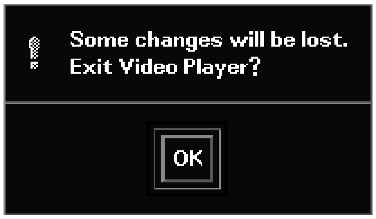
ICS 105: Project in HCI

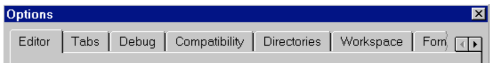
Design Process

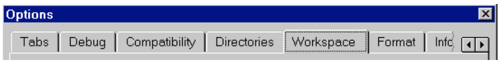
bad design is all around…



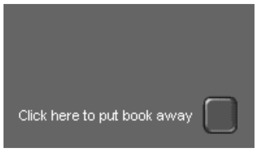
Some changes will be lost. Exit Video Player?

bad design is all around…



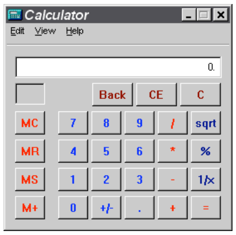


bad design is all around…

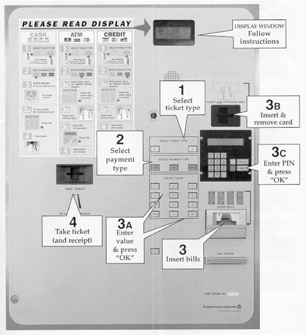


Click here to put book away

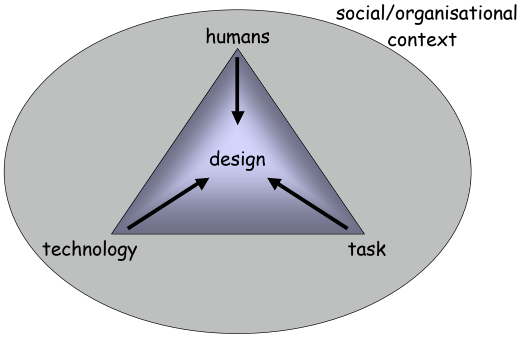
bad design is all around…



bad design is all around…



what is HCI?



social/organisational context

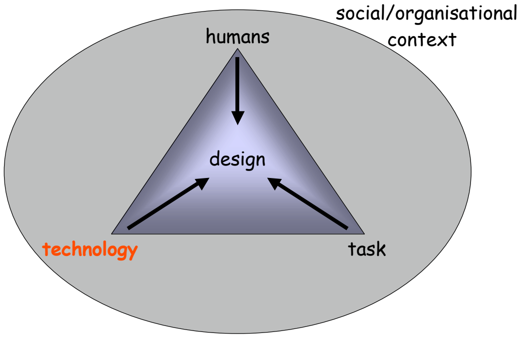
humans

design

technology

task

what is HCI?



social/organisational context

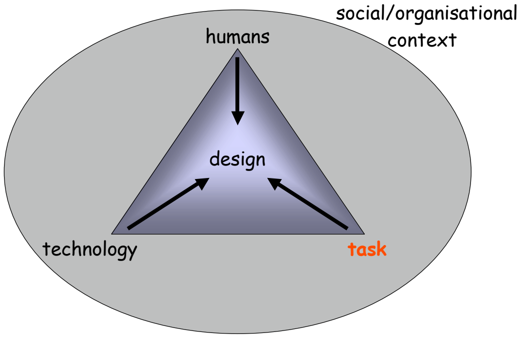
humans

design

technology

task

what is HCI?



social/organisational context

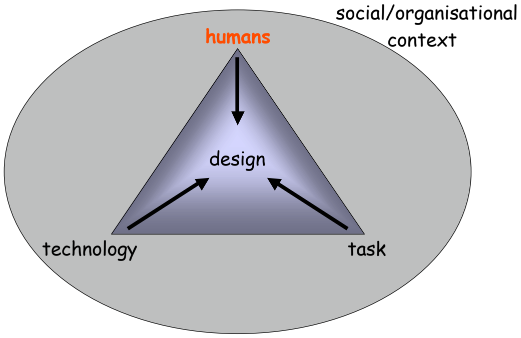
humans

design

technology

task

what is HCI?



