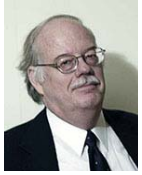
Model-based Evaluation



David E. Kieras

University of Michigan

Powered by Naval Research TM/SM

Introduction to the Session

Using cognitive architectures in user interface design is a form of usability evaluation based on models of the user.

This session presents the basics of Model-Based Evaluation, and the critical properties of models that are suitable for this purpose.

Approaches to Developing Usable Systems

Ensuring Usability: Standard Human Factors Process

What's Good and Bad about
Standard Human Factors Process

Ensuring Usability: The Engineering Model Approach

Engineering Model Process

Two Traditions in Engineering Models

Models are Simulations of
Human-Computer Interaction

Models in Science versus Design

Models in Science

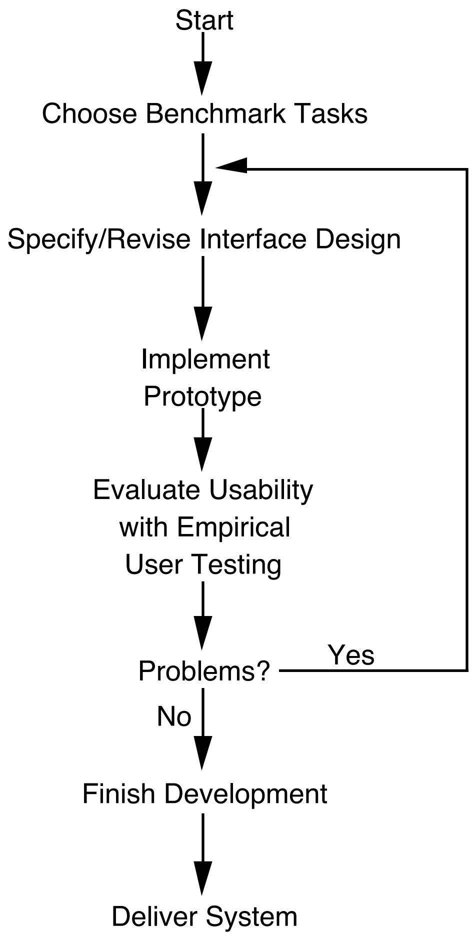
Models in Design

Ensuring Usability: Standard Human Factors Process

Early use of guidelines, empirical user testing of prototypes.

Identify problems that impair learning or performance.

Compare user performance to a specification.



What's Good and Bad about Standard Human Factors Process

What's good

Definitely works if used thoroughly enough!

Much accumulated experience.

What's bad

Slow and expensive.

- Research effort in HCI: tighten the iterative design loop.
- But, there are unavoidable time and cost demands:

  E.g., in expert domains, subjects are too few and their time is too valuable for thorough user testing to be done.

  E.g. design to support transfer of training.

No systematic way to accumulate or analyze design experience.

- No representation of how the design "works" to ensure usability.
- Any change to product or user's task might produce a new usability situation - what aspects of design are still valid?
- Usability knowledge resides heavily in the intuitions of user interface experts.

  Cf. Master Builders during the Middle Ages.

  Guidelines are usually too vague for non-experts.

Only psychology used: Experimental methodology.

- How to run experiments on human behavior and draw valid conclusions.
- But surely something else would be useful from 100+ years of research!

Ensuring Usability: The Engineering Model Approach

Use model predictions instead of user testing.

Engineering model process:

1. Describe the interface design in detail.
2. Build a model of the user doing the task.
3. Use the model to predict execution or learning time.
4. Revise or choose design depending on prediction.

Get usability results before implementing prototype or user testing.

A critical issue in expert domains, transfer of training

Engineering model allows more design iterations.

Doesn't guarantee perfection or completeness.
Provides some low-cost, highly informative iterations.

Model summarizes the interface design from the user's point of view.

Represents how the user gets things done with the system.
Components of model can be reused to represent design of related interfaces.

