

TDT4252 / DT8802

Enterprise Modelling and Enterprise Architecture

Sobah Abbas Petersen

Adjunct Associate Professor
sap@idi.ntnu.no

Evaluating Enterprise Models

TDT4252/DT8802, Spring 2014



NTNU – Trondheim
Norwegian University of
Science and Technology

Lecture Today

- Evaluating Enterprise Models

Based on the following article:

A7: S. A. Petersen, “Using Competency Questions to Evaluate an Agent-based Model for Virtual Enterprises”, in *Processes and Foundations for Virtual Organisations*, L. Camarinha-Matos and H. Afsarmanesh (Eds.), Kluwer Academic Publishers, 2003, p.261-270.



Model Evaluation - general

- There are several ways to evaluate models:
 - **Competency questions**: Can the model answer queries as though it has common sense about the enterprise? (Fox & Gruninger)
 - An example is available from: (A7) S. A. Petersen (2003) “Using Competency Questions to Evaluate an Agent-based Model for Virtual Enterprises”.
 - **Characteristics** of Enterprise Models(Fox & Gruninger).
 - **Principles** of Enterprise Models (Vernadat)

A6: Evaluating Enterprise Models (2)

Characteristic	Description
Functional completeness	Can the DEM represent the information necessary for a function to perform its task?
Generality	To what extent is the DEM shared between the diverse activities in the enterprise, e.g. finance and manufacturing?
Efficiency	Does the DEM support efficient reasoning?
Perspiciuity	Is the DEM easily understood by users so that it can be applied consistently and interpreted across the enterprise?
Precision granularity	Is there a core set? Does the representation support reasoning at various levels of abstraction and detail?
Minimality	Does the DEM contain the minimum number of objects necessary?

Enterprise Modelling Principles (1)

- Principle of separation of concerns:
 - It would be unrealistic to consider an enterprise as a whole. It must be analysed piece by piece.
- Principle of functional decomposition:
 - Major functions structured as sub-functions.
- Principle of Modularity:
 - To facilitate management of change.
- Principle of Genericity:
 - Important to define standard building blocks as generic classes to factor common descriptive attributes and behaviours.
- Principle of reusability:
 - To reduce modelling efforts and increase modularity.

Ref: Vernadat



Enterprise Modelling Principles (2)

- Principle of separation of functionality and behaviour:
 - Functionality: things to be done; behaviour: the way things are done.
- Principle of process and resource decoupling:
 - Consider separately the things being done (processes) and the agents performing them (resources).
- Principle of conformity:
 - Deals with syntax and semantics of the model to really accurately represent what it's supposed to model.

Ref: Vernadat



Enterprise Modelling Principles (3)

- Principle of model visualisation:
 - To easily communicate models, e.g. simple graphical formalisms.
- Principle of simplicity vs. adequacy:
 - The modelling language to be rich enough to express what needs to be expressed, yet simple enough so that users can easily learn to use it.
- Principle of management of complexity:
 - Must permit the representation of systems of great complexity.
- Principle of rigor of representation:
 - The model must neither be ambiguous nor redundant.
- Principle of separation of data and control:
 - To support real-time systems. The process is operated not by the availability of data, but by events.

Ref: Vernadat



A7: An example of using Competency Questions

- The paper is based on these ideas:
 - What is a Virtual Enterprise (VE)?
 - Agent-based Model
 - Competency Questions as an evaluation technique
 - Examples

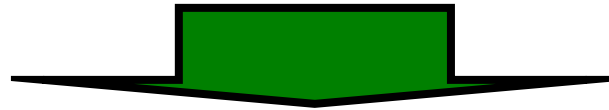
A6: Competency Questions

(Fox & Gruninger)

- We should be **able to query the model and obtain answers** to support the organisation.
- Commonsense queries:
 - Ability to answer queries as though the model has a common sense understanding of the enterprise.
 - Such an understanding often represents knowledge about the enterprise acquired over a relative short time and does not require expert knowledge.
 - Examples of such knowledge: organisational structure, roles, goals and resources would enable the deduction of what resources a person might allocate based on his/her role in the organisation.
 - We refer to it as **shallow-level processing**: retrieval that **requires a small number of deductions to answer the query**.
 - Requires a set of rules of deduction, axioms.