social/organisational context

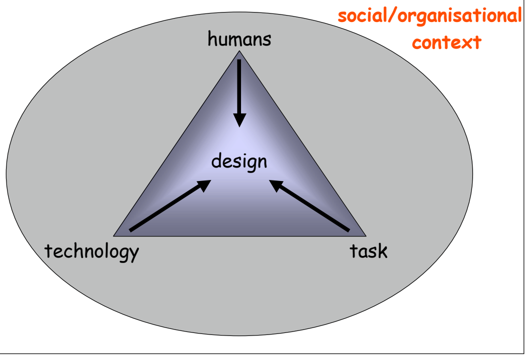
humans

design

technology

task

what is HCI?



social/organisational context

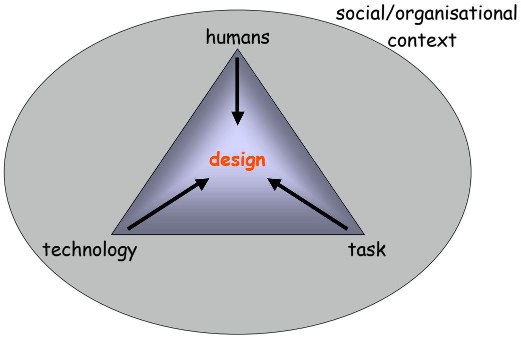
humans

design

technology

task

what is HCI?



social/organisational context

humans

design

technology

task

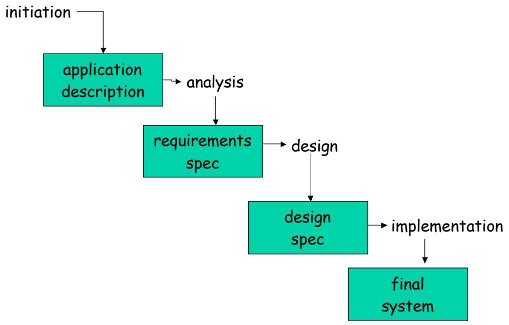
engineering for usability

- usability doesn’t just happen by accident
- usability as a feature of the engineering process
  - design process to ensure usable software
  - take usability as central, not tacked-on
    - remember, usability != interface

usability

- ease of learning
  - do it faster the second time around…
- recall
  - remember how to do things from one session to next
- productivity
  - perform tasks quickly and efficiently
- minimal error rates
  - reduce number
  - if they occur, good feedback for better recovery
- high user satisfaction
  - users confident of success

waterfall model



initiation

application description

analysis

requirements spec

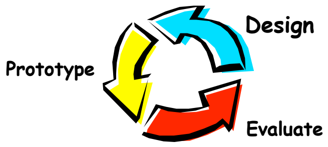
design

design spec

implementation

final system

user-centered design



Design

Prototype

Evaluate

comparing the approaches

- waterfall
  - structured
  - customers, not users
  - limited feedback
    - errors carry forward
    - expensive to fix
- iterative
  - focused on users
  - continual evaluation
    - finds errors sooner

prototyping approaches

- rapid prototyping
  - build, test, throw it away
  - key focus: collecting requirements information
- incremental prototyping
  - building large systems in stages
  - key focus: reducing delays
- evolutionary prototyping
  - “design in place”
  - key focus: balancing prototyping with development
- the problems of premature commitment

prototyping

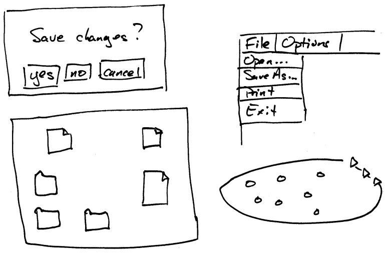
- the central problems of prototyping
  - the system is incomplete
    - what parts to build and use?
    - how to make them “work”?
  - getting to real user experience
    - creating a persuasive situation

prototyping

- degree of integration
  - horizontal prototype
  - vertical prototype
- interaction style
  - chauffeurred prototype
  - wizard-of-oz
- how polished?
  - low-fi prototype
  - high-fi prototype

