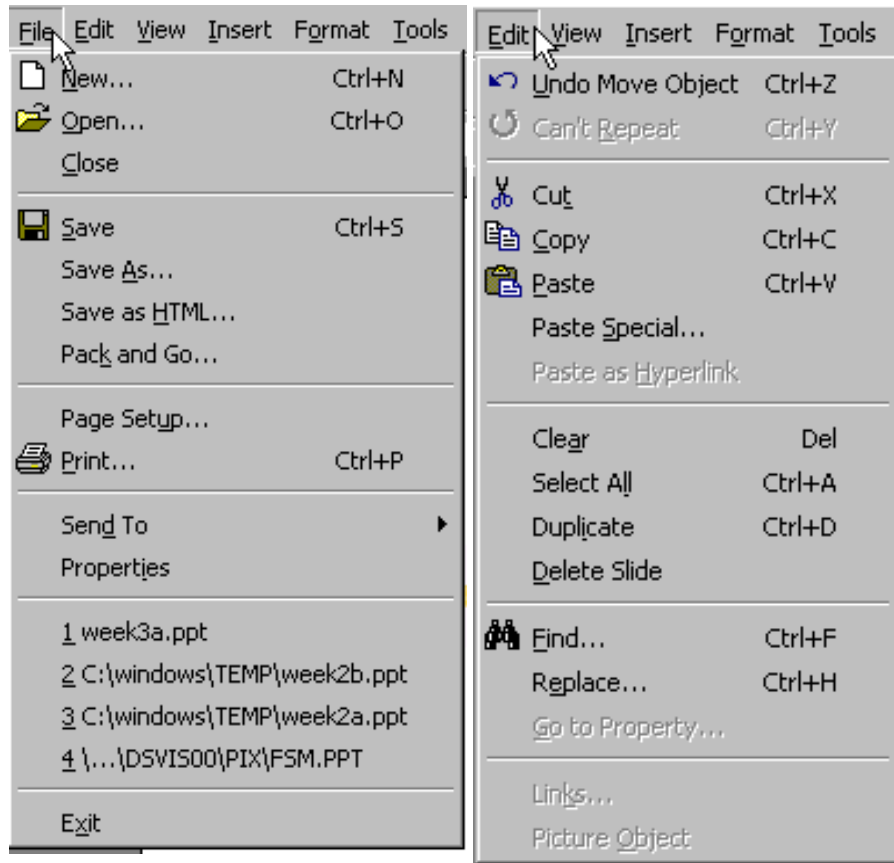
- ☑ E.g., selecting from palette makes it obvious what modes available
- ☑ Selection highlighted
- ☑ Current mode shown in tracking symbol ("+")

Be Consistent



- ⌘ Consistency allows users to reuse knowledge across and within applications
- ⌘ Failure to be consistent can lead to errors, frustration

Be Consistent



⌘ Follow conventions for menu layout

- ☑ "File" has "New", "Open", "Close", "Save", "Save As"
- ☑ "Edit" has "Undo", "Redo", "Cut", "Copy", "Paste"

⌘ Follow conventions for keyboard accelerators

- ☑ ^P - print
- ☑ ^C - copy
- ☑ ^X - cut
- ☑ ^V - paste
- ☑ ^A - select all
- ☑ ^F - find

Be Consistent



- ⌘ Use consistent date format
- ⌘ Use consistent screen layout
- ⌘ Use same navigation commands from screen to screen
- ⌘ Use consistent colour (if colour matters)
- ⌘ Use consistent dialogues
 - ☑ E.g., dialogues for printing, saving a file should always be the same

General UI Design Guidelines

[Baecker et al., Nielsen]



- ⌘ Know the User
- ⌘ Minimize Memorization
- ⌘ Optimize Operations
- ⌘ Engineer for Errors
- ⌘ Avoid Modes
- ⌘ Be Consistent

Detailed Design Guidelines

[Helander, ed., Handbook of HCI]



- ⌘ Use of Text
- ⌘ Placing and Sequencing of Information
- ⌘ Presenting Text
- ⌘ Graphics

Use of Text



⌘ How much information?

