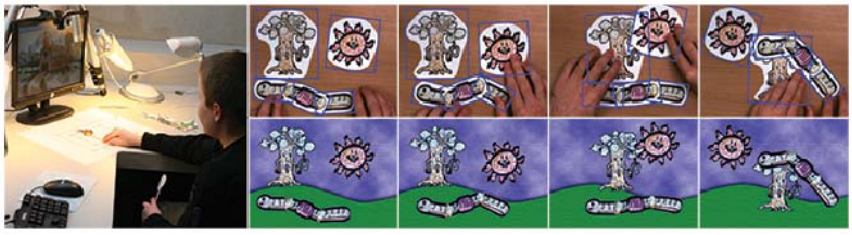
Human Information Processing (Perception)

CS160: User Interfaces

Maneesh Agrawala

Video Puppetry



Barnes, Jacobs, Sanders, Goldman, Rusinkiewicz, Finkelstein, Agrawala, SIGGRAPH Asia 2008

Individual Programming Assignment
(due Mar 4)

Project Management/To-Do List

Tasks have the following properties:

- Task Name
- Percentage Completed (0-100%)
- Start and End date
- Priority
- List of people assigned to the task
- URL related to the task

Checklist view

- Include checkbox to automatically set completion percentage to 100%
- You should be able to see the completion percentage

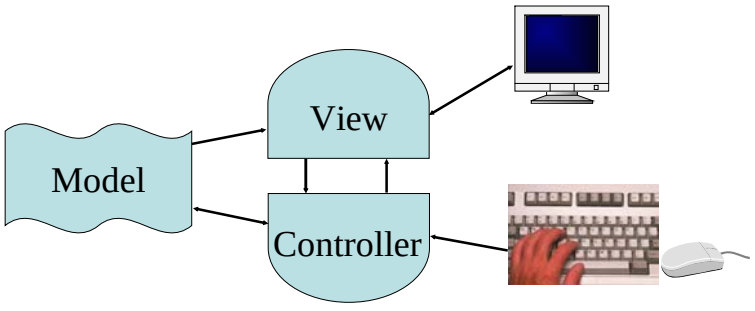
Timeline view

Magic lens: http://dohistory.org/diary/exercises/lens/index.html

Review: Model-View-Controller

Architecture for interactive apps

- Model: Info the application manipulates
- View: Visual display of the model
- Controller: Receives input & decides what they do



Review: Creating Lo-Fi Prototypes

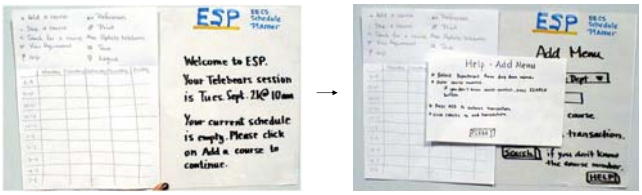
Set a deadline
- Don't think too long - build it!

Draw a window frame on large paper
- **Draw at a large size, but use correct aspect ratio**

Put different screen regions on cards
- Anything that moves, changes, appears/disappears

Ready response for any user action
- e.g., Have those pull-down menus already made

