

Predictive Engineering Models of Interface Usability: Cognitive Modeling and Human-Computer Interaction

Anthony J. Hornof

Department of Computer and Information Science
University of Oregon
hornof@cs.uoregon.edu
<http://www.cs.uoregon.edu/~hornof/>

Presented at CHIFOO - 3/21/01

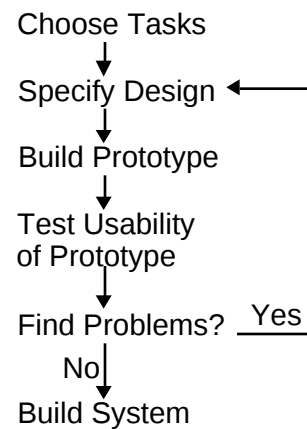
Main points of this talk

An overview of how cognitive modeling is already being used to help design usable interfaces

Cognitive models of menu search that show how people look for things on a computer screen

A proposed visual search prediction tool that designers could use to automatically predict the usability of their screen layouts and web pages

Human Factors Approach to Building Usable Systems



Problems with this approach:

- Slow
- Expensive
- Does not explain

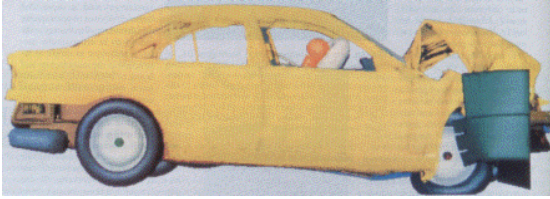
why

Solution:

Incorporate engineering

Engineering Models

Computer simulation:



Physical test:

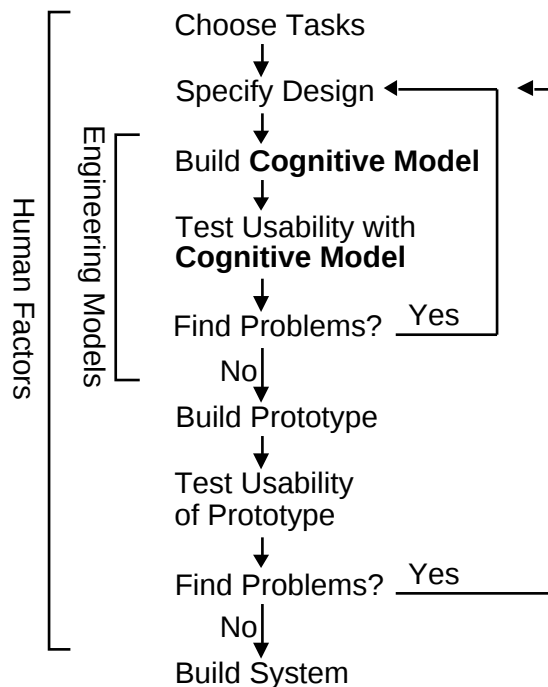


[Adapted from Thomke et al. 1999]

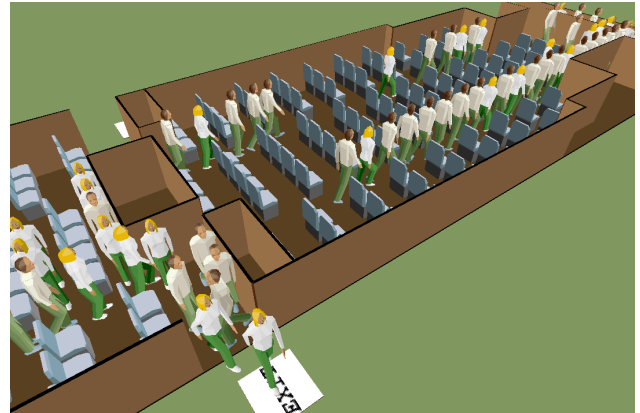
Why Engineering Models Are Great

- Faster
- Cheaper
- Explain *why* a design is be
- Can contribute to a design specification

Human Factors + Engineering Models = How to Build Usable Systems



Engineering Models



Galea et al. - <http://fseg.gre.ac.uk/exodus/air.html>

What is a cognitive model?

"A cognitive model is a computer program that behaves like a human being.

It may emulate the perceptual, cognitive, and/or motor processes people go through to complete a task.

It may take the same amount of time that people take to perform a task.

It may make the same type of errors that people make.

It may take the same amount of time and require the same type of experience to learn to perform a task.

It may do the same inefficient fumbling for a solution to a difficult problem.

