Knowledge Modelling and Management

Part A (3)

Yun-Heh Chen-Burger
http://www.aiai.ed.ac.uk/~jessicac/project/KMM
Case Study:
Social Security Services
Real-Life Case Study
Social Security Services
In Netherlands (SSS)

- Two most important types of social securities:
  - General assistance benefits;
  - End-of-the-line type of benefits.
- 60,000 people supported by general assistance benefits.
- To qualify for benefits, each applicant is screened in detail.
- Screening rules are codified and derivable from volumes of laws and regulations.
- Considerable backlog of applications and growing.
- Results in long queues in office and delay of final decision making from initial client intake.
- Such delay has caused complaints and caught media attention.

Source: chapter 3 of [1]
Rationale: Problem - Opportunity Formulation

- The Secretary of the Directorate therefore has suggested to use knowledge systems to reduce backlog.

- **Rationale:**
  - The applicable laws and regulations are so complex, it takes too much time for the staff to reach a decision.
  - Therefore, if we can provide a knowledge system that stores the needed legal decision-making knowledge, the decision process can be speeded up.
  - As a result, the backlog is reduced!!
### Organisational Model OM-1

**Organisation Model: Problems and Opportunities Worksheet OM-1**  
**Initial contextual and problem analysis**  
**Example Worksheet for Housing Application**

| Problems and Opportunities | *Assessment of individual applications takes too much time, creating back log to be processed.*  
|                           | *There is not sufficient staff for handling urgent cases.*  
| Organizational Context    | *Mission: Enable people to take responsibility to find a proper home.*  
|                           | *External factors: National regulations.*  
|                           | *Strategy: Provide high quality housing at a reasonable price.*  
| Solutions                | Solution 1: Develop an automated system to speed up application assessment.  
|                           | Solution 2: Set up a training program for application assessment.  

[Table taken from Chapter 10, p241.]
Initial Assessment of the Case

- Problem areas clearly identified.
- Objectives are clear – reduce backlog.
- Have a sense of direction for solution.
- Clear benefits for the organisation.
- Sufficient information for OM-1!!
Quick Review: Organisation Model Variant Aspects OM-2

For each identified problem-opportunity area in OM-1, describe important organisational-related characteristics:

– Organisational structure;
– People involved and their roles;
– Culture and power – organisational practice; social and interpersonal skills; relationships and networks.
– Resources involved – e.g. information systems, equipment and materials, technology/patents and rights;
– Processes – identify of processes, may use e.g. UML activity diagram, or other business process models (more in OM-3);
– Knowledge involved (more in OM-4);
Organisational Structure (SSS)

1. Central Office
   - Support
   - Archive Department
   - Coding Department
   - Finance Department
   - Computer Centre (external)

* Aggregation relation
1 * One to many relations
       ----- Association/External Link

Branch Office
   - Team
   - Support
   - External Visiting
   - Training
   - Archive Section
   - Test Section

p.s. there are 16 branch offices
Organisational Chart and Power and Influence

Directorate

Branch Director

Team Chief

Tester

Chief External

Secretary

Chief Payments

Regulations Expert

Team Member

Formal power relation/influence

Strong informal relation/influence

Weak informal relation/influence
Resources Analysis

- Limited computers available.
- “Outsourced” computing capacity – use external computer center.
- Only some terminals are connected to the external central computer center.
- Some experiments with the use of PC with limited applications – e.g. producing letters of notification.
- Insufficient office space in some branches for receiving applicant intake.
Assumed main process need automation: i.e. the decision-making process for approving benefit applications.

Process breakdown:
- Intake,
- Archiving,
- Decision-making,
- Notifying,
- Reporting,
- Paying,
- Quality control.

An UML Activity Diagram is used.
Knowledge Intensiveness Analysis

Primary Process

- Intake
- Decision Making
- Notifying
- Reporting
- Paying

Secondary Process

- Archiving
- Quality Control

Knowledge intensive task
Insufficient decision-making knowledge
Primary Process

Intake

Decision Making

Notifying

Reporting

Paying

* Knowledge intensive task

* Insufficient decision-making knowledge

Secondary Process

Archiving

Quality Control

Time Cost Analysis

Time costing 10%

Time costing 40%

Time costing 30%

10% of time
Relative time spend for activities as cost – cost-effectiveness Analysis
Two Possible Solutions

● Knowledge based system solution – original proposal.

● Archiving and reporting system solution – proposed after cost-effectiveness analysis.
Feasibility Evaluation
Business Feasibility

- A knowledge system will not in itself solve the time inefficiency problem.
- Higher benefits in speeding up Archiving and Reporting processes.
- More PCs may be needed
  - Decentralised computing may replace centralised computing:
  - Some branches may have space issues.
- The interactions between different roles may be changed.
  - Remote auditing and query is possible
- The interactions between different organisational units may be changed.
  - Remote reporting and archiving is possible
Technical and Project Feasibility Evaluation and Conclusion

- **Technical Feasibility:** Grey decision-making process and non-verbal tacit knowledge is used, but hard to formulate.

