GTA: Groupware Task Analysis – Modeling complexity

Gerrit C. van der Veer, Bert F. Lenting, Bas. A.J. Bergevoet

Recall the design representation classification framework
GTA – Groupware Task Analysis

- Discusses important topics related to task-based design
  - traditional task-based design method
  - task-based design and CSCW and ethnography
- Provides the conceptual basis for a task modelling language and approach supporting groupware
  - The goal of task analysis is to build a “complete” task model
  - The model is a tool for designing complex technology for supporting CSCW
  - Hence, the modelling language needs important concepts from the field of CSCW
- Important question: does IS imply CSCW?

Three roles of task models

- Task model 1:
  - describe the current situation
  - not part of design itself, but a prerequisite
- Task model 2:
  - describe a future situation, i.e. the desired work processes
  - first step in a design process for improving work processes
- Abstract UI model:
  - describe the conceptual structure and behaviour of the UI
  - dialog structure must be designed to support the tasks
  - may include a revised domain model
Design of a new system or UI is often triggered by
- desire to improve inefficient and costly processes, e.g. slow processes for handling invoices
- new technology that supports new ways of working, e.g. the ability to scan and interpret invoices

Task model 1 (current tasks) helps in two ways:
- ensure that existing goals are met by new system
- evaluate future system as it must be an improvement to justify the cost

“In all cases where a 'current' version of the task situation exists, it pays off to model this.”
From task model 1 to task model 2

- **Restructure tasks using (new) technology**
  - solve specific problems ("this process is too slow")
  - meet requirements ("this shouldn't take more than 10 min.")

- **Ref. relasjon mellom IS og CIS**
  - which parts of an IS (processes) should be supported by computer technology

- **Three sources for making design decisions**
  - problem description: task and domain model
  - wishes for future business goals and work processes
  - technological possibilities

---

“The requirements as stated by the ‘client’ (the instance that ‘pays’ the design team for improving the task situation) have to be clearly distinguished from the actors' problem specifications.

... economic considerations, time constraints, and usability specifications for acceptance testing

... In as far as clients’ specifications are in conflict with proposed solutions to problem specifications, the designers will have to negotiate with the client.”
Traditional representation techniques

The data recording is a person oriented record, where the BOOK DIRECTOR (BD) is the person being followed. The recording starts when the BOOK DIRECTOR enters his office which is outlined in the figure below.

Person A sits behind a computer and is working with it. Persons B and C are negotiating. The black rectangles on the right of the figure indicate three windows. The door to the office is in the upper left corner.

The BOOK DIRECTOR then walks to A and says: "Hi Richard, did you manage to have those invitations printed yet? A replies: 'I'll go to the printer this afternoon; I'm making some final adjustments now."

The BOOK DIRECTOR says: 'That's fine."

1:30 The BOOK DIRECTOR walks to his mailbox, placed on top of a cabinet, and gets his mail. He collects all ordering forms from teachers and puts back the other mail.

4:15 The BOOK DIRECTOR takes the ordering forms and puts an order behind a computer (see figure). The computer is already operational and shows a menu. He opens a disk box containing about 40 disks, and searches through the floppy disks. He takes a disk labeled 'Book inventory' out of the box and puts it into the computer's disk drive.

4:47 The BOOK DIRECTOR starts up a word processor application from the menu and opens a file from the floppy disk. This is done by selecting a menu option ‘Open’ and selecting the file to be opened. An inventory of all book titles that have to be ordered appears on the screen.

5:14 The BOOK DIRECTOR takes one ordering form and looks at it. He positions the cursor on the right place in the inventory by moving the mouse and starts entering the data from the ordering form.