In all, the point is to have the computer behave like a human, not simply to get the job done with the least effort or in the least time."

(Bonnie John, 1998)

A Cognitive Model Using the Keystroke Level Model (KLM)

Card, Moran and Newell (1983)

Task: Open a new file when the previous opened file was on a different disk

Device: Macintosh System 7.1



Procedure:	Operator
1. Move cursor to "Task 8"	P
2. Press mouse button	B
3. Move mouse down to "Desktop"	P
4. Release mouse button	B
5. Scan list for new drive	M
6. Move cursor to new drive	P
7. ...	...

Predicted Task Time

$$6(1.2) + 18(1.1) + 6(1.1) = 15.6 \text{ sec}$$

A major challenge in building a computational model

**Everything must be machine-readable:
The device, the task, and the cognition**

// Visual objects on the display

Visual_object: Trash
Label is Trash.
...

// Task is to delete the file "Work/file1.txt"

Task_item: T1
Name is First.
Type is delete_file.
Filename is "file1.txt".
Enclosing_directory is "Work".
Next is T2.
...

Method_for_goal: drag object using <object>, <destination>

Step 1. Look_for_object_whose Label is
<object>
and_store_under <target>.
Step 2. Point_to <target>.
Step 3. Hold_down Mouse_Button.
...

(Kieras, 1999)

A Cognitive Model Using GOMS

Card, Moran and Newell (1983)

GOMS: A methodology that involves organizing a KLM in terms of Operators, and Selection Rules. Knowledge is proceduralized, and more closely represents human procedural knowledge.

Task: Delete a folder or file.

Device: Macintosh Finder

Procedural Knowledge:

Method for goal: delete an object.

Step 1. Accomplish goal: drag object to trash.

Step 2. Return with goal accomplished.

Method for goal: drag item to destination.

Step 1. Locate icon for item on screen.

Step 2. Move cursor to item icon location.

Step 3. Hold mouse button down.

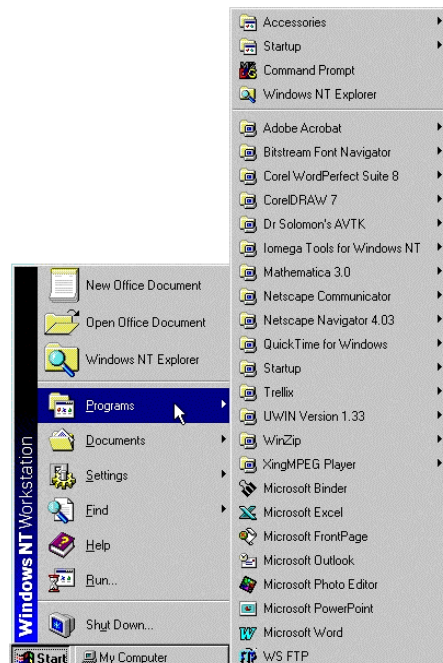
Step 4. Locate destination icon on screen.

Step 5. Move cursor to destination icon.
...

Predicts: Task time, learning time, etc.

The Research Goal: Predict Visual Search Performance

Example: How long will it take someone to find "Netscape Navigator?" How will they do it?



The Research Goal: Predict Visual Search Performance

Example: How long will it take someone to find the link to tax forms? How will they find it?



The Menu Task Modeled

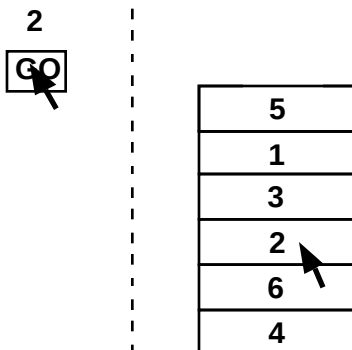
Experiment run by Erik Nilsen (1991)

Menus:

- 3, 6, or 9 numerical digits.
- Randomly re-ordered every trial.

Procedure:

- Study target digit.
- Click on "GO".
- Menu appears.
- Point to and click on target.



Progress Towards the Goal: The Menu Models

Substantial progress towards this goal has already been made.

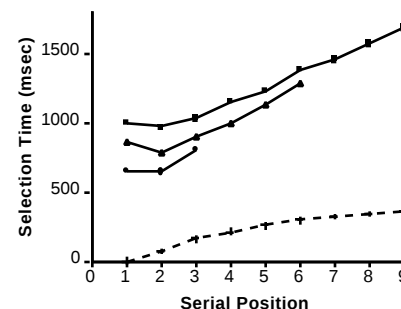
Models have been constructed that account for the perceptual, cognitive, and motor processes that people use for searching simple menus.

