Chapter 18
Business Ontologies

Peter Rittgen

18.1 Introduction

Ontologies are typically divided into foundational (or top-level), domain and application ontologies (Bugaite and Vasilecas, 2005). Foundational ontologies cover the most general concepts that can be expected to be common to all domains, such as "individuals" vs. "universals" or "substantials" vs. "moments". They are therefore domain-independent. Domain ontologies are tailored for a specific area of human activity, e.g. medicine, electrical engineering, biology or business. Application ontologies further restrict attention to a particular activity in a domain, e.g. the diagnosis of lung diseases in medicine or a computer-based order handling system in business. Figure 18.1 shows the level architecture and names a few examples on each level.

It can be argued, though, whether three levels of ontology are adequate to cover the whole breadth of ontological endeavors. In the business domain, for example, we can identify a number of dimensions that justify further ontological levels. Let us consider a few examples. We distinguish between private-sector and public-sector organizations. Each organization belongs to some industry (banking, car manufacturing, retail, etc.) and it is divided into functional units such as procurement, production, marketing, sales and so on. Along the hierarchy we have the strategic, tactical and operational levels. In addition to these we might also consider a level below the application domain level, the personal level that takes into account, e.g. the way in which an individual uses a particular information system for a particular task which is often different from the way others use the same system for the same or a similar task (Carmichael et al., 2004; Dieng and Hug, 1998; Haase et al., 2005; Huhns and Stephens, 1999).

P. Rittgen (✉)
Vlerick Leuven Gent Management School, Leuven, Belgium; University of Borås, Borås, Sweden
e-mail: Peter.Rittgen@vlerick.com

R. Poli et al. (eds.), *Theory and Applications of Ontology: Computer Applications*,
DOI 10.1007/978-90-481-8847-5_18,

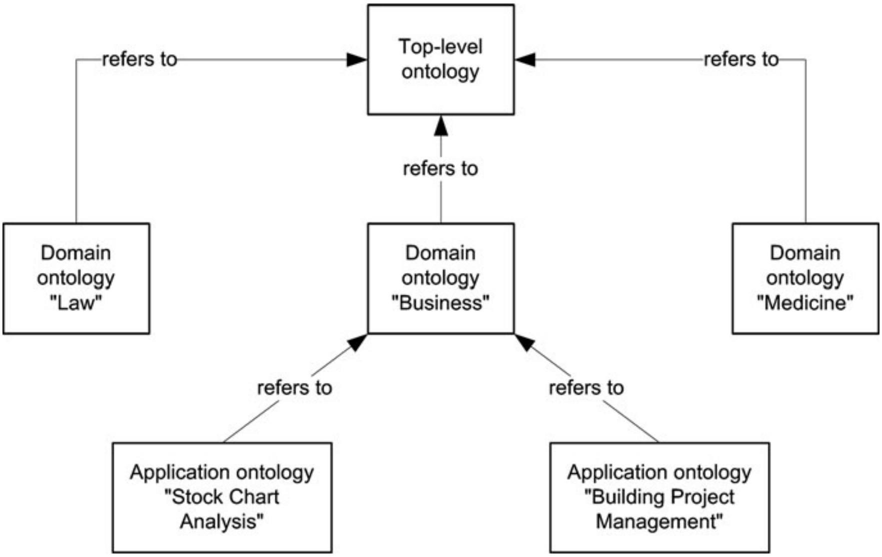

Fig. 18.1 Ontology levels

18.1.1 Domain-Level Ontologies

The diversity of phenomena along all these dimensions makes it difficult to find an adequate level of abstraction that fits the whole business domain. In organizational theory a number of metaphors have been suggested to understand and explain organizational behavior at a high level of abstraction. Metaphors establish a link between a source field and a target field and explain phenomena in the target field in terms of the source field. For organizational theory as a target field the following source fields have been proposed: the machine metaphor (Scott, 1997), living systems (biology) (Kendall and Kendall, 1993), open systems (Flood, 2005), the brain metaphor (Gareth, 1997), learning systems (Senge, 1990), social networks (Davern, 1997), complex adaptive systems (Anderson, 1999), autopoietic social systems (Luhmann, 1990) and so on. A metaphor is a vehicle for explaining a target domain in terms of a source domain, e.g. the steering of a ship (source) as a metaphor for the steering of a company (target). Using a metaphor therefore implies a shift of domain. Existing ontologies for the source domain can therefore be transferred to the business domain.