- ☒ 25%-60% of display should be used

⌘ Reducing information

- ☒ Make appropriate use of abbreviations
- ☒ Avoid unnecessary detail
- ☒ Use concise wording
- ☒ Use familiar data formats
- ☒ Use tabular formats with headings

Use of Text



⌘ Grouping information

- ⌘ Highlighting relationships among data

- ⌘ Can use

 - ⌘ Format

 - ⌘ Colours

 - ⌘ Outlining

 - ⌘ Highlighting

 - ⌘ Font

Use of Text



Cisc 124 Grades

John Smith received a grade of 47

Jen Wu received a grade of 92

Mike Jenkins received a grade of 86

Nick Graham received a grade of 63

⌘ Hard to extract information from this format

☒ What grade did a particular person get?

☒ What is highest/lowest grade?

☒ How many people failed?

Use of Text



Cisc 124 Grades	
<u>John Smith</u>	47
<u>Jen Wu</u>	92
<u>Mike Jenkins</u>	86
<u>Nick Graham</u>	63

Use of Text

Cisc 124 Grades	
<u>John Smith</u>	47
<u>Jen Wu</u>	92
<u>Mike Jenkins</u>	86
<u>Nick Graham</u>	63

- ⌘ Tabular format eases reading
- ⌘ Removed Unnecessary Detail
- ⌘ Colour and font emphasize vertical relationship
 - ☒ Group names together and group grades together
- ⌘ Multiple fonts, text colour, underlining are ugly and don't add any information
- ⌘ Insufficient contrast between colours in title

Use of Text



Cisc 124 Grades

Nick Graham	63
-------------	----

Mike Jenkins	86
--------------	----

John Smith	47
------------	----

Jen Wu	92
--------	----

Use of Text



Cisc 124 Grades

Nick Graham	63
Mike Jenkins	86
John Smith	47
Jen Wu	92

- ⌘ Colour now used to group name and grade together
- ⌘ Alphabetical ordering aids looking up names

Presenting Text



⌘ Case

- ☒ Use conventional upper/lower

⌘ Spacing

- ☒ With **constant-width fonts**, use ragged-right
- ☒ With proportional-width fonts, use justified

⌘ Spacing between lines

- ☒ letter height : total space between lines
 - ☒ 1:2 faster than 1:4
 - ☒ 1:2.7 faster than 1:1.3
 - ☒ ...implies 1:2 to 1:2.7 optimal

Presenting Text



⌘ Line length

- ☒ 40 - 60 characters optimal
- ☒ Inter-line spacing : Line length = 1:30

⌘ Font

- ☒ Just because you have 1,000 fonts doesn't mean you have to use them all
- ☒ Sans-serif easier to read on displays
- ☒ Serif more traditional for longer texts

Graphics



⌘ A picture is worth 1,000 words

⌘ Represent

- ☒ Real world images

- ☒ Complex real-world systems

- ☒ Numerical data

 - ☒ Graphical data to represent overall trend; textual for real analysis

- ☒ Direct manipulation objects

Design of Interactive Systems



- ⌘ Design is a creative process
- ⌘ Goal is to design a system that is
 - ⌘ Appropriate to users in their context
 - ⌘ *cf* User Needs Analysis
 - ⌘ Allows users to perform their tasks
 - ⌘ *cf* Task Model
- ⌘ Balance usability properties
 - ⌘ Ease of learning, ease of use

Conceptual/Mental Models

[Norman, Design of Everyday Things]