What is a Virtual Enterprise?

- A Virtual Enterprise (VE) has the following characteristics:
 - Goal-oriented
 - Autonomous entities that collaborate
 - Distributed and heterogeneous
 - Has a limited lifetime



A team of partners (human beings, organisations or software agent) that collaborate to achieve a specific goal.

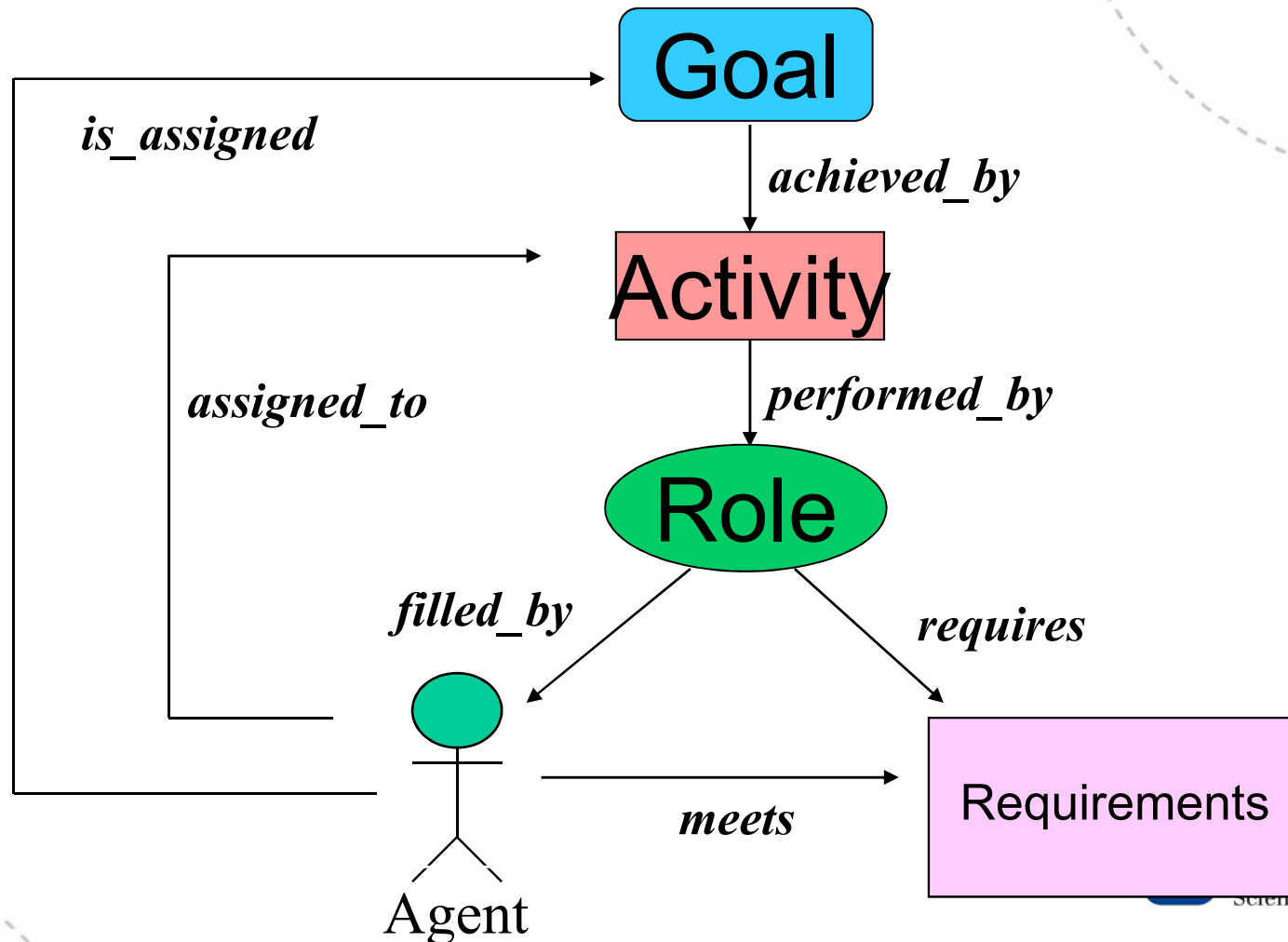
What is an Agent?