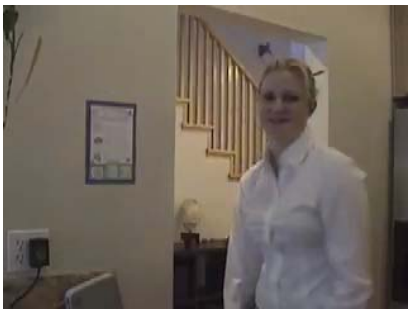
Use photocopier to make many versions

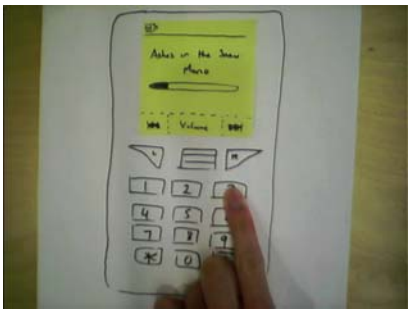


Review: Video Prototyping

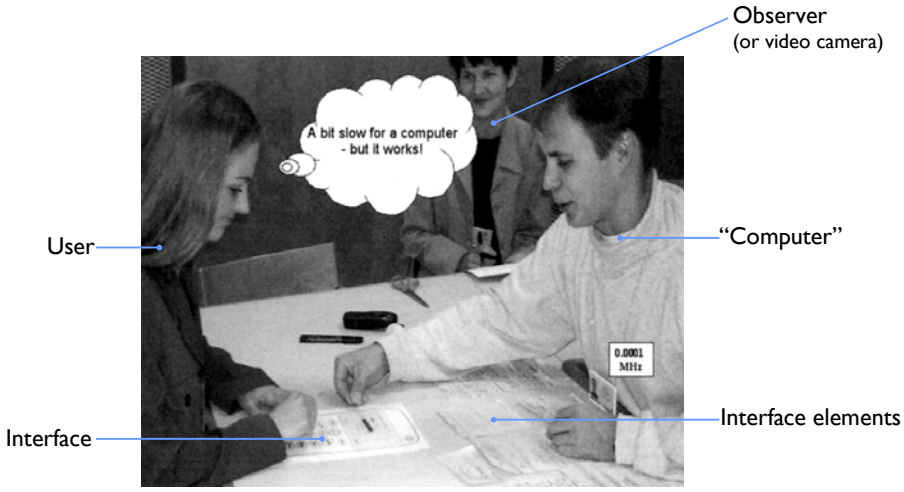
- Illustrate how users will interact w/ system
- Quick to build
- Inexpensive

Better illustrates context of use than a storyboard





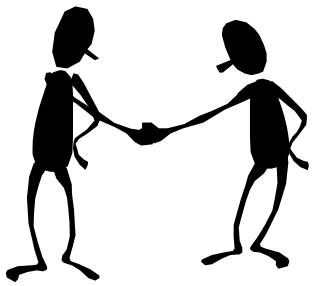
Review: Testing Prototypes



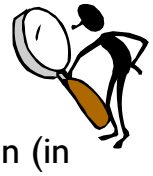
10 steps to better evaluation

1. Introduce yourself

   some background will help relax the subject.



10 steps



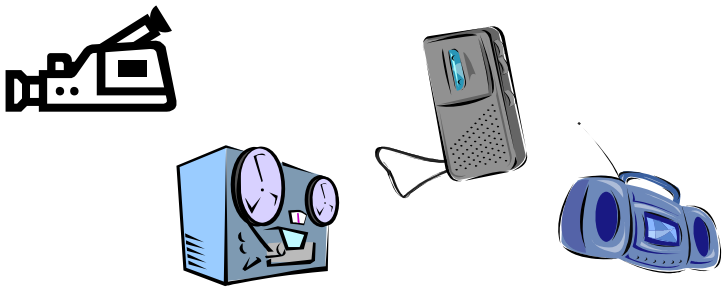
2. Describe the purpose of the observation (in general terms), and set the participant at ease
   - You're helping us by trying out this product in its early stages.
   - If you have trouble with some of the tasks, it's the product's fault, not yours. Don't feel bad; that's exactly what we're looking for.

10 steps (contd.)

3. Tell the participant that it's okay to quit at any time, e.g.:
   - Although I don't know of any reason for this to happen, if you should become uncomfortable or find this test objectionable in any way, you are free to quit at any time.

10 steps (contd.)

4. Talk about the equipment in the room.
   - Explain the purpose of each piece of equipment (hardware, software, video camera, microphones, etc.) and how it is used in the test.



10 steps (contd.)

5. Explain how to "think aloud."
   - Explain why you want participants to think aloud, and demonstrate how to do it. E.g.:
   - We have found that we get a great deal of information from these informal tests if we ask people to think aloud. Would you like me to demonstrate?

10 steps (contd.)

6. Explain that you cannot provide help.

10 steps (contd.)

7. Describe the tasks and introduce the product.
   - Explain what the participant should do and in what order. Give the participant written instructions for the tasks.
   - Don't demonstrate what you're trying to test.

10 steps (contd.)

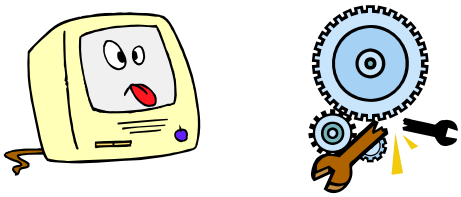
8. Ask if there are any questions before you start; then begin the observation.

10 steps (contd.)

9. Conclude the observation. When the test is over:
   - Explain what you were trying to find.
   - Answer any remaining questions.
   - discuss any interesting behaviors you would like the participant to explain.

10 steps (contd.)

