Three Roles of Conceptual Models in Information System Design and Use

Roel Wieringa

Department of Mathematics and Computer Science
Vrije Universiteit
De Boelelaan 1081
1081 HV Amsterdam
The Netherlands
uucp: roelw@cs.vu.nl

This paper attempts to draw together results from information systems research, linguistic theory, and methodology in order to present a unified framework in which to understand conceptual models. Three different roles of conceptual models (CM's) in the design and use of information systems (IS's) are investigated. The descriptive role of a CM is that it is an abstract representation of the universe of discourse (UoD) of the IS; the normative role of a CM is that it contains prescriptions for the behavior of entities in the UoD. A third role of CM's emerges when a computer is viewed as a symbol-manipulating machine capable of performing speech acts like commanding and promising. These acts are commands or promises only against a background of shared conventions, which is stored in a shared CM. A CM playing this role is called institutional. This paper is an abstract of Wieringa [1989].

1. Introduction

The aim of conceptual modeling is to specify an explicit conceptual model (CM) of a universe of discourse (UoD). The concepts of CM and UoD are defined in section 2 below; here, we assume a preliminary understanding and sketch the general picture of the relation between a CM, its UoD, and an information system (IS) that implements the CM (figure 1.) This general picture deviates somewhat from the view of CM's in the standard three-level IS architecture and allow us to isolate the problem studied in this paper more clearly than can be done in the standard view.
The relation between the CM and the UoD is preliminarily taken to be one of abstraction. The CM itself is an abstract mathematical structure, which is a logical model of a theory, i.e. a set of sentences in a formal language, called a CM *schema* or CM *specification* (CMS). The CMS is a linguistic entity which can be entered in a computer, which, like any universal Turing machine, can then simulate the CM, using the CMS as instructions. We thus view the implementation as a symbol-manipulating machine.

The CM as preliminarily defined above provides a formal basis for common understanding of the UoD. This includes anything relevant about the entities in the UoD, static as well as dynamic, but nothing that is irrelevant. A secondary role of the CM is that, through its CMS, it defines the allowable ways in which information about the UoD can be stored and manipulated. This gives us two of the four roles listed for CM's in the ISO report (van Griethuysen [1982], p. 1-9). The third role listed is to provide a basis for interpretation of external and internal syntactical forms which represent information about the UoD. If this interpretation is done by people, then this, too, is one of the roles of the CM as construed in this paper, but if the interpretation is executed by the computer, then it is not a role playable by the CM as construed in this paper but by another kind of model, which we here call a *data model*. A data model describes the structure of the data in the store of a machine in an implementation-independent manner. For example, an object with 128 attributes (e.g. an employee) will be a single object in the CM if it represents a single entity in the UoD, but it may be divided over several records on disk, in which case these records are abstractly represented as different (implementation) objects in the data model. A single object in the CM may thus correspond to several objects in the data model. The data model also can play the fourth role listed in the ISO report, which is to provide a basis for mappings between the internal and external schemas. The data model thus functions as a central schema which is the mapping target between the internal and external levels, while the CM is (in our preliminary understanding) an abstract UoD description. The same separation of concerns between a CM and a data model has been proposed by Kent [1980]. Adding the internal and external levels, we get the overall architecture of figure 2.

![Figure 2](attachment:image.jpg)

The data model is a conceptual model of a database, just as the CM is a conceptual model of a UoD (usually an enterprise). Correspondingly, just as the CM is a logical model of a CM schema, so the data model is a logical model of a data model schema (usually called a database schema). Corresponding to each external view of a data model, there is an
application model (AM), which contains the abstract representations of the entities of interest for a particular application. The arrows in figure 2 represent definability in the corresponding schemas. For example, each predicate in an AM specification must be definable of predicates of the CMS, and the CMS and data model schema must be definable in terms of each other, to guarantee equivalence.

Compared to the ANSI/SPARC architecture (Tsichritzis & Klug [1978]), the data model corresponds to the "conceptual view" of the database and the data model schema corresponds to the "conceptual database schema." However, in that report too, we find the two roles of a model of the UoD and of a model of the database played by a single model, the conceptual view.

Having separated the role of central schema which acts as a target for mappings in the database from the role of model of the UoD, we can now restrict us to the second role. If the CM is a basis for common understanding of the UoD, what is the relation of that understanding to the UoD, and which consequences do these different relations have to the role of a CM in the development and use of an IS? We can answer this question now without any reference to the mappings from internal and external schemas, which are extraneous to the UoD. After defining the concepts of UoD and CM more precisely in sections 2 and 3, we will find as the first role that of a representation of the possible entities, and their possible states, processes and interaction, in the UoD (section 4). This agrees completely with the role of models in the empirical sciences. After a review of the different kinds of integrity constraints in CM's (section 5), we will find a second important role, that of a normative model for a UoD (section 6). This is a reversal of the relation between the CM and the UoD, for in case of a mismatch, the UoD must adapt itself to the (normative) CM, not vice versa. It is important, therefore, to be explicit about whether a particular CM plays this role. If conceptual modeling results in an implemented IS, which is used in the UoD, then the computer is one of the possible actors in the UoD. It may perform physical actions like opening and closing elevator doors, but also actions like blocking a bank account or granting a parking permit. This latter type of action requires description and justification by explicit rules in a publicly shared CM in order to succeed. A CM containing this kind of rules is playing an institutional role (section 7). It is an important task in conceptual modeling to formulate explicitly the rules which make a CM play this role. Section 8 gives a brief summary of the paper and mentions some related work.

2. The universe of discourse

Definition 1.