***But*, current models can only predict a few aspects:**

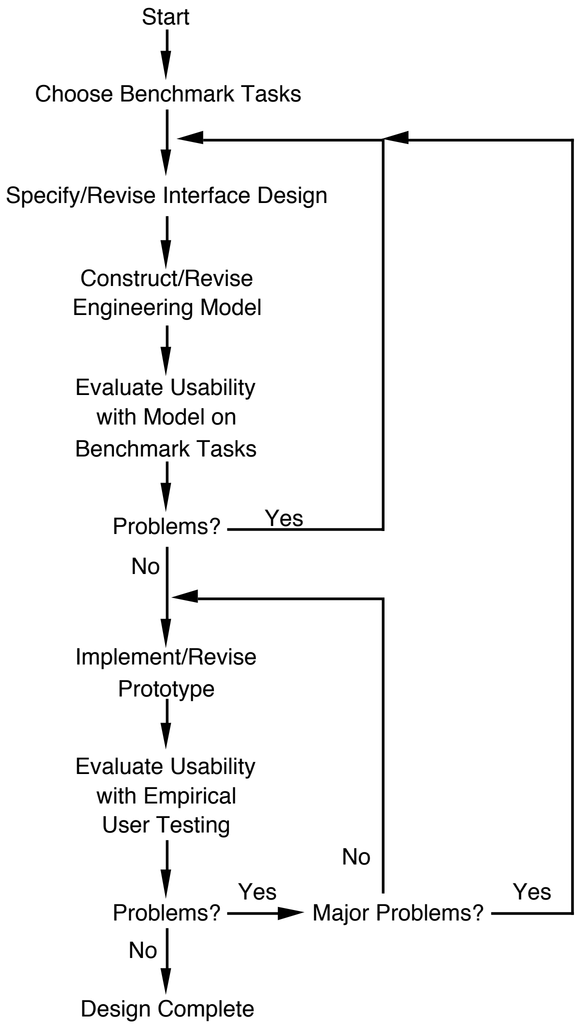
Time required to execute specific tasks.
Relative ease of learning of procedures, consistency effects.

Some user testing still required

Assess aspects not dealt with by an analytic model.
Protection against errors, oversights, in the analysis.

Engineering Model Process

Use model first, user testing for final check, loop back if necessary.


Start
Choose Benchmark Tasks
Specify/Revise Interface Design
Construct/Revise Engineering Model
Evaluate Usability with Model on Benchmark Tasks
Problems?
Yes
No
Implement/Revise Prototype
Evaluate Usability with Empirical User Testing
Problems?
Yes
Major Problems?
No
Yes
No
Design Complete

Two Traditions in Engineering Models

The Human Factors Tradition.

Represent task in terms of human performance elements.

- Parameters for task element completion time, error rate.
- Analysis/prediction of workload, task timelines, etc.

Task performance predicted by calculation or simulation runs of the task.

- Prime example: task network models.

Main Strength and Weakness

- Developed and used in context of practical design and analysis.
- Lack modern, systematic theoretical and scientific base.

Implication:

- Models of human task performance work well enough to justify more complete development and fielding.

The Cognitive Psychology Tradition.

Computational architectures for human cognition and performance.

- Many approaches in psychology and AI.

Usability predicted by performance of a simulation model for the task built with the architecture - a simulated human doing the task.

