

# Interacção Humana com o Computador

## Aula 6



Departamento de Informática  
UBI 2024/2025

João Cordeiro  
[jpcc@ubi.pt](mailto:jpcc@ubi.pt)



# HUMAN-COMPUTER INTERACTION

THIRD  
EDITION

DIX  
FINLAY  
ABOWD  
BEALE

## Chapter 5

# Interaction Design Basics



# Interaction Design Basics

---

- **Design:**
  - What is it? Interventions, goals, constraints;
- **The design process**
  - What happens when
- **Users**
  - Who they are, what they are like ...
- **Scenarios**
  - Rich stories of design
- **Navigation**
  - Finding your way around a system
- **Iteration and prototypes**
  - Never get it right first time!



# Interactions and Interventions

---

## Design interactions not just interfaces

not just the immediate interaction

e.g. stapler in office - technology changes interaction style

- **Manual:** write, print, staple, write, print, staple. ...
- **Electric:** write, print, write, print, ..., staple



## Designing interventions not just artifacts

not just the system, but also ...

- documentation, manuals, tutorials
- what we say and do as well as what we make



# Interaction Design Basics

---

## What is design?



# What is design?

## Achieving goals within constraints

- **Goals** - purpose
  - Who is it for?
  - Why do they want it?
- **Constraints**
  - Materials, platforms, time, and energy
- **Trade-offs**
  - Good vs. optimum!





# Golden rule of design

---

## Understand your Materials!





# For Human-Computer Interaction

---

## Understand your materials

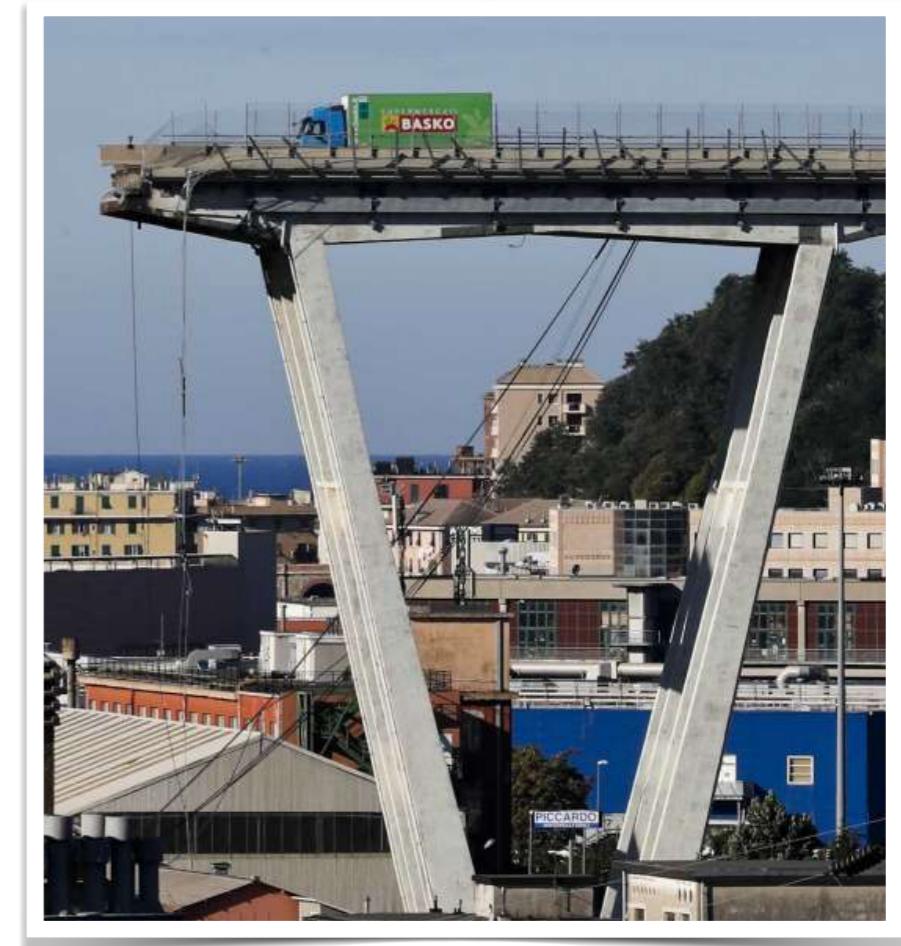
- Understand computers
  - Limitations, capacities, tools, platforms
- Understand people
  - Psychological, social aspects
  - Learn about human error
- and their interaction ...





# To err is human

- Accident reports ..
  - air-crash, industrial accident, hospital mistake
  - enquiry ... **blames** ... ‘**human error**’
- But ...
  - Concrete lintel breaks because too much weight
  - Blame ‘lintel error’ ???  
... No!!! Design error!!!  
We know how concrete behaves under stress
- Human ‘error’ is normal!
  - We know how users behave under stress, so design for it!
- Treat the user at least as well as physical materials!