But metaphors also imply some severe restrictions. By viewing organizations as, e.g. living systems we fail to capture those parts of organizational behavior that are not found in biology. Established approaches to a business ontology draw therefore on a number of different related theories to develop a richer picture of the domain. Theoretical contributions can come from communication theories, e.g. Speech Act Theory (Austin, 1975; Searle, 1997, 1999) and Theory of Communicative Action (Habermas, 1984); social theories, e.g. Actor Network

Theory (Law, 1992; Walsham, 1997) or Structuration Theory (Giddens, 1986); economic theories, e.g. Agency Theory (Jensen and Meckling, 1976; Ross, 1973) or Transaction Cost Economics (Klein et al., 1978; Williamson, 1981, 1983, 1998) and others.

Examples of existing approaches to a general ontology of the business domain are: Core Enterprise Ontology (CEO) (Bertolazzi et al., 2001), Edinburgh Enterprise Ontology (EEO) (Uschold, King, Moralee and Zorgios, 1998), Toronto Virtual Enterprise (TOVE) (Fox and Gruninger, 1998), Business Model Ontology (BMO) (Osterwalder, 2004), e^3value Ontology (EVO) (Gordijn, 2004), Socio-Instrumental Pragmatism (SIP) (Goldkuhl, 2002, 2005) and Enterprise Ontology (EO) (Dietz, 2006).

CEO introduces a framework that only specifies the concepts that are common to the whole business domain. They are grouped into four areas: Passive entities (business objects); active entities (actors, agents); transformations (actions, processes); and conditionals (business goals and rules, constraints and states). It is up to the ontology designer to build the actual ontology refining the basic concepts into the specific ones of the respective application domain.

EEO takes a different approach. Instead of just providing a framework they actually specify a supposedly complete repository of detailed enterprise terms. The ontology user therefore only has to choose the ones s/he needs for the particular application. EEO provides both a natural-language and semi-formal definition for all the terms ranging from the foundational concepts of the meta-ontology (e.g. entity, relationship, actors) to the domain concepts that are divided into 4 main areas: activities, organization, strategy and marketing.

TOVE is similar to EEO in that they also try to capture a complete enterprise terminology in a number of ontologies. The base ontology is a general Activity ontology from which more specific ontologies are derived such as Organization Ontology and Resource Ontology. But contrary to EEO the concepts and relations are defined in a completely formal way including theorems and proofs of important properties such as soundness and completeness.

BMO is based on the balanced scorecard and defines four so-called pillars: Product, customer interface, infrastructure management and financial aspects. These pillars are then refined into a set of nine building blocks: Value proposition (concerning the product); target customer, distribution channel and relationship (concerning the customer interface); value configuration, capability and partnership (infrastructure management); and cost structure and revenue model (financial aspects). The important difference to the above mentioned approaches is that BMO takes a wider view of a business including also concepts that lie outside the focus of the actual enterprise but which have considerable impact on it such as the ones related to the customer interface. BMO also introduces the concept of value (of a product or service) whereas other business ontologies are often restricted to the cost perspective.

After having discussed a few business ontologies on a general level we shall describe two examples in greater detail. They are: Socio-Instrumental Pragmatism (Goldkuhl, 2002, 2005) and Enterprise Ontology (Dietz, 2006). They represent two diagonally opposed ends of the spectrum spanned by the dimensions scope and degree of elaboration. This allows the reader to get an impression of the bandwidth of the approaches. The former has a wide scope that covers any kind of social behavior including business action. But it is not yet a full-blown ontology but rather a basic taxonomy complemented by a set of relations between the basic concepts. Enterprise Ontology, on the other hand, is exactly the opposite: It is a highly elaborated and formalized ontology that provides a substantial level of detail regarding concepts and their relations, and also a set of axioms defining the semantics. But its scope is much narrower as it makes a clear commitment to a very specific conceptualization of the business world, thereby excluding other points of view. Enterprise Ontology has been criticized for that (Goldkuhl and Lind, 2004; Verharen, 1997) but due to its rigidity this approach can nevertheless serve as an illustrative example of a business ontology. The chosen example ontologies are described in the sections Socio-Instrumental Pragmatism and Enterprise Ontology, respectively.

18.1.2 Application-Level Ontologies

As such a general ontology of the business domain cannot be used directly in any concrete business application. It is therefore necessary to have at least one more level, the application ontology. Some researchers suggest additional levels, e.g. task ontologies (Guarino, 1998). But instead of introducing a multitude of levels we propose to interpret them as different domain ontologies instead because most of the interesting problems already occur in the case of a second level. So we only abstract from the complexity levels that do not contribute to our discussion. We do not argue that a reduction to three levels is indeed sufficient. According to this definition, a domain ontology can be task-specific, company-specific etc. We illustrate most of the following discussions in the context of information systems where most of the business ontology research is located.

