An Application of Task Analysis to the Development of a Generic Office Reference Model

Presented at Interact '90
Cambridge, August 1990

Technical Report No. 111
Jill Hewitt
John Hobson
John Sapsford-Francis

November 1990
AN APPLICATION OF TASK ANALYSIS TO THE DEVELOPMENT OF A GENERIC OFFICE REFERENCE MODEL

Jill HEWITT, John HOBSON, John SAPSFORD-FRANCIS
Hatfield Polytechnic, College Lane, HATFIELD, Herts. AL10 9AB
Tel: 0707 279327 e-mail: comjail@hatfield.uk.ac

The roles of task analysis in the software engineering life cycle are considered, and a method is described which is suitable for capturing the high level communication tasks in offices. The contribution of this method to the building of a generic office model and its role in the generation of scenarios for future early requirements analysis are discussed.

1. THE GENERIC OFFICE REFERENCE MODEL

Work carried out with the Human Factors Division, British Telecom*, over the past year has led us to the design of a multi-perspective office reference model. It incorporates four views: the high level organisational (goal) view, the task view, the linguistic view and the operational requirements (support services) view. The task analysis described in this paper contributes to this model at the organisational and task levels. The model itself and the interfaces between the views have been described elsewhere (Watkinson 1990). The role of the model is to provide a benchmark for the evaluation of the potential effects of new systems and products in a very early requirements analysis. Scenarios of office activities may be generated from the model, they are used either as the basis for further investigations or to formulate hypotheses about possible future activities in selected areas.

1.1 The Role of Task Analysis

Task Analysis techniques have been shown to be useful in various stages of the Software Engineering Life Cycle. TAKD (Johnson & Diaper 1989) and TKS (Johnson et al. 1988) aim to capture some of the different types of knowledge that are recruited in task behaviour and to provide a generic task model which can be used as the basis for subsequent system design. TKS considers the broader concept of roles, made up of collections of tasks; this concept also plays a major part in the early stages of CORE (SD-Scicon), a more traditional requirements analysis technique, where viewpoints are considered as a starting point of the analysis. Although Suchcliffe (1988 (a)) describes task analysis as synonymous with requirements analysis, there do appear to be major differences between the approaches of (say) TKS and JSD (Suchcliffe 1988 (b)) in that TKS allows for consideration of cognitive tasks, whereas JSD is more concerned with modelling communicating processes and their data.

Other task analysis methods concentrate more on the cognitive limitations of tasks and are employed in the process of interface design; methods such as GOMS (Card et al. 1983) and TAG (Payne & Green 1989) allow for a specification of user-system interaction and provide the designer with a way of evaluating alternatives.

In the context of the generic office model, a method was required that allowed capture of high level task descriptions so that they could be viewed in the context of a company’s organisational structure. The more detailed descriptions of user-system actions and the design of any particular new system were not to be considered.

2. THE METHOD

The analysis in this section is based on data collected from a variety of offices, mainly in the form of recorded interviews and conversations with key company personnel.

2.1 The High Level View

The high level view which has been developed seeks to elicit and describe the structure of relationships between task performers (Checkland 1981), and provide a referential framework for further analysis. The basis of the method is the same semi-structured interview used for the more detailed task analysis, enhanced with formal organisation charts, job descriptions and plans of office layout.

The method seeks to create an overall picture of an office environment showing the inter-relationships between the office personnel and the flow of information between them, it thus provides a frame of reference for any subsequent task analysis or systems analysis and gives the essential organisational overview when considering a change to some part of the system.

It is important to be able to relate the analysis back to the original transcript of the semi-structured interview, subsequent analyses can then be more easily cross-related. Figure 1 gives an example of a structure chart developed from a paragraph of transcription which is an interview with an accountant's secretary (Liz) talking about her boss, Alec. This has been used as an example throughout the text.

*Research Project A114929/P (IOD) Hatfield Polytechnic and British Telecom.
Sponsored by: Human Factors Division, British Telecom Research Laboratories, Martlesham Heath, Ipswich IP3 7RE.
Interview 1.
Para 4.
E: I suppose right he gets the post opens it delegates to whoever his
next assistant in line em like he's got someone working for him - em he
deals with his own clients. Em he'd dish out the post to him and anything
general to me that's not worth bothering with ......