# To err is human

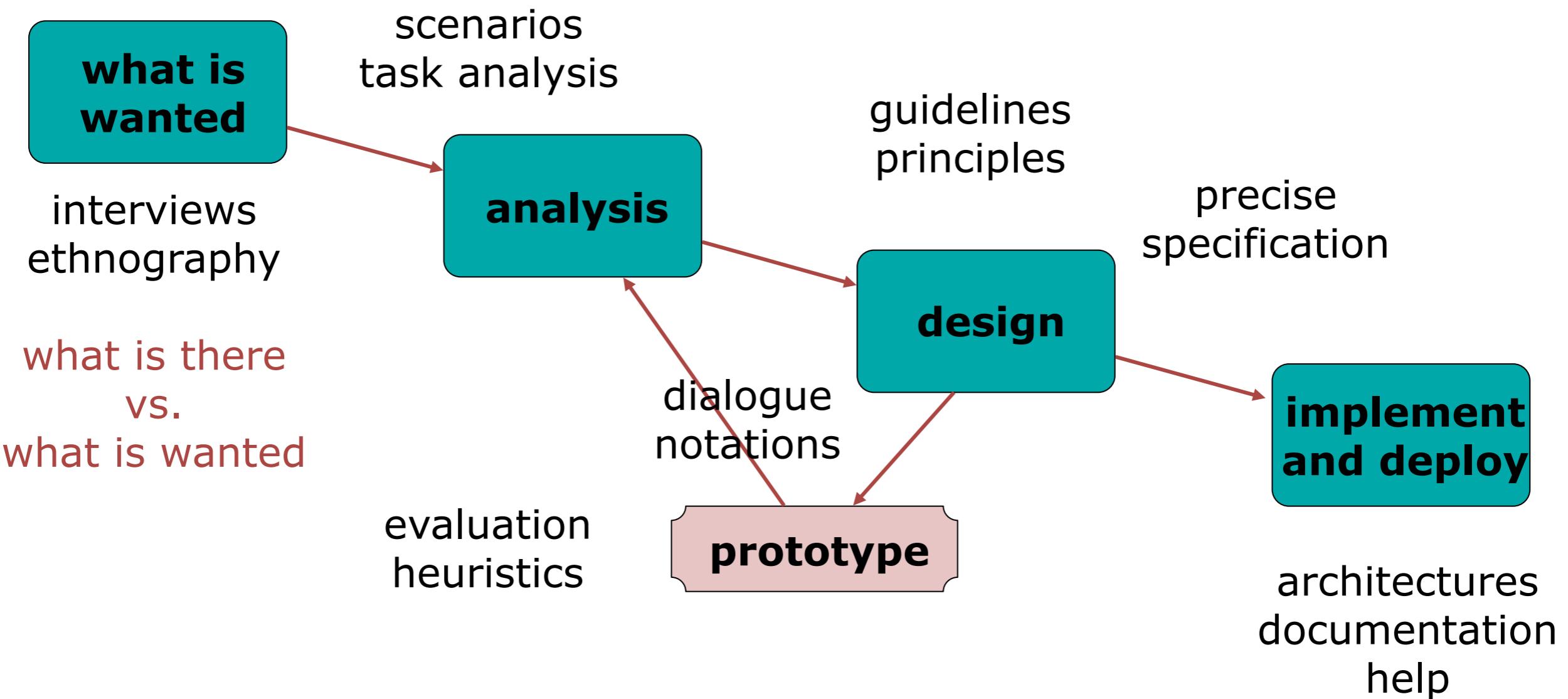
---

# the user





# The Process of Design





# Software Engineering

---



Como o cliente explicou



Como o lider de projeto entendeu



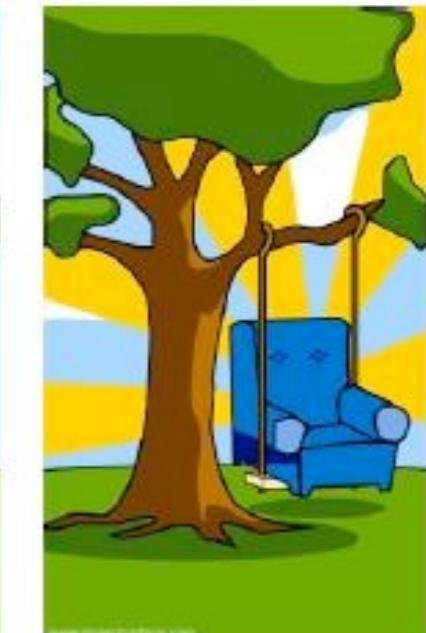
Como o analista planejou



Como o programador codificou



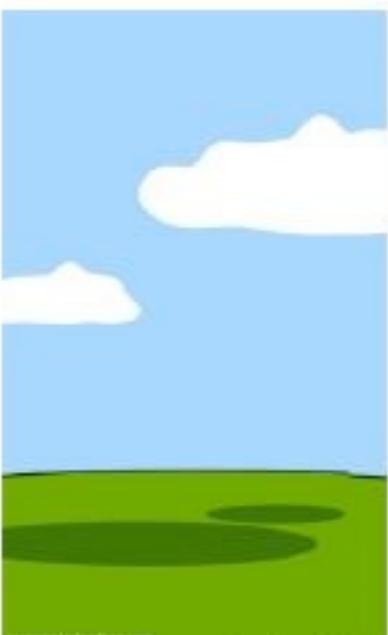
O que os beta testers receberam



Como o consultor de negocios descreveu



Valor que o cliente pagou



Como o projeto foi documentado



O que a assistencia tecnica instalou



Como foi suportado



Quando foi entregue



O que o cliente realmente necessitava



# Steps ...

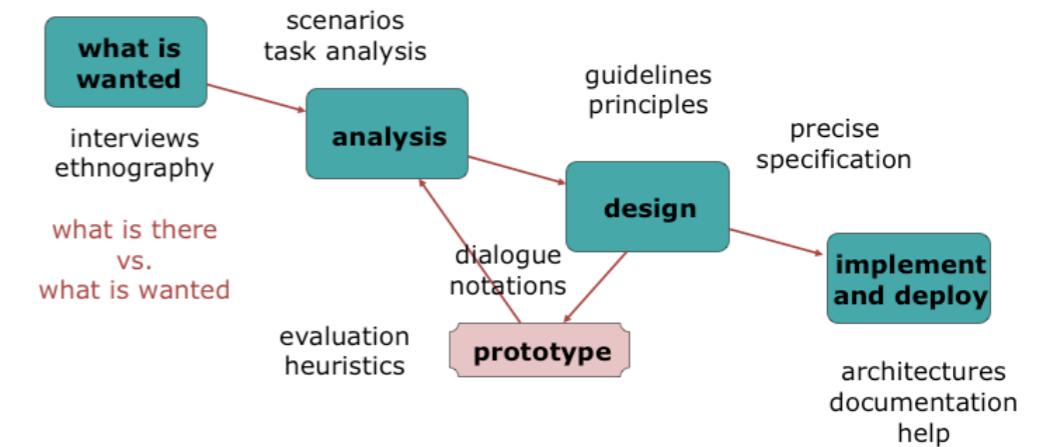
- Requirements
  - What is there (user context) and what is wanted ...

- Analysis
  - Ordering and understanding

- Design
  - What to do and how to decide; guidelines;
  - Heuristics <= cognitive models.

- Iteration and prototyping
  - Getting it right ... and finding what is really needed!

- Implementation and deployment
  - Making it and getting it out there.





# ... but how can I do it all ! !

- Limited time ⇒ design trade-off
- Usability?
  - Finding problems and fixing them?  
this is easy! X
  - Deciding what to fix? ✓
- Attention: "*A perfect system is badly designed*"
  - too good ⇒ too much effort in design





# Know your user

## A Commandment!

- Who are they?

- more than one
- artifacts used.



- Probably not like you!

- old ≠ new. (e.g. triangle exercise)



- Watch them

- what they doing it? (introspection difficulty!)

- Talk to them

- why they do!



- Use your imagination

- *participation design.*



# Cultural Probes

---

- **Direct observation**
  - Sometimes hard
    - in the home
    - psychiatric patients, ...
- **Probe packs**
  - Items to prompt responses.
    - e.g., glass to listen at wall, camera, postcard
  - Given to people to open in their own environment they record what is meaningful *to them*.
- **Used to ...**
  - Inform/guide interviews, prompt ideas, en-culture designers.





# Persona

- **Description of a user type**
  - Not necessarily a real person
- **Use as a surrogate user**
  - Example:



What would Betty think when ...

- **Details matter**
  - Makes her ‘real’





# Persona - an example:

---



Betty is 37 years old. She has been Warehouse Manager for five years and worked for Simpkins Brothers Engineering for twelve years. She didn't go to university, but has studied in her evenings for a business diploma. She has two children aged 15 and 7 and does not like to work late. She did part of an introductory in-house computer course some years ago, but it was interrupted when she was promoted and could no longer afford to take the time. Her vision is perfect, but her right-hand movement is slightly restricted following an industrial accident 3 years ago. She is enthusiastic about her work and is happy to delegate responsibility and take suggestions from her staff. However, she does feel threatened by the introduction of yet another new computer system (the third in her time at SBE).





# Persona - an example:

---



Betty is 37 years old, She has been Warehouse Manager for five years and worked for Simpkins Brothers Engineering for twelve years. She didn't go to university, but has studied in her evenings for a business diploma. She has two children aged 15 and 7 and does not like to work late. She did part of an introductory in-house computer course some years ago, but it was interrupted when she was promoted and could no longer afford to take the time. Her vision is perfect, but her right-hand movement is slightly restricted following an industrial accident 3 years ago. She is enthusiastic about her work and is happy to delegate responsibility and take suggestions from her staff. However, she does feel threatened by the introduction of yet another new computer system (the third in her time at SBE).