When we take a look at the application-ontology level we discover that the idea of having a separate ontology for every application is fraught with a severe problem as many individuals and organizations make use of several applications within the context of a single task or business process. Let us consider two of the solutions that have been proposed to solve this problem. The first one, a bottom-up approach, aims at integrating the affected application ontologies, each of which could have been developed independently, to derive a higher-level domain ontology for the task or the specific organization. An example of this is given in (Corbett, 2003).

The second solution is top-down. It assumes the existence of a library of ontologies that is used to build an application ontology (e.g. on the task level) by re-using existing domain ontologies (e.g. on the business process level). Systems that support this are called ontology library systems. Examples of such systems are WebOnto (Domingue, 1998), Ontolingua (Farquhar et al., 1997) and SHOE (Heflin and Hendler, 2000).

18.2 Socio-Instrumental Pragmatism

Socio-Instrumental Pragmatism (SIP) is an ontology that combines social, communicative and instrumental aspects of business behavior. The basic assumption behind SIP is that business action is performed by a human being (possibly on behalf of an organization) with the help of some instrument (tool) to create a result for another human being or organization. The social aspect consists in the action being directed from one human being to another. The instrumental aspect aims at the technical (or material) dimension of this action. The instrument can be anything from an axe to a complex IT (Information Technology) system. As social action requires communication between individuals the communicative element is also present.

SIP is based on six ontological concepts (Goldkuhl, 2002): Human being, human inner world, human action, sign, artifact and natural environment. They are defined in the following paragraphs.

Human being is the most fundamental concept because they are the actors in the social world described; they act in the world based on meanings and perceptions that they derive from the world.

The *human inner world* represents the knowledge that a human being has acquired over time about themselves and the external world; this inner world is intended to be seen as part of the human being.

Human action is an important aspect of the human being; it can be overt, which means that the actions are intended to intervene in the external world, thus trying to change something about it. And they can be covert, when they are aimed to change some other human being's inner world; covert actions try to change knowledge that resides in the human inner world.

Signs are the result of communicative actions; for instance, when we write a note saying, "I will be at the store", the writing of the note is by itself a communicative action, but the note created is a sign which will mean something to the person who will read it.

Artifacts are things which are not symbolic and not natural but which are material and artificially created. Examples of artifacts are cars, clothes, a knife, etc. The difference between signs and artifacts is that while signs are intended to mean something to someone (symbolic), artifacts perform material actions. For instance, a human might use a knife (artifact) to cut some carrots, i.e. artifacts are needed to perform material actions.

Natural environment means the objects present in the environment that are not artificially created by humans (e.g. trees).

Figure 18.2 shows the different realms of the world according to the SIP ontology.

18.2.1 Restructuring the Taxonomy

In an attempt to formalize the conceptual system of SIP and the inter-concept relations, we have developed a meta-model in the form of a class diagram of the UML

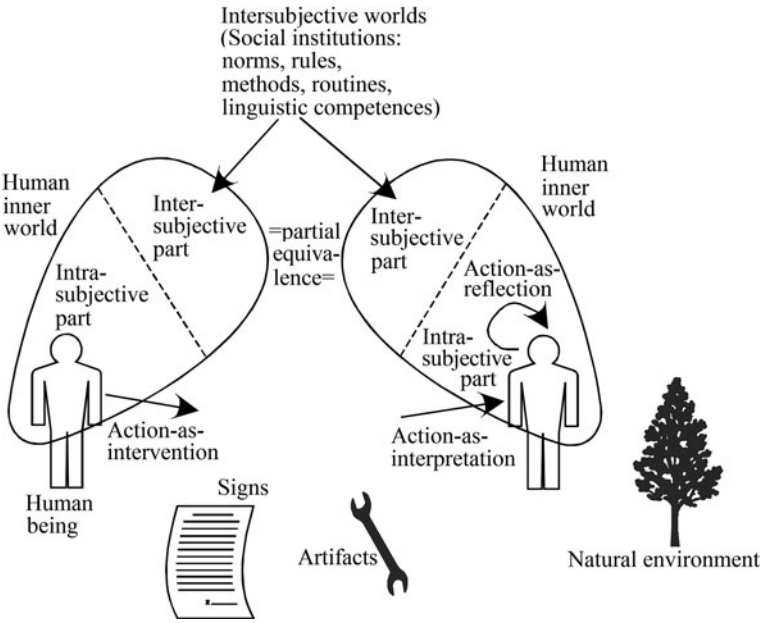

Fig. 18.2 Realms of the world in SIP (Goldkuhl, 2002)

(Unified Modeling Language). A meta-model defines a language for modeling, i.e. it specifies the terms of that language and the way in which they are related and thereby indirectly also defines the semantics to some degree by setting constraints that restrict the use of the language to the admissible models and excluding the ones that are not meaningful. Meta-models can be seen as one way of representing an ontology where the terms of the modeling language are the concepts of the ontology.