5:29 The BOOK DIRECTOR enters the code of the lecture, the name of the lecture, the teacher's name, the book title, the author, the publisher and the year in which the book has been published. All this information is read from the ordering forms filled in by teachers.
The ethnographic approach is unique in its attention to all relevant phenomena in the task domain that are not explicitly verbalizable by (all) experts, including knowledge and intentions that are specific for some actors only, conflicting goals, cultural aspects that are not perceived by the actors in the culture, temporal changes in beliefs, situational factors that are triggers or conditions for strategies, and non-physical objects like messages, stories, signatures and symbols, of which the actors may not be aware of their functions in interaction.

How to relate this to task-based design of UI (and IS)?

Formal representations

- Supports communication among designers
- Tracability throughout the design process
- Make interpretations (and conflicts) explicit

“... the analyst is not a passive outside observer, but active in shaping the task model ... different beliefs or conflicting knowledge and goals should be explicitly reflected in the task model 1 and be considered in specifying task model 2 where conflict might be either resolved by ... a unified solution of the conflict, or, alternatively, by providing multiple possibilities ...”
Formalising observations

<table>
<thead>
<tr>
<th>Task-object</th>
<th>Update inventory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kind-of</td>
<td>Update-task</td>
</tr>
<tr>
<td>Initial State</td>
<td>Textlines 1 2 3, M</td>
</tr>
<tr>
<td>Final State</td>
<td>Textlines 1 2 3, M+k</td>
</tr>
<tr>
<td>Goal</td>
<td>Add 1 Order</td>
</tr>
<tr>
<td>Condition</td>
<td>Access 7 (file) = true</td>
</tr>
<tr>
<td>Structure</td>
<td>LOOP, SEQ, arguments (get, read, enter)</td>
</tr>
</tbody>
</table>

Methode Analytique de Description (MAD)

- One of many task modelling methods
- Task hierarchy with constraints (conditional and temporal decomposition):
  - OR, COND – alternative sub-tasks
  - LOOP – repeated sub-tasks
  - SET, SEQ – unordered and ordered sub-tasks
  - PAR, SIM – parallel sub-tasks
Conceptual framework for GTA

- **Actors**
  - actors are classified based on their characteristics (knowledge, experience, (cap)abilities, ...)

- **Roles**
  - roles are directly related to the tasks that they perform
  - roles are assigned by the organisation

- **Organisation**
  - defines the relation between actors and roles
  - often modelled as composite actors and/or roles

Conceptual framework for GTA

- **Tasks**
  - tasks are the main concept, not goal (they usually have a one-to-one correspondence)
  - traditional task tree, with relation between siblings

- **Domain model**
  - traditional object-oriented analysis model
  - type hierarchy and semantic relations

- **Environment - Universe of Discourse**
  - objects – actual objects represented in the CIS
  - actors – those who perform tasks (and are often represented in the CIS)
DUTCH

DUTCH’ design process

Task Model 1
- ethnography
- psychological knowledge
- acquisition/hermeneutics
- problem analysis/specification
- specification/negotiation
- validity analysis
- documents/artifacts

Task Model 2
- usability measuring
- early evaluation
- specifications
- feedback

Scenario
Simulation
Prototype
UVM
Functionality
Dialog
Representation
Design Space
Implementation
UAN – User Action Notation

- Dialog and functions, as experienced by the end-user
- Possible user and UI actions for a specific design
- Should be complemented by concrete representations of the UI, e.g. sketches, prototypes, etc.

Tasks and design - tasks and UAN

- Extension of UAN
- Mental and physical actions
- Connection between ontologies
- Not a software model of, but relation between task model and abstract UI model
Framework for usability

- Usability
  - Efficiency
  - Effectiveness
  - Satisfaction
  - Learnability
  - Errors/Safety
  - Manoeuvrability
  - Task Completion

- Usage Indicators
  - Performance Speed
  - Consistency
  - Shortcuts
  - Feedback
  - Selection
  - Undo

- Means
  - Task Conformance
  - Flexibility

- Knowledge
  - User Model
  - Design Knowledge
  - Task Model

→ has an impact on
→→ is a source for improving