10. Use the results.
    - When you see participants making mistakes, you should attribute the difficulties to faulty product design, not to the participant.



Advantages of Low-Fi Prototyping

Takes only a few hours
- No expensive equipment needed

Can test multiple alternatives
- Fast iterations
  - Number of iterations is tied to final quality

Can change the design as you test
- If users are trying to use the interface in a way you didn't design it – go with what they think! Adapt!

Especially useful for hard to implement features
- Speech and handwriting recognition

Drawbacks of Lo-Fi Prototyping

Evolving the prototype requires redrawing
- Can be slow (but reprogramming usually slower)

Lack support for "design memory"

Force manual translation to electronic format

Do not allow real-time end-user interaction

Caveats

There is a down-side to the informal design approach:

Often hard to involve ***paying*** clients as subjects – they treat the fidelity of the interface as a sign of development effort

Mitigators: involve them early and often, correspond with the same people, explain the process up front (set expectations)

LoFi Prototype (due Mar 16)

Low-Fidelity Prototype

- Identify project mission statement
- Create low-fidelity prototype that supports 3 tasks
  - 1 easy, 1 moderate, 1 difficult task
- Create a video prototype showing (cameras next class)
  - How it supports the 3 tasks
  - Context in which is will be used (back story)
- Test the prototype with target users
  - No one from this class
  - Not your friends

Topics

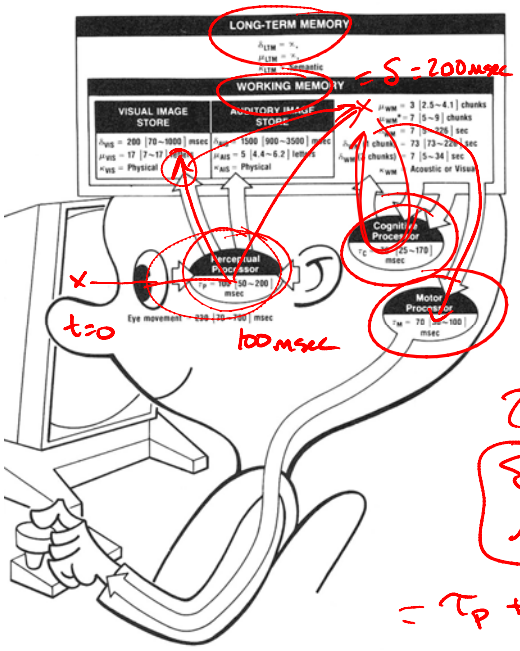
- The Model Human Processor
- Memory

The Model Human Processor

Why Model Human Performance?

Why Model Human Performance?

To predict impact of new technology/interface
- Apply model to predict effectiveness
- We could build a simulator to evaluate user interface designs



Human Info. Processor
- Perceptual
- Cognitive
- Motor (will discuss later)
- Working memory
- Long-term memory

Unified model
- Probably inaccurate
- Predicts perf. well
- Very influential

τ = cycle time

δ = decay time

μ = capacity

$= \tau_P + \tau_C + \tau_M$

Perceptual Processor

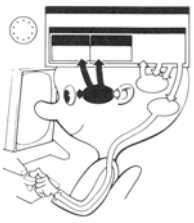
Physical store from our senses: sight, sound, touch, …
- Code directly based on sense used
  - Visual, audio, haptic, … features

Selective
- Spatial
- Pre-attentive: color, direction…

Capacity of visual store
- Example: 17 letters

Decay time for working memory: 200ms

Recoded for transfer to working memory
- Progressive: 10ms/letter



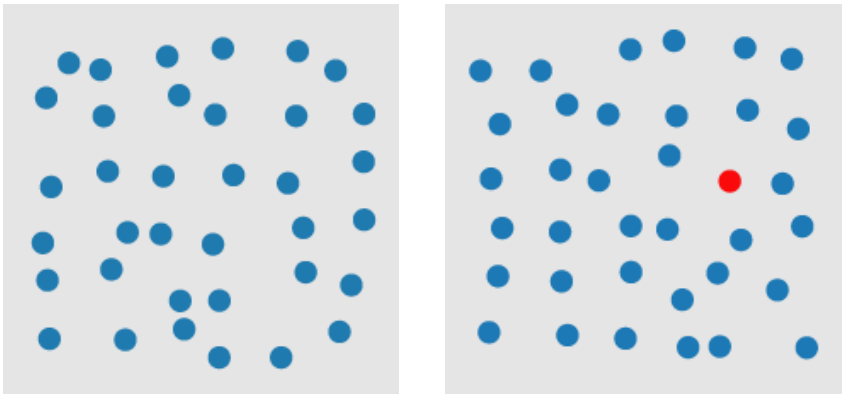
How many 3's

```
12817687561389765469845069856049828267 62
98098584582245098564589450984509809435 85
90910302099059595957725646750506789045 67
88457898098216776548763649085609129496 86
```