The universe of discourse (UoD) is the set of possible entities which currently are of interest to the relevant people. Entities of interest which are people are called information subjects.

This definition is based on the ISO definition of UoD, with the following adaptions.

1. It emphasizes that the entities are of interest for the "relevant" people, thus pointing out the problem of determining who is relevant, and who should therefore be consulted in the development of the CM.

2. It replaces the emphasis the ISO puts on the time of existence of the entities by an emphasis on the fact that we are talking about a set of possible entities. Each of these
entities, however, is taken to include the events and processes which these entities execute.

3. It points out that only the set of possible entities which are currently thought to be of interest are included in the UoD. System boundaries could not be drawn if everything which possibly might become of interest should be included in the system.

There is an ambiguity in the word "of" in "UoD" in that it does not make clear whether discourse is *about* the universe but does not take place in it, or whether discourse takes place *in* it but is not about the universe in question, or whether it is both in and about the UoD.

**Definition 2.**

Discourse about a universe is called *external* to the UoD if it is not conducted by the information subjects in the UoD, otherwise it is called *internal*. A universe is called *physical* if discourse about it is external to it, otherwise it is called *social*.

The choice of the term social is motivated by that fact that the only type of entity able to talk about its universe is man. Examples of physical UoD's are the universe of a CAD/CAM system, or of foraging animals in wildlife area; examples of social UoD's are human organizations. This must be qualified by the remark that by definition, a UoD is physical if the information subjects do not conduct discourse about the UoD of an IS. This makes the concepts of social and physical UoD are relative to a discourse. If information analysts conduct discourse about a human organization but do not share their discourse with the people in the organization, then with respect to that discourse the organization is a physical UoD.

3. Conceptual models

**Definition 3.**

*A conceptual model* (CM) of a UoD is an abstract entity which embodies a common understanding among the relevant people of the UoD.

Some of the relevant people are information analysts, database (DB) designers, programmers, DB administrator, users, specialists about the UoD, and, in a social UoD, information subjects. In a social UoD users, specialists and information subjects often are the same persons.

A CM embodies a common understanding of the UoD, but it need not embody a unique understanding or even an understanding thought to be reasonable by all parties concerned. It is important to realize that a CM is an object of common understanding and must be defined through a process of *negotiation* between various groups of people. In social UoD's one of these is the group of people living in the UoD. This makes conceptual modeling of, and in, social UoD's considerably more haphazard than conceptual modeling of physical UoD's, since there will exist different versions of reality ("CM's-in-use") in different groups of relevant people. In this negotiating process, many aspects of the UoD will be thoughtfully left out and others may be thoughtfully distorted by the relevant people, and those not only irrelevant details.

In social UoD's there is often a non-empty intersection of the CM and the UoD. For example, the concept of employee is a universal concept which exists in the UoD, but exists in the UoD only because it exists in a shared CM of the UoD. An individual employee like John Doe, on the other hand, is clearly separate from his representation in a CM. But an individual conceptual object like the permission to John Doe to park his car in the company
parking lot, is a conceptual object in the CM as well as in the UoD. Other examples are numbers and truth values, which, if they exist at all, exist as conceptual entities in a social UoD and in a CM used in that UoD. In what follows, we ignore any overlap between CM's and social UoD's.

4. Descriptive models, abstract objects, and object identity.

Definition 4.

A descriptive model of a UoD is an abstract system which represents the entities in the UoD at a certain level of approximation. ■

Remarks:

1. A description presupposes a concept of matching between description and described. What matching is, precisely, is left unanalyzed in this paper, but given this concept, a descriptive model's representing the UoD consists of the descriptive model being true or false of the UoD. This implies that if there is a mismatch between the descriptive model and the UoD, the descriptive model is wrong and should be corrected.

2. A descriptive model represents the UoD at a certain level of abstraction, so that, if it is true, it can only to be said true at a certain level of approximation. Parts of the UoD are not modeled at all, and of those parts which are modeled irrelevant details are not represented, and of those aspects which are represented, some are represented with a "rounding error." The abstraction relation between a descriptive model and its UoD is analogous to the relation between ideal point masses and real masses, or between ideal gasses and real gasses in natural science.

3. A descriptive model is defined to be an abstract system. This means that it has a state space and a current state which moves through the state space as described by a state evolution function. We only consider discrete state spaces and talk about state transition functions, which assign a (next-state, output) pair to each (state, input) pair. (In nondeterministic systems there is a set of (next-state, output) pairs for each (state, input) pair.)

The last point must be elaborated, for a CM is a system at two levels of aggregation, that of individual objects and that of a domain of existing and communicating objects. First, a particular employee entity e in the UoD is represented in a descriptive model by an abstract object o, which is an abstract system with a state space and state transition function (cf. figure 3, where the dashed arrow represents abstraction).

The state space of o is defined by the attributes of o, and a particular state is a vector of attribute/value pairs, e.g. (name: John, address: a, salary: 3500). Now, it has been argued elsewhere that attributes must be attributes of something which is itself not an attribute. We therefore postulate an underlying identity for e, which is represented in the descriptive model


2. Khoshafian & Copeland summarize the arguments for CM's. These duplicate part of the theorizing which has been done in analytic philosophy about the identification of individuals, see Loux [1970] for an overview of these. Wieringa [1989] briefly compares the two research traditions, which do not seem to have had any contact.
Descriptive Model UoD

Figure 3. Abstract object and the UoD entity it represents.

by a surrogate $s$, which identifies the abstract object representing $e$ in a certain state. Thus,

$$ o = (s, (\text{name} : \text{John}, \text{address} : a, \text{salary} : 3500)) $$

represents an employee in a certain state, and

$$ o' = (s, (\text{name} : \text{John}, \text{address} : a, \text{salary} : 4500)) $$

represents the identical employee in a different state. $o$ and $o'$ are mathematically different objects, $o \neq o'$. We can imagine $o$ and $o'$ as two points in the state space determined by $(\text{name} : \text{String}, \text{address} : \text{Address}, \text{salary} : \text{N})$ and labeled by $s$. The two points are different positions in this state space, but they are identified by the same label $s$.