Aspects of these models, such as the search strategies that people use, can now be applied to more general visual search tasks.

This is finer grain, higher fidelity modeling than KLM and GOMS.

The modeling was done using the EPIC cognitive architecture.

Nilsen's Observed Data



Qualitative features of observed data:

- Shorter menus are faster.
- Slope is linear, ^a 100 msec per item.
- Serial position 1 is higher than position 2.

Slope not accounted for by mouse movement time predicted by Fitts' Law.

The EPIC Cognitive Architecture

Kieras and Meyer (1994)

Framework for building models of human performance

Resembles Model Human Processor except

- EPIC runs on a computer.
- EPIC has programmable cognitive processor
- EPIC synthesizes more of the human performance and cognitive psychology literature.

Fixed architecture

Input to EPIC:

- Task environment.
- Perceptual parameters.
- Cognitive strategy.

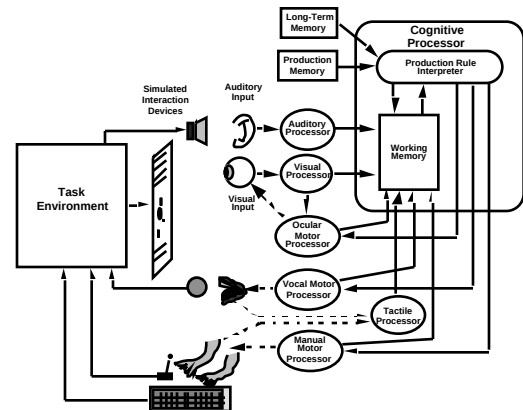
Output from EPIC:

- Trace of processing.
- Time to execute task.

All of the production rules for the Serial Processing Systematic Search menu model

```
CHOICE-START
START-CURSOR-TRACKING
LOOK-AT-GO-BOX
VERIFY-GO-BOX
MOVE-CURSOR-TO-GO-BOX
MOVE-GAZE-TO-TARGET-PRECUE
GET-TARGET-PRECUE
MOVE-GAZE-BACK-TO-GO-BOX
PRESS-MOUSE-BUTTON-TO-SHOW-MENU
PUNCH-MOUSE-BUTTON-TO-SHOW-MENU
PREPARE-POINT
FIX-GAZE-ON-TOP-ITEM
TARGET-IS-NOT-LOCATED-SACCADE-ONE-ITEM
TARGET-IS-LOCATED-MOVE-GAZE-AND-CURSOR-TO-TARGET
RELEASE-MOUSE-BUTTON-ON-TARGET
PUNCH-MOUSE-BUTTON-ON-TARGET
CLEANUP-STEP-CLEANUP
CLEANUP-TARGET-OBJECT
CLEANUP-CURRENT-ITEM
CLEANUP-PRECUE
CLEANUP-TARGET-TEXT
```

The EPIC Cognitive Architecture



The actual code of two production rules

```
;; TARGET-IS-NOT-LOCATED-SACCADE-ONE-ITEM
IF
(
  (GOAL DO MENU TASK)
  (STEP VISUAL-SEARCH)
  (WM CURRENT-ITEM IS ?OBJECT)
  (VISUAL ?OBJECT IS-ABOVE ?NEXT-OBJECT)
  (NOT (VISUAL ?OBJECT IS-ABOVE NOTHING))
  (MOTOR OCULAR PROCESSOR FREE)
)
(
  (VISUAL ?OBJECT LABEL ?NT) ;; Wait for text to appear.
  (NOT (WM TARGET-TEXT IS ?NT)) ;; It is not the target text.
)
THEN
(
  (DELD (WM CURRENT-ITEM IS ?OBJECT))
  (ADDD (WM CURRENT-ITEM IS ?NEXT-OBJECT))
  (SEND-TO-MOTOR OCULAR MOVE ?NEXT-OBJECT)
)
)

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
(TARGET-IS-LOCATED-MOVE-GAZE-AND-CURSOR-TO-TARGET
;; Decides you found the item during the visual sweep.
IF
(
  (GOAL DO MENU TASK)
  (STEP VISUAL-SEARCH)
  (WM CURRENT-ITEM IS ?TARGET-OBJECT) ;; To distinguish from the precue.
  (VISUAL ?TARGET-OBJECT LABEL ?T) ;; Wait for text to appear.
  (WM TARGET-TEXT IS ?T) ;; It is the target text
  (WM CURSOR IS ?CURSOR-OBJECT)
  (MOTOR OCULAR PROCESSOR FREE)
  (MOTOR MANUAL PROCESSOR FREE)
)
)
THEN
(
  (DELD (STEP VISUAL-SEARCH))
  (ADDD (STEP RELEASE OR PUNCH MOUSE BUTTON))
  (SEND-TO-MOTOR OCULAR MOVE ?TARGET-OBJECT)
  (SEND-TO-MOTOR MANUAL PERFORM POINT RIGHT ?CURSOR-OBJECT
?TARGET-OBJECT)
)
)
)
```