# Interaction Design

---

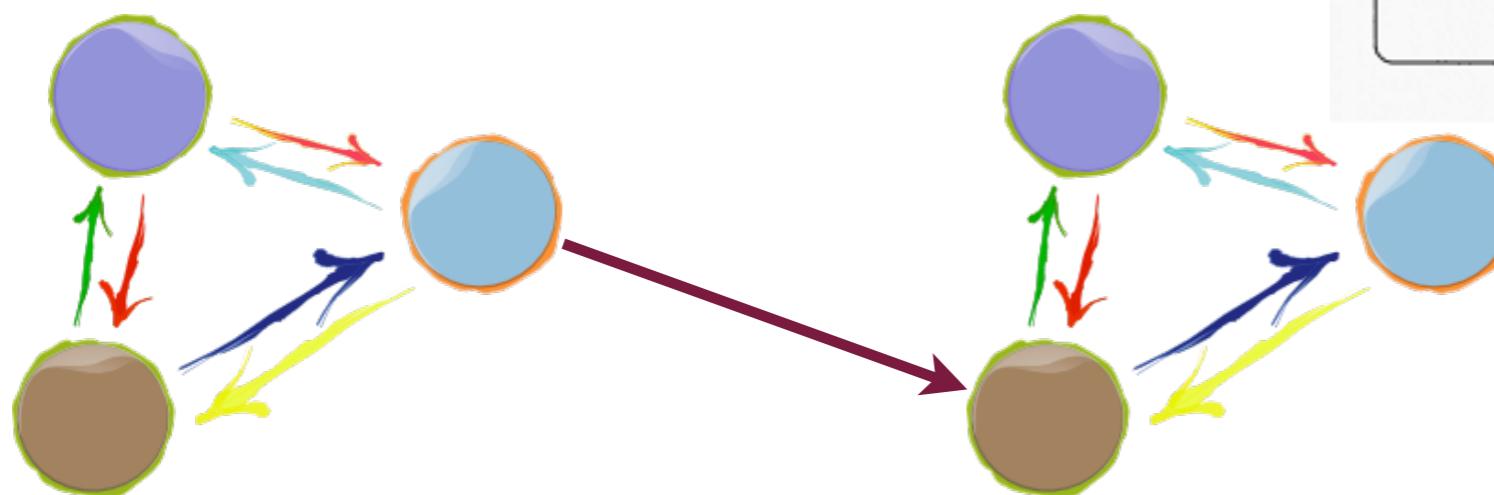
## Scenarios

stories for design  
use and reuse



# Scenarios

- Stories for design
  - communicate with others
  - validate other models
  - understand dynamics
- Linearity
  - time is linear - our lives are linear; but interaction might not be!



**STORYBOARD**

NAME: A PORTRAIT OF ENVY PAGE: 4 OF     
BY: E. GARCIA FOR: Some DATE: 7-3-07

	ACTION: (CONT'D) <u>Jerry approaches Gillian's desk</u> <u>Gillian is typing</u> AUDIO: <u>keys typed on</u> COMMENTS: _____
	ACTION: (over monitor) <u>Gillian looks up at Jerry (takes mug)</u> AUDIO: <u>"Thanks!"</u> COMMENTS: <u>from above</u>
	ACTION: <u>J: "I figured you were busy."</u> <u>OTS Gillian @ Jerry</u> AUDIO: <u>D</u> COMMENTS: <u>from below centered</u>



# Scenarios ...

---

- **What will users want to do?**
  - how would they react to this ...
- **Step-by-step walkthrough**
  - what can they **see** (sketches, screen shots)
  - what do they **do** (keyboard, mouse etc.)
  - what are they **thinking**?
- **Use and reuse throughout design**



# Also play and act ...

---

- Mock up device
- Pretend you are doing it
- Internet-connected Swiss Army Knife ...



use toothpick as stylus 😊



# Also play act ...

---

- Mock up device
- Pretend you are doing it
- Internet-connected Swiss Army Knife ...

but where is that thumb?



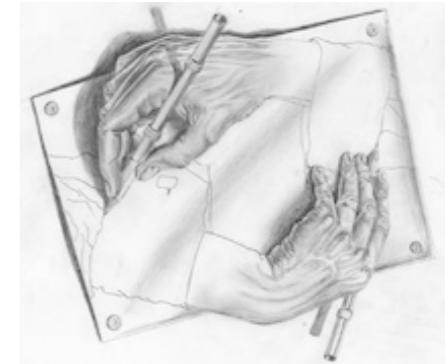
Maybe a voice interface?



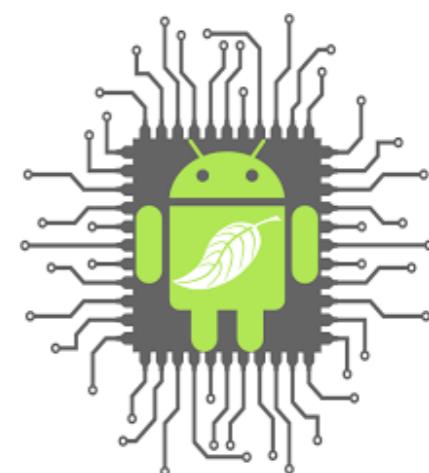
# With scenarios, explore the depths

---

- **Explore interaction**
  - what happens when



- **Explore cognition**
  - what are the users thinking
- **Explore architecture**
  - what is happening inside





# Use scenarios to ...

---

- **Communicate with others**
  - designers, clients, users
- **Validate other models**
  - ‘play’ it against other models
- **Express dynamics**
  - **screenshots** - appearance (static view)
  - **scenario** - behaviour (active understanding)



# Linearity vs Complexity of life

**Scenarios - one linear path through the system**

## Pros:

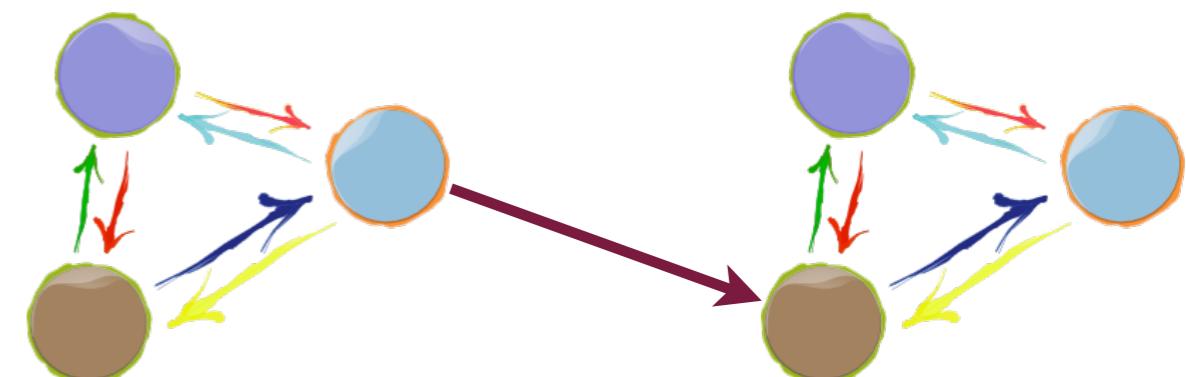
- Life and time are linear
- Easy to understand (stories and narrative are natural)
- Concrete (errors less likely)

## Cons:

- No choice, no branches, no special conditions
- Miss the unintended

- So:

- Use several scenarios
- Use other methods





---

# **Navigation Design**

**local structure - single screen**

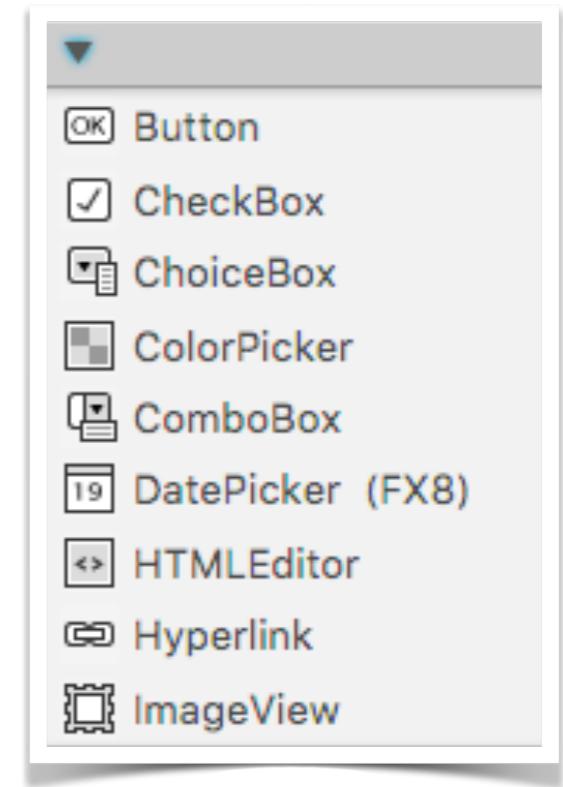
**global structure - whole site**



# Four Levels of Design

---

- **Widget choice**
  - menus, buttons, labels etc.
- **Screen design**
  - find things, grouping.
- **Application navigation design**
  - what will happen ...
- **Environment**
  - other apps, OS