The identity $s$ of an abstract object $o$ contains three pieces of information (cf. Strawson [1970], who attributes these features to individuals in general):

1. it identifies $o$ as an instance of a universal concept,
2. it distinguishes $o$ from all other possible instances of the universal concept, and
3. it identifies the same instance of the universal concept in different states.

$s$ is thus a principle of object instantiation, identity and persistence. This exemplifies the close connection "between the identity [...] of an entity and its persistence, between its persistence and its existence, and between its existence and its being the kind of thing it is." (Wiggins [1980], pp. 54-55. See also pp. 60, 62, 86) "The kind of thing it is" is another way of saying "under which universal concept it falls."

The relevance of this for IS's can be illustrated by the following example. If we count the number of passengers a transport company has carried in one week, the answer must mention the universal of which we are counting instances: "1000 persons" is not the same answer as "1000 passengers", if a passenger is the kind of entity which is individuated by buying a ticket. Thus, the three properties of surrogates provide a principle of counting instances of a universal concept. Incidentally, this implies that the objects are classified by universal concepts whose names are count nouns rather than mass nouns. Examples of count nouns are "person," "employee," "project." Instances of these concepts are discrete, countable objects. Examples of mass nouns are "water," "profit," "wood." Instances of these universal concepts must be measured instead of counted. See Gabbay & Moravcsik [1973] for more in this.

Apart from a state space, a system has a state transition function. We do not go into specification languages for state transition functions here; examples can be found, among others, in Sernadas et al. [1987], Wieringa & van de Riet [1988].
Turning to the domain level of aggregation, we must specify a system, which we may call the conceptual database, whose state at any moment is given by 1. the identities of the existing objects and 2. the state of each existing object. The state transition function of the descriptive model is given by 1. how objects are created and destroyed, 2. which processes existing objects execute, and 3. how the existing objects communicate. Again, we refer to the literature listed above and the references mentioned there for proposals of ways to specify this.

5. Integrity constraints

The state space and state transition function of each abstract object, and the creation, deletion, and communication patterns of objects, are integrity constraints (IC's) built into the specification itself, i.e. one can only specify a descriptive model by specifying these constraints. In addition to these built-in IC's there are general statements which can be attached to universal concepts like employee, person and project. For example, to Person the statement "all persons have an age" may be attached. We want to distinguish six kinds of IC in this section, illustrated in table 1.

<table>
<thead>
<tr>
<th>Type</th>
<th>Static</th>
<th>Dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Necessary</td>
<td>analytical</td>
<td>An employee must be hired before s/he can be fired.</td>
</tr>
<tr>
<td></td>
<td>age ∈ N</td>
<td></td>
</tr>
<tr>
<td>Empirical</td>
<td>age &lt;150</td>
<td>Animal of species X always feeds in the morning.</td>
</tr>
<tr>
<td>Deontic</td>
<td>The balance of a bank account should not be less than n.</td>
<td>A library user should return a borrowed book after at most 6 weeks.</td>
</tr>
</tbody>
</table>

Table 1.

We discuss the examples in the left column in detail.

The constraint age ∈ N is an analytical statement, i.e. one which follows from the meaning of the symbols occurring in it. Analytical truths are necessarily true in all states of the descriptive model.³ Analytical truths are not empirical in that they cannot be falsified by events in the UoD; however, they can be construed as empirical statements about discourse about the UoD and can therefore be falsified by that discourse. Two interesting properties of analytical IC's are the following.

³ See Kripke [1977], [1980] for subtle differences between necessity and analyticity. At least the statement that all analytical statements are necessarily true emerges as a necessary truth from that discussion.
1. A change of an analytical constraint is a change of the meaning of some of the symbols occurring in it. For example, we may decide to measure temperature in degrees Kelvin instead of degrees centigrade. The analytically constraint \( \text{temp} \in \mathbb{R} \) then becomes the analytically true (in the revised language) constraint \( \text{temp} \in \mathbb{R} \wedge \text{temp} \geq 0 \) (by definition, degrees Kelvin start at 0).  

2. An analytical constraint can be used to check if the current state of an implementation implements a possible state of the CM. It cannot be used to check a state of the UoD, for it is necessarily true of any state of the UoD (given the meanings attached to the symbols in a discourse). If in the implementation a record has a negative value in its age field, the implementation does not represent a possible state of the UoD. (If the implementation represents a possible state, it may represent the correct state.)

Consider now the example \( \text{age} \leq 150 \). This is not a necessary truth at all but depends upon contingent factors like the state of our general health. It is an empirical truth about our UoD which our model, as an abstraction of the UoD, should satisfy. The situation is analogous to empirical laws in natural science. Like the famous Boyle/Charles law for ideal gasses \( PV/T = C \), the law \( \text{age} \leq 150 \) defines a subset of the total set of all logically possible states of the world.

1. Barring change of meaning, a change of empirical constraint is a change in the way the UoD behaves.

2. If the implementation violates an empirical constraint, the implementation or the constraint may be wrong. Empirical statements are falsifiable, which means that if they are in conflict with the UoD, the UoD is right and they are wrong.

