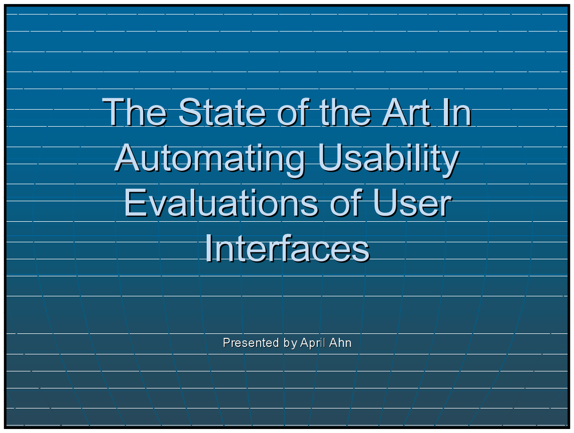
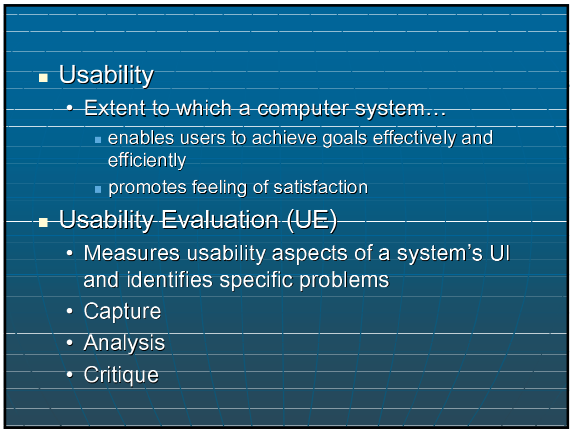
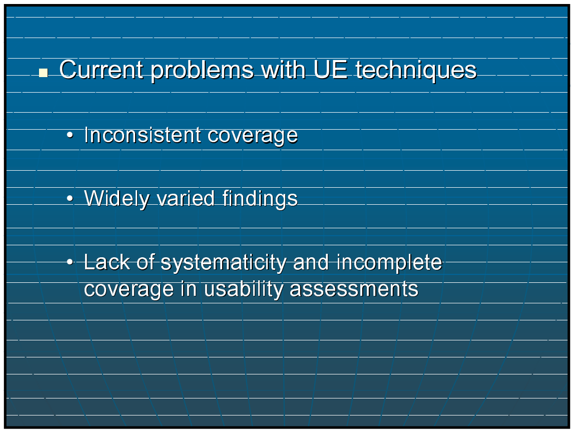


# The State of the Art In Automating Usability Evaluations of User Interfaces

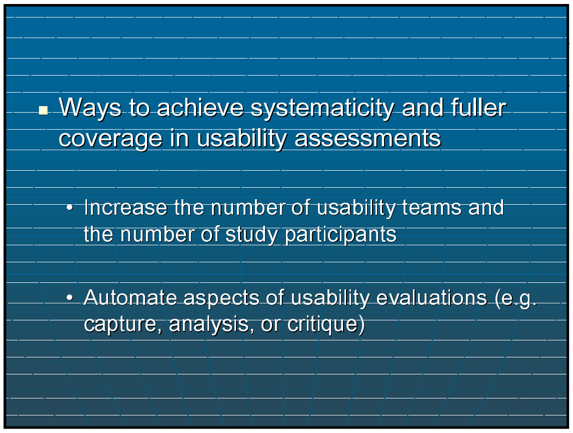
Presented by April Ahn



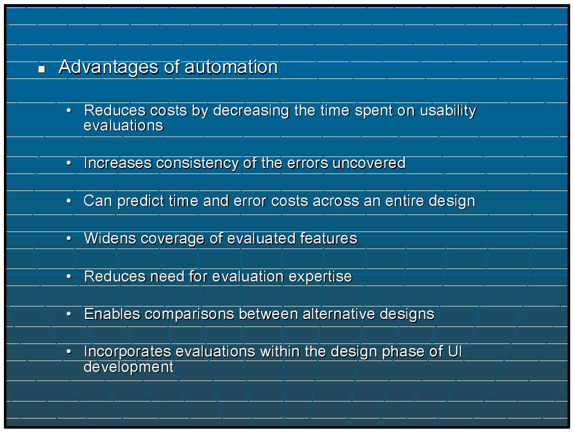
- Usability
  - Extent to which a computer system…
    - enables users to achieve goals effectively and efficiently
    - promotes feeling of satisfaction
- Usability Evaluation (UE)
  - Measures usability aspects of a system's UI and identifies specific problems
  - Capture
  - Analysis
  - Critique



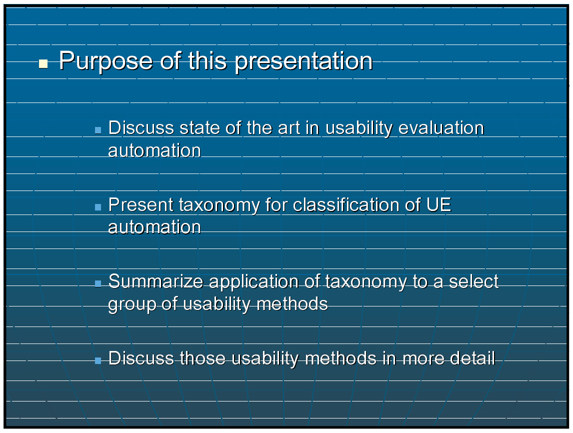
- Current problems with UE techniques
  - Inconsistent coverage
  - Widely varied findings
  - Lack of systematicity and incomplete coverage in usability assessments



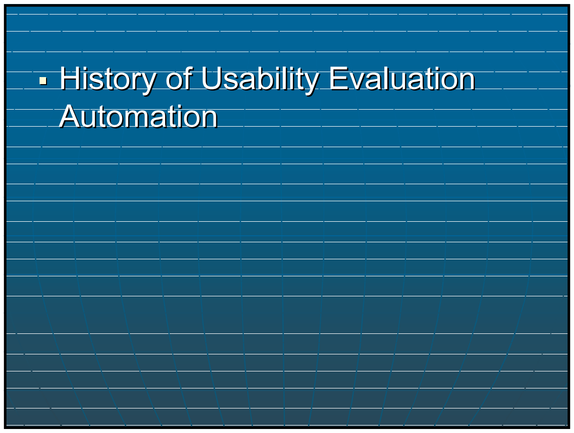
- Ways to achieve systematicity and fuller coverage in usability assessments
  - Increase the number of usability teams and the number of study participants
  - Automate aspects of usability evaluations (e.g. capture, analysis, or critique)



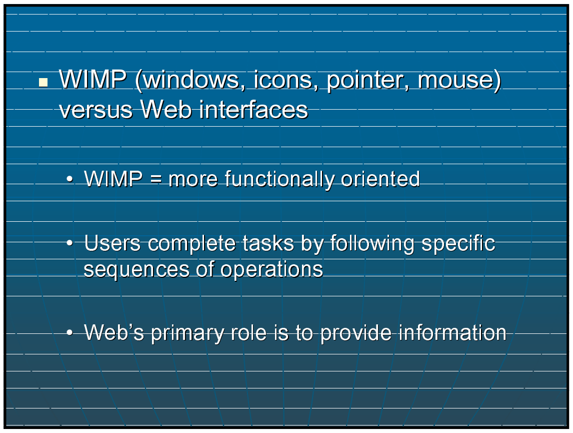
- Advantages of automation
  - Reduces costs by decreasing the time spent on usability evaluations
  - Increases consistency of the errors uncovered
  - Can predict time and error costs across an entire design
  - Widens coverage of evaluated features
  - Reduces need for evaluation expertise
  - Enables comparisons between alternative designs
  - Incorporates evaluations within the design phase of UI development