# Think about structure

- **Local**

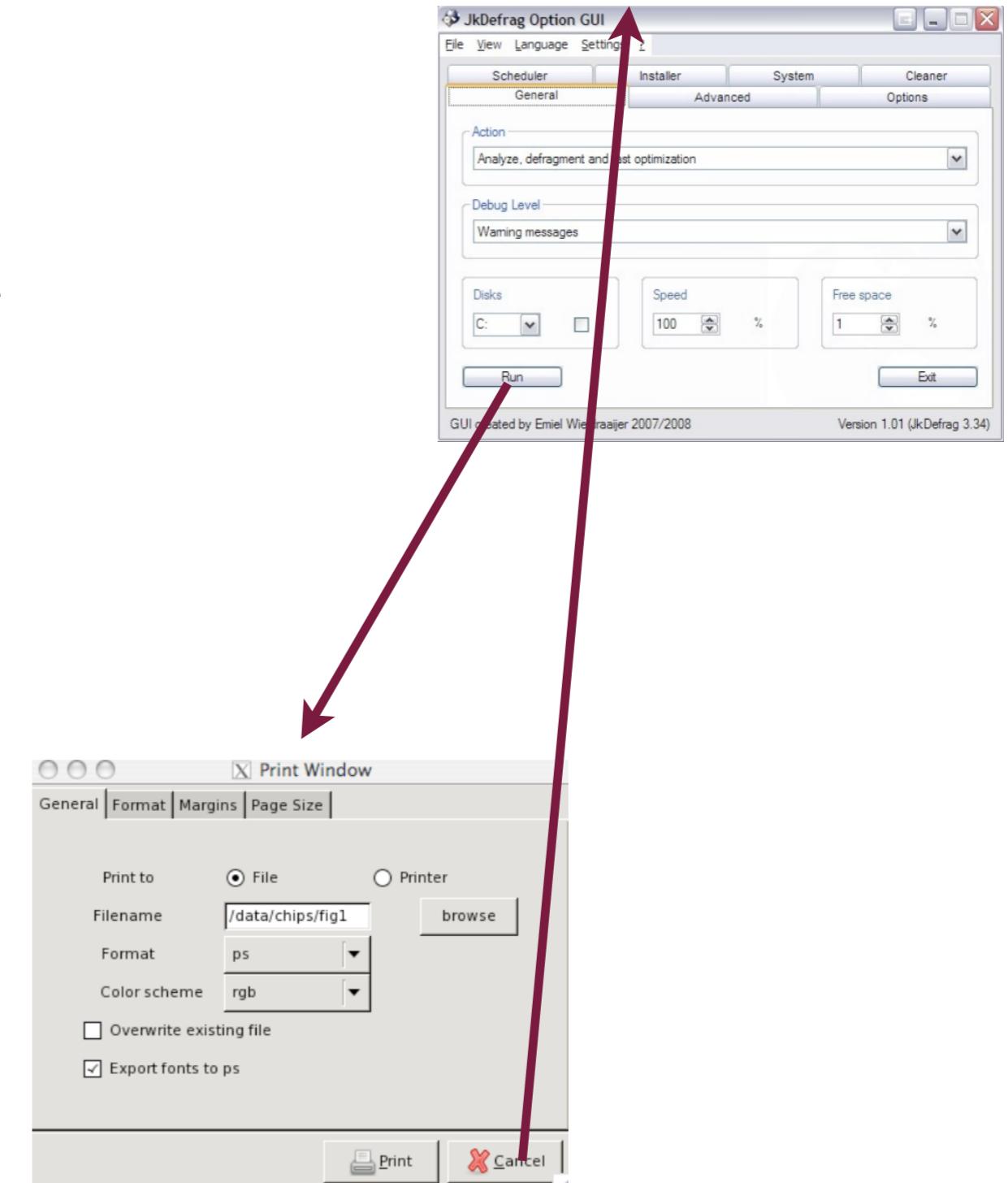
- looking from this view out

- **Global**

- structure of navigation between views/screens

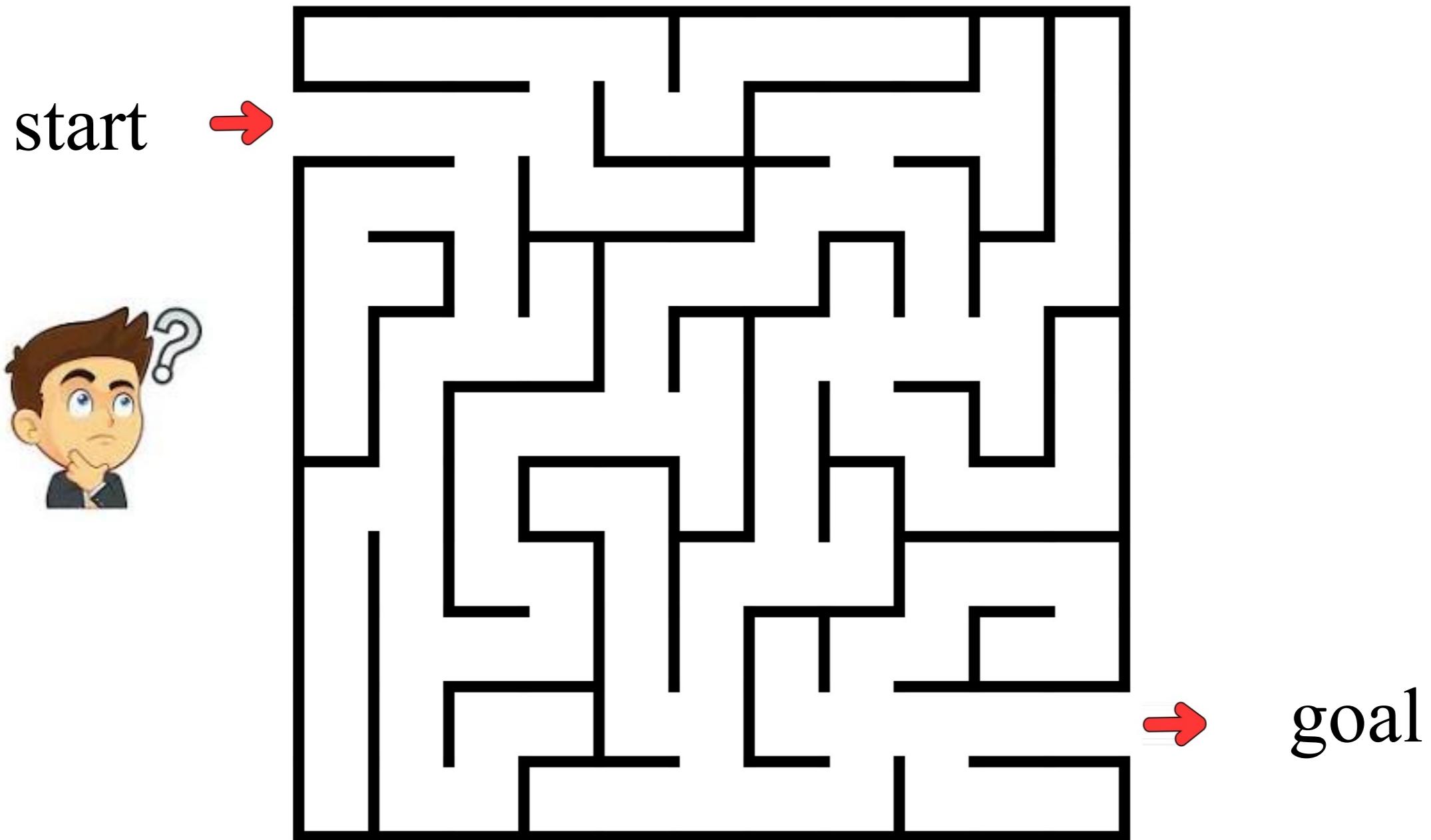
- **Wider still**

- interaction with other applications





# Goal seeking



**Do not turn it into a maze!!!**

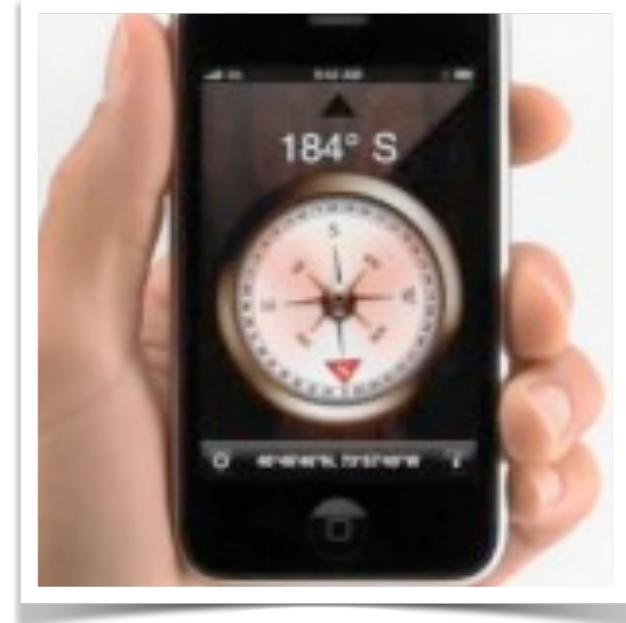


# Four golden rules

---

## A Sense of Location

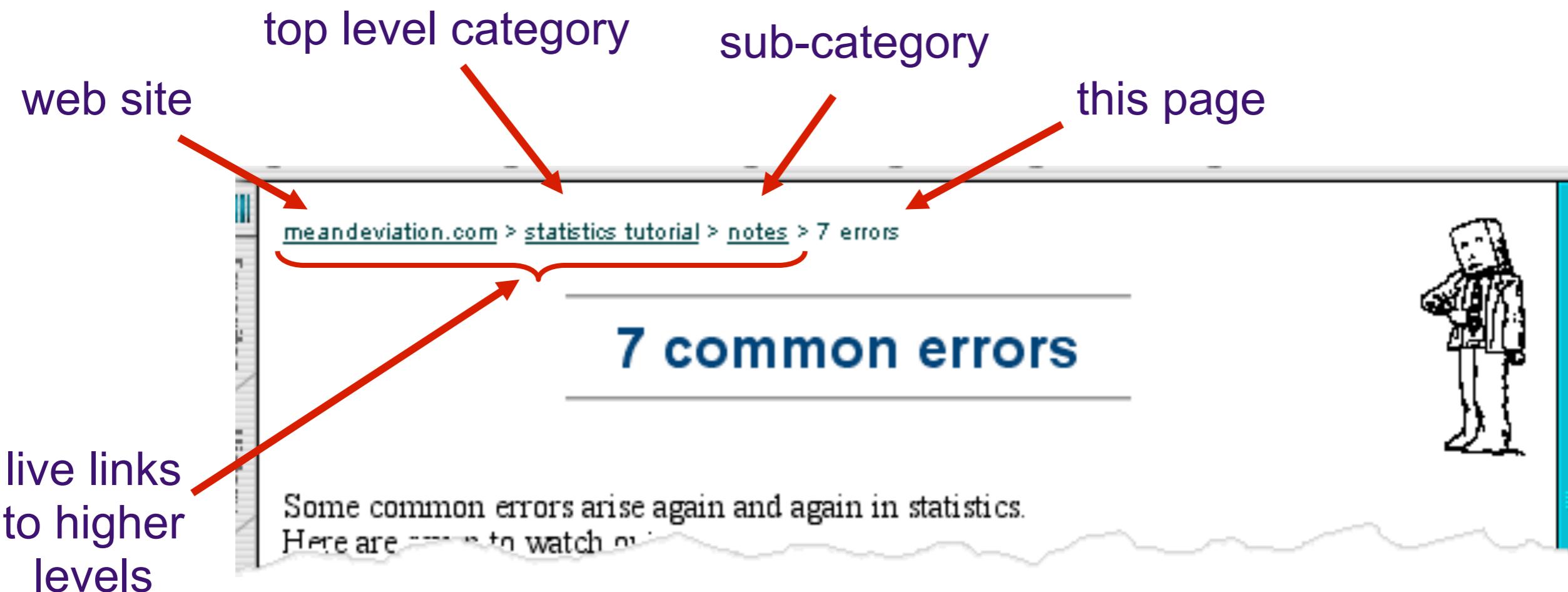
- knowing where you **are**
- knowing what you **can do**
- knowing where you **are going**
  - or what will happen
- knowing where you've **been**
  - or what you've done





# Where you are

shows path through web site hierarchy





# What can you do? Are you lost?





# Location and Modes

- Lock to prevent accidental use ...
  - remove lock - ‘c’ + ‘yes’ to confirm
  - frequent practiced action
- **But:** if lock forgotten
  - in pocket ‘yes’ gets pressed
  - goes to phone book
  - in phone book ...
    - ‘c’ - delete entry
    - ‘yes’ - confirm
  - ... oops !