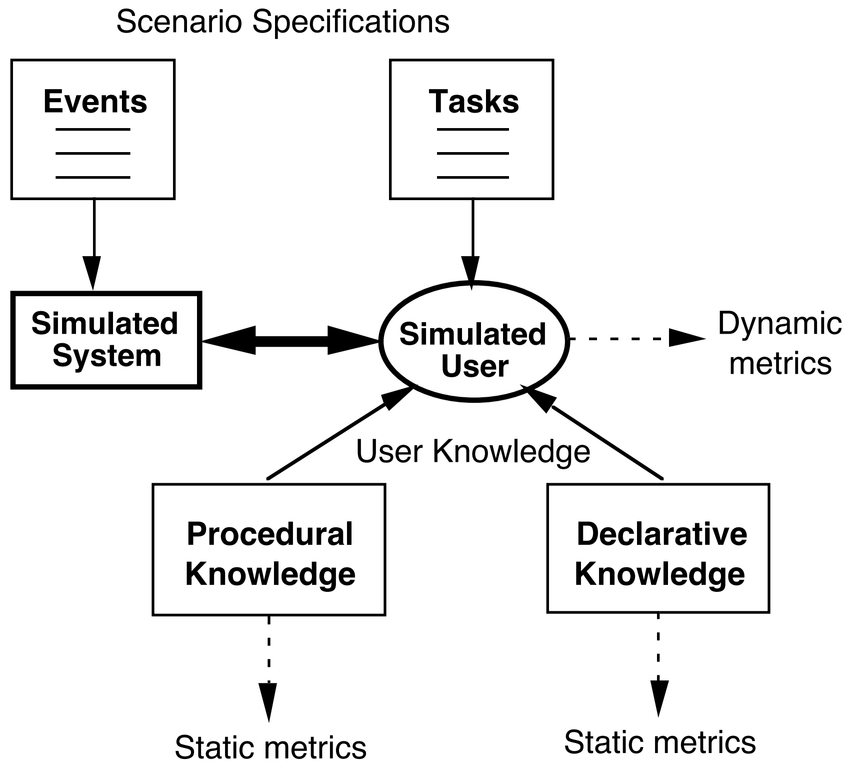
Main Strength and Weakness.

- Strong scientific basis.
- Little practical experience.

Challenge is to combine the best from both traditions.

Models are Simulations of Human-Computer Interaction

Differences in how each type of model fits this picture.



Procedural knowledge - how-to procedures; executable.

Declarative knowledge - facts, beliefs; reportable.

Models in Science versus Design

In both:

Constructing a model or theory of what humans will do in a situation.

Questions in both:

What does the model predict?
What information do we have to put in before we get a prediction out?
How accurate is the prediction?

How are they different?

Models in Science

How good is the model as an account of what humans really do?

Usually in a laboratory experiment.

Comparison of model to empirical data on human behavior.

"Testing" the model.

Model failures can be especially informative scientifically; not really a disaster at all.

Scientific standards on quality of the model, data, and the comparison are quite demanding.

Model must be a clear and precise scientific statement.

Data must be collected using well-controlled and documented procedures, and analyzed with proper statistical procedures.

Comparisons must be quantified and rigorous.

Real-world situations often impose practical barriers to scientific data-collection, so standards are hard to meet outside the laboratory.

Models in Design

What does the model say about what people will do when they use the system we are building?

Will this design work well enough?
What alternative designs might be better?

Can't test model against data if the system is not built yet!

Even if built, “good” data can be impractical to obtain.

Validating model against data is not part of the “normal” use of a model.

It is supposed to be already validated.

But calibration of a component or parameter estimation might be needed.

If you have enough data about the actual performance of a prototype, why spend effort on the model?

Model is known to be limited and approximate, but can still be useful to design effort.

But only if it is understood what model actually says, and what its limits are.

The Critical Role of Psychological Constraints

Modeling System Must Provide Psychological Constraints

Example of Psychological Constraints

Cognitive versus Perceptual-Motor Constraints

Modeling System Must Provide Psychological Constraints

Modeling human performance requires representing human psychology.

Human abilities, limitations.

Time course, trajectory of behavior.

Evaluation of a proposed design must be a routine design activity, not a scientific research project.

Most design problems should be engineering design:

- Use existing interface concepts to best deliver a set of functionality.
- Use human performance in the same way as other engineering performance criteria.

Need to be able to build models without inventing psychological theory.

- Designer rarely has the time, training, or resources.

Modeling system must provide the psychology automatically.

Constrain what the model can do, so modeler can focus on design questions, not psychological basics.

If model can be programmed to do any task at any speed or accuracy, something's wrong!

Of course, science is never complete, but need to include as much as possible.

Example of Psychological Constraints

Pointing and typing take documented amounts of time.

Model of the human should not be able to point instantly or type at implausible speed.

Can't point with the mouse and type text at the same time.

Constraint on the hands.

Modeling system should make it impossible for this to happen.