Output from EPIC when running the menu models



Serial Processing Models

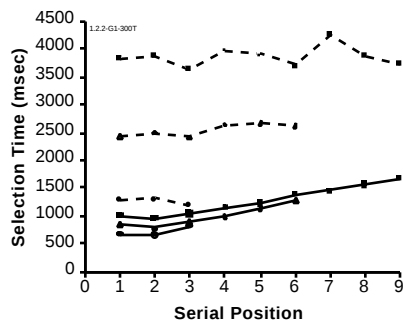
Strategy:

- Move eye to next item.
- Decide if it is the target.
 - If yes, point and click.
 - If no, move on.

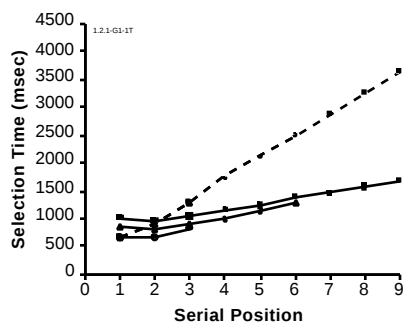
Random vs systematic search

Serial Processing Models

Random Search



Systematic Search



Parallel Processing Models

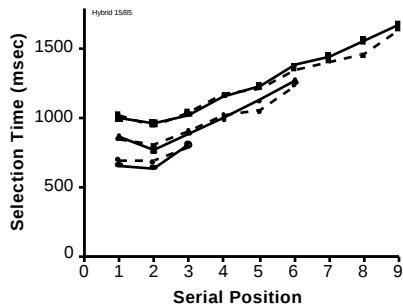
Strategy:

- Move eye around menu quickly.
- Move items to working memory in parallel.
- Watch for target to appear.
- When it appears, point and click.

Random versus systematic search

One versus three items fitting in fovea

Dual Strategy / Varying Distance Hybrid Model



Model represents belief that:

- Parallel processing of menu items.
- 50% random / 50% systematic search.
- Far from screen 85% of trials, near 15%.

Fits observed data, plausible explanation

Another set of models were built to explain Nilsen's ordered menu data.

Major Conclusions of the Randomly Ordered Models Have Been Validated With Eye Movement Data

People process menu items in parallel

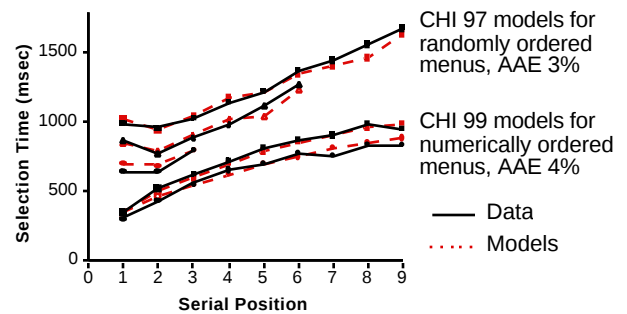
- Average saccade length of 2.21 menu items [Aaltonen et al., CHI 98]
- Too few fixations for serial processing. [Byrne et al., CHI 99]

Mixture of random and systematic search

- Top-to-bottom scan paths on some trials; more random scan paths on other trials; mixture of both on other trials. [Aaltonen et al., CHI 98]
[Byrne et al., CHI 99]

The models can also be fine-tuned based on eye movement data that is now available.

New Insights into Menu Search



People do not serially process one menu item at a time, as has been proposed by many researchers.

Search is both random

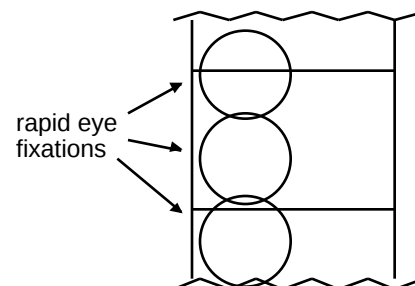
The top menu item takes longer to select because of an interaction between more than one item fitting into the fovea and random search.