- An agent can be defined as a computer system that has the following properties:
 - Autonomy
 - Social ability
 - Reactivity
 - Pro-activeness
- Goal-oriented
- Designed to operate in distributed environments
- Handle sophisticated interactions such as negotiations

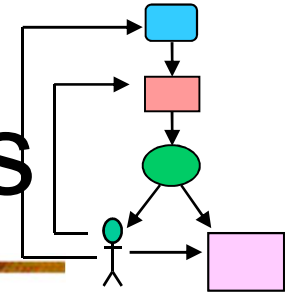
Why Agents & Virtual Enterprises?

- Human beings or organisations can delegate responsibilities to agents.
- Agents can provide a flexible means of modelling the VE in terms of cooperative work.
- Agents are computational entities. Thus the resulting model can be easily and efficiently converted to the required computational support.
- Agents can handle sophisticated interactions such as coordination & negotiations.

Agent-based Model of a VE

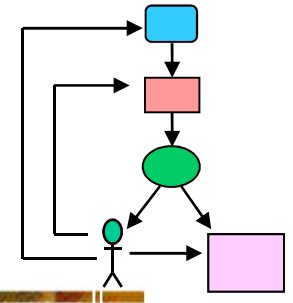


Agent-based Model: Contents



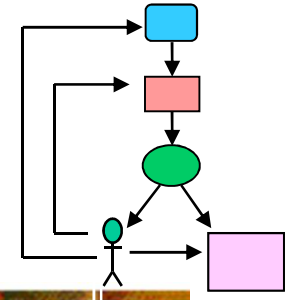
Entity	Attributes	Relationship
Goal , e.g. Produce some pens	goal(goal_name, product_area, deadline, max_cost).	achieved_by(goal, activity).
Activity , e.g. Design pens	activity(name, start_date, completion_date).	performed_by(activity, role).
Agent , e.g. Michael	agent(name, address, goals, skills, availability).	filled_by(role, agent).

Requirements of the Model



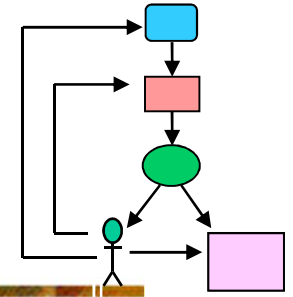
- A model is no longer just an information repository. But, it must be able to:
 - **Answer queries** about what is explicitly represented in as well as implied by the model.
 - Provide decision-making support.

The Need for Evaluation of Models

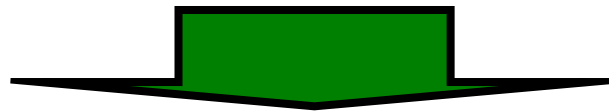


- We need to know when the modelling is complete.
- We need to be able to judge the correctness of representation.

Competency Questions

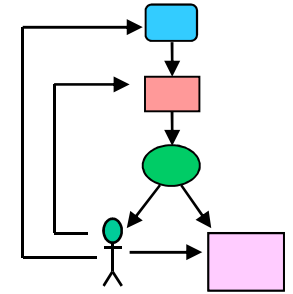


- Proposed by Fox & Gruninger.
- **Competency of a model** is its ability to answer a set of questions.
- The requirements for a model can be formulated as a set of questions that the model is required to answer.