Modeling system should generate the timing involved in completing both tasks, such as time to move hands between keyboard and mouse.

Advantage of computational tools for modeling:

Can represent the subtle details and interactions, and apply them as needed.

Much more easy than hand work or general-purpose tools (e.g. spreadsheet).

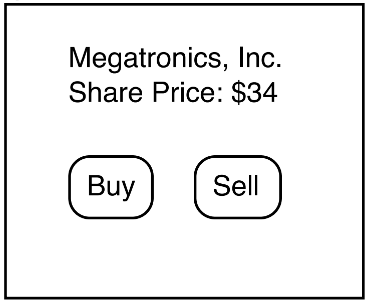
But of course, you have to learn and use a complicated piece of software.

Cognitive versus Perceptual-Motor Constraints

What dominates a task?

Heavily cognitive tasks: Human “thinks” for most of the time.

- E.g. Online stock trading system.



Megatronics, Inc.
Share Price: $34

But many HCI tasks are dominated by perceptual-motor activity.

A steady flow of physical interaction between human and computer.

Time required depends on:

- Computer's behavior - determined by the design.
- Human characteristics - many well-documented properties.

Implication

Modeling perceptual-motor aspects is often practical, useful, and relatively easy.

- Many modeling systems can capture it.

Modeling purely cognitive aspects of complex tasks is often difficult, open-ended, and requires research resources.

- Try to avoid with good task analysis to identify design requirements.
  - E.g. Make required information easily available.
  - E.g. Choose functionality that eliminates difficult cognitive tasks.

Brief History of Constraints in Psychological Theory

1879-1950s. Verbal and Mathematical Theory

1960s. Box Models of Human Information Processing

1970s. Computer Simulations of Box Models

1980s. Cognitive Architectures

1990s. Embodied Cognitive Architectures

State of the Scientific Literature about Constraints

1879-1950s. Verbal and Mathematical Theory

Verbal theory.
Theory expressed in prose, hopefully very carefully.
Difficult to make rigorous deductions and predictions.

Mathematical models.
Usually based on probability theory.
Works well for very simple phenomena.
Breaks down for complex, sequential, or qualitative phenomena.

1960s. Box Models of Human Information Processing

Flowcharts showing information flow between assumed components, or stages of processing - the boxes.

Basic idea: humans processed information in stages, organized around various subsystems.

E.g. Visual perception, short-term memory, long-term memory, decision-making.

Problem is not the representation, but the lack of rigor in what the boxes dc

Each box behaved according to verbal or mathematical theory.

No restrictions on what each box did, or requirement that it was well-defined.

1970s. Computer Simulations of Box Models

More flexibility than mathematical models, more rigor than verbal models

Symbolic computation allowed expression of qualitative ideas.

E.g. semantics, plans, strategies, etc.
Beginning of interaction with Artificial Intelligence research.

Human Operator Simulator (HOS, late 1970s)

Describe information processing stages in a task procedure language.

Apply appropriate math model of an elementary psychological phenomenon to predict time and accuracy for that stage.

- “Micro models”
- Obsolete now, difficult to apply.

Right idea, but ahead of its time.

1980s. Cognitive Architectures

Cognitive Architecture: A fixed set of components and mechanisms tha represent basic human abilities and limitations.

Architecture represents many constraints based on the scientific literature.

Task strategy can only be represented and implemented in certain ways allowed by the architecture.

Architecture supplies actual performance timing and effects.

In scientific work, architectures are relatively easy to understand - as parsimonious as possible.

Scientific standard is that nothing must be hidden - all spelled out.

Outside the scientific research literature, quite a jumble.

GOMS models are based on simplified architectures.

Less flexible than either full-fledged architectures or box-model approaches.

But considerably easier to work with.

1990s. Embodied Cognitive Architectures

Cognitive architectures began to incorporate perceptual and motor systems and their constraints.

Earlier: Cognition was "disembodied."

Outside world directly known to cognition which could directly act on the world.
No constraints from perceptual or motor mechanisms.

Now: Cognition has to work through the peripheral systems, and these limit what and when information is available and when actions can be taken.