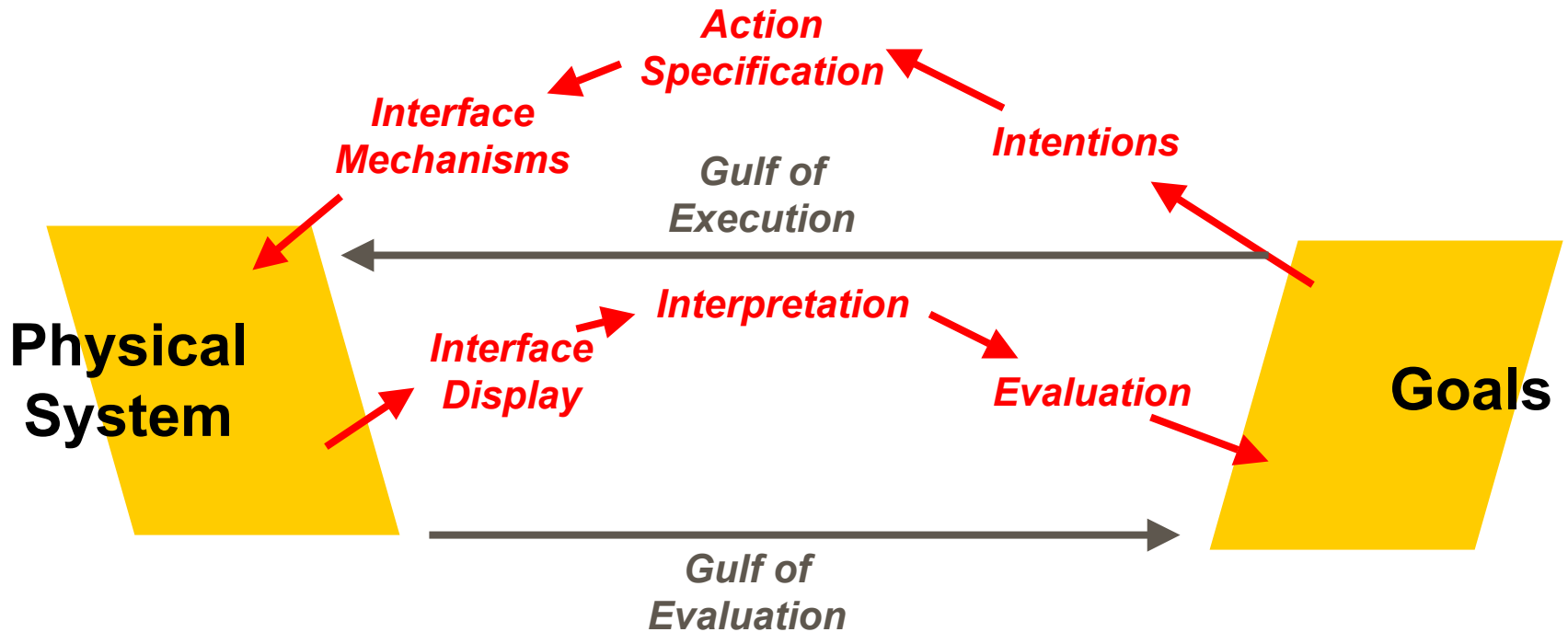
- ⌘ Users interacting with systems wish to achieve some goal
- ⌘ Must find ways of achieving this goal via the mechanisms of the system
- ⌘ Must understand system in order to
 - ⏏ Go from goal to sequence of actions to achieve goal
 - ⏏ Determine whether goal has been achieved by examining system state

Norman's Gulfs of Execution/Evaluation



- ⌘ Example: pouring water into a bathtub
- ⌘ Goal: have water flow at desired rate, temperature
- ⌘ Goal expressed in terms of *psychological variables*
 - ☒ Temperature of water
 - ☒ Flow rate of water
- ⌘ Mechanisms available are *physical variables*
 - ☒ Hot tap
 - ☒ Cold tap
- ⌘ User must work out how to adjust hot and cold tap to achieve desired values of psychological variables, and must touch/see/hear water to determine whether these values achieved