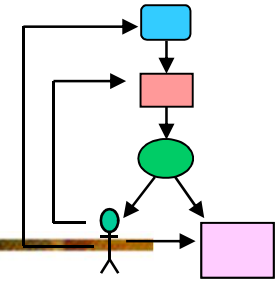
Competency Questions

Formulating Competency Questions



- A challenge in using Competency Questions is the design of an appropriate set of questions.
- How would you obtain appropriate competency questions:
 - The purpose of your model.
 - Literature.
 - Other similar cases.

Case Studies



- **Company A: Small Consulting Company**, which maintains a database of highly qualified consultants in various fields and draws upon these resources to form project teams.
- **Company B: Student Project Groups**: A company that operates in the oil and offshore industry, on a global scale. The particular scenario that we analysed was the selection of several groups of students who will work together as teams during their summer holidays.
- **Company C: Building Construction Project**: Company C is hired by a customer to assist them to evaluate bids in the selection of a contractor for large scale projects in various domains.

Example1: Do all agents contribute to the goals of the VE?

```
performs(role, activity, agent) :-
```

```
    filled_by(role, agent),
```

```
    performed_by(activity, role).
```

```
is_assigned(agent, goal) :-
```

```
    performs(role, activity, agent),
```

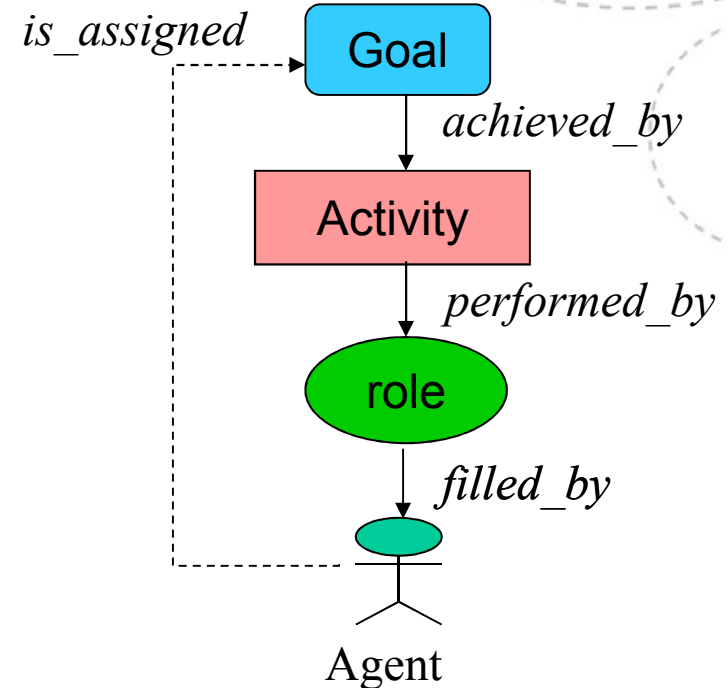
```
    achieved_by(activity, goal).
```

```
for all agents
```

```
    if      (is_assigned(agent, goal))
```

```
    then    (all agents contribute to the
```

```
        goals of the VE).
```