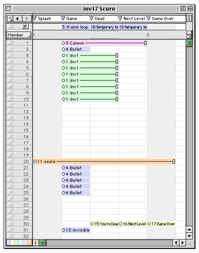
prototyping

- “low-fi” prototypes



prototyping

- high-fi prototypes
  - computer-based prototypes
  - Macromind Director
    - commonly used at Apple
    - basic metaphor
      - stage
      - actors
      - score

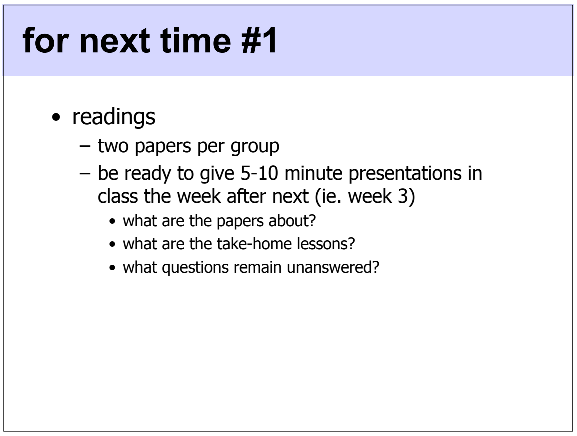


prototypes

- trade-offs
  - “high-fi” prototypes can be more impressive
    - getting customer buy-in
      - note – this is NOT the purpose of UI prototypes!
    - more persuasive
      - good: users don’t need to suspend belief
      - bad: too “polished” can make people less critical
  - “low-fi” prototypes are easier to build
    - faster iteration

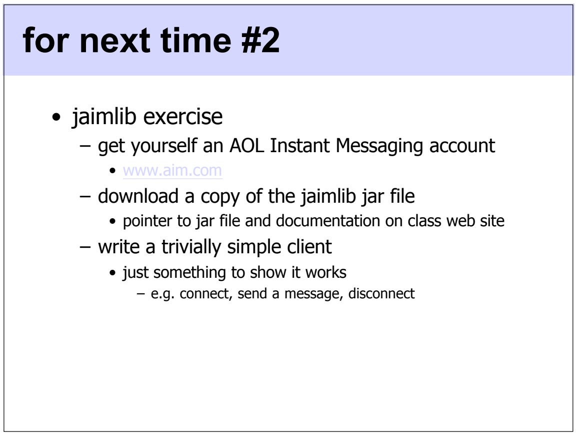
immediate relevance…

- we don’t have the luxury of time in 105
  - 10 weeks is very short
  - but iteration is critical
    - we need to create more than one system
    - we need some evaluation input
  - so we’ll do some lo-fi prototyping
  - timetable:
    - week 2 -- swing fundamentals (and reading)
    - week 3 -- ideas, analysis (and discussion of readings)
    - week 4/5 -- designing and prototyping



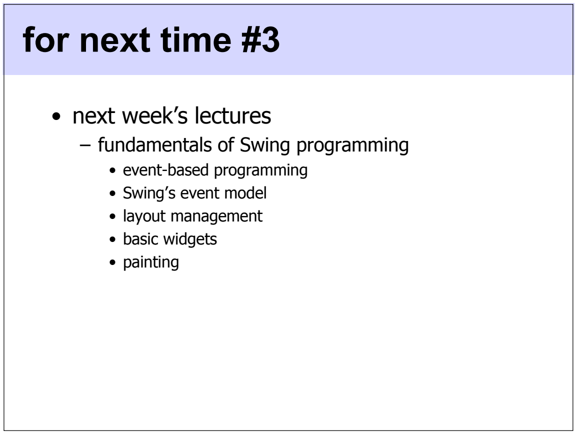
for next time #1

- readings
  - two papers per group
  - be ready to give 5-10 minute presentations in class the week after next (ie. week 3)
    - what are the papers about?
    - what are the take-home lessons?
    - what questions remain unanswered?



for next time #2

- jaimlib exercise
  - get yourself an AOL Instant Messaging account
    - www.aim.com
  - download a copy of the jaimlib jar file
    - pointer to jar file and documentation on class web site
  - write a trivially simple client
    - just something to show it works
      - e.g. connect, send a message, disconnect



for next time #3

- next week's lectures
  - fundamentals of Swing programming
    - event-based programming
    - Swing's event model
    - layout management
    - basic widgets
    - painting