In order to use empirical IC's as constraints on the allowable states of an implementation of a DB, we should formulate them so weak that it is absolutely certain that they are true of all states of the UoD during the useful life of the descriptive model (e.g. 10 or 15 years). Thus, though \( \text{age} < 150 \) is an empirical generalization about the age of human beings, we can treat it as if it were a necessary truth and therefore use it to constrain the possible states of an implemented DB.

The example constraint that the balance of a bank account should not be less than a number \( n \) is not an analytical truth following from the meaning of the symbols used in the statement, nor is it an empirical generalization about bank accounts. It is a rule instituted by agents in the UoD to constrain the possible states of the UoD. We will call these constraints deontic. In general, deontic IC's express obligations, permissions and prohibitions for agents in the UoD. Like empirical constraints, deontic constraints can be violated by the UoD. But where in the case of empirical constraints this is a novel way for the UoD to behave which is logically possible and ethically neutral, in the case of a deontic constraint this would be a case of behavior which is logically and physically possible but ethically inadmissible (at least

---

4. We should note here the difficulties of finding a criterion which distinguishes analytical statements, whose truth-value can be determined by an analysis of the meanings of the symbols occurring in it, from empirical statements. Analytical statements are immune to falsification by events in the UoD and empirical statements are not. However, as Quine argues, immunity to falsification is not a property of a statement in isolation but depends, among other things, on the degree of entrenchment of the statement in a theory. See Davidson [1984], Quine [1951]. Suppe [1977] gives a good summary.
according to the rule being violated).

1. A change of a deontic constraint is a change in the norms pertaining to the UoD. For example, we may decide to allow an even more negative balance.

2. A deontic constraint can be used to check if the agents in the UoD behave in a permissible way. With the constraint in hand, proper action can be taken if a balance exceeds a negative limit. Deontic IC's do not constrain the implementation, but the UoD The implementation must be able to implement a CM which correctly represents the fact that the UoD violates deontic constraints.

**Definition 5.**

A descriptive law is a necessary truth about the UoD and a prescriptive or deontic law is a normative statement about the UoD. A statement is analytically true if its truth follows from the meaning of the symbols occurring in it, it is empirically true if it is a truth about the UoD.

**Definition 6.**

An IC is a prescriptive law for an implementation.

The relevance of the distinction between necessary and normative laws is that violation of prescriptive laws about the UoD do not require any corrective action in an implementation of a CM, but that violations of descriptive laws are violations of IC's and do need corrective actions in the implementation. If, as a matter of fact, a violation of a prescriptive law in the UoD is corrected, then the violation is still part of the history of the UoD, but if an implementation error has been corrected, the erroneous state is not part of the history of the UoD and is usually obliterated from the history of the implementation.

**Definition 7.**

A law is static if it concerns each state of the UoD, and it is dynamic if it concerns at least two states of the UoD, and it is called a communication law if it concerns the interaction between entities in the UoD.

The examples of dynamic laws in table 1 are self-explanatory.

**6. Normative models and direction of fit**

**Definition 8.**

The direction of fit between two entities is an arrow which points to the entity to which the other entity must adjust itself in case there is a mismatch between the two. If the direction of fit is from A to B, then B is called normative for A.

The definition presupposes that there is a meaningful concept of matching between the two

---

5. Life is not so clear-cut as this definition suggests. The distinction between analytical and empirical statements has been attacked vigorously in the past thirty years (see note 4) and that between empirical statements and normative statements is not uncontroversial either (cf. Austin [1962], p. 149 and Searle [1969]).

6. This classification is analogous to van Fraassen’s [1970] classification of empirical laws into laws of coexistence (like the Boyle/Charles law PV/T=C) laws of change, (like F = m.a), and laws of interaction, which say how entities interact with each other when they are composed into larger systems.

entities. In this paper we encounter two such concepts: mismatch between description and described, and mismatch between prescription and the entity whose behavior is prescribed. In each case, the direction of fit is in the direction in which we can find "the boss," which is the entity to whom the other must adjust. The UoD is normative for a descriptive model DM of it (figure 4).

Definition 9.
A normative model (NM) for a UoD is a system whose states and behavior is normative for the UoD. ■

Figure 4. Direction of fit for descriptive models.

Figure 5. Direction of fit for normative models.

Note that we speak of a normative model for a UoD, in contrast to a descriptive model of a UoD.

By "norms" we mean rules which specify what is permitted, forbidden or obliged in certain situations in the UoD. Continuing to view a CM of the UoD as a set of possible states and a state transition function, permissions and prohibitions can concern states (it is permitted or prohibited to be in a state) as well as actions (it is permitted or prohibited to do something). Obligations, on the other hand, can only concern actions. An obligation is always an obligation to do something before a specific point in real time.

A deontic law constrains the UoD, not an implemented DB which must represent a state of the UoD. In fact, it may be desirable that the DB be able to represent a violation of a deontic law. In this case, the deontic law should not be implemented as a DB constraint at all; rather, the DB should be able to specify that a deontic law is being violated.

Many examples of IC's have been given which actually are deontic laws for the UoD. Examples of deontic laws presented as necessary truths of the UoD are

(1) "All suppliers supply at least one part." (Reiter [1984], p. 196)
(2) "No company must supply two different departments with item 1." (Nicolas [1982], p. 235)
(3) Only increases in salary are allowed (Weber et al. [1983], p. 126)
(4) "During the month of January only, a car may be declared to have been produced in the previous year." (Griethuysen [1982], p. B-1).
Other examples can be readily found. The UoD of the last example has been used in quite a number of papers.

