Human Computer Systems

Task Analysis

Task Analysis

- Study the way people perform tasks with existing systems
- techniques for task analysis
  - decomposition into subtasks, classification of task knowledge , lists of things used and actions performed
- sources of information
  - existing documents, observation, interviews
- using task analysis to design
  - manuals & documentation, new systems

Task Analysis

- Task analysis is the process of analysing the way people perform their jobs - they things they do, what they act on, and what they need to know to do their jobs
- example (ref Preece) : Housekeeping :-
  - in order to clean the house
    - get the vacuum cleaner out
    - fix the correct attachment
    - hoover the rooms
    - when dustbag is full, empty it
    - put the vacuum cleaner and tools away
  - need to know about vacuum cleaners, attachments, dustbags, where they are stored, rooms etc

3 Approaches to Task Analysis

- Task Decomposition
  - look at splitting task into subtasks, look at the order in which they are performed
- Knowledge-based techniques
  - look at what users need to know about the objects and actions involved in a task, and how that knowledge is organised
- Entity-relation-based analysis
  - object-based approach, emphasis on identifying actors & objects, relationships between them & actions the perform

Task Analysis

- Concerned with existing systems & procedures
- one of the main purposes is to help in producing training materials & documentation
- another is when a new system is introduced , task analysis contributes to the statement of requirements of the system

Scope of Task Analysis

- Wide scope
- includes:
  - tasks directly related to computer
  - tasks not related to computer e.g. retrieve document from filing cabinet
- similar to traditional systems analysis but recognising the importance of the user

Scope of Task Analysis

- Could use GOMS-like notation to represent task decomposition, but difference lies in the intention of the models
- Goals-oriented models
  - purpose is to understand the cognitive processes as a person undertakes a task - granularity normally small
- Task analysis emphasis
  - observe the user from the outside, and includes actions like retrieve document i.e observe behaviour rather than their mental state

Task Decomposition

- Hierarchical Task Analysis (HTA)
  - OUTPUTS are a hierarchy of tasks and subtasks, and a plan describing in what order and under what circumstances subtasks are performed

Hierarchical Task Analysis (HTA)

0. in order to clean house
   1. get vacuum cleaner out
   2. fix attachment
   3. Clean rooms
      - 3.1 clean hall
      - 3.2 clean bedroom
   4. Empty dust bag
   5. Put vacuum cleaner & attachments away

Plan 0: do 1 - 2 - 3 - 5 in that order. When dust bag full do 4

Plan 3: do any of 3.1, 3.2 etc in any order depending on which rooms need cleaning

Hierarchical Task Analysis (HTA)

- Indentation denotes levels of hierarchy
- plans labelled by task to which they correspond e.g. plan 3 refers to 3.1 - 3.2
- only plans for subtasks which are decomposed
- could produce a more detailed plan eg
  - plan 3 : do 3.1 every day, do 3.2 once a week, when visitors are due do both 3.2 and 3.2

Producing an HTA

- Iterative process
- decide what subtasks must be accomplished in order to perform the main task
- refer to various sources e.g. direct observation, expert opinion, documentation etc
- look at each subtask and subdivide if necessary etc

Producing an HTA...

- Apply some form of stopping rule to avoid going on indefinitely - depends on purpose of task analysis
- example: chemical plant - high level decomposition:

0. in an emergency
    1. Read the alarms
    2. Work out appropriate action
    3. Perform corrective action

Producing an HTA...

- If aim is to install computer monitoring of plant then we would expand task 1 & 3
- if aim is to produce on-line manuals then task 2
- a rule for training materials is the P x C rule
  - if the probability of making a mistake in the task (P) multiplied by the cost of the mistake (C) is below an agreed threshold, then stop expanding i.e simple tasks don't need expanding (because nobody needs training) unless they are critical

Producing an HTA...

- Stopping point where task contains complex motor responses (e.g. mouse movement) or where it involves internal decision making
- task hierarchy can be represented diagramatically as well as textually (Dix pps. 265- 268 2nd ed, 515-519 3rd ed)
- after first attempt at HTA, examine for errors or omissions e.g. by asking domain expert

Knowledge-based Analysis

- Begins by listing all objects & actions involved in the task - build taxonomies of these
- similar to hierarchical descriptions in biology e.g. mammals, reptiles etc
- aim is to understand the knowledge needed to perform a task