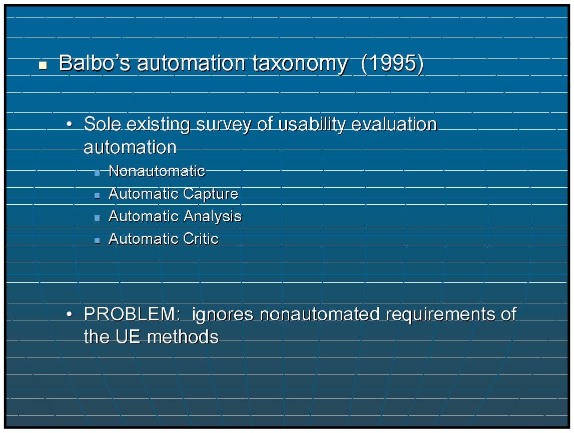
- Purpose of this presentation
    - Discuss state of the art in usability evaluation automation
    - Present taxonomy for classification of UE automation
    - Summarize application of taxonomy to a select group of usability methods
    - Discuss those usability methods in more detail



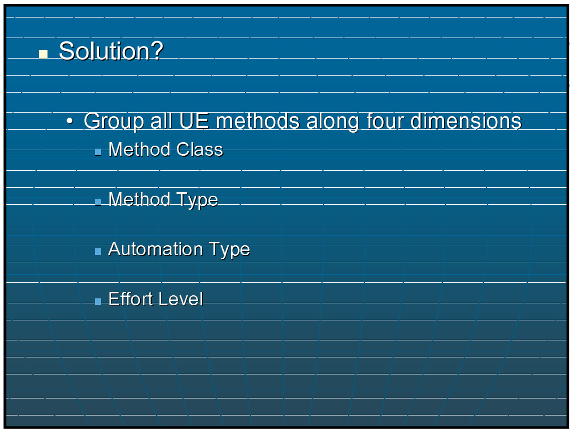
- History of Usability Evaluation Automation



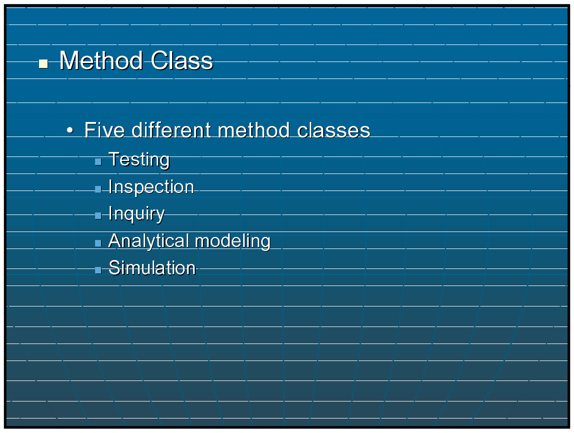
- WIMP (windows, icons, pointer, mouse) versus Web interfaces
  - WIMP = more functionally oriented
  - Users complete tasks by following specific sequences of operations
  - Web's primary role is to provide information



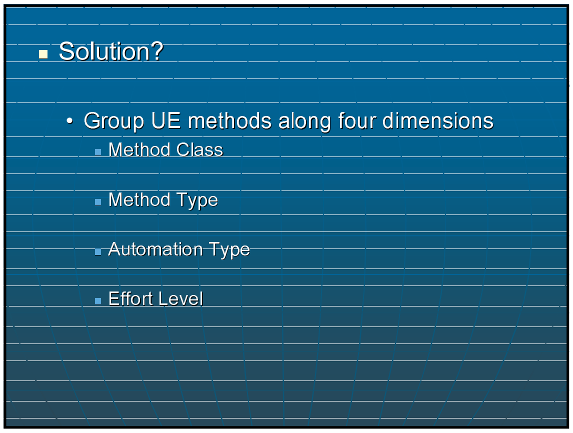
- Balbo's automation taxonomy  (1995)
  - Sole existing survey of usability evaluation automation
    - Nonautomatic
    - Automatic Capture
    - Automatic Analysis
    - Automatic Critic
  - PROBLEM:  ignores nonautomated requirements of the UE methods



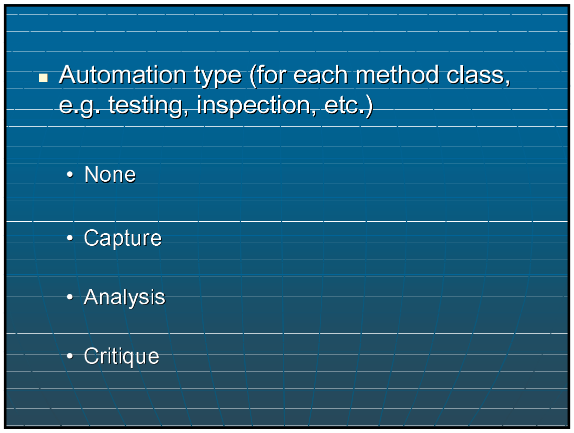
- Solution?
  - Group all UE methods along four dimensions
    - Method Class
    - Method Type
    - Automation Type
    - Effort Level



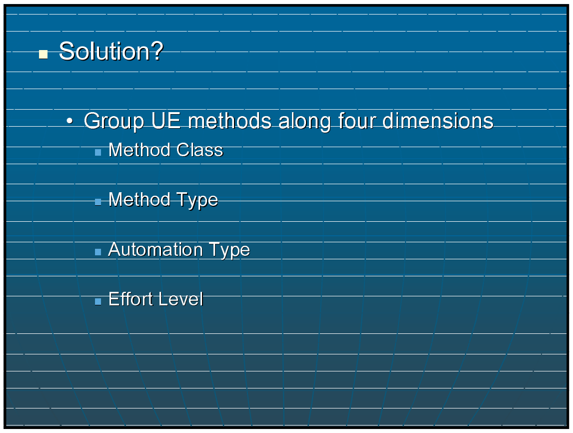
- Method Class
  - Five different method classes
    - Testing
    - Inspection
    - Inquiry
    - Analytical modeling
    - Simulation



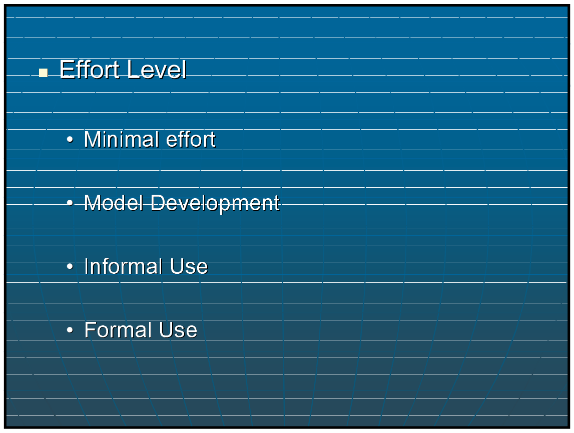
- Solution?
  - Group UE methods along four dimensions
    - Method Class
    - Method Type
    - Automation Type
    - Effort Level



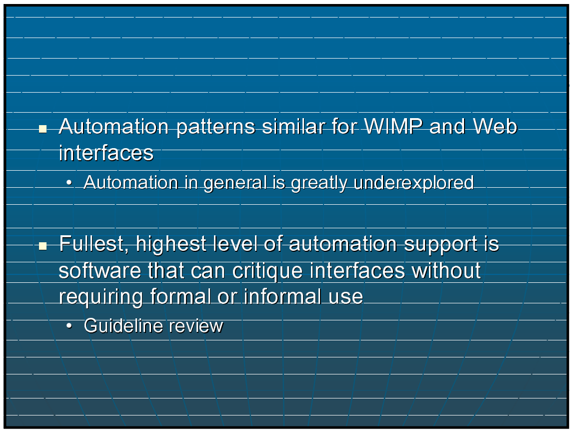
- Automation type (for each method class, e.g. testing, inspection, etc.)
  - None
  - Capture
  - Analysis
  - Critique