When a CM is a normative model, the norms embodied in it have been negotiated by, or at least with, the people within the UoD, but if the UoD is physical, they are simply imposed upon the UoD. The information analyst should play the role of observer in this process, or at most of advisor if s/he is an organization specialist.

Classification in a normative model is a normative activity, which determines which sets of rules apply to the object classified. For example, if a meeting of students and teachers is classified as a course, then the rules for course become applicable, such as that each course should have one teacher responsible for it. If it is on the other hand classified as a seminar, quite different rules may become applicable.

If a CM plays the role of descriptive model as well as of normative model, then to each universal concept, descriptive as well as prescriptive laws have been attached. For example, Cars ought to be registered, Courses ought to have one Teacher responsible for them, Employees ought to be at their work at certain times. Figure 6 shows this situation with respect to CM's.

![Figure 6](image)

**Figure 6.** Combined descriptive and normative model.

The CM is now related to the UoD with two directions of fit. If the UoD is physical, the CM may be negotiated by the UoD specialists in such a way that it can be implemented and the implementation connected to the UoD by a feedback loop. In this loop, information passes between the UoD and the implementation in the opposite direction to that of the direction of fit. An example is an elevator system, which receives descriptions of the state of the buttons, elevators, etc. and issues motor and door commands to achieve optimal service according to some norms fixed in the CM.

If the UoD is social, the CM is part of the UoD and describes and prescribes the behavior of the UoD in a process of social discourse and control. The CM as negotiated by the inhabitants of the UoD in this case is usually not fit for implementation, for at least the following reasons.

1. Descriptions of what is the case are not agreed upon. Not only may there be many different points of view, these views may differ to such an extent that no underlying common CM exists.

2. Not only opinions about what is the case may be irreconcilable, opinions about what ought to be the case usually differ even more.

---

8. Good politicians are usually masters in the game of suggestive classification.
3. Closely connected to this is that the CM as it is actually used in discourse taking place in the UoD, the CM-in-use, is often hardly explicit, and explicating it to the extent needed for IS development may be a never-ending process.

4. Even if a common descriptive and normative CM can be negotiated in finite time, it will in general cover so many aspects of the UoD that implementation is prohibitively expensive. This leads to a simplified CM, which descriptively models an explicated CM-in-use which in turn is descriptive as well as normative (figure 7). By agreement and in certain circumstances, the simplified CM may be used as a descriptive and normative model of and for the UoD, instead of the CM-in-use it is derived from.

Figure 7. A simplified version of a CM-in-use may be defined and used.

Arrows 1 and 2 connect a CM-in-use to a UoD, and arrow 3 indicates that $CM_2$ is an abstraction, and therefore a simplification, of $CM_1$. A CM for IS development is usually like $CM_2$ a simplification of one or more real CM's. $CM_2$ must satisfy the following requirements.

1. There is a balance between its usefulness and the simplification it embodies,
2. it describes less as forbidden than the CM-in-use forbids (it does not stand in the way of human actors in the UoD), and
3. it describes more as possible than what is really possible in the UoD (it is always able to represent the current state of affairs). 9

CM's which combine the descriptive and normative role can be used to built IS's which

---

9. cf. Griethuysen [1982], p. 2-14, where the opposite rules are formulated. The first rule formulated is that the IS should not be able to represent impossible states of the UoD, where I claim that it merely should be able to represent all possible states (and may, in the interest of simplicity of the implementation, be able to represent some impossible states of the UoD as well). The abstraction from the UoD to the CM involves loss of knowledge, which in terms of epistemic logic means that the set of possible states increases. The second rule formulated in the ISO report is that what is forbidden in the UoD should be represented as forbidden in the IS, whereas I maintain that the IS should merely be able to represent all permitted states of the UoD as permitted (and may represent some forbidden states as permitted as well). Here, too, there is a decrease of knowledge when we go from the UoD to the IS. My reason for putting the rules this way is that I would like an IS to be as unobtrusive as possible.
register data about the UoD as well as control the UoD by reducing the difference with the norms represented in the CM. This requires the connection of the IS to the rest of the UoD by means of a feedback loop, through which descriptive information flows from the UoD to the IS, and prescriptive information from the IS to the UoD. In a physical UoD, this can be realized by physical means. Elevator systems, radar systems etc. are controlled in this way. In social UoD’s, the entities to be controlled may be people, or may be conceptual entities like bank account or obligations. These cannot be controlled by physical means, simply because they are not physical objects, or because ethical considerations prohibit it. There is however, an important mechanism by which social UoD’s may be controlled by computer, discussed in the next section.

7. Institutional models

The distinguishing feature of social UoD’s is that discourse about it is conducted, among others, by entities living in it. Now, linguistic agents in a linguistic universe may perform speech acts. I suggest that speech acts are the mechanism by which the feedback loop to a social UoD is closed. Implementations belong to the linguistic actors in social UoD’s, and influence social UoD’s by speech acts. I give a brief introduction to some concepts of speech act theory and then apply them to conceptual models.

A speech act is not a proposition but the utterance of a proposition. The utterance of a proposition may have the force of an assertion, denial, argument, commitment, promise, command, directive, apology, declaration, etc. These forces are called illocutionary forces in speech act theory (Searle [1969], [1970], Searle & Vanderveken [1985]). When a proposition is uttered, the utterance has a propositional content, which is the proposition uttered, and an illocutionary force. It is impossible to utter a proposition without an illocutionary force.

Searle & Vanderveken [1985] give a formal analyses of illocutionary force, of which we take up three elements. First, like any act, a speech act can succeed or fail. For example, a command to leave the room fails if the speaker is not in a position to command the hearer, and a priest may attempt to excommunicate someone, but will be unsuccessful if proper procedure is not followed. If a speech act is not successful, it fails.