First attempt at car taxonomy

Motor controls

- steering *steering wheel, indicators*
- engine/speed
  - direct *ignition, accelerator, foot brake*
  - gearing *clutch, gear stick*
- lights
  - external *headlights, hazard lights*
  - internal *courtesy light*
- wash/wipe ....
- heating ....etc

Knowledge-based Analysis

- As with HTA difficult to know when to stop
- best to list everything then refine by removing unnecessary items
- to move from the list of objects, consult domain expert - some classifications may already exist
- another technique is to give the user cards with items on them and ask for them to be sorted into related piles, then name piles - hence get user view of structure

Task Analysis for Knowledge Description (TAKD)

- Uses special form of taxonomy called task descriptive hierarchy (TDH)
- either/or branches (XOR)
- AND - when object must be in several categories
- OR - could fall onto more than 1 category but not necessarily all

TAKD examples

Wash/wipe **AND**

- function **XOR**
  - wipe
    - *front wipers, rear wipers*
  - wash
    - *front washers, rear washers*
- position **XOR**
  - front
    - *front wipers, front washers*
  - rear
    - *rear wipers, rear washers*

TAKD examples

Kitchen item **OR**

- preparation
  - *mixing bowl, plate, chopping board*
- cooking
  - *frying pan, casserole, saucepan*
- dining
  - *plate, soup bowl, casserole, glass*

TADK Uniqueness Rules

- Completed TDH can distinguish between any two specific objects (previous example fails test because can't distinguish soup bowl from glass)
- TADK would require a refinement to example until all pairs can be distinguished from each other

Kitchen item **AND**

/__ shape **XOR**

/ |__ dished

/ | *mixing bowl, casserole, soup bowl, glass*

/ |__ flat

/ *plate, chopping board, frying pan*

/__ function **OR**

{__preparation

{ *mixing bowl, plate, chopping board*

{__cooking

{ *frying pan, casserole, saucepan*

{__dining **XOR**

|__ for food

| *plate, soup bowl, casserole ....etc*

TAKD

- / represents AND
- | represents XOR
- { represents OR
- labels eg AND not normally used in normal TDH as they are implied by the way the tree is drawn
- at this point when each object has a unique path (s) in the hierarchy is a term in the knowledge representation grammar (KRG)

Knowledge Representation Grammar (KRG)

- Built up using / for AND branches
- ( ) for XOR
- { } for OR similar to diagram
- eg plate :

  kitchen item/shape(flat)/function{preparation, dining(for food)}

  - ie a kitchen item whose shape is flat AND its function is preparation OR dining for food

Actions

- We can carry out the same process for actions

kitchen job **OR**
|__ preparation
| beating, mixing
|__ cooking
| frying, boiling, baking
|__ dining
| pouring, eating, drinking

Tasks

- Produce generic descriptions of simple tasks using the object and action taxonomies
- KRG terms do not use the complete KRG description of each action & object - uses generic description
- one method - annotate tree with number of times each object/action is mentioned by expert, or used during observation period - small number then don't use lower level dictinctions eg haven't bothered with for food/for drink distinction (generification)

KRG sentences

- ‘cut’ level depends on number of different KRG sentences for simple tasks - if unmanageable then suggests more generification required - if all tasks represented by 2 or 3 sentences then level of abstraction too great - and again purpose of analysis, circumstances etc apply

Links

- generic categories for actions & objects are linked eg action of beating an egg in a mixing bowl

kitchen job(preparation(beating))

using a kitchen item/shape(dished)/function{preparation}

- however level of detail not well matched - need to be more specific about action of beating

kitchen job(preparation)

using a kitchen item/shape(dished)/function{preparation}

Links

- Or more generic description

  kitchen job(preparation)

  using a kitchen item/function{preparation}

- or if we watch a chef beating eggs in soup bowl

  kitchen job(preparation(beating))

  using a kitchen item/shape(dished)

Entity-relationship- based techniques

- ER modelling is a technique usually encounted in database design or object-oriented programming
- can be used for task analysis - major difference is the kinds of entities modelled
- in task analysis we are interested in a wide range of non-computer entities including
  - physical objects,
  - the actions performed on them and
  - the people who perform them

ER- based techniques

- Like knowledge-based approaches, the cataloguing of objects and actions is central to this method
- emphasis on relationships between actions & objects, rather than on similarities between different objects
- (ref Dix p274-279 & worked exercise p279-280 2nd
- P 525-530 3rd ed)

Further reading

- Sources of information (Dix p280-287 2nd ed, 532-538 3rd ed)
- Uses of task analysis (Dix p287-291 2nd ed, p532-541 3rd ed)