Extremely powerful constraints.

State of the Scientific Literature about Constraints

Scientific psychology ("Experimental Psychology") has been underway since 1879.

Tremendous collection of empirical data on human abilities.

However, very spotty coverage due to quirks of the research culture.

Basic research:

Strong tendency to avoid applied issues, so mostly artificial tasks used.
Pursuit of small-scale theoretical ideas, not comprehensive theory.
Emphasis on "odd" phenomena, rather than mainstream phenomena.
Pursuit of qualitative phenomena and hypothesis testing, rather than determining quantitative functions and parameters of human performance.

Applied research (especially in Human Factors):

"Putting out fires" instead of developing science base.
Extremely specific experiments or user tests with little generalizability.
Acceptance of "quick & dirty" rules of thumb, instead of developing theory.

For every relevant phenomenon about human performance, there is a haphazard mixture of useful results, pointless results, and missing results.

Future progress should be better:

Needs of cognitive architectures should guide much more focused basic research.
Trying to apply cognitive architectures shows where theory is lacking.

Modeling Approaches

Three Current Approaches

Task Network Models - before detailed design.

Cognitive Architecture Models - packaged constraints.

GOMS models - relatively simple and effective.

Three Current Approaches

Differ in constraints, detail, when to use.
More about each one in remaining talks.

Task Network Models - before detailed design.

Cognitive Architecture Models - packaged constraints.

GOMS models - relatively simple and effective.

Task Network Models - before detailed design.

Essentially, stochastic PERT charts.
PERT (U.S. Navy, 1950s)
SAINT (Chubb, 1981)

Connected network of tasks:
Connection means that one is a prerequisite of the other.
Both serial and parallel execution of tasks.
Final time to a complete computed from chain of serial and parallel tasks.

Tasks can be any mixture of human and machine tasks.

Each task characterized by a distribution of completion times, and arbitrary dependencies and effects:
Completion time distribution can be from any source - e.g. SME opinion.
Arbitrary tests for prerequisite conditions
Arbitrary effects on state of simulation or other tasks
Workload metrics can be computed and aggregated.

Get results with Monte-Carlo simulation runs.

Can be used before detailed design, because of extreme generality:
If no built-in human performance constraints, no dependence on design details.

Cognitive Architecture Models - packaged constraints.

A Cognitive Architecture:

A fixed set of components and mechanisms that represent basic human abilities and limitations - task independent.

- Based on scientific work on human cognition and performance.
- Change only if pervasive and hoped to be permanent.

The architecture is “programmed” with a strategy to perform specific tasks.

- Architecture provides constraints on the form and content of the strategy.

Architecture + specific strategy = a model of a specific task.

To model a specific task:

Do a task analysis to arrive at the human’s strategy for doing the task.

“Program” the architecture with its representation of the task strategy.

Run the model using some task scenarios.

Result is predicted behavior and time course for that scenario and task strategy.

Goal is comprehensive psychological theory, so these are quite complex; used mostly in a research settings.

Term “cognitive architecture” now used outside psychological theory - especially in new “cognitive system” movement.

E.g. a framework for building “smart” software.

Here, restricted to software systems that support models of actual humar cognition and performance.

GOMS models - relatively simple and effective.

A key model-based methodology based on simplified cognitive architectures.

Definition of GOMS model:

An approach to describing the knowledge of procedures that a user must have in order to operate a system

- Proposed by Card, Moran, & Newell (1983)

Goals - what goals can be accomplished with the system
Operators - what basic actions can be performed
Methods - what sequences of operators can be used to accomplish each goal
Selection Rules - which method should be used to accomplish a goal

Different levels of GOMS models - more later.

Well worked out, quite practical, but limited due to simplifications.

Often in the "sweet spot" - lots of value for modest modeling effort.

Summary

Models of the user can partially eliminate the need for empirical user testinc of interface designs, with potential savings in cost and time.

Such models need to include critical features of human psychological limitations and abilities in the form of constraints on what the model can dc

Three general approaches to models are task networks, cognitive architectures, and GOMS models, which will be described in later sessions.