Second, like any act, a successful speech can be successful in that the effect is reached, but in a defective way. For example, a priest may defectively excommunicate someone, even though the excommunication succeeded. The defect may be that the grounds for excommunication were insufficient or that faulty evidence was used; but if proper procedure is followed, excommunication took place.

Third, each speech act has conditions of success, which state how it should be performed in order to succeed. What the conditions of success of an act are depends mainly on one part of illocutionary force, called the illocutionary point of the speech act. There are only five illocutionary points, which differ mainly in their direction of fit. We give them, together with some conditions of success.

1. An illocutionary force has the descriptive point iff the proposition expressed is presented as representing an actual state, event or process in the UoD. The direction of fit is from the proposition to the UoD, for if there is a mismatch, the expressed proposition is false. A successful and nondefective performance of a descriptive speech act requires that the speaker has reasons that count in favor of the truth of the expressed
proposition. Examples of illocutionary forces with the descriptive point are state, deny, inform, and respond.

2. A speech act has the *directive* point if it is an attempt to make agents in the UoD to do things. The direction of fit is from the UoD to the expressed proposition, for if there is a mismatch, the UoD is supposed to adapt itself. A successful and nondefective performance requires at least that the agent is capable of doing what s/he is directed to do. Directive speech acts also create reasons for the agent to do what s/he is directed to do. Examples of directive performatives are ask, command, request, permit, and forbid. Some of these have special conditions of success. For example, to command a person, the speaker must be in a *position of authority* over the hearer.

3. The direction of fit of a *commissive* speech act is the same as that of a directive, except that the future course of action is supposed to be executed by the speaker. Examples of commissives are promise, refuse, guarantee.

4. A *declarative* speech act changes the UoD by saying that it is changed. For example, you may be *declared* to be a credit-worthy customer, the research proposal may be *approved*, a chairperson may be *appointed*, a meeting may be *adjourned*, etc. Successful declarations are by definition true and they have changed the state of the world; whence the double direction of fit. All declaratives, except supernatural and purely linguistic declaratives, require an nonlinguistic institution for their successful performance. For example, the declaration of credit-worthiness presupposes the institution of buying and selling, opening a meeting presupposes the institution of meetings, etc. Note that all declaratives, except supernatural and purely linguistic declaratives, achieve their purpose by the fact that *the speaker has the authority to perform the declarative act*, and invokes this authority in uttering the speech act. For example, only the (acting) chairperson has the authority to open a meeting, etc.

5. The point of *expressive* speech acts, finally, is to express the state of the speaker. Examples are thanking, welcoming, greeting. There is no question of a match between an expressive and the UoD, so expressives have null direction of fit. However, like other speech acts, they can be insincere.

Note that this concept of direction of fit applies to speech acts, while in section 6 we applied it to CM's. This could be done by concentrating on the common element with speech acts, that of *matching* of two entities.

Speech act theory has been applied to conceptual modeling, by several researchers (Auramäki et al. [1988], Flores & Ludlow [1981], Kimbrough et al. [1984], Lehtinnen & Lyytinen [1986], Winograd & Flores [1986]). However, these all use speech act theory to define a formal logic of events to be represented in the CM and specified in the CMS. We

10. An example of a supernatural declaration is God's pronouncement "Let there be light!" An example of a purely linguistic declaration is the definition of a term. Neither needs a nonlinguistic institution to achieve its purpose.

11. Some declaratives have descriptive content as well, as for example the statement of an umpire that the ball is out. In descriptive declaratives, a state of affairs is ruled to be the case after a fact-finding procedure. The umpire's statement that the ball is out makes it so in the context of the game, regardless of the precise facts of the matter. Similarly, if a judge decides, on appeal, that you are guilty, then for legal purposes you are guilty.
apply speech act theory on the other hand to the acts of an implementation itself, whether or not the implementation works according to an advanced logic. Our application proceeds in three steps. First, we note that implementations can perform speech acts; second, we note that this may require the previous delegation of authority to the implementation to perform such an act; third, we point out that this requires an institutional context in which these acts can be performed, and that it is a role of conceptual models to provide this context.

**Definition 10.**

1. **An implementation** of a CM is a machine which can store a CMS and manipulate data according to it.
2. **Automation** is the delegation of tasks performed in a UoD to a machine.

To have an implementation able to perform certain tasks is thus not sufficient to automate a task. For automation, an explicit delegation of the task must take place, so that the implementation can replace a certain actor in the UoD. The implementation thereby becomes part of the UoD. The application of speech act theory to this situation runs as follows.

1. The machine we are interested in are computers, which are symbol-manipulating devices. They are thus capable of speech acts. Speech acts may be uttered by being spoken aloud, but may also be uttered by being written down on a paper, or by being typed onto a keyboard, or by being printed on a screen. A request by an implementation for a password is a speech act with a directive point, and each answer to a query is a speech act with a descriptive point. If a DB refuses to perform a certain action (e.g. because a wrong password has been given) then it performs a speech act with the commissive point, and if it prints "Hello" on the screen it performs an expressive speech act. Finally, a computer may, after considering the data, declare a UoD entity to be in a state. Speech acts with every possible illocutionary point may thus be performed successfully and nondefectively by a computer. In other words, speech acts may be implemented in a computer.