- **Project Feasibility:**
  - Knowledge systems vs. efficient reporting and archiving solutions:
    - Reporting and archiving solution looks favourably.

- **Proposed actions:**
  - Not to take the KS solution, but to
  - Simplify the workflow for reporting and archiving process, so can
  - Resolve the bottleneck problems of reporting and archiving.
Discussion: Regarding Actor and In-depth Task Analysis

- In the provided UML activity diagram – when used on its own - it is unclear as who are involved in those tasks. It is therefore unclear as when those tasks are automated how they will affect certain individuals.

- Lack of detailed process, task descriptions and analysis – a task model is needed.

- Lack of in-depth analysis of tasks from the view point of organisational operations.

- Organisational impacts: it is unclear how the organisation may be impacted, although we can guess that external regulations expert may change their interactions with the tester – as machines would provide some of the information – agent and OTA (Organisation-Task-Agent) model is needed.

- We can also image that there may be less dependency and interactions between the branch and the external computer center in the to-be model, as some processing are now done locally in branch offices – OTA model is needed.
An Overview of the Models

A road map for carrying out knowledge-oriented organization and task analysis.

OTA: Organisation-Task-Agent Model

Source: AIFB [16]
Task, Agent and OTA

Concluding Statement

These models are used, if a Knowledge System is decided to be built
What is a Task in CommonKADS

- Task is a subpart of a business process that:
  - Represents a goal-oriented activity adding value to the organization;
  - Handles inputs and delivers desired outputs in a structured and controlled way;
  - Consumes resources;
  - Requires (and provides) knowledge and other competences;
  - Is carried out in according to given quality and performance criteria;
  - Is performed by responsible and accountable agents.
Task Model
TM-1

- Task Analysis
- Refinement of OM-3 (process breakdown).
- Analysis from 3 different aspects: the 3-D view:
  - **Functional** view:
    » task decomposition, I/Os, I/O flows, data flows
    » may use UML activity, business process, or data flow diagrams to describe a task model;
  - **Static information structure**:
    » information content and structure of objects – may use **UML class diagrams**;
  - **Control** or dynamic view:
    » May use temporal links and control flow over sub-tasks
    » may use activity and/or state transition diagrams.
Elements Included in TM-1

- Name of task
- Organisation where the process is carried out
- Dependencies between processes (I/O)
- Objects handled (I/O objects)
- Timing and control (record frequency and duration, pre- and post-conditions)
- Agents involved
- Knowledge and competence involved
- Resources involved (staff time, systems, equipments, materials, financial, budgets)
- Quality and performance measures used
Task Model 2 (TM-2)
Process Criticality and Bottleneck Analysis

- Knowledge item analysis
- Refinement from OM-4 (knowledge assets).
- Below questions may be asked for each identified task:
  - How often do you carry out this task?
  - How long does it take?
  - Whom do you talk to when carrying out the task? (e.g. to gain necessary knowledge)
  - What do you need in order to start with it?
  - What happens to the organisation, if it goes wrong?
  - What may go wrong, and what do you do then?
  - How do you know that the task is successfully concluded?
– Other Bottleneck and criticality analysis:

» How many processes need information produced by this process so that they can start/finish their own executions? This information may be derived via control and data flow constraints

» How many processes need to be completed before this process can start/finish its execution?

» What data is required for this process? Is this data difficult/complicated to obtain? What is the accuracy of this data? Is this data important for this process?

» Can some constraints be relaxed, in case this process can not be executed successfully, without compromising overall organisational operations?

• May use task scenarios to gain further understanding.
Agent Descriptions
Agent Model (AM-1)

- Analysis from the view point of an agent involved in a task.
- Understanding their roles, involvements, competence, and how they collaborate with other agents in their tasks, inc:
  - Organisation position;
  - Involvements in tasks;
  - Communication with other agents;
  - Has/use knowledge in tasks;
  - Other competences;
  - Responsibilities and constraints.
- To show how systems are used by an agent, UML Use-Case diagrams are useful.
Organization-Task-Agent Models (OTA-1)
Concluding Statement

- Impacts and changes in organisation (as-is vs. to-be models)
  - Inc. aspects of organisational structure, processes, resources, people, knowledge, culture and power.

- Impacts and changes on specific task/agent:
  - Task layout, resources needed, training needed, performance and quality criteria, staffing, individual positions, responsibilities, authorities, constraints, required knowledge and communication.

- Attitudes and commitments (from stakeholders): is it sufficient?