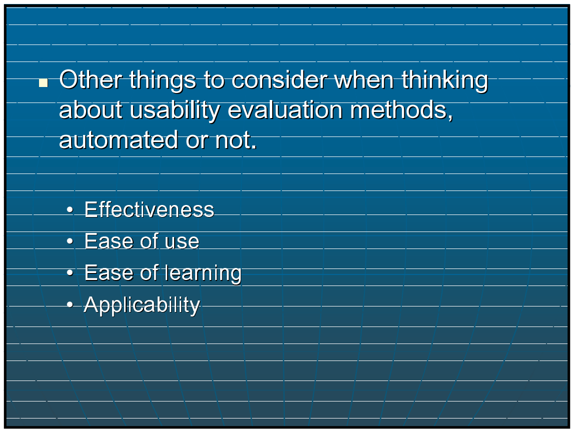
- Solution?
  - Group UE methods along four dimensions
    - Method Class
    - Method Type
    - Automation Type
    - Effort Level



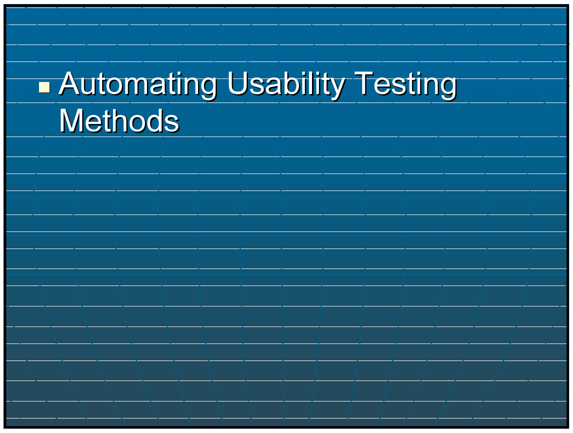
- Effort Level
  - Minimal effort
  - Model Development
  - Informal Use
  - Formal Use



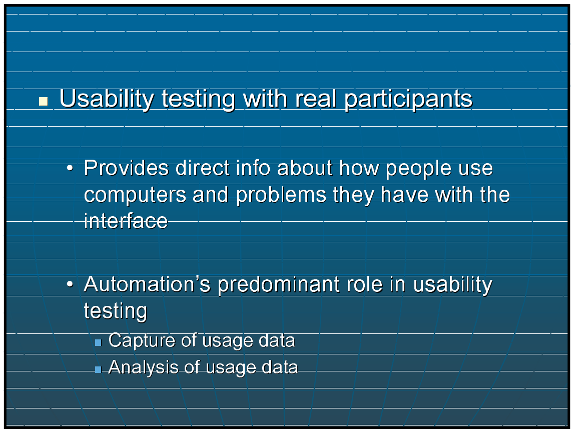
- Automation patterns similar for WIMP and Web interfaces
  - Automation in general is greatly underexplored
- Fullest, highest level of automation support is software that can critique interfaces without requiring formal or informal use
  - Guideline review



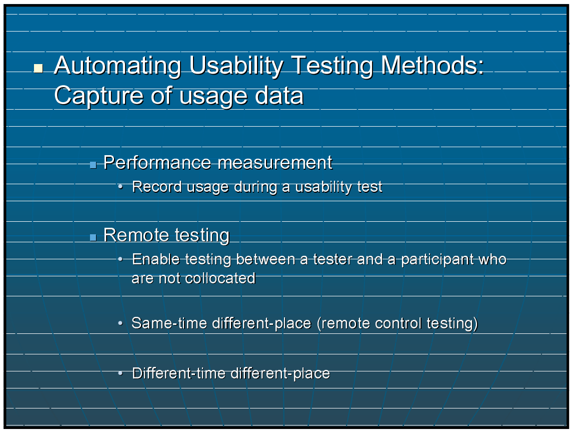
- Other things to consider when thinking about usability evaluation methods, automated or not.
  - Effectiveness
  - Ease of use
  - Ease of learning
  - Applicability



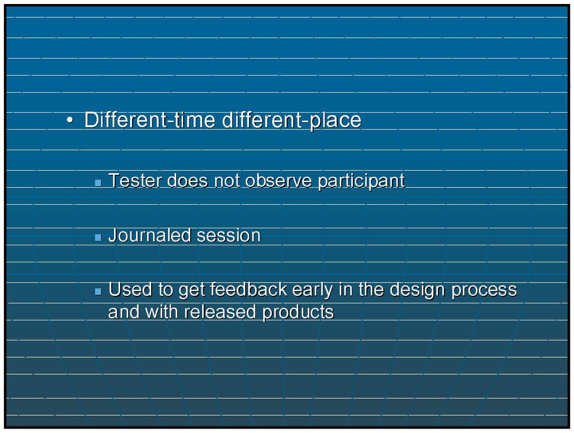
- Automating Usability Testing Methods



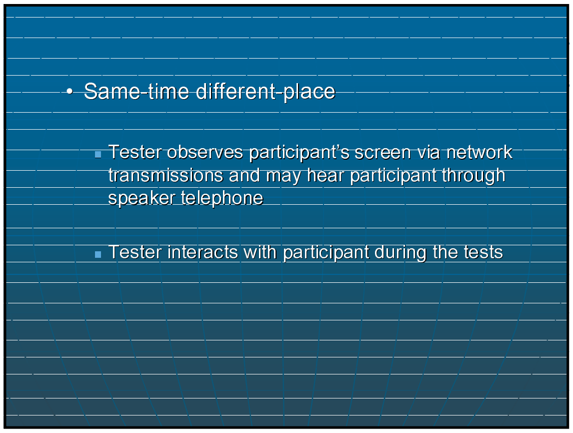
- Usability testing with real participants
  - Provides direct info about how people use computers and problems they have with the interface
  - Automation’s predominant role in usability testing
    - Capture of usage data
    - Analysis of usage data



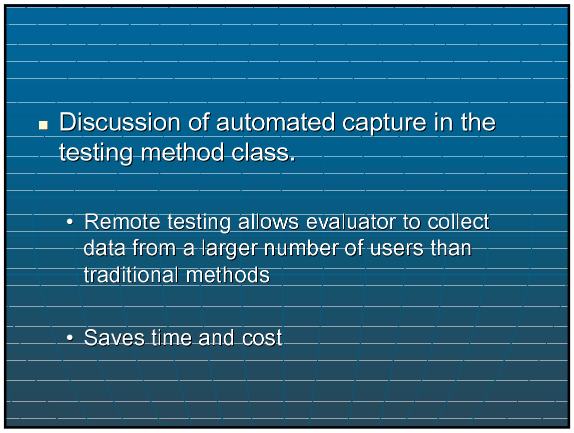
- Automating Usability Testing Methods: Capture of usage data
  - Performance measurement
    - Record usage during a usability test
  - Remote testing
    - Enable testing between a tester and a participant who are not collocated
    - Same-time different-place (remote control testing)
    - Different-time different-place



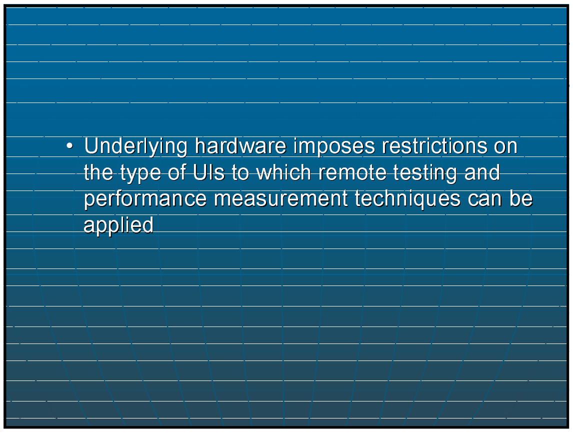
- Different-time different-place
  - Tester does not observe participant
  - Journaled session
  - Used to get feedback early in the design process and with released products