2. However, automation requires additionally that tasks be delegated, and delegation of some speech acts to a machine may require previous delegation of the authority to perform those acts to the machine. A neat example of this is a system of traffic lights, to which we have delegated the authority to direct us to stop. Without this previous delegation (or with a massive ignoring of this previous declaration as is the case in Amsterdam), traffic lights tell us nothing. Declaring customers credit-worthy, blocking bank accounts etc. are directive and declarative speech acts which similarly require previous delegations of authority. Note, incidentally, that the previous delegation of authority to a machine is itself a speech act which in its turn usually requires special authority to be performed. This authority must itself have been delegated beforehand, etc.

3. All of this requires an institutional context of rules which define when these acts are performed successfully and nondefectively. These rules are called constitutive (Searle [1969], pp. 33 ff.). To be in agreement with our terminology of laws in section 5, we will also call them constitutive laws (on a par with descriptive and deontic laws). An example of these laws are the rules of soccer. Like deontic laws, these prescribe the behavior of certain entities, i.e. the players of the game. But additionally, the entities whose behavior is prescribed do not exist at all outside the context of these laws, and
their behavior cannot be described without presupposing these laws. A person is a keeper, and can be described as such, only in the context of the rules of soccer. Similarly, to call someone a customer is to presuppose the institutions of buying and selling, without which there would not even be a customer; and to appoint someone chairperson presupposes the rules governing the social institution of meetings. The same constitutive laws which create and govern the institution of meetings are also presupposed when the chairperson opens a meeting.

In the case of the soccer player, there are two facts, the brute fact that somebody is a person, and the institutional fact that somebody is a keeper. Next to the brute fact that a ball passes between two poles, the institutional fact that a goal is made exists (provided the umpire decides so). \( \text{age}(p) = 30 \) is a physical fact, but \( \text{name}(p) = \text{John} \) is an institutional fact relying on the institution of personal names. (The institution of language is left out of consideration here; if we add this, all linguistic interaction has an institutional component.) An institutional fact that does not correspond to a brute fact is \( \text{balance}(\text{acc}) = 3000 \). A brute fact that does not correspond to an institutional fact is the fact that elevator doors are open.

**Definition 11.**

A constitutive law is a normative statement about the UoD which holds for entities that exist only because these laws exist, and can be described only by presupposing these laws.

Note that constitutive laws are also deontic. Most examples of deontic constraints given in section 5 are also constitutive laws, or at least part of a constitutive system of laws (which jointly create, describe and prescribe a kind of object and its behavior).

Constitutive laws always have the form "A counts as B in context C," where A and B are events, processes, or entities. For example, to declare a customer credit-worthy is to count him as credit-worthy, and the pronouncement by the chairperson of certain words in the context of the beginning of a meeting counts as the opening of the meeting.

We have now found a third important role of CM's, which is that of providing a system of constitutive laws for a UoD.

**Definition 12.**

An institutional model is a CM containing constitutive laws.

Since an institutional model creates facts just by the fact that it presents them to exist, it has a double direction of fit with the UoD (figure 8).

![Figure 8](image-url)

We emphasize again that there are several CM's-in-use in the UoD, all of which may play descriptive, normative and institutional roles, and some of which may actually
contradict each other. During conceptual modeling, one CM is defined to be the CM for the IS, and this will in general be an abstraction and simplification from the CM’s-in-use, and may very well be in contradiction with some CM’s-in-use.

Also, it must be emphasized that we are interested in the institutional role of a CM with respect to an implementation, and not at the way speech acts may be formally represented in an implementation. However, it is quite usual to represent constitutive laws as exceptionless necessary truths, as is done with many deontic and empirical laws. For example, the law that all persons with a student number are students is a translation of a constitutive law into a necessary truth.

We can now look at the way feedback loops can be realized in social UoD’s. Two kinds of automation can be in play here, which we will call ontological and performative.

**Definition 13.**
The declaration that an implementation contains certain entities is called *ontological automation*. The declaration that an implementation has the authority to perform certain speech acts is called *performative automation*. ■

To illustrate the first kind of automation, take the obligation of a library user to return a book. The question whether this obligation is an objective entity, existing independently of the human mind, is metaphysical and does not concern us here. But the question whether the obligation of a particular user consists of an implementation being in a certain state is a practical question which must be resolved during conceptual modeling. If the obligation is an entity independent of the state of the implementation, then the implementation can at most contain a *surrogate* which represents the obligation but is different from it. This is apparent from the fact that the implementation can be wrong in representing an obligation to exist, and must then be corrected. If on the other hand the existence of the obligation is identified with the implementation’s being in a certain state, then the implementation does not contain a surrogate for the obligation and cannot be in error.

To put it differently, looking at the state of the implementation is in all cases at least one of the ways in which the existence of the obligation can be determined. If there are also other ways, if none of the ways of finding out the existence of the obligation is privileged, and if these ways may yield different results, then apparently the existence of the obligation is independent of these ways of finding out whether it exists, and the obligation is an independent entity. If on the other hand querying the implementation is the *only* way to find out whether the obligation exists, then we have no choice but to identify the obligation with a state of the implementation. This is an identification of ontology (what exists) with epistemology (how we find out what exists). (Of course, there may be legal escape procedures to undo facts thought to be established beyond doubt, but these do not concern us here.)