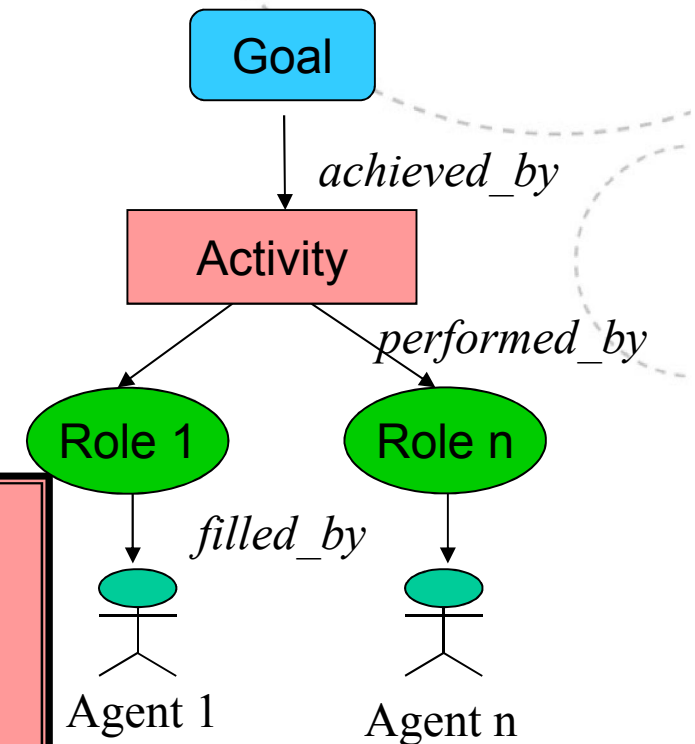


Example 2: How do we know when VE formation is complete?

- VE formation is complete when all roles are filled by agents.
- All goals are assigned activities, all activities have roles that perform them.

for all roles

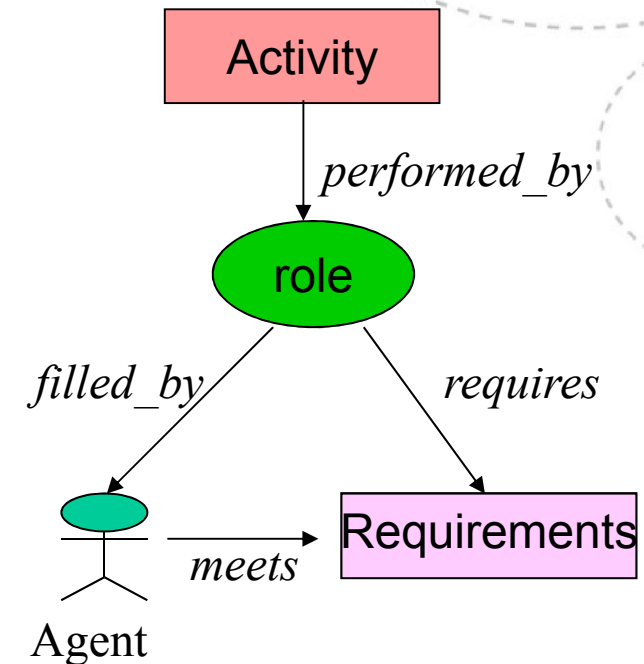
if (performs(role, activity, agent))
then (VE formation is complete).



Example 3: What are the skills required to perform a particular activity?

performed_by(activity, role).
 filled_by(role, agent).
 requires(role, requirements(Skills,
 availability, price)).

if (performed_by(activity, role) and
 requires(role, requirements(**Skills**,
 availability, price))
 then (skills required for activity = **Skills**).



Evaluating Models

- The Competency Questions evaluate the competency of a model.
- For a complete evaluation, other characteristics of the model must also be evaluated.
- Fox et. al. proposed a set of characteristics to evaluate enterprise models.

Considering other characteristics

<i>Characteristic</i>	<i>Our Agent-based Model</i>
Generality	Can be applied to VEs in different domains.
Transformability and efficiency	Since model is described in formal logic, it can easily be transformed into a computer-based model that can be used for efficient reasoning.
Perspiciuity	Since the model can be described in semi-formal syntax, the users can easily understand it.
Granularity and scalability	Can be enhanced to cover the complete lifecycle of the VE.
Extensibility	Currently, the model is very generic. It can be enhanced to include concepts such as "change", using rules.

Summary

- Presented an agent-based model to support the formation of VEs.
- Presented a few examples of how Competency Questions have been used to evaluate the model.