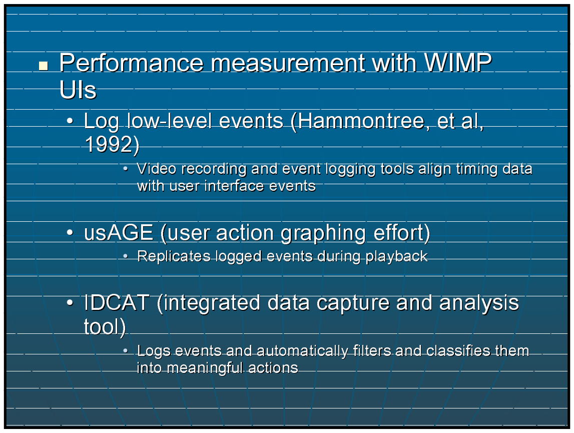
- Same-time different-place
  - Tester observes participant’s screen via network transmissions and may hear participant through speaker telephone
  - Tester interacts with participant during the tests



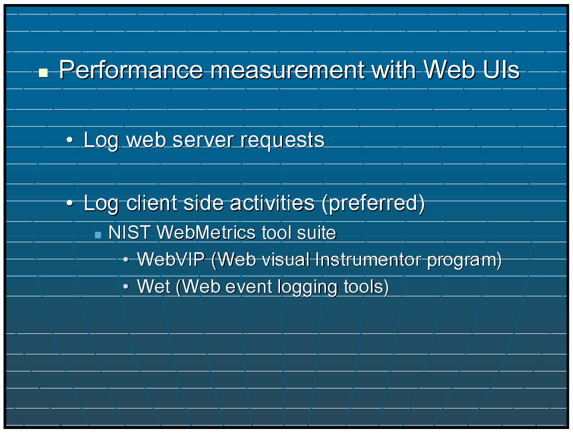
- Discussion of automated capture in the testing method class.
  - Remote testing allows evaluator to collect data from a larger number of users than traditional methods
  - Saves time and cost



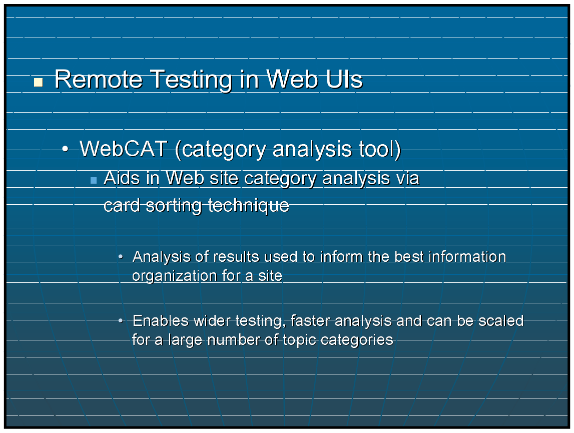
- Underlying hardware imposes restrictions on the type of UIs to which remote testing and performance measurement techniques can be applied



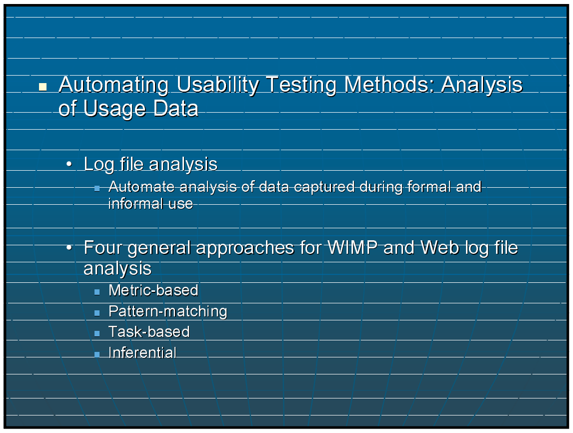
- Performance measurement with WIMP UIs
  - Log low-level events (Hammontree, et al, 1992)
    - Video recording and event logging tools align timing data with user interface events
  - usAGE (user action graphing effort)
    - Replicates logged events during playback
  - IDCAT (integrated data capture and analysis tool)
    - Logs events and automatically filters and classifies them into meaningful actions



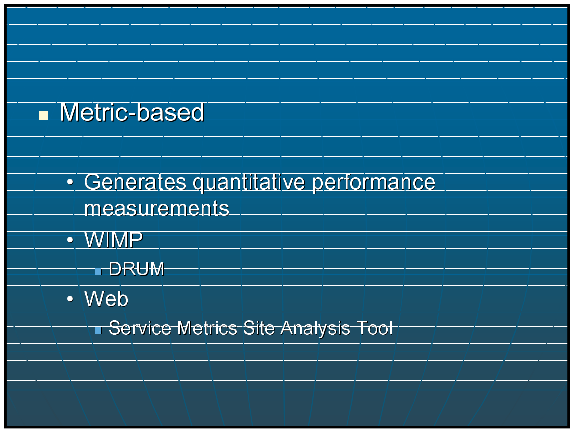
- Performance measurement with Web UIs
  - Log web server requests
  - Log client side activities (preferred)
    - NIST WebMetrics tool suite
      - WebVIP (Web visual Instrumentor program)
      - Wet (Web event logging tools)



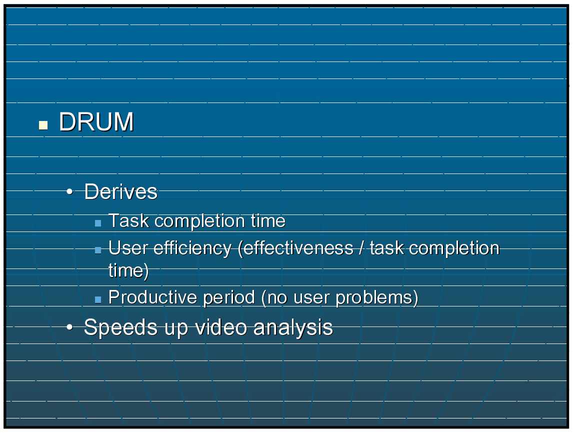
- Remote Testing in Web UIs
  - WebCAT (category analysis tool)
    - Aids in Web site category analysis via card sorting technique
      - Analysis of results used to inform the best information organization for a site
      - Enables wider testing, faster analysis and can be scaled for a large number of topic categories



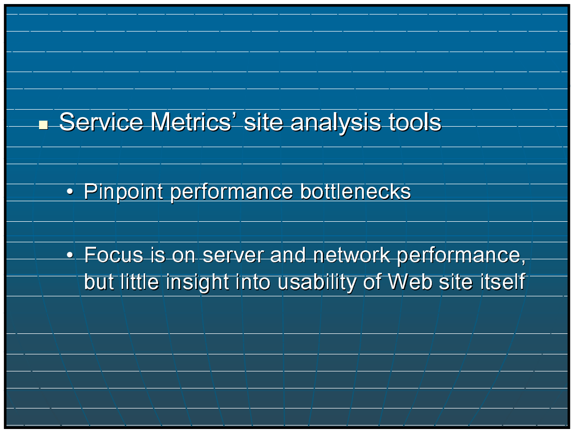
- Automating Usability Testing Methods: Analysis of Usage Data
  - Log file analysis
    - Automate analysis of data captured during formal and informal use
  - Four general approaches for WIMP and Web log file analysis
    - Metric-based
    - Pattern-matching
    - Task-based
    - Inferential