How many 3's

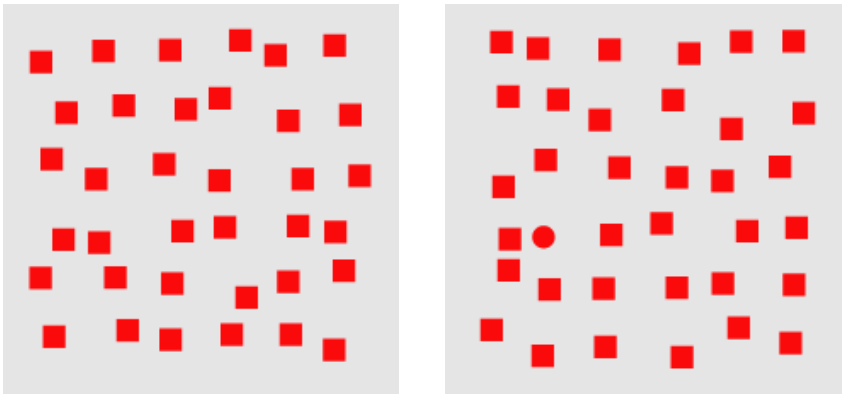
128176875613897654698450698560498282676298098584582245098564589450984509809435859091030209905959595772564675050678904567884578980982167765487636490856091294968 6

Visual Pop-Out: Color



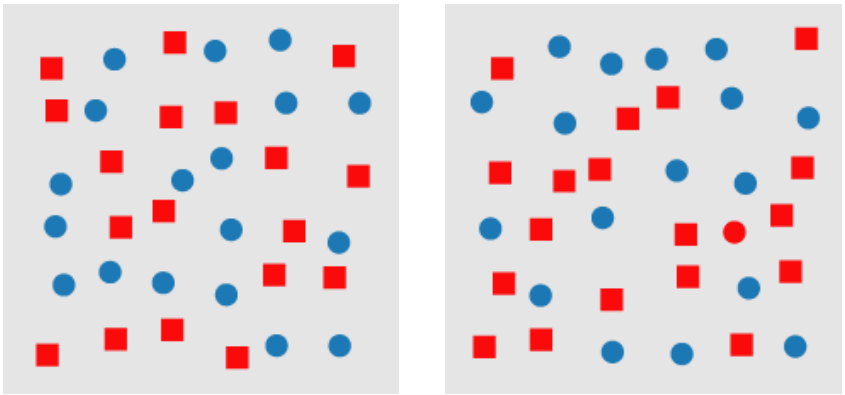
http://www.csc.ncsu.edu/faculty/healey/PP/index.html

Visual Pop-Out: Shape



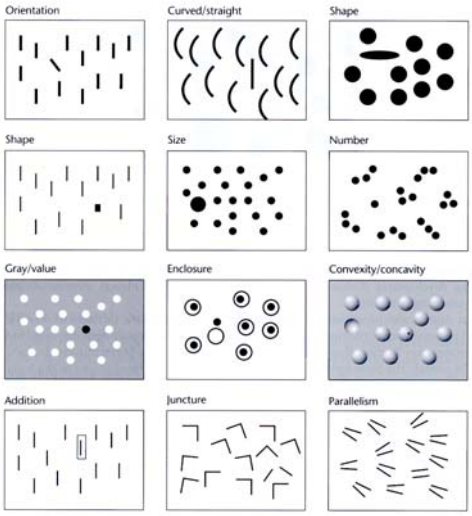
http://www.csc.ncsu.edu/faculty/healey/PP/index.html