# Four golden rules

---

## A Sense of Location

- knowing where you are
- knowing what you can do
- knowing where you are going
  - or what will happen
- knowing where you've been
  - Or what you've done





# Structure

---

## Global

**between screens  
within the application**



# Beware the big button trap

---



things

other things



more things

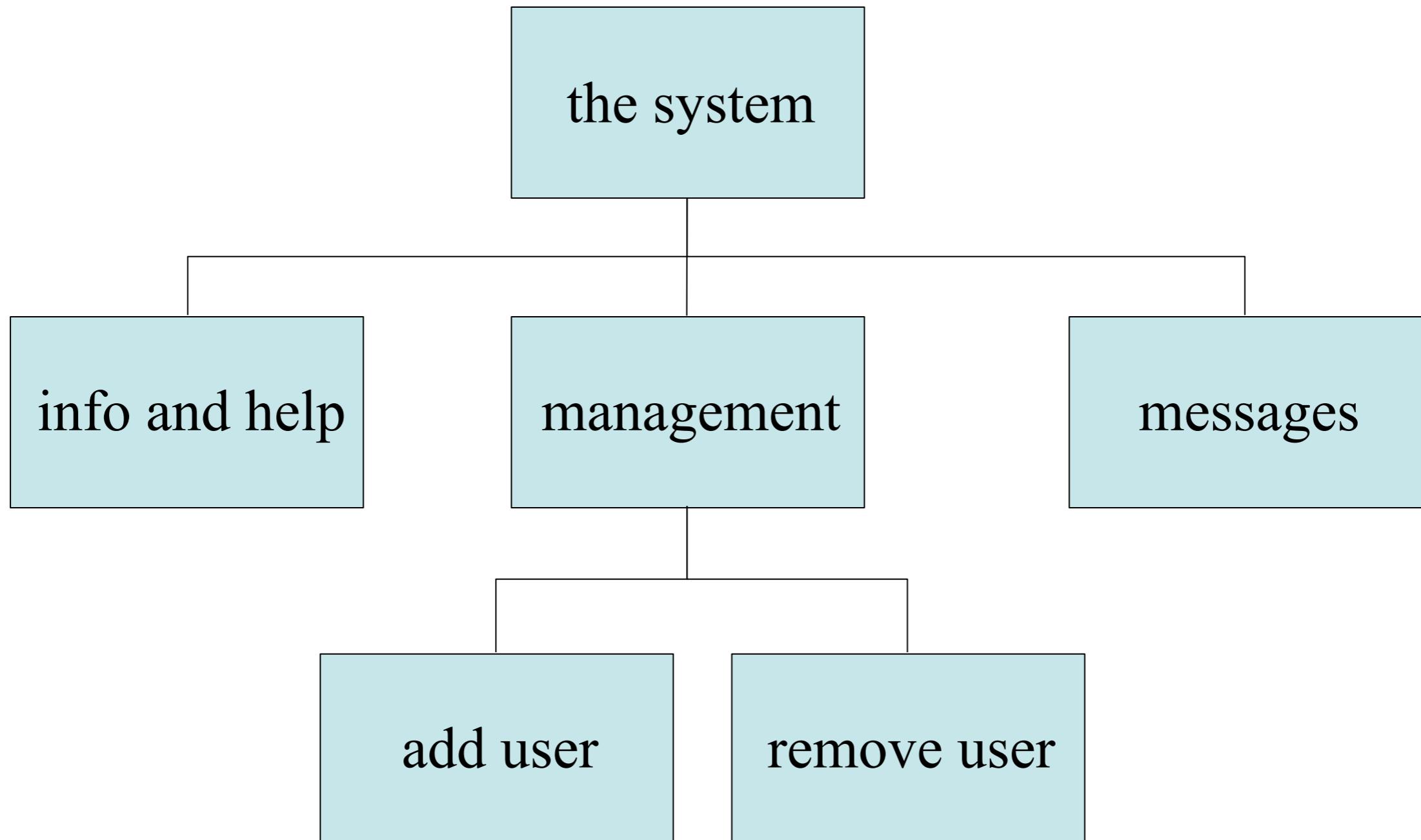
the thing from  
outer space

- **Where do they go?**
  - Lots of room for extra text!
  - Sacrifice form for the sake of function!