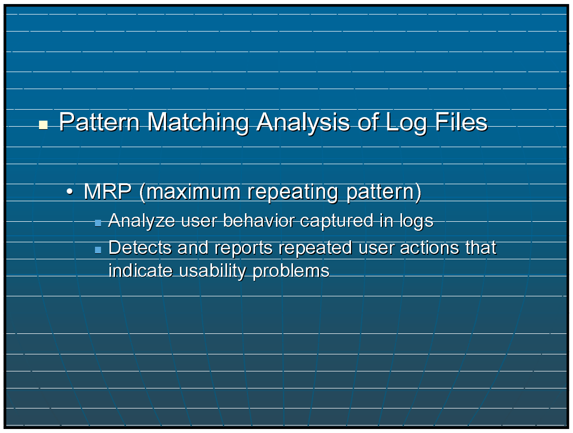
- Metric-based
  - Generates quantitative performance measurements
  - WIMP
    - DRUM
  - Web
    - Service Metrics Site Analysis Tool



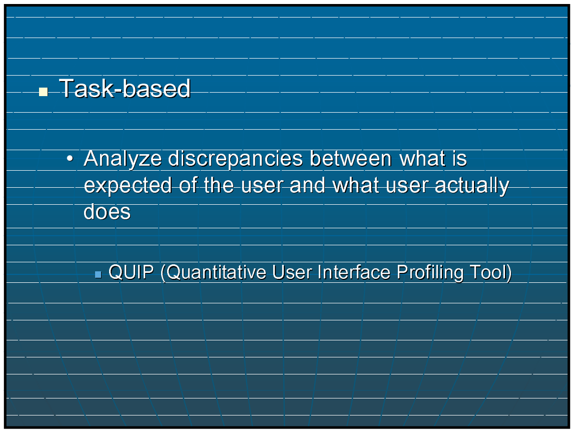
- DRUM
  - Derives
    - Task completion time
    - User efficiency (effectiveness / task completion time)
    - Productive period (no user problems)
  - Speeds up video analysis



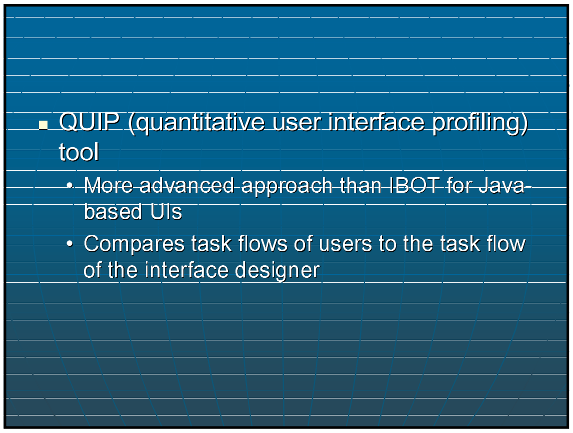
- Service Metrics’ site analysis tools
  - Pinpoint performance bottlenecks
  - Focus is on server and network performance, but little insight into usability of Web site itself



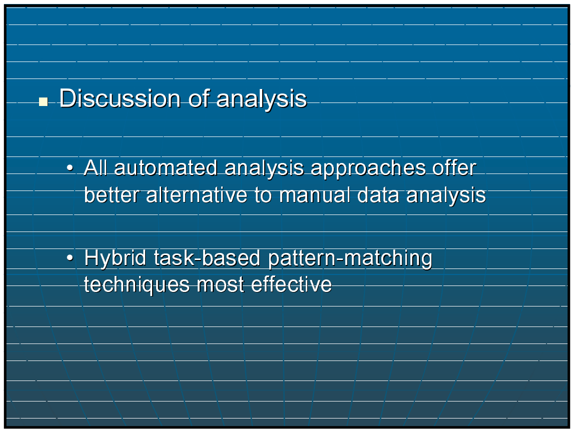
- Pattern Matching Analysis of Log Files
  - MRP (maximum repeating pattern)
    - Analyze user behavior captured in logs
    - Detects and reports repeated user actions that indicate usability problems



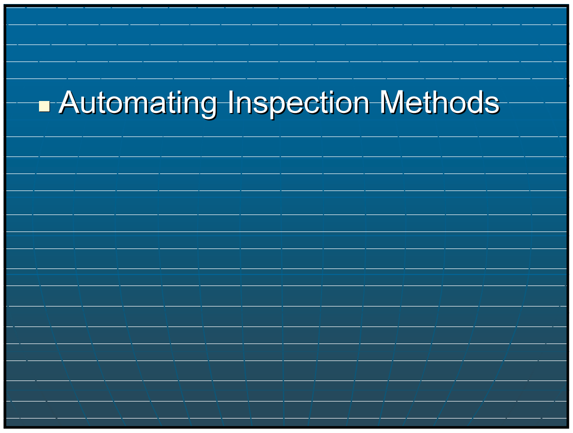
- Task-based
  - Analyze discrepancies between what is expected of the user and what user actually does
    - QUIP (Quantitative User Interface Profiling Tool)



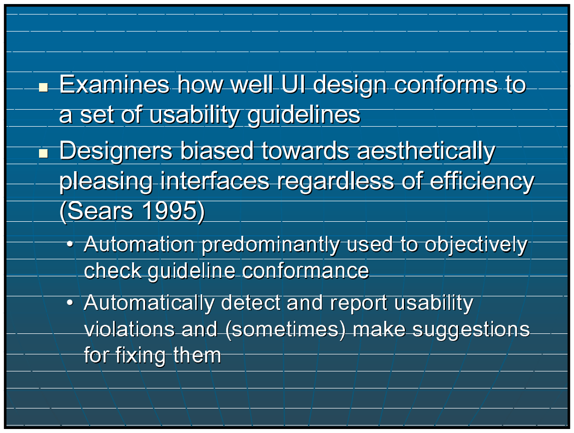
- QUIP (quantitative user interface profiling) tool
  - More advanced approach than IBOT for Java-based UIs
  - Compares task flows of users to the task flow of the interface designer



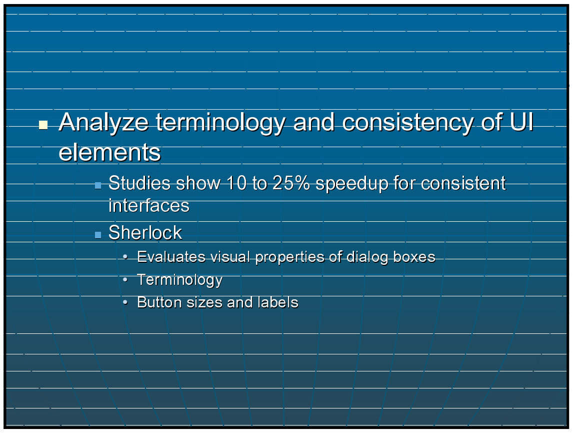
- Discussion of analysis
  - All automated analysis approaches offer better alternative to manual data analysis
  - Hybrid task-based pattern-matching techniques most effective



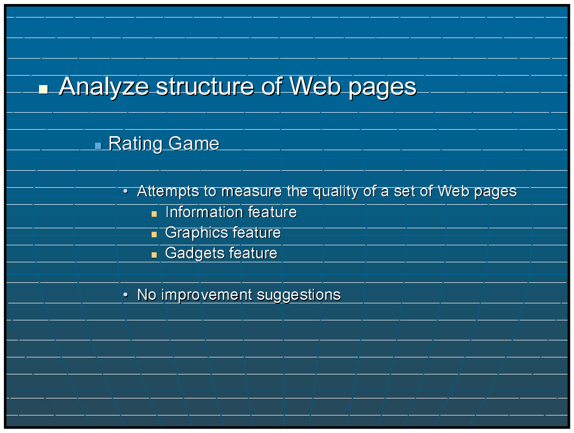
- Automating Inspection Methods



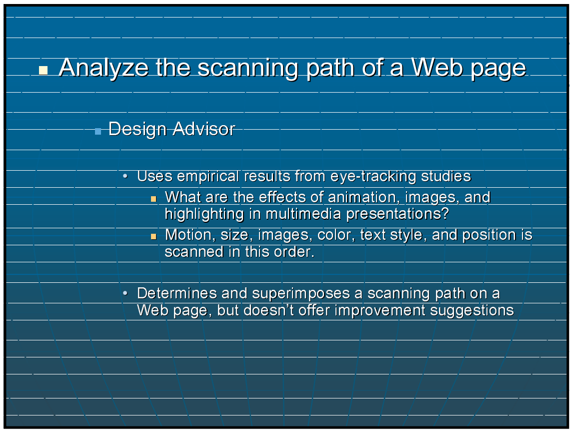
- Examines how well UI design conforms to a set of usability guidelines
- Designers biased towards aesthetically pleasing interfaces regardless of efficiency (Sears 1995)
  - Automation predominantly used to objectively check guideline conformance
  - Automatically detect and report usability violations and (sometimes) make suggestions for fixing them