Feature Conjunctions



http://www.csc.ncsu.edu/faculty/healey/PP/index.html

Preattentive Features

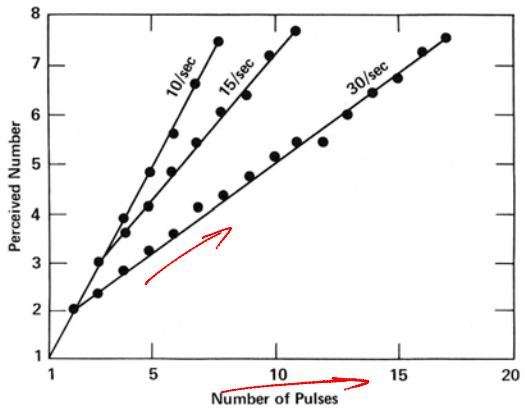


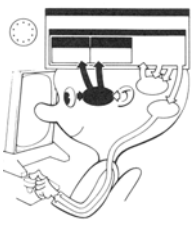
[Information Visualization. Figure 5. 5 Ware 04]

Perceptual Processor

Cycle time

- Quantum experience: 100ms
  - Percept fusion





Perceptual Processor

Cycle time
- Quantum experience: 100ms
  - Percept fusion
- Frame rate necessary for movies to look continuous?
  - time for 1 frame < Tp (100 msec) -> 10 frame/sec.
- Max. morse code rate can be similarly calculated

Perceptual causality
- Two distinct stimuli can fuse if the first event appears to *cause* the other
- Events must occur in the same cycle

Perception of Causality [Michotte 46]

Michotte demonstration 1. What do you see? Most observers report that the red ball **hit** the blue ball. The blue ball moved "**because** the red ball hit it." Thus, the red ball is perceived to "**cause**" the red ball to move, even though the balls are nothing more than color disks on your screen that move according to a program.

http://cogweb.ucla.edu/Discourse/Narrative/Heider_45.html

Perception of Causality [Michotte 46]

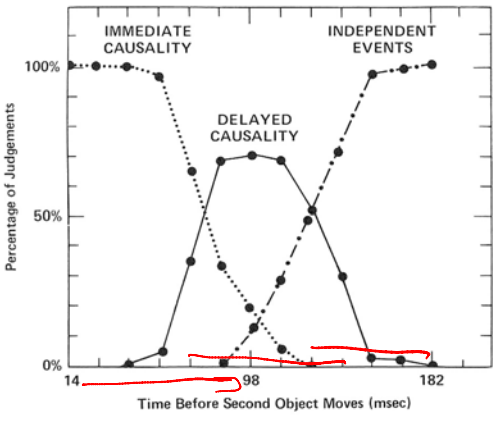
Michotte demonstration 1. What do you see? Most observers report that the red ball **hit** the blue ball. The blue ball moved "**because** the red ball hit it." Thus, the red ball is perceived to "**cause**" the red ball to move, even though the balls are nothing more than color disks on your screen that move according to a program.

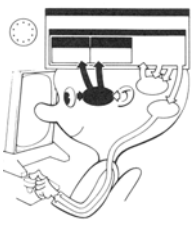
http://cogweb.ucla.edu/Discourse/Narrative/Heider_45.html

Perceptual Processor

Cycle time

- Quantum experience: 100ms
  - Causality

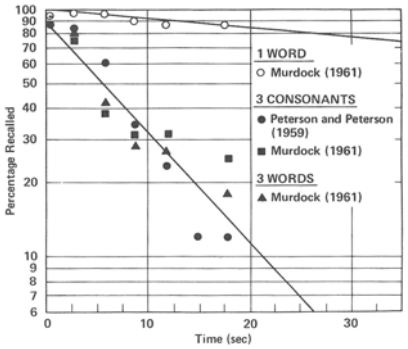


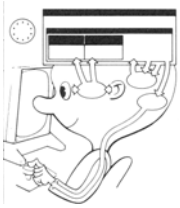


Working Memory

Access in chunks
- Task dependent construct
- 7 +/- 2 (Miller)

Decay
- Content dependant
  - 1 chunk 73 sec
  - 3 chunks 7 sec
- Attention span
  - Interruptions > decay time



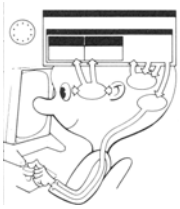


Long Term Memory

Very large capacity
- Semantic encoding

Associative access
- Fast read: 70ms
- Expensive write: 10s
  - Can also move from WM to LTM via rehearsal