![Diagram](image)

Figure 1. The Structure Chart derived from part of an interview

Cross referencing is provided by the numbering, thus 1.4.3
refers to interview 1, paragraph 4, line 3.

These structure charts give a useful visual overview of the
roles of individuals, they can be overlaid to show all the
activities relating to an individual, derived perhaps from
several interviews. A complete analysis of the office revealed
a large number of structures in which Liz was involved,
highlighting her pivotal role in the day to day running of the
office. They provide a useful mechanism for cross-checking
information in any subsequent interviews and can highlight
areas of interest where formal and informal structures differ.

These discrepancies between formal and informal structures
were found to fall into two categories; the first is a set of
tasks which are natural extensions of the job description and
are part of a job expansion, for example a secretary using a
word processor to create letters where previously a typewriter
was used. The second category are tasks which bear no
obvious relationship to the job description. This second set
of tasks may be further sub-divided into those which would
generally be expected to be done by someone less qualified,
and those tasks which have developed through some change
in the system.

An example of a less-skilled task is that of Alec opening the
post; this was seen to be a hang-over from the early days of
the office when the first person in opened all the mail, but a
closer investigation may discover a covert role in that it
provides him with knowledge of the performance of other
partners. The role of computer expert undertaken by one of
the other partners provides a set of tasks which would not
have been included in his job description, but which provide
him with a certain amount of control over the administration
of the partnerships.

Any proposed changes to the office would need to
acknowledge such covert control mechanisms if they were to
be successfully implemented and it is important that
subsequent task analysis takes such possible covert goals into
account.

2.2 The Task View

Following the high level structural analysis, a more detailed
task analysis is carried out with the aims of: gaining a better
insight into the tasks as viewed by individuals, identifying
roles as collections of tasks fulfilled by individuals, providing
a basis for comparing activities between roles and between
companies and providing a generic model against which
future requirements models may be compared.

The method used draws on those developed by Diaper and
Johnson, but with important differences. In particular, we
recommend the incremental building of a generic task model
and the maintaining of close references back to the source
material; this allows actual data to be used by subsequent
scenarios rather than the more sanitised generic items, thus
giving them a more 'real' flavour.

The first step is to translate the task performance descriptions
into standard 'sentences' of the form:

Performer: Action, On Object (With Object),
From/initial state -> To/ final state

This standard form allows the controlled inclusion of
contextual information, which is critical if one is to make
meaningful generalisations. The sentences are then grouped
by task. From the paragraph given in Figure 1 we can
determine the 'sentences' shown in Figure 2.

In the case of this example, the 'with' object is not required,
although it does occur in other sentences such as:

liz: send,letter (fax), alec -> client
liz: send,letter (post), alec -> client

The next stage is to build the Generic Task Representations
(GTR's), as shown in Figure 3. This may involve the ana-
lysis in the inference of some of the sub-tasks, actions and
objects, although these should be checked where possible
with the task performer. Inferred items are shown in italics in
the figure. Some generic terms may be introduced, and new
terms added to cope with the inferred structures. Figure 4
shows extracts from a thesaurus which is developed to keep
track of generic names their pseudonyms and origins;
Figure 2.

GTR: Handle incoming post
Plan: Handle Incoming Post (only 1 plan in this case)
Goal: Overt: Sort Incoming Post
   Cover: Check progress on cases
   Overt Subgoals:
      get new post
      identify post owner
      distribute post
   Cover Subgoals:
      get new post
      identify post owner
      scan letter
      note progress
      distribute post
Subtask: get new post
   manager: get post, post(_,post-box) --> self
Subtask: identify post owner
   manager: open, post(_) --> _
   manager: read, subject (_), opened_letter --> self
Subtask: scan letter
Subtask: note progress
Subtask: distribute post
   Condition: subject is manager's client
   manager: allocate, post(_,self) --> self
   Condition: subject is general post
   manager: allocate, post(_,self) --> secretary
   Condition: default
   manager: allocate, post(_,self) --> assistant

Figure 3 A Generic Task Representation

Figure 5 gives an example of the context information that would need to be recorded for each generic task.