Norman's Gulfs of Execution/Evaluation

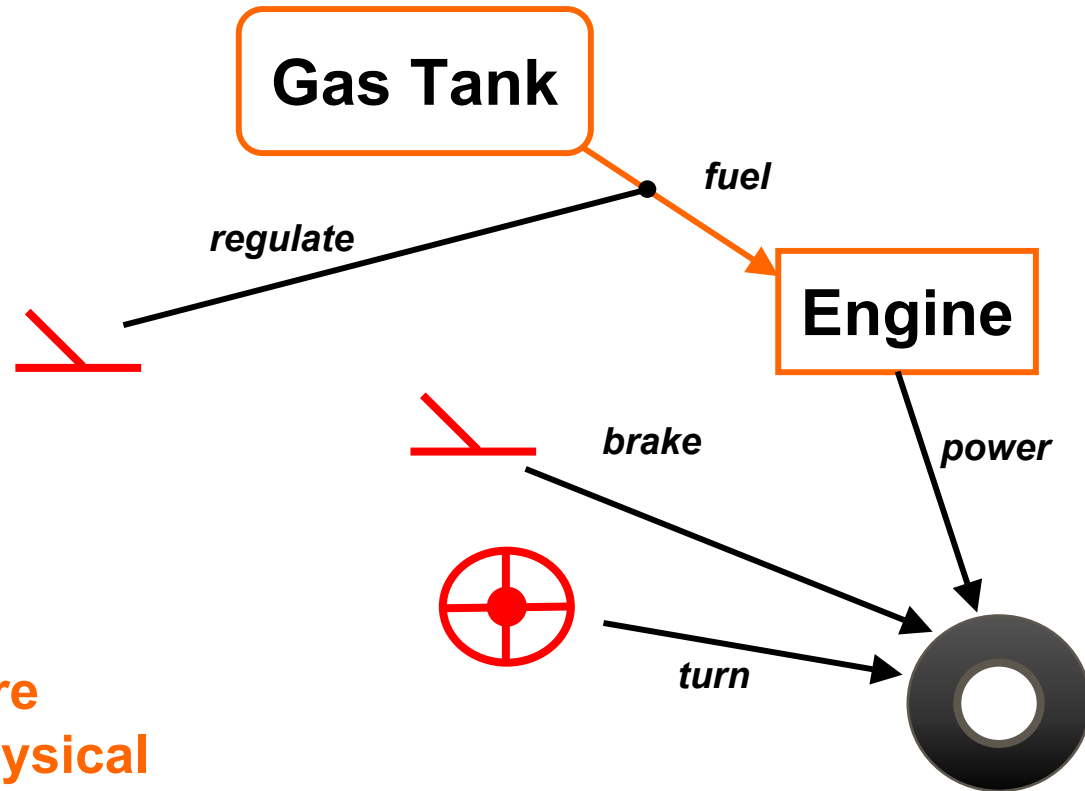


Conceptual Models



- ⌘ In order to accomplish goals with a system, the user needs to understand:
 - ☒ What actions on the system are possible
 - ☒ What the effects of a particular action will be
 - ☒ How to view the state of the system to evaluate the results of actions
- ⌘ Therefore users need a model of how the systems work
 - ☒ Conceptual models allow full, complete, correct understanding of system
 - ☒ Used by engineers, teachers

Conceptual Model of Car



Question: What are psychological, physical variables involved in driving a car?

Conceptual Model of Car

⌘ Helps answer questions like:

- ☒ What are effects of controls (accelerator, brake, steering wheel) on speed, direction of car?
- ☒ What is effect of depressing accelerator and brake at same time?
- ☒ What happens if car runs out of gas?

⌘ Explains what we need to know to *operate* a car, not to build one

☒ Therefore not an implementation model

- ☒ "The 'steering wheel' turns the 'steering column', which acts on a gear in the 'steering rack' (for newer cars with rack and pinion steering) which connects to a steering arm on each side of the car to each 'wheel hub'. Older cars (like my Alfa and the RX-7) have a 'rotating ball' assembly that moves a steering arm back and forth, which pulls one wheel while pushing the other side out" [Dave Phillips, Car Expert]