# hierarchical diagrams

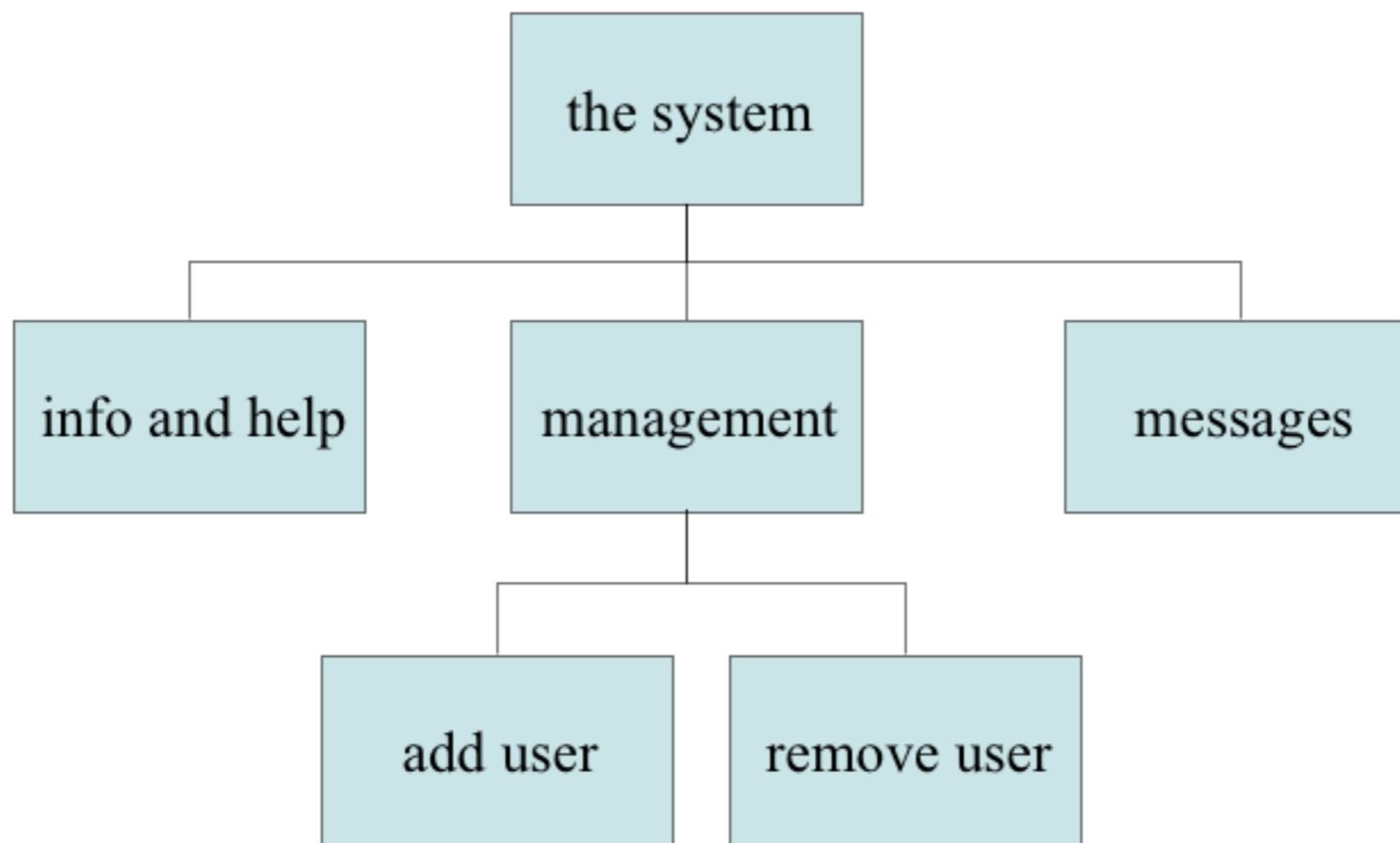
---





# Hierarchical diagrams ctd.

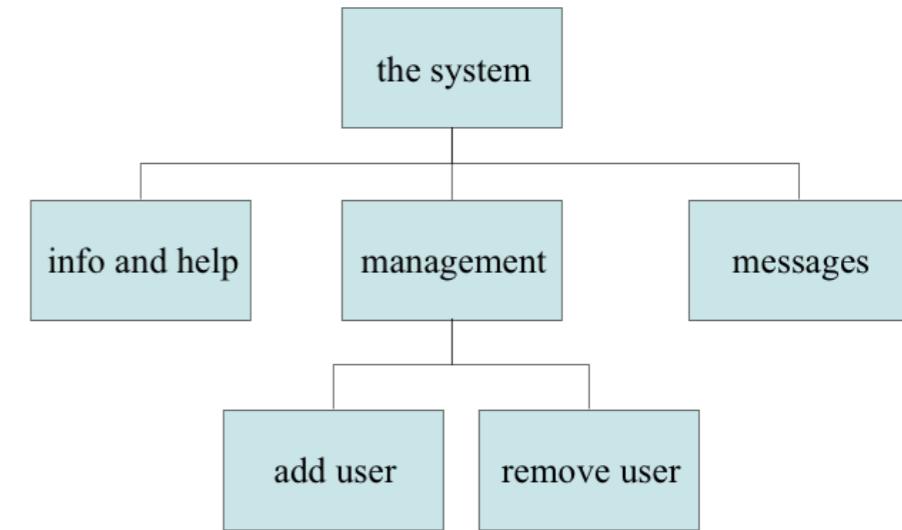
- **Parts of application**
  - screens or groups of screens
- **Typically functional separation**





# Navigating hierarchies

- **Deep is difficult!**
  - people find simpler than most
- **Misuse of Miller's  $7 \pm 2$** 
  - short term memory, not menu size, not visual search.  
see also: <http://www.hcibook.com/e3/online/menu-breadth/>
- **Optimal?**
  - many items on each screen
  - but structured within screen
  - the naturalness of the classification, which depends from knowing the user.





# Navigating hierarchies

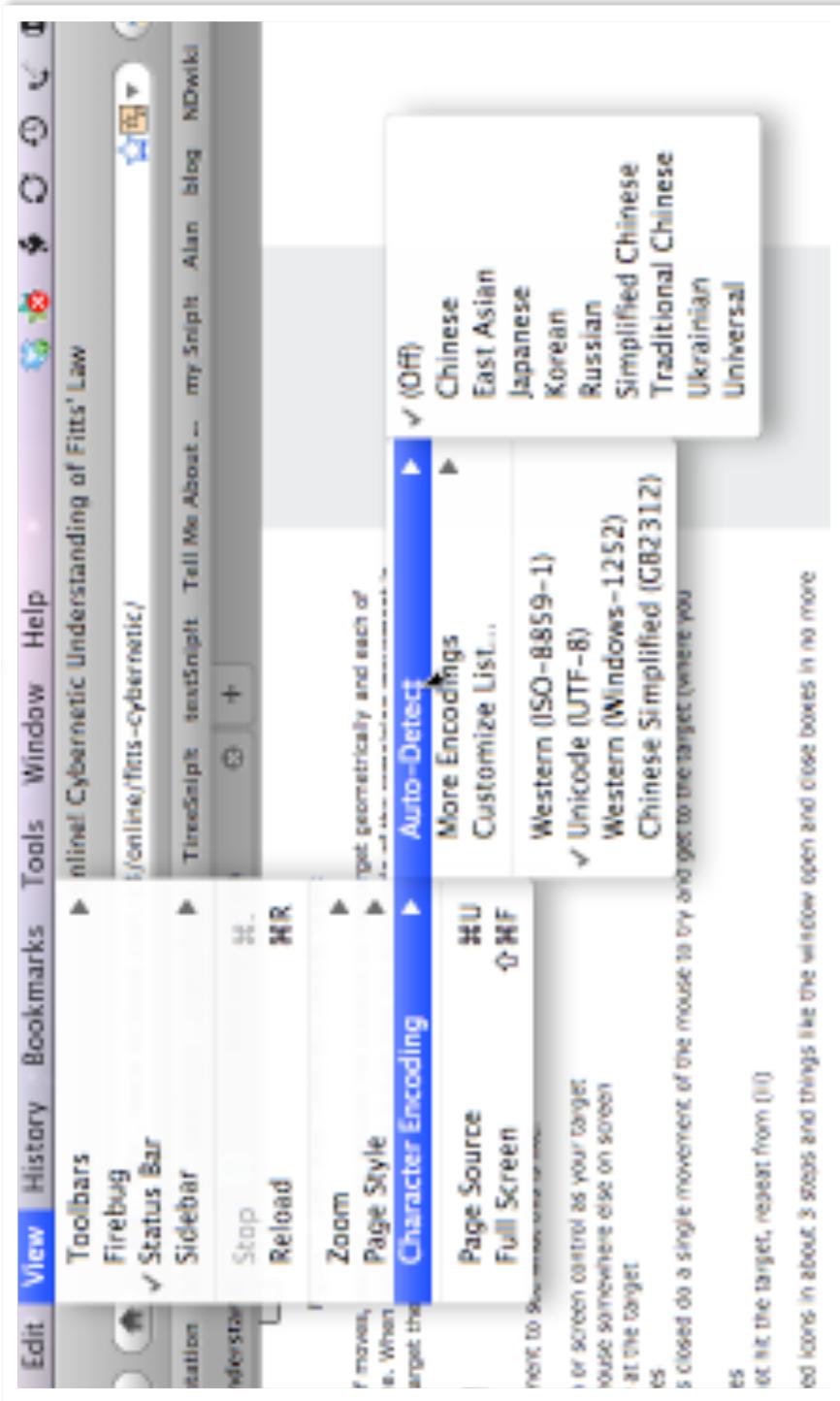
The screenshot shows a web browser window with the following interface details:

- Menu Bar:** Edit, View, History, Bookmarks, Tools, Window, Help.
- Toolbar:** Includes icons for Home, Stop, Reload, Zoom, Page Style, Character Encoding, and others.
- Address Bar:** Displays the URL: <http://online/fitts-cybernetic/>.
- Content Area:** Shows a snippet of text related to Fitts' Law and Cybernetics.
- Character Encoding Submenu (View > Character Encoding):**
  - Auto-Detect (selected)
  - More Encodings
  - Customize List...
  - Western (ISO-8859-1)
  - Unicode (UTF-8) (selected)
  - Western (Windows-1252)
  - Chinese Simplified (GB2312)
- Page Source:** A link to view the page source code.
- Full Screen:** A button to switch to full screen mode.

