HUMAN-COMPUTER INTERACTION THIRD EDITION

DIX FINLAY ABOWD BEALE

Lecture 13
Models 5 – Task Analysis

Lecturer: Gerald Weber

Based on Dix et al.
Chapter 15

What is Task Analysis?

Methods to analyse people's jobs:

- what people do
- what things they work with
- what they must know

An Example

- in order to clean the house
  - get the vacuum cleaner out
  - fix the appropriate attachments
  - clean the rooms
  - when the dust bag gets full, empty it
  - put the vacuum cleaner and tools away
- must know about:
  - vacuum cleaners, their attachments, dust bags, cupboards, rooms etc.

Approaches to task analysis

- Task decomposition
  - splitting task into (ordered) subtasks
- Knowledge based techniques
  - what the user knows about the task and how it is organised
- Entity/object based analysis
  - relationships between objects, actions and the people who perform them
- lots of different notations/techniques

general method

- observe
- collect unstructured lists of words and actions
- organize using notation or diagrams

Differences from other techniques

Systems analysis **vs.** **Task analysis**

system design - focus - the user

Cognitive models **vs.** **Task analysis**

internal mental state - focus - external actions

practiced `unit' task - focus - whole job

Task Decomposition

Aims:

- describe the actions people do
- structure them within task subtask hierarchy
- describe order of subtasks

Variants:

- Hierarchical Task Analysis (HTA)
  - most common
- CTT (CNUCE, Pisa)
  - uses LOTOS temporal operators

Textual HTA description

Hierarchy description ...

0. in order to clean the house
    1. get the vacuum cleaner out
    2. get the appropriate attachment
    3. clean the rooms
        3.1. clean the hall
        3.2. clean the living rooms
        3.3. clean the bedrooms
    4. empty the dust bag
    5. put vacuum cleaner and attachments away

... and plans

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

Plan 3: do any of 3.1, 3.2 or 3.3 in any order depending on which rooms need cleaning