The menu length effect results from the random component in the search strategy.

Implications for menu design guidelines

The models propose that people make a high speed "maximally efficient foveal sweep."

Such a search, if made down the left edge of a menu, would miss everything but the left-justified keywords, such as if someone is looking for "Spelling" in the menu below:



For high speed search, left-justified is more important than grammatically similar.

Implications for fundamental search strategies to be used in a predictive search tool

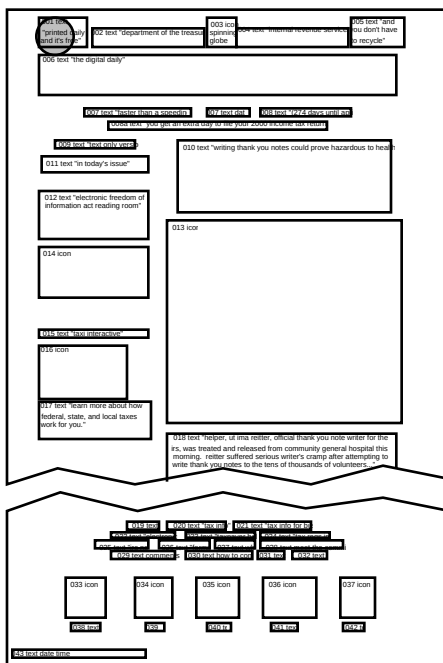
The Research Goal: Predict Visual Search Performance

Example: How long will it take someone to find the link to tax forms? How will they find it?



How the tool will work

The tool will provide a machine-readable description of the screen layout



The Research Goal: Predictive Modeling of Visual Search

A tool that would be incorporated with screen layout design software



Future tool (left) and current download time predictions in Claris HomePage 3.0 (right).

INPUT to the tool: a screen layout and a task.

OUTPUT from the tool: how long a person would take to accomplish the task.

The predictive visual search modeling could also be incorporated into future versions of GLEAN.

How the tool will work (continued)

An actual EPIC model will be automatically constructed and run.

Impose preset visual recoding delays for each type of item.

Impose a general visual search strategy that includes:

- Moving the eyes as fast as possible, without waiting to evaluate each item.
- Maximally efficient foveal sweeps.
- Parallel consideration of items.
- Decide where to move eyes next based on proximity, whether or not the items have been in the fovea yet, and primary features such as size, shape, and color.
- Occasional random jumps.
- Halt the search as soon as the target text or the icon corresponding to the target text is found.

Previous research demonstrates the research goal is plausible

Understanding Cognitive Information Engineering (UCIE)

- Lohse (1993)
- System that predicts the time required to answer a specific question based on information presented in a line graph, bar graph, or table.
- Timing parameters to eye fixations and other component processes.
- UCIE predicts total task execution time by summing the time required for all component tasks.
- Only built and validated for graphs and tables, and does not predict performance for more general visual search tasks that arise when using a computer.

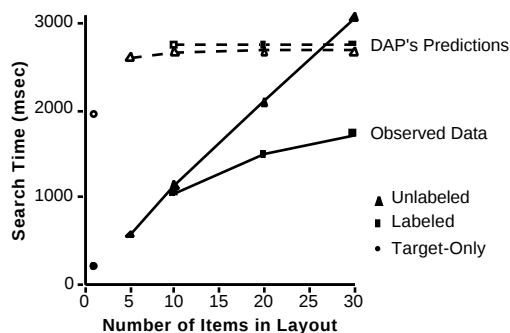
The Display Analysis Program (DAP) (Tullis, 1988)

- Predicts average search time for any alphanumeric computer screen layout

Visual search predictions need to incorporate the search strategies

DAP does not predict that adding group labels will speed search:

X5X	REJ		X1X	ZAM		X2X	DAG
X3X	PEW			POF			MUG
	TAH			TIK			MOL
	KOD			KIC			POK
	ZIN			BER			TOR
	WIW						
X4X	BUK		X5X	BEY		X6X	LIF
	COG			SEH			REH
	TAW			LUN			TEX
	HOJ			YID			PIG
	TES			REJ			DEK



Display Analysis Program (DAP) (Tullis, 1988)

To: Atlanta, GA

From: Asheville, NC Fares: First: 92.57 Coach: 66.85
 Departs: 7:20a Arrives: 8:05a Flight: PI 299
 Departs: 10:10a Arrives: 10:55a Flight: PI 203
 Departs: 4:20p Arrives: 5:00p Flight: PI 259

