

# *Task Analysis*

Overview, utility  
Types of task analysis  
Sources and use

# Task Analysis

- Analyzing and describing how people do their jobs/work
  - > Go to their environment
- Examine users' tasks to better understand what they need from interface and how they will use it

# Components

- Three key components to include in discussing how people work
  - ✓ Activities
  - ✓ Artifacts
  - ✓ Relations
- Don't just focus on computer system artifacts and interactions
- Study related processes and objects in the environment that people may use and involve
  - ➔ Example: office env---papers, whiteboards, etc.

# Task Analysis Focus

- Focus on observable behaviors
  - ✓ What are the practices, methods, steps, objects, ..., used?
- Observe users, what they do, less so how they do it
- Not on internal cognitive state of user (more on that next week)

# Input & Output

- Gather data:
  - Documentation
  - Interviews
  - Observation
  - Surveys/questionnaires
  - Automatic data recording/tracking
- Represent Data:
  - Lists, outlines, matrices
  - Narratives
  - Hierarchies & Networks
  - Flow charts

# Data to be Gathered

- Information about users
- Description of environment
  - Where the tasks will be performed
- Major goals of the job
  - What will result in a successful end state?
- User preferences & needs
  - Before they even start: coffee, pen, notebook, log sheets...

# Data to be Gathered ...

- Tasks & Subtasks:
  - Physical
  - Cognitive
  - Communication
- Conditions under which these tasks are done
- Results/outcomes of tasks
- Requirements to perform task:
  - Information
  - Communication with others
  - Equipment

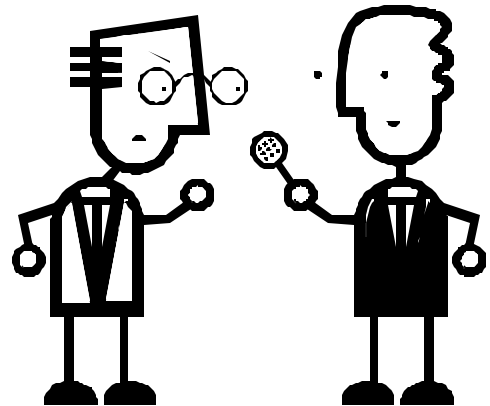
# Data Gathering Tools: Docs

- Documentation
  - Often contains description of how the tasks should be done (rather than how they are currently being done)
  - Standards
  - Manuals
  - Histories
  - Best Practices
- Domain Expert Description
  - Expert describes how process should work, how tasks should be done
  - “Knowledge-based” discovery



# DGT: Interviews

- Interviews:
  - Structured
    - Efficient
    - Require training
  - Unstructured
    - Inefficient
    - No training
  - Semi-structured
    - Good balance
    - Often appropriate



# Semi-structured Interviews

- Predetermine data of interest
- Plan for effective question types

How do you perform task x?

Why do you perform task x?

Under what conditions do you perform task x?

What do you do before you perform...?

What information do you need to...?

Who do you need to communicate with to...?

What do you use to...?

What happens after you...?

What is the result or consequence of...?

What is the result or consequence of NOT...?

- See: Gordon & Gill, 1992;  
Graesser, Lang, & Elofson, 1987

# DGT: Observation

- Observation
  - In site, watch users do what they do
  - Record with videotape
    - To watch later, or again
  - Take lots of notes, sketches
  - May require coding the video later
  - Focus on *specific* task-relevant behaviors in notes, but later convert to abstract subtasks

# DGT: Questions, Think-aloud

- Questionnaires
  - Exploratory vs. confirmatory
  - Open-ended vs. categorical (exhaustive)
    - What do you need to perform..? (list)
    - Which of the following is most important to perform...? (select)
  - If you ask it, use it. If you won't/can't use it, don't ask it.
- Think-aloud protocol
  - Person talks about what they are doing, while they are doing it (or just before or after)
  - Observer can ask probe questions
  - Why did you just do that?
- Note: Probe questions affect performance, as does thinking aloud.

# DGT: Logging

- Automatic tracking
  - Keystroke/mouse click monitoring
  - Timers
  - Logs
  - Physical location/movement trackers
    - Cell phones
    - Aware Home

# Representing Data: Outlines

- Lists, outlines, matrices
  - Use expanding/collapsing outline tool
  - Add detail progressively
  - Know in advance how much detail is enough
  - Can add linked outlines for specific subtasks
  - Good for sequential tasks
  - Does not support parallel tasks well
  - Does not support branching well

# RD: Narratives

- Narratives
  - Describe tasks in sentences
  - Often expanded version of list or outline
  - More effective for communicating general idea of task
  - Not effective for details
  - Not effective for branching tasks
  - Not effective for parallel tasks

# RD: Hierarchies

- Hierarchical Task Analysis (HTA)
  - Graphical notation & decomposition of tasks
  - Tasks as sets of actions
  - Tasks organized into plans
    - Clusters of subtasks with a preferred order and prerequisite conditions
- Example Task Clusters
  - Fixed sequence
  - Optional tasks
  - Waiting events
  - Cycles
  - Time-sharing
  - Discretionary



# RD: Networks

- Network / Entity-Relationship Diagrams
  - Objects/people with links to related objects
    - Stress relationship between objects and actions
  - Links described functionally and in terms of strength
    - Task: Develop design for final project
      - objects - pens, paper, drawing tools, etc.
      - actors - Mary, Bob, Sally
      - composite objects - the “team”
  - Often list attributes, actions of objects

**Object:** pen    **simple**

**Attribute:**

color: red

writing: on/off

**Object:** Mary    **actor**

**Actions:**

M1: make a sketch

M2: organize meeting

# RD: Flow Charts

- Flow Chart of Task Steps
  - Combines Entity-relationship (network) with sequential flow, branching, parallel tasks.
  - Includes actions, decisions, logic, by all elements of the system
  - Abstracted
  - Mature, well-known, good tools