As another example take a bank account, which either is a conceptual entity existing independently from the state of an implementation (which however is set up to faithfully represent this independent state of affairs), or it is nothing but the state of an implementation. Similarly, a button is represented by a surrogate in an elevator control system, but a request for service, signaled to the system by pushing a button, may be usefully identified with its representation in the system. The reason for this is that without a representation in the implementation, the request is considered not to have taken place, i.e. there is no request.
Ontological automation is one of the ways to populate the intersection of the CM and the UoD, the implementation being the means by which this population is realized. Another way is to simply replace an UoD entity by part of the implementation. For example, an electromechanical elevator control system may be replaced by a computerized one. Now, if an entity in the UoD is an entity in an implementation, then a feedback loop can be implemented mechanically, because the entity is now under the physical control of the implementation. Provided there is a CM describing and justifying the feedback loop, this works also in social UoD’s. For example, an ontologically automated bank account may be blocked by an implementation, because it resides in the implementation. In as far as the account is also an entity in the UoD, the control loop with the UoD has been closed. Note incidentally that, because the implementation entity and the UoD entity are the same, there is no delay between the sending and the receival of the control signal. Both events are the same event looked at in a different way. We therefore have infinite "transmission" speed here.

Turning to the second kind of automation, which we called performative, examples abound. An example is the declaration, by a machine, that a certain employee now has a parking permit for the company parking lot, that a customer is credit-worthy, that a bank account is blocked (which also is an illustration of ontological automation), etc. It is important to note that often there are no technical changes to the machine at all when automating performatively. For example, granting a parking permit to a person can be automated by projecting a form on a screen in which personal data are filled in, together with an indication whether the permit is granted. After checking the data the machine registers a permit or a refusal. This action of the machine can be seen as either the registration of a permit/refusal as granted by an employee, or as a grant given by the machine itself on the basis of the data entered by the employee, or, as an intermediate position, as a directive to the person entering the data to grant to permit to the employee. Which of these is the case cannot be determined by inspecting the program but the institutional context in which the program operates. Depending upon whether performative automation took place, the output of a machine may be construed as an advice to a person to perform a certain speech act, or as the performance of the speech act itself, whose validity may or may not be checked by the user.

8. Summary and conclusions

Three roles of a CM in the design and use of an IS have been defined and discussed, the descriptive, normative, and institutional roles. For social UoD’s, the CM actually specified as the result of an information analysis process is usually a simplification of the CM in use in the UoD. In particular, it may not contain any of the rules which say when speech act succeeds or fails. Still, especially in social UoD’s, the normative and institutional roles of a CM require careful consideration during the automation process. The designer of the CM must be aware whether the constraints s/he specifies are analytical truths, empirical generalizations, or deontic laws for the UoD, for these case require different actions to be undertaken when violations occur. The designer must also be aware of the nature of the acts performed by the implemented IS. If the IS requires special authority to perform certain acts, then this authority must be transferred to the IS by a declaration which is public and which proceeds according to publicly known and accepted rules, and the procedure to be followed must be laid down in a system of constitutive laws.
Application of concepts of deontic logic to CM's has been undertaken by Lee [1988] and Wieringa et al. [to appear]. Lee uses several variations of deontic logic, some of which contain paradoxes, as pointed out in Wieringa et al. [to appear]. The current paper arose out of the philosophical analysis given in the last mentioned, which provides a framework for the application of a deontic variant of dynamic logic to the specification of CM's. What is added in this paper is a more principled analysis of the descriptive and normative roles of a CM, and an analysis of the institutional role played by CM's.

Speech act theory has been applied to conceptual modeling by Auramäki et al. [1988], Lehtinen & Lyytinen [1986] and Winograd & Flores [1986], but these consider the computer still as an electronic analogue of a piece of paper which transmits speech acts performed by people, whereas I propose the analysis of the acts of a computer itself as speech acts. This does not require an advanced logic, but does require an analysis of the CM of the UoD in which the machine functions.

9. References

Auramäki E., E. Lehtinen, K. Lyytinen [1988]
"A Speech-Act-Based Office Modeling Approach," Transactions on Office Information Systems, 6(2), 126-152.

Austin J.L. [1962]
How to Do Things with Words, Harvard University Press.

Davidson D. [1984]

Flores F., J.J. Ludlow [1981]

Fraassen B.C. van [1970]

Gabbay D., J.M. Moravcsik [1973]

Gallaire H., J. Minker (eds.) [1978]
Logic and Databases, Plenum Press.

Griethuysen, J.J. van [1982]

Harré R. [1970]
Principles of Scientific Thinking, MacMillan.

Hesse M.B. [1963]
Models and Analogies in Science, Sheed and Ward.
Kent W. [1980]


Khoshafian S.N., G.P. Copeland [1986]


Kripke S.A. [1977]


Kripke S.A. [1980]

Naming and Necessity, Revised and enlarged edition, Basil Blackwell.

Lee R.M. [1988]

"Bureaucracies as Deontic Systems," ACM Transactions on Office Information Systems, 6(2), 87-108.

Lehtinen E., K. Lyytinen [1986]


Loux M.J. (ed.) [1970]

Universals and Particulars: Readings in Ontology, Doubleday.

Nicolas J.M. [1982]

"Logic for Improving Integrity Checking in Relational Databases," Acta Informatica 18, 227-253.

Quine W.V.O. [1951]


Reiter R. [1984]


Searle J.R. [1969]


Searle J.R. [1970]


Searle J.R., D. Vanderveken [1985]

Foundations of Illocutionary Logic, Cambridge University Press.


Suppe F. [1977]

The Structure of Scientific Theories, 2nd edition, University of Illinois Press.

Strawson P.F. [1970]


Tschritzis D., A. Klug (eds.) [1978]


Weber W., W. Stucky, J. Karszt [1983]

"Integrity Checking in Data Base Systems," Information Systems, 8 (2), 125-136.

Wieringa R.J., Meyer J.J Ch., H. Weigand [to appear]


Wieringa R.J., R.P. van de Riet [1988],


Wieringa R.J. [1989]


Wiggins, D. [1980]

Sameness and Substance, Basil Blackwell.

Winograd T., F. Flores [1986]