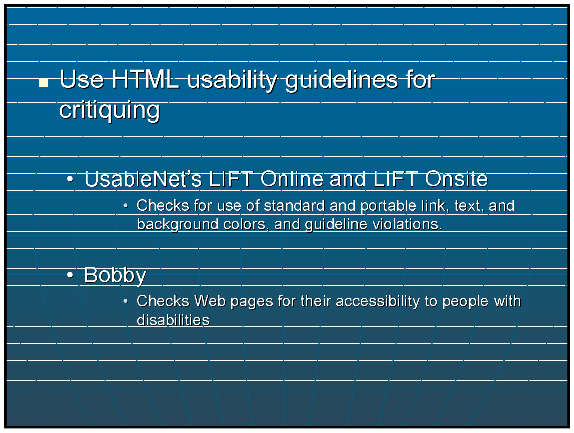
- Analyze terminology and consistency of UI elements
  - Studies show 10 to 25% speedup for consistent interfaces
  - Sherlock
    - Evaluates visual properties of dialog boxes
    - Terminology
    - Button sizes and labels



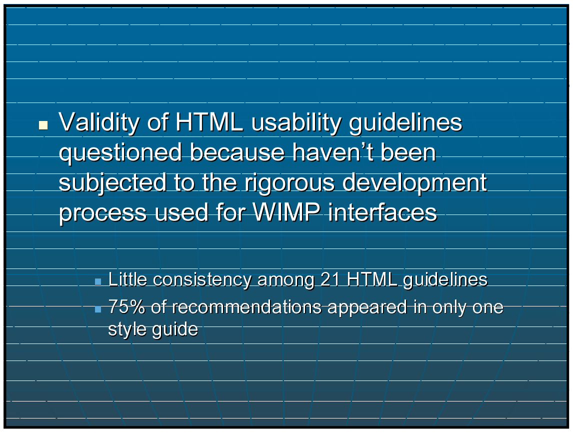
- Analyze structure of Web pages
  - Rating Game
    - Attempts to measure the quality of a set of Web pages
      - Information feature
      - Graphics feature
      - Gadgets feature
    - No improvement suggestions



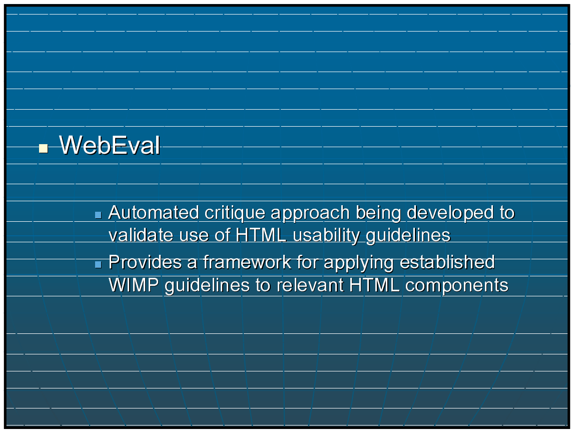
- Analyze the scanning path of a Web page
  - Design Advisor
    - Uses empirical results from eye-tracking studies
      - What are the effects of animation, images, and highlighting in multimedia presentations?
      - Motion, size, images, color, text style, and position is scanned in this order.
    - Determines and superimposes a scanning path on a Web page, but doesn't offer improvement suggestions



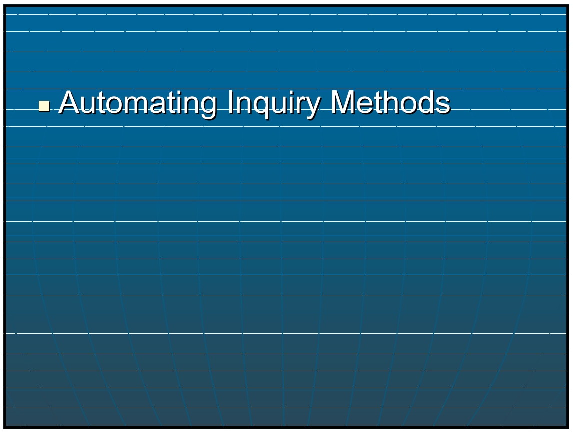
- Use HTML usability guidelines for critiquing
  - UsableNet's LIFT Online and LIFT Onsite
    - Checks for use of standard and portable link, text, and background colors, and guideline violations.
  - Bobby
    - Checks Web pages for their accessibility to people with disabilities



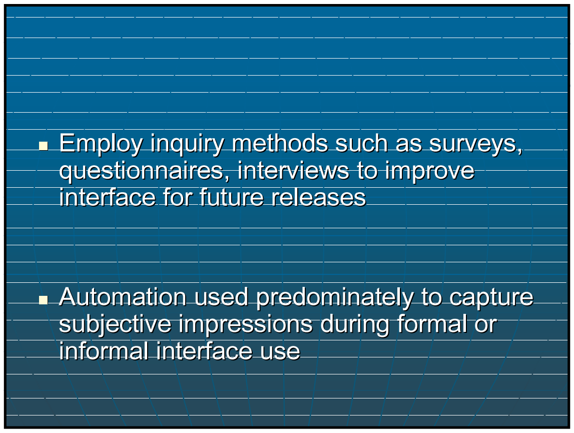
- Validity of HTML usability guidelines questioned because haven't been subjected to the rigorous development process used for WIMP interfaces
  - Little consistency among 21 HTML guidelines
  - 75% of recommendations appeared in only one style guide



- WebEval
  - Automated critique approach being developed to validate use of HTML usability guidelines
  - Provides a framework for applying established WIMP guidelines to relevant HTML components

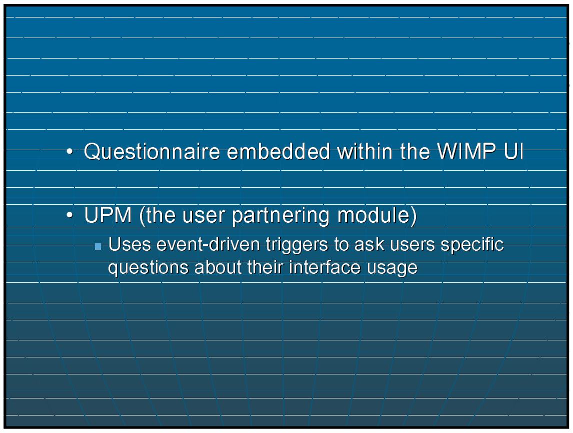


- Automating Inquiry Methods



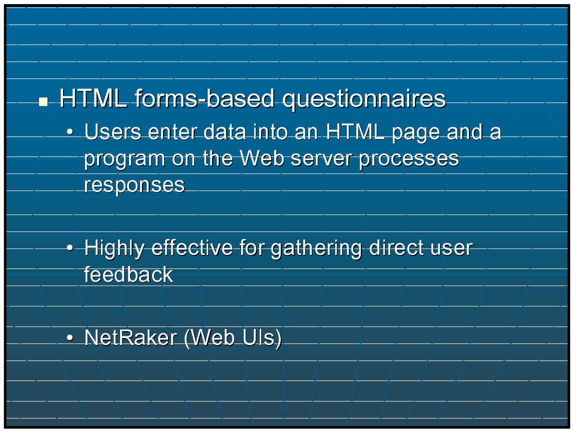
- Employ inquiry methods such as surveys, questionnaires, interviews to improve interface for future releases