Context at the time of acquisition key for retrieval



Cognitive Processor

Cycle time: 70ms
- Can be modulated

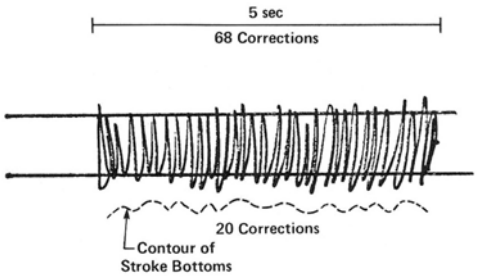
Typical matching time
- Digits: 33ms
- Colors: 38ms
- Geometry: 50ms…

Fundamentally serial
- One locus of attention at a time
  - Eastern 401, December 1972
    - Crew focused on landing gear indicator bulb,
    - Aircraft is loosing altitude (horn, warning indicator…),
    - Aircraft crashed in the Everglades
    - see "The Human Interface" by Raskin, p25
  - But what about driving and talking?

Motor Processor

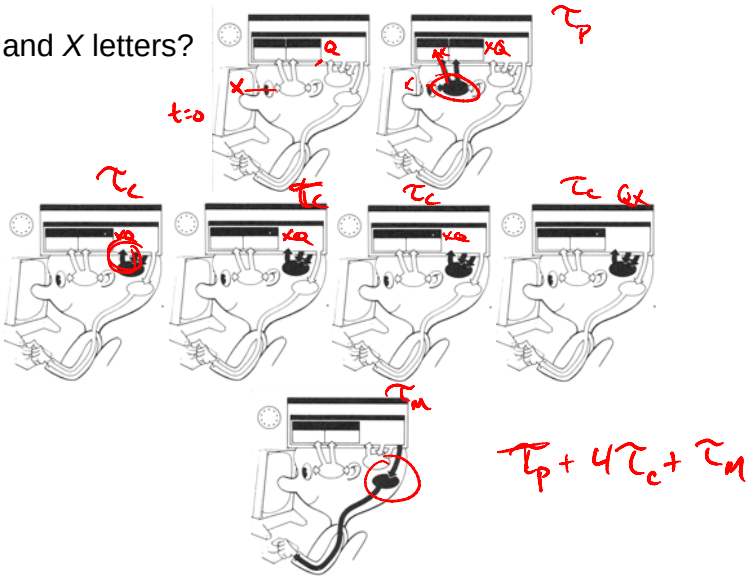
Receive input from the cognitive processor

Execute motor programs
- Pianist: up to 16 finger movements per second
- Point of no-return for muscle action



Recognize-Act Cycle

Are *Q* and *X* letters?



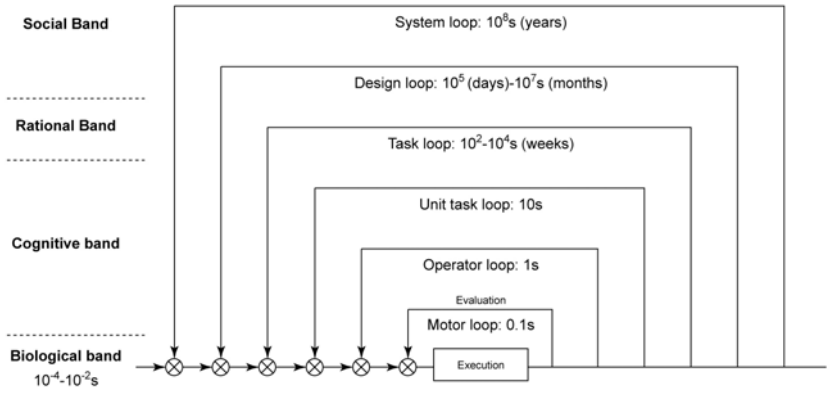
Cognitive Processor

Page 70 of Card Moran and Newell

- Clocks starts when 2nd letter is flashed
- Move letter x into visual store WM
  T_p
- Recognize the symbol as codes
  + T_c
- Classify the codes as letters
  + T_c
- Match the fact that they are both letters
  + T_c
- Initiate motor response
  + T_c
- Process motor command
  +T_m
- Approx 450 (180-980) ms

Human Interaction Loops

(Newell)



Principles of Operation

Interface should respect limits of human performance
- Preattentive features pop-out
- Events within cycle time fuse together
- Causality

Recognize-Act Cycle of the cognitive processor
- On each cycle contents in WM initiate cognitive actions
- Cognitive actions modify the contents of WM

Discrimination Principle
- Retrieval is determined by candidates that exist in memory relative to retrieval cues
- Interference by strongly activated chunks
  - Two strong cues in working memory
  - Link to different chunks in long term memory