From: Austin, TX Fares: First: 263.00 Coach: 221.00
 Departs: 8:15a Arrives: 11:15a Flight: EA 530
 Departs: 8:40a Arrives: 11:39a Flight: DL 212
 Departs: 2:00p Arrives: 5:00p Flight: DL 348
 Departs: 7:15p Arrives: 11:26p Flight: DL 1654

From: Baltimore, MD Fares: First: 209.00 Coach: 167.00
 Departs: 7:00a Arrives: 8:35a Flight: DL 1767
 Departs: 7:50a Arrives: 9:32a Flight: EA 631
 Departs: 8:45a Arrives: 10:20a Flight: DL 1610
 Departs: 11:15a Arrives: 12:35p Flight: EA 147
 Departs: 1:35p Arrives: 3:10p Flight: DL 1731
 Departs: 2:35p Arrives: 4:16p Flight: EA 141

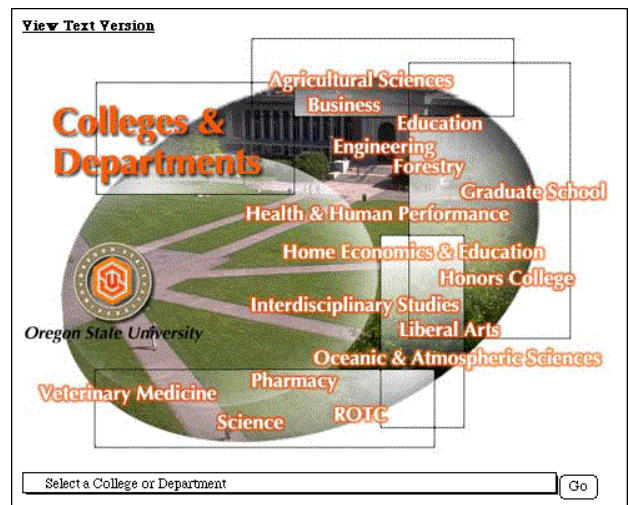
~ 34.1 62.6 12 5.6 101 7.50

How it works

- Takes alphanumeric screen as input.
- Automatically computes measurements of overall density, local density, number of groups, group size, number of data items, layout complexity.
- Predicts mean search time based on these raw measurements.

A Predictive Tool Might Have Helped to Improve This Design

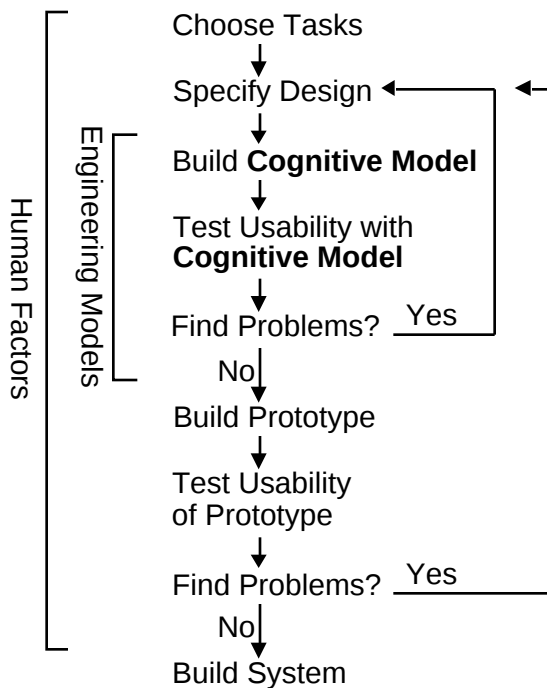
Task: Find the department of psychology.



[<http://www.orst.edu/mc/coldep/coldep.htm> 12/6/98]

The predictive tool would probably predict a long search time for this task with this layout.

Human Factors + Engineering Models = How to Build Usable Systems



Four important points about cognitive modeling and HCI:

1. Aspects of human performance can be predicted using cognitive modeling.
2. A cognitive model is built based on analysis of the task, device, and the cognitive (including perceptual, memory, and motor) processing necessary to accomplish the task using the device.
3. Methodologies exist—and continue to be developed—for building predictive engineering models of human performance, models that can be used to predict the usability of computer interfaces before conducting the essential but expensive user observation studies.
4. Cognitive architectures are particularly useful for building detailed analytical and predictive models. Cognitive architectures represent the “hardware” of human performance, the invariants, and provides an excellent framework for building predictive models.