Generalization is carried out by associating terms that are used in similar ways then finding a generic term that can be substituted for each (Diaper 1989). Because we are dealing with sparse information it may be necessary to infer some generic terms. Thus the generic term 'manager' is substituted for 'Alec' etc. As new analyses are undertaken, the analyst should refer back to generic terms already introduced, adding new ones only where necessary, thus gradually building up the generic model.

Some tasks may be cognitively complex, in these cases the tasks are not analysed further. Such is the case with 'scan letter' and 'note progress'. It is not clear what Alec is looking for nor how he notes progress.

A role consists of a group of Generic Task Representations, so that for an individual we can identify a number of roles. The decision as to which role a task 'belongs' to may be somewhat arbitrary, but roles provide a useful mechanism for comparison across individuals and across companies and allow the construction of a generic role model.

In our analysis of five companies we were able to identify generic roles such as Typist (Transcribe Dictation, Copy Typing, Photocopying, Filing and Distributing Documents), and Personal Assistant (Arrange Meetings, Answer Phone, Deal with Callers, Reply to General Mail). These roles formed part of the work of each senior secretary interviewed, but they each also played other roles which were not generally applicable, e.g. Trainer (of other secretarial staff) or Database Manager (of staff cv's). It was thus easier to
As the analyses proceed, a generic task model is built up which can be related at the role level to a higher level organisational view and at the task level to the operational requirements (e.g., communication services) needed to support the tasks. A linguistic analysis at the task level can give further information about dialogues which adds to the richness of the generic office model. This model should be used as the starting point for new analyses, thus providing a framework for the first structured interviews and a benchmark for the new analysis. Ideally, the iterations should be repeated until the information gained from a new analysis does not make substantial changes to the generic office model, a stable state thus being reached.

3. SCENARIO BUILDING

As identified in the introduction, the generic office model is to be used in establishing requirements for new systems where actual customers have not yet been identified. We may want to answer such general questions as "What will be the effect of this innovation on an office organisation?" or "What can be built in order to improve this aspect of an office organisation?" In both cases, the generation of a scenario can help to clarify ideas by providing a "snapshot" of activities in the area under review. For example if we were developing an electronic diary management system that could be accessed over the telephone we could extract from the generic office model all those roles containing tasks that were related to scheduling and recording appointments. These could be used to generate a "typical office" scenario involving the use of diaries, which could be enriched by the inclusion of actual dialogues and task information. A hypothesis about the proposed new system could then be
made and a new scenario developed to show how the
electronic system could be used. This would form the basis
of early discussions about the functionality of the proposed
system, and would provide valuable metaphors to be used in
the design of the first user interface prototype.

4. COMPUTERISATION OF THE ANALYSIS

Carrying out this kind of task analysis for a group of organi-
sations requires continual cross referencing throughout a
large body of data. The computerisation of parts of the task
analysis process has been considered. The use of a database
is favoured for maintaining 'sentences' of actions and objects
drawn from the interview material as well as the thesaurus
and context table. This should allow sophisticated questions
that can extract, for example, all instances of the use of a
particular object or a particular action. This should help
considerably in the generalisation process.

The use of a Hypermedia environment would allow data to be
stored in its original form giving easy access for researchers
to browse through it, perhaps listening to recorded material or
viewing photographs of office layout.

We are also investigating the possibility of setting up a simu-
lation of parts of the model. This would provide an important
check on the completeness and consistency of the model and
greatly facilitate the process of scenario building and testing.

5. CONCLUSIONS

We have developed a method of task analysis that contributes
to the multi-perspective office reference model whilst main-
taining useful links back to the source data to enable realistic
scenarios to be evoked. This approach has required the
development of new procedures and notations. The use of a
standard 'sentence' as a contextual structure for actions and
their objects makes it possible to genericify actions and objects
using limited context. The brevity and consistency of the
'sentence' improves its comprehensibility. Our approach
provides support for searching and cross referencing in the
generalisation process.

6. REFERENCES

Human Computer Interaction. Lawrence Erlbaum Associates,
Hillside, New Jersey.

Wiley.

CORE - SD-Seicon, Pembroke House, Pembroke Broadway,
Camberley, Surrey, GU15 3XD.

Diaper D. (1989) Task Analysis for Human-Computer
Interaction Chapter 4. Ellis Horwood.