N.B. only the plans denote order

Generating the hierarchy

1 get list of tasks

2 group tasks into higher level tasks

3 decompose lowest level tasks further

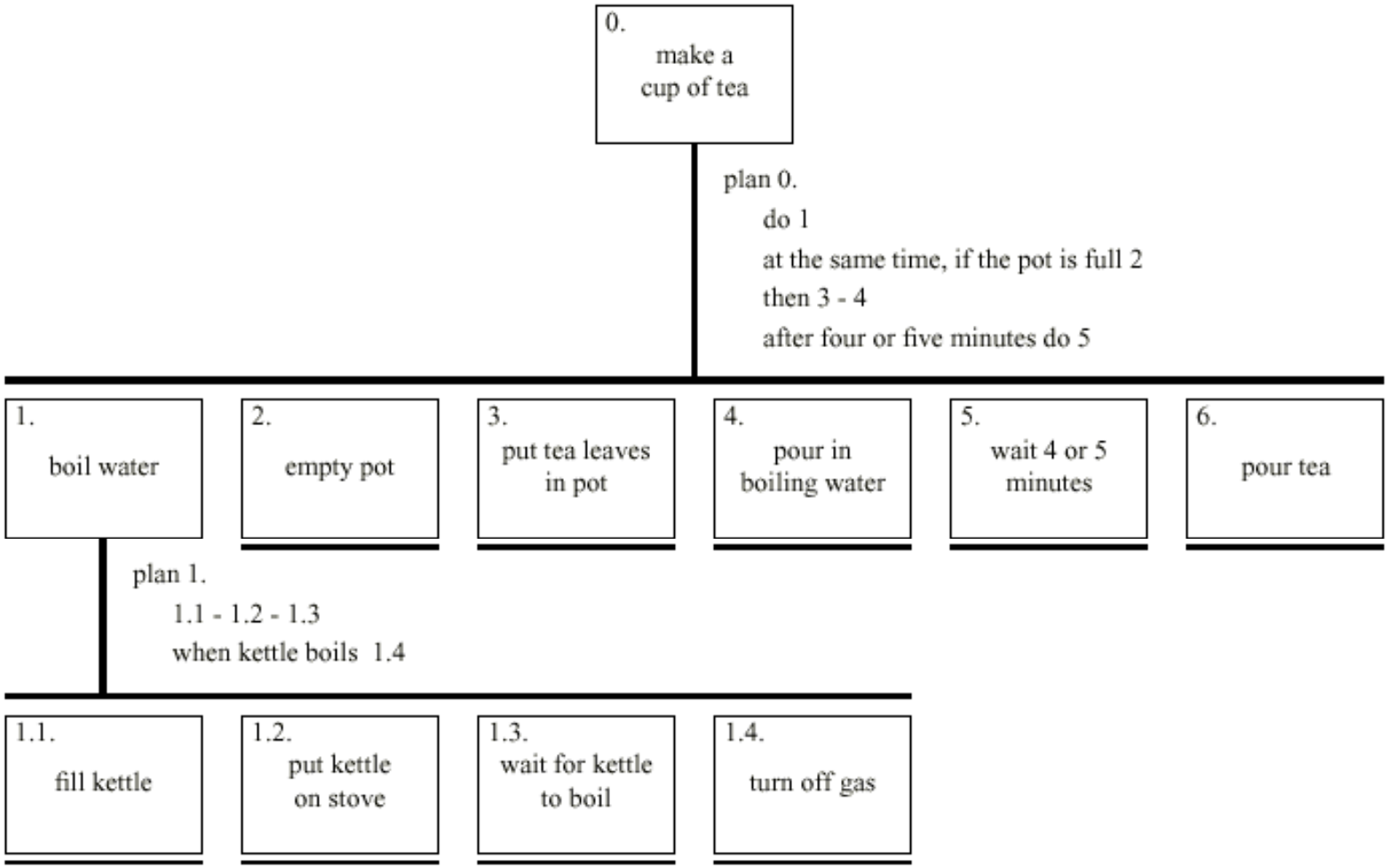
Stopping rules

- How do we know when to stop?
- Is “empty the dust bag” simple enough?
- Purpose: expand only relevant tasks
- Motor actions: lowest sensible level

Tasks as explanation

- imagine asking the user the question:
  what are you doing now?
- for the same action the answer may be:
  typing ctrl-B
  making a word bold
  emphasising a word
  editing a document
  writing a letter
  preparing a legal case

Diagrammatic HTA



Refining the description

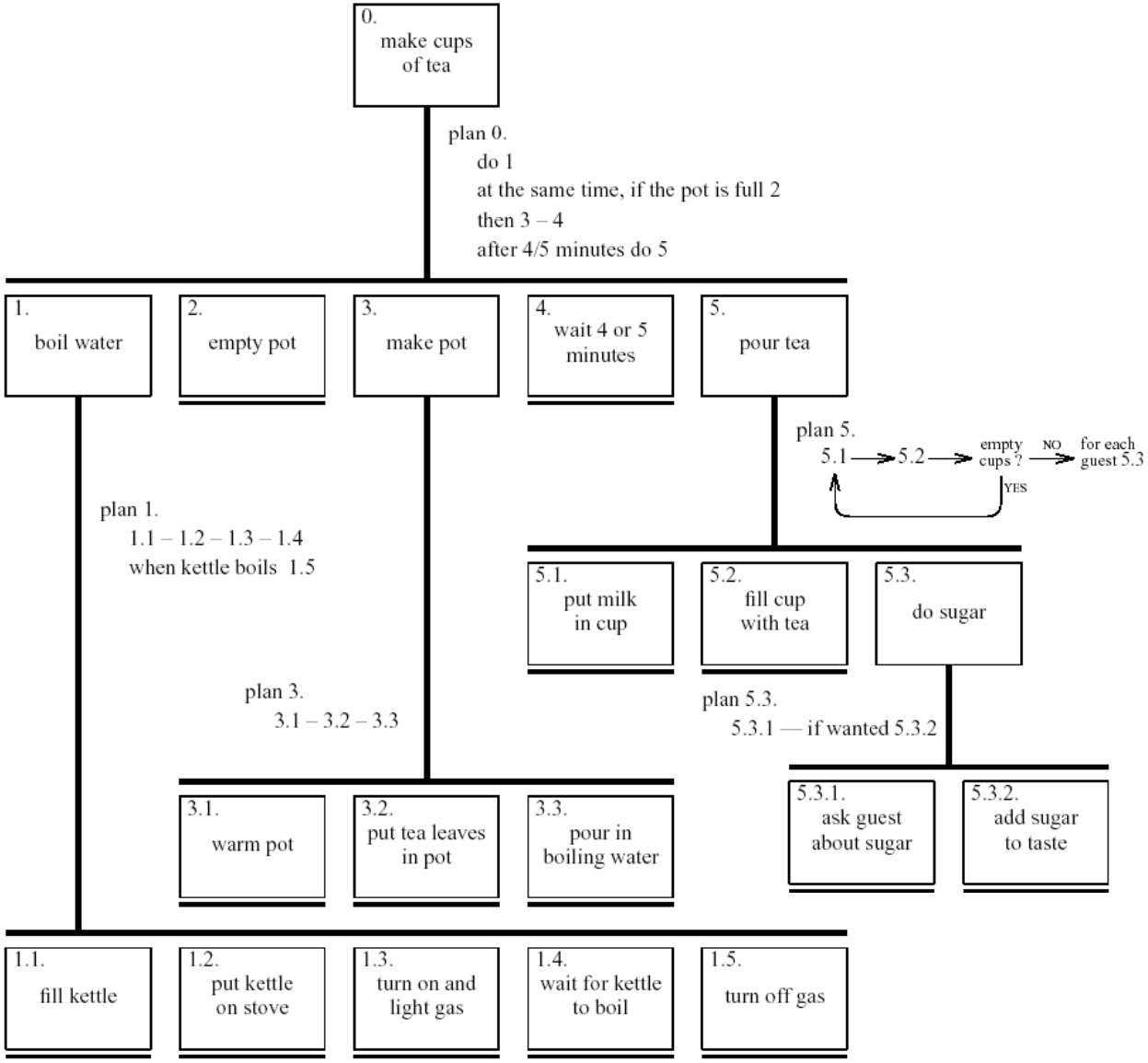
Given initial HTA (textual or diagram)

How to check / improve it?