A UML Class Diagram is a language that allows for the expression of meta-models (as well as object-level models of a domain). It offers classes, associations and generalization as meta-concepts (among others). Classes are collections of objects that share common attributes and behavior and they can be used to represent concepts of an ontology. Associations are relations between classes which refer to the concept relations in the ontology. Generalization declares one class as the generalization of another thereby introducing a taxonomical relation in terms of the ontology. For further details about UML see (OMG, 2005, 2006).

The meta-modeling process involved a restructuring of the taxonomy suggested by (Goldkuhl, 2002). Human action has been generalized to (social) action and human being to actor. The other concepts become sub-concepts of object. This leaves us with four main concepts: Actors, agents, objects and actions, where agent is a new concept that was introduced to handle artifacts that can perform actions.

18.2.1.1 Actors

Actors are the main entities in our model, and they can perform either as locutor or addressee within the communicative context. When actors perform actions that are directed towards another actor we speak of social actions. They can be performed either in a human-human relation or in a human-artifact-human relation. When performing as locutor the actor is trying to change some aspect of the world by means of his/her actions. For instance, when a person pays the phone bill she is trying to avoid the interruption of her phone service. When performing as addressee the actor receives and interprets an action directed to him and can act himself as a consequence of that action. Taking our example the addressee will be the phone company, which at the moment of receiving the payment will not make any attempt to interrupt the customer's phone service.

Besides locutor and addressee we can distinguish between organizational actors and human actors. The former is an actor that performs as an agent on behalf of the organization; the latter performs an action on behalf of herself.

18.2.1.2 Objects

An object may be physical or conceptual and it may be formed by other objects or related to them, but every object is unique (Embley et al., 1992). Under the object concept we have artificial and natural objects. Artificial are those that are created by human beings, natural objects are those created by nature and found in the environment. Among the artificial objects we have artifacts (material objects) and signs (can be material or immaterial). Artifacts are created to extend actors' capabilities. An artifact is seen as a tool. Signs on the other hand are not tools but messages in a static phase waiting to be interpreted by actors or artifacts. A message can take either a physical form (a written text) or a non-physical form (an utterance) (Goldkuhl, 2002).

We can distinguish between 4 different types of artifacts: static, dynamic, automated and multi-level. Static artifacts are those that cannot perform any operation by themselves, e.g. a stone, a knife or an axe. Dynamic objects are those capable of performing some operations by themselves but they need constant control by a human being to function properly, for example a car or a driller. Automated artifacts are those that can operate entirely by themselves and only need to be started by an actor. Here we can mention a washing machine as an example (Goldkuhl and Ågerfalk, 2005).

Multi-level artifacts are those that have a mix of capabilities and can perform either as static, dynamic or automated artifacts depending on the circumstances. Multi-level artifacts have an important property which is the capability of creating and interpreting signs. They lack consciousness and are ruled by a pre-defined set of instructions that serve as a guide to perform the pre-defined actions they do. IT systems are an example of multi-level artifacts. Signs can be created either by human beings or artifacts, and every sign can be interpreted by human beings only, by artifacts only, or by both (Goldkuhl, 2005). A written note is a sign; an utterance performed by an actor is another example of a sign as well as a ticket printed by a system in an electronic store.

18.2.1.3 Actions

The objective of human actions is to change something in the world. They can be communicative or material. The main difference between these two types of actions lies in the fact that communicative actions are intended to change knowledge. Knowledge is implicitly meaningful to someone; and knowledge handling is an exclusive characteristic of actors within an IS (Information System). On the other hand, material actions are aimed at material conditions and aspects of the world which are meaningful to someone. They are intended to change something physical in the external world. As a human characteristic, knowledge can be learned through actions, either communicative actions (for instance, a conversation) or material actions (e.g. when studying an object). Knowledge is the result of the actor's interpretation of both communicative and material actions, and it can be acquired in a social context (from other actors transferring knowledge e.g. in a classroom) or in a non-social context (a person reading a book on her own) (Goldkuhl, 2001).

We can divide actions into i-actions (intervening actions) and r-actions (receiving actions). I-actions are those intended to make a change in the external world, e.g