- Proposed actions:
  - Improvements, supporting facilities, parallel project actions, expected results/cost/benefits, when to re-consider the proposals.
Summary: Organisational Context and Task Analysis

- Organisation Model (OM)
  - OM-1: problems-opportunities, goals, solutions;
  - OM-2: variant (main) aspects of an organisation;
  - OM-3: process breakdown;
  - OM-4: knowledge assets and relations with people;
  - OM-5: Feasibility study and decision making.

- Task Model (TM)
  - TM-1: detailed task description – process detail analysis;
  - TM-2: task criticality and bottleneck analysis – knowledge item analysis;

- Agent Model (AM)
  - Agent position, involvements, responsibilities;
  - communication patterns, has knowledge, constraints.

- Organisation-Task-Agent (OTA) Model
  - Evaluation and conclusions: impact and changes in organisation, task/agent impact and changes, attitudes and commitments; proposed actions and prospects.
Additional Information
The CommonKADS (Knowledge Acquisition Data System) methodology for the KBS development is a result of the KADS-II project (“An Advanced and Comprehensive Methodology for Integrated KBS Development”), which was a part of ESPRIT 2 (European Strategic Programme for Research and Development in Information Technology) project. It has been developed over a period of 1990 – 1994;

However some ideas were derived directly from the former KADS project (“A Methodology for the Development of Knowledge-Based Systems”; 1985 - 1990).

At the moment, it is one of the most widely used methodology for KBS development in both research and commercial applications.

It is still the European de facto standard for knowledge analysis and knowledge-intensive system development.

Source: [21a]
Different Levels of Formality of Models

- **Informal model**: explained using natural language with or without graphical or other visualisation aid.

- **Semi-formal model**: explained using well-defined (graphical) notations complimented with informal (textual) descriptions, i.e. in natural language.

- **Formal model**: use well-defined notations only. This form of representation is suitable to use as a basis to provide automation.

- Models of different levels of formalities are often chosen to describe different domains to serve different purposes. They are often used together when describing a highly informal domain, which is often the case where CommonKADS method is applied.
Roles of KE and KM

- The role of knowledge engineering and knowledge system in KM:
  - Knowledge engineering offers effective methodologies for KM, including KA, high level conceptual modelling, formulisation and automation.
  - Knowledge systems offer important options for automation of KM activities.

- Areas that KE has contributed to:
  - Knowledge-oriented organisation analysis.
  - Task and agent analysis.
  - Conceptual modelling methods and transition to automation.
  - Intuitive graphical visualisation method – suitable for non-technical personnel.
  - Support sharing and reuse of knowledge, information and system components (from task models).
  - Example applications:
    » May be used in KM (quick scans) workshops, IT strategy scoping and feasibility studies.
    » Support early stages of SE project: requirement engineering, conceptual modelling of knowledge.

- **Ultimate goal of KM:** to deliver knowledge at the right time, right place, right shape, with needed quality and the lowest possible cost.
Subject Areas Related to KM

- Business management
- Organisation theory
- Business process re-engineering and improvement
- Information management
- Knowledge system development
- Community based discussion forum
- Corporate Intranet environment
- Document management system
- Email management system
- Knowledge economy
- Knowledge ecology
- Community of practice
- Social network analysis
UML Class Diagram

Organisational resources, information and data may be described in UML Class Diagrams
A class includes three parts:
- class name
- attributes and their types (e.g. string, number, integer, Boolean, date, universal)
- operations, parameters used and their types.

Association
- Relationship: An attribute whose value is not an atomic values, but indicate a relationship with objects (of other classes)
- May be bi- or uni-directional, e.g. married-to, owned-by
- Cardinality: e.g. 0-1, 1-0, 1-many, many-1.
- May indicate the “roles” of objects that are involved in a relationship, e.g. husband and wife, employer and employee.

Association Class
- Important associations may be “upgraded” to become a class, as it contains important attributes, e.g. the works-for relationship may be described in a class.

Source: chapter 14 in [1]
Associations and Object

- **Generalisation**: super-class, sub-class
  - Agent vs. human and computer-program
  - Human vs. man and woman
  - Arrow-headed link hanging from the super-class

- **Aggregation**: part-whole relation
  - Physical and conceptual aggregates
  - Audio system vs. CD-player, record-player, tape-deck, speaker, amplifier, head-phones, etc.
  - Can specify cardinality, e.g. must have one amplifier, but may have 2-4 speakers and 0-1 CD-player, etc.
  - Empty diamond-head hanging from the “whole” side of the association

- **Composition**: necessary part-whole relation
  - E.g. table and table-leg
  - Solid diamond-head hanging from the “whole” side of the association

- **Objects** – instances of a class
Main Reference


[source: http://www.aiai.ed.ac.uk/~jessicac/project/KMM]