- Automation used predominately to capture subjective impressions during formal or informal interface use



- Questionnaire embedded within the WIMP UI

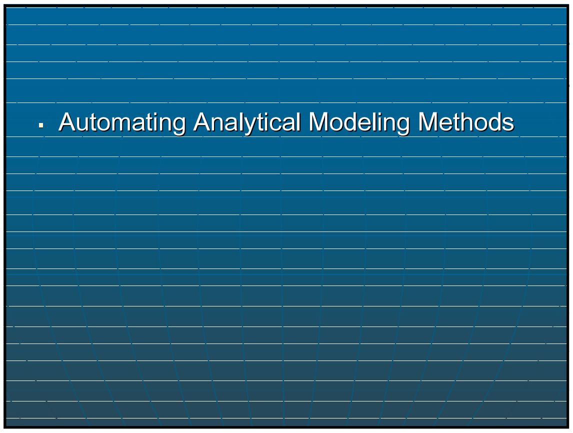
- UPM (the user partnering module)
  - Uses event-driven triggers to ask users specific questions about their interface usage



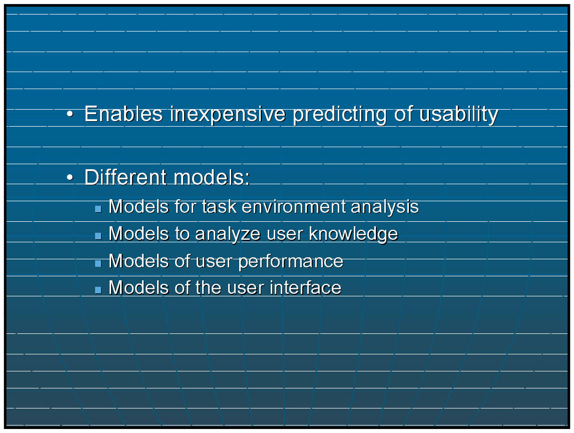
- HTML forms-based questionnaires
  - Users enter data into an HTML page and a program on the Web server processes responses

  - Highly effective for gathering direct user feedback

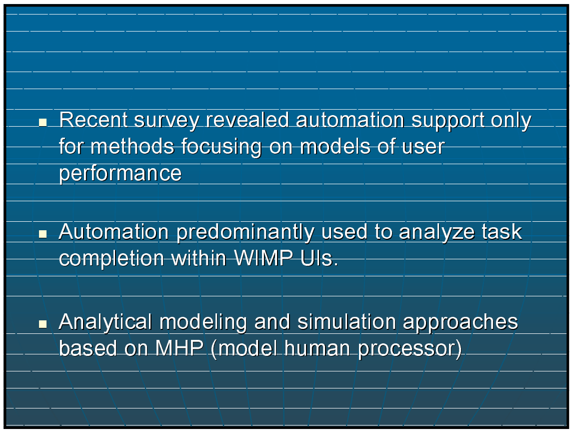
  - NetRaker (Web UIs)



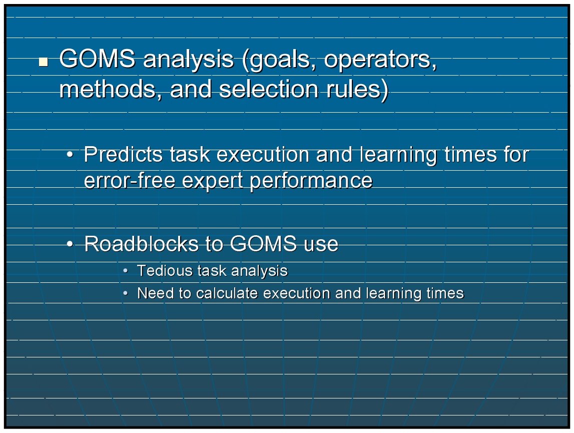
- Automating Analytical Modeling Methods



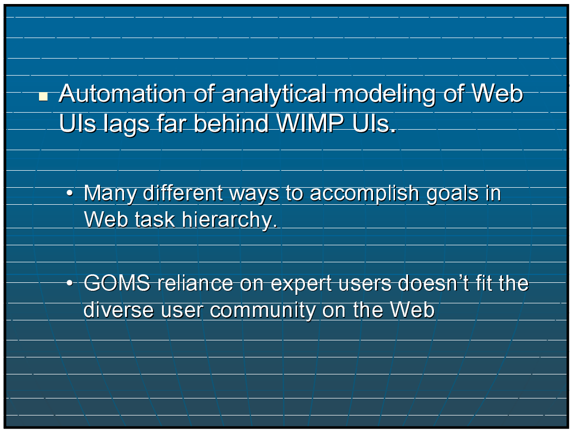
- Enables inexpensive predicting of usability
- Different models:
  - Models for task environment analysis
  - Models to analyze user knowledge
  - Models of user performance
  - Models of the user interface



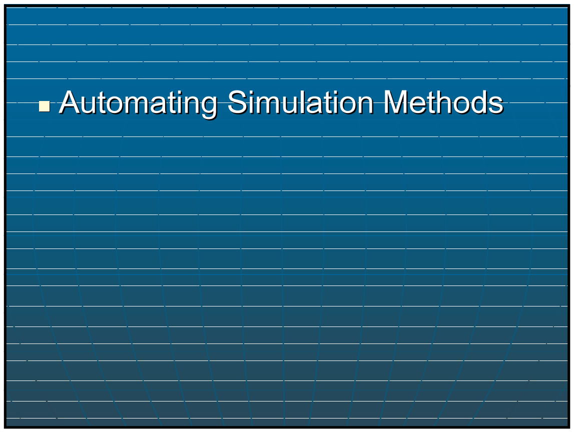
- Recent survey revealed automation support only for methods focusing on models of user performance
- Automation predominantly used to analyze task completion within WIMP UIs.
- Analytical modeling and simulation approaches based on MHP (model human processor)



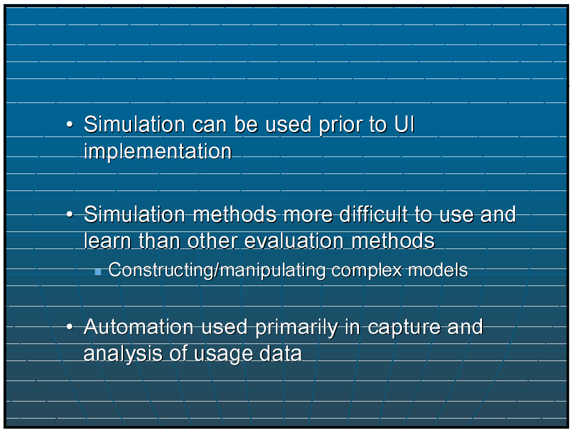
- GOMS analysis (goals, operators, methods, and selection rules)
  - Predicts task execution and learning times for error-free expert performance
  - Roadblocks to GOMS use
    - Tedious task analysis
    - Need to calculate execution and learning times



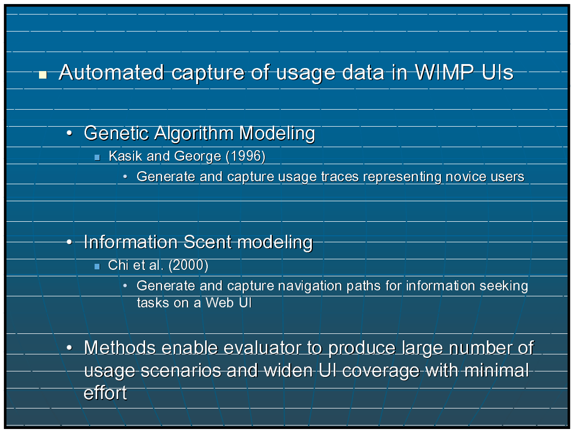
- Automation of analytical modeling of Web UIs lags far behind WIMP UIs.
  - Many different ways to accomplish goals in Web task hierarchy.
  - GOMS reliance on expert users doesn't fit the diverse user community on the Web