Some heuristics:

paired actions	e.g., where is `turn on gas'
restructure	e.g., generate task `make pot'
balance	e.g., is `pour tea' simpler than making pot?
generalise	e.g., make one cup ….. or more

Refined HTA for making tea



Types of plan

fixed sequence	-	1.1 then 1.2 then 1.3
optional tasks	-	if the pot is full 2
wait for events	-	when kettle boils 1.4
cycles	-	do 5.1 5.2 while there are still empty cups
time-sharing	-	do 1; at the same time ...
discretionary	-	do any of 3.1, 3.2 or 3.3 in any order
mixtures	-	most plans involve several of the above

Knowledge Based Analyses

Focus on:

- Objects – used in task
- Actions – performed

+ Taxonomies –
represent levels of abstraction

Knowledge-Based Example …

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
  - wipers — *front wipers, rear wipers*
  - washers — *front washers, rear washers*
- heating — *temperature control, air direction, fan, rear screen heater*
- parking — *hand brake, door lock*
- radio — *numerous!*

Task Description Hierarchy

Three types of branch point in taxonomy:

- XOR – normal taxonomy
  object in one and only one branch
- AND – object must be in both
  multiple classifications
- OR – weakest case
  can be in one, many or none

```
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
```

Larger TDH example

```
kitchen item AND
/____shape XOR
/    |____dished  mixing bowl, casserole, saucepan,
/    |            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
          |____for drink  glass
```

N.B. ‘/|{’ used for branch types.

More on TDH

Uniqueness rule:

- can the diagram distinguish all objects?

e.g., plate is:

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

nothing else fits this description

Actions have taxonomy too:

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

Sources of Information

Documentation

- N.B. manuals say what is *supposed* to happen but, good for key words and prompting interviews

Observation

- formal/informal, laboratory/field (see Chapter 9)

Interviews

- the expert: manager or worker? (ask both!)

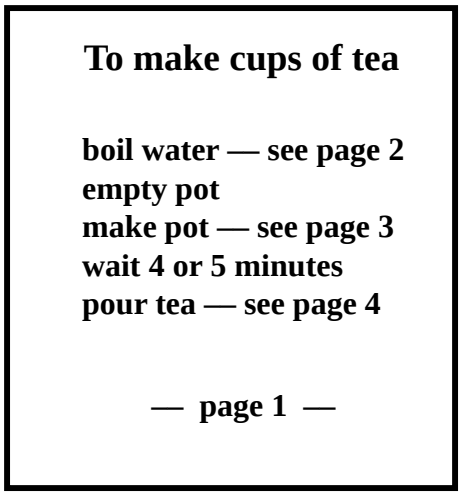
Uses – manuals & documentation

Conceptual Manual

- from knowledge or entity–relations based analysis
- good for open ended tasks

Procedural 'How to do it' Manual

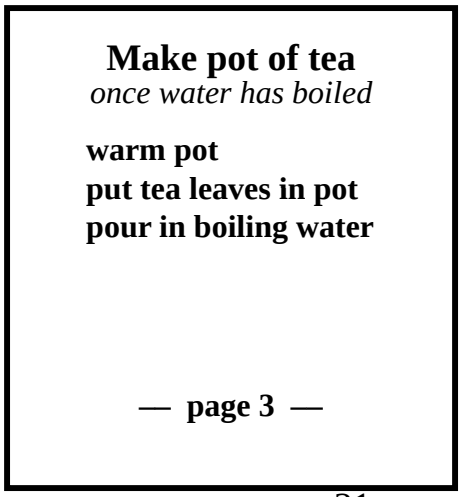
- from HTA description
- good for novices
- assumes all tasks known



To make cups of tea

boil water — see page 2
empty pot
make pot — see page 3
wait 4 or 5 minutes
pour tea — see page 4

— page 1 —



Make pot of tea
once water has boiled

warm pot
put tea leaves in pot
pour in boiling water

— page 3 —

Uses – requirements & design

Requirements capture and systems design
- lifts focus from system to use
- suggests candidates for automation
- uncovers user's conceptual model

Detailed interface design
- Taxonomies suggest menu layout
- object/action lists suggest interface objects
- task frequency guides default choices
- existing task sequences guide dialogue design

NOTE. task analysis is never complete
- rigid task based design $\Rightarrow$ inflexible system