see also: <http://www.hcibook.com/e3/online/menu-breadth/>



# Navigating hierarchies

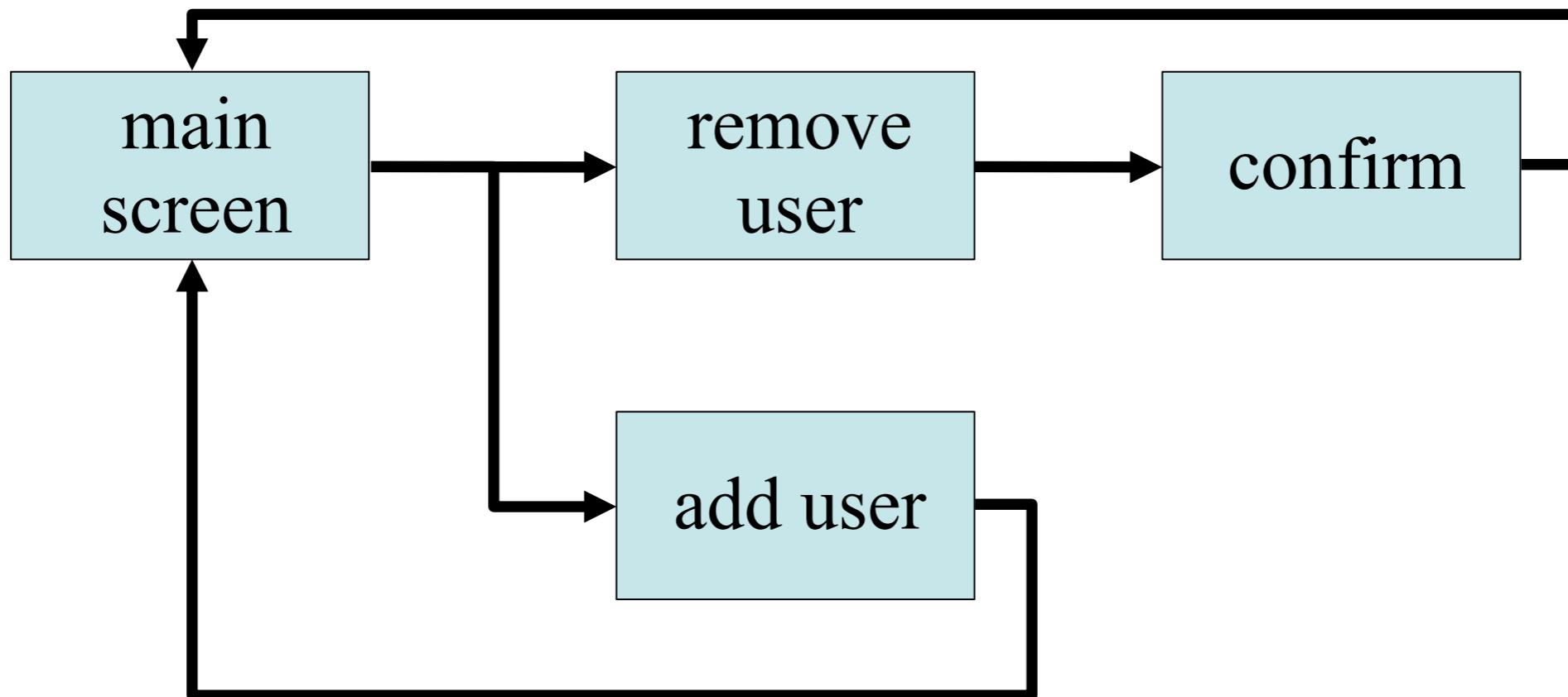


## Menu Selection Loop

1. visually find the necessary menu item
2. select the item (with mouse or finger, depending on the device)
3. wait for the system to show the next level of the menu
4. start again at 1 until you get to the bottom of the hierarchy



# Network diagrams

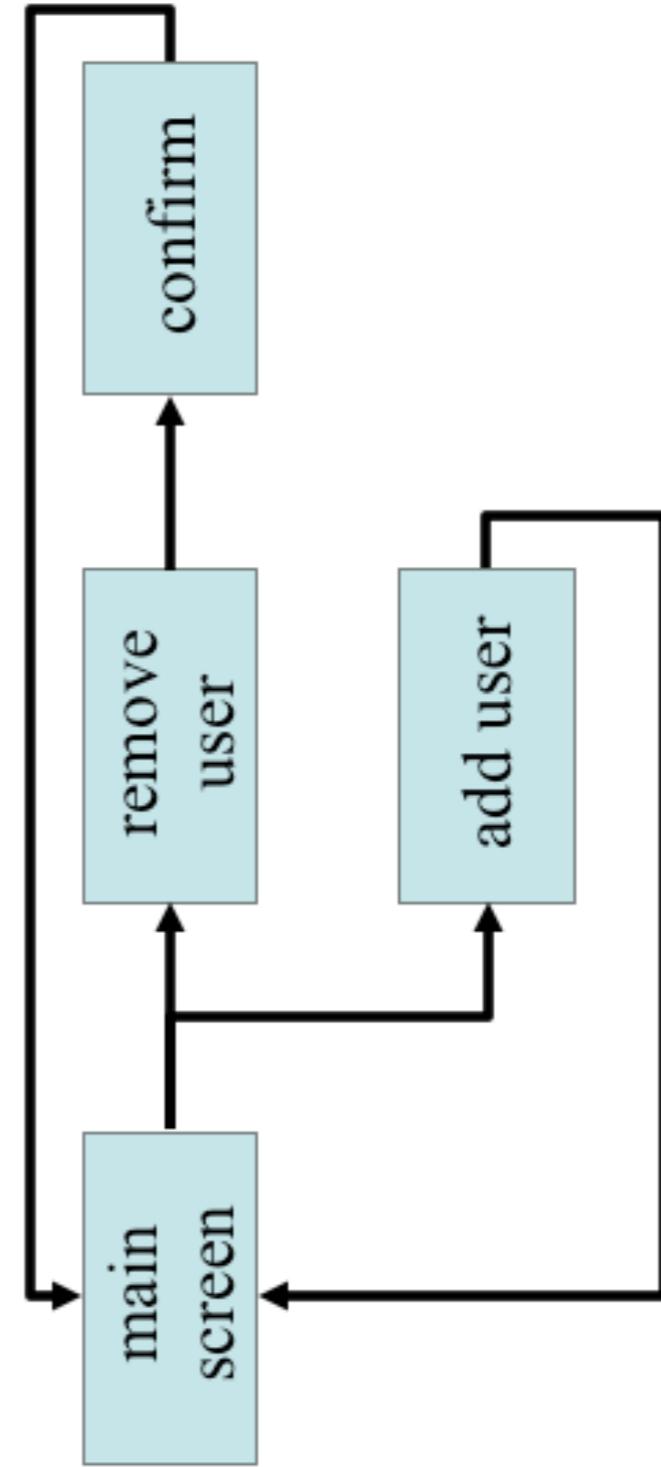


- Show different paths through the system and even loops.



# Network diagrams

- What leads to what
- What happens when
- Including branches
- More task oriented than hierarchy





# Structure

---

**Wider still**

**From one screen looking out**



# Wider still - interact with other sys.

- **Style issues:**

- platform standards, consistency

- **Functional issues**

- e.g. cut and paste

- **Navigation issues**

- embedded applications

- links to other apps ... the web

- ▶ example: view PDF in browser.