Mental Models



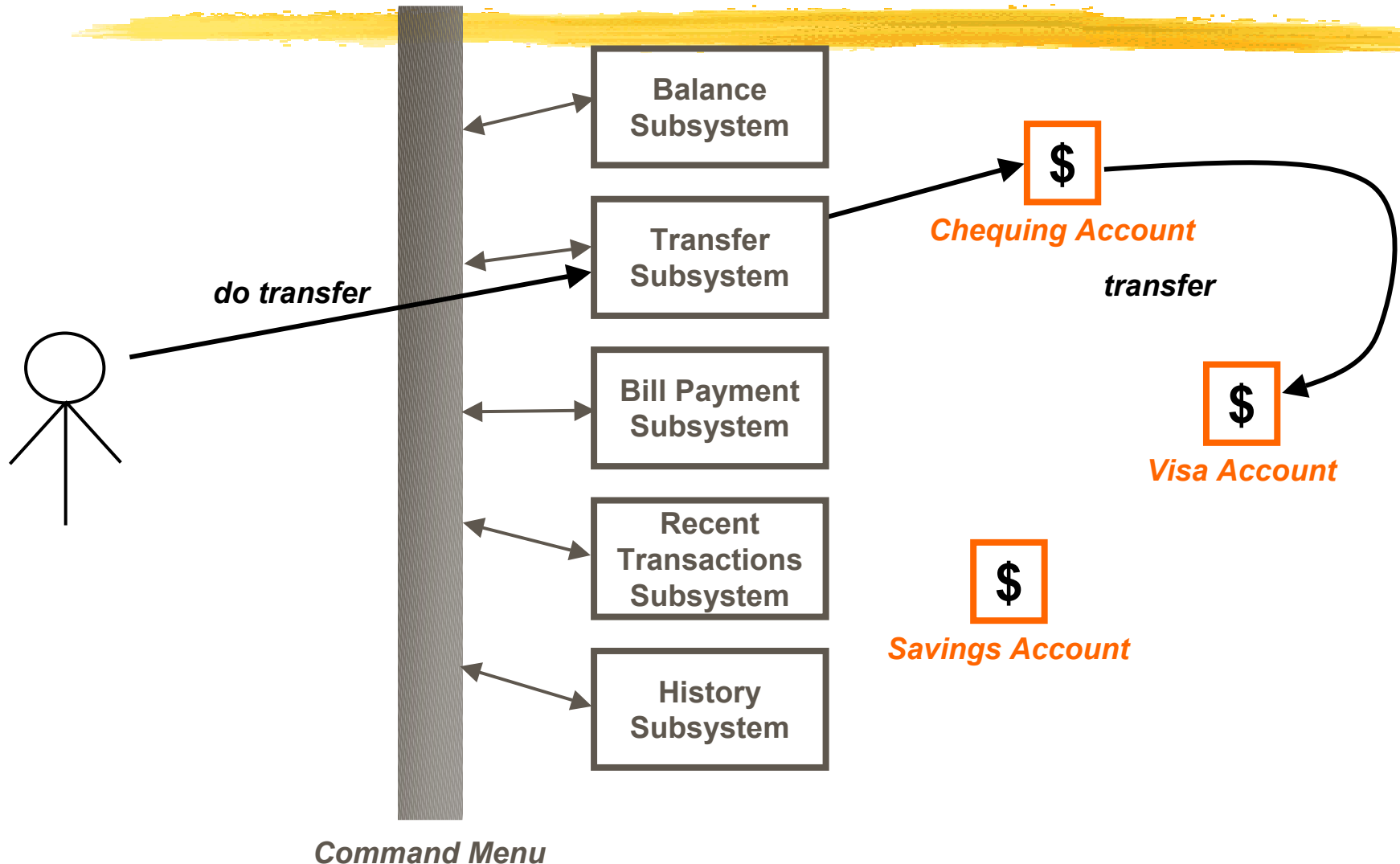
- ⌘ In learning about a system, users are trying to develop an understanding of its conceptual model
 - ☒ Through training, books, experimenting with system
 - ☒ User's model called *mental model*
- ⌘ If conceptual model not visible to users, they will have to guess
- ⌘ Incorrect mental models lead to difficulties in establishing task-action mapping

Mental Models



- ⌘ How user understands user interface
- ⌘ Predictive, explanatory
- ⌘ Often incorrect, incomplete
- ⌘ Parsimonious -- kept small enough to understand what features the user needs/uses
- ⌘ Superstition -- if something doesn't work, easier to develop superstition than to extend mental model
- ⌘ People are loathe to extend their mental models

Conceptual Model of Banking System



Developing Mental Model



⌘ Navigation structure very clear

☒ Main menu makes this part of conceptual model visible

⌘ Familiar concept of *account* revealed in conceptual model

⌘ Need to experiment with system to discover that visa is an account not a bill

☒ Therefore, pay visa via transfer, not bill payment

⌘ Distinction between “history” and “recent transactions” unclear

☒ From experimentation, history gives more transactions

☒ Recent transactions may be updated faster after a transaction is performed (but this may be just superstition)