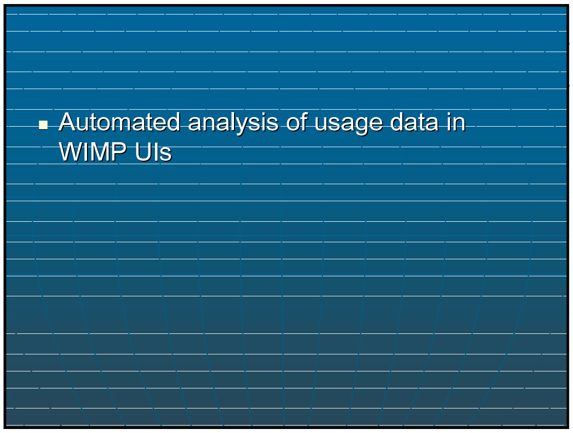
- Automating Simulation Methods



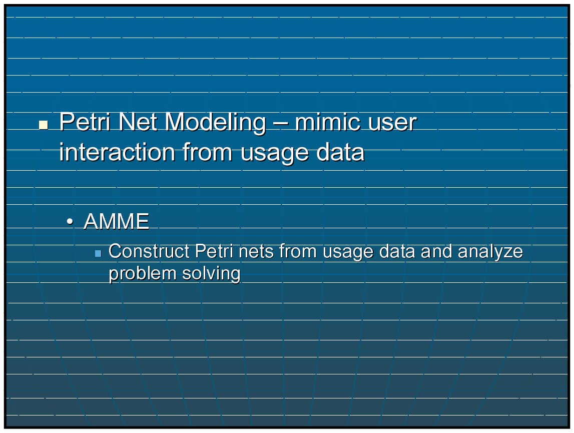
- Simulation can be used prior to UI implementation
- Simulation methods more difficult to use and learn than other evaluation methods
  - Constructing/manipulating complex models
- Automation used primarily in capture and analysis of usage data



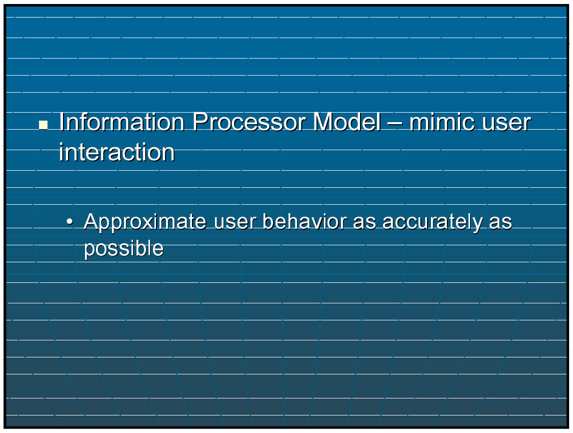
- Automated capture of usage data in WIMP UIs
  - Genetic Algorithm Modeling
    - Kasik and George (1996)
      - Generate and capture usage traces representing novice users
  - Information Scent modeling
    - Chi et al. (2000)
      - Generate and capture navigation paths for information seeking tasks on a Web UI
  - Methods enable evaluator to produce large number of usage scenarios and widen UI coverage with minimal effort



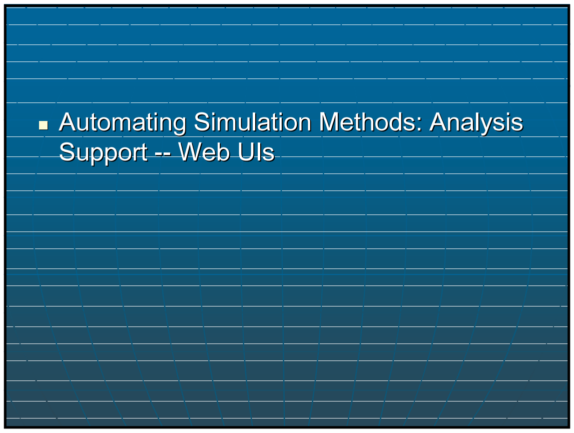
- Automated analysis of usage data in WIMP UIs



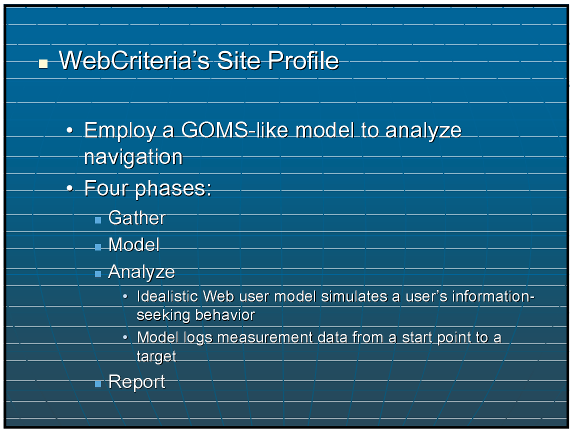
- Petri Net Modeling – mimic user interaction from usage data
  - AMME
    - Construct Petri nets from usage data and analyze problem solving



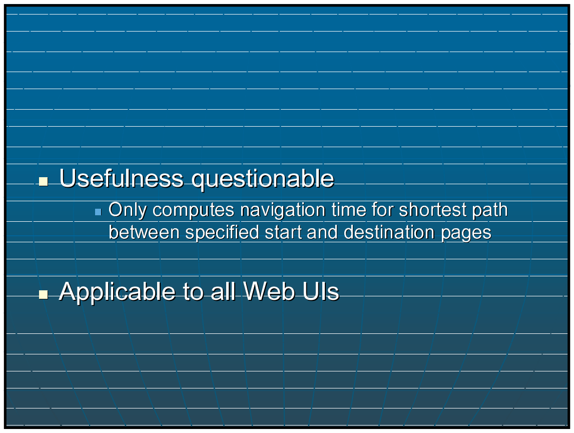
- Information Processor Model – mimic user interaction
  - Approximate user behavior as accurately as possible



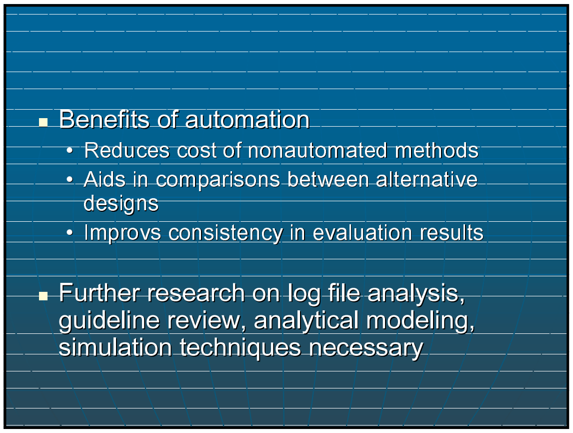
- Automating Simulation Methods: Analysis Support -- Web UIs



- WebCriteria's Site Profile
  - Employ a GOMS-like model to analyze navigation
  - Four phases:
    - Gather
    - Model
    - Analyze
      - Idealistic Web user model simulates a user's information-seeking behavior
      - Model logs measurement data from a start point to a target
    - Report



- Usefulness questionable
  - Only computes navigation time for shortest path between specified start and destination pages
- Applicable to all Web UIs



- Benefits of automation
  - Reduces cost of nonautomated methods
  - Aids in comparisons between alternative designs
  - Improvs consistency in evaluation results
- Further research on log file analysis, guideline review, analytical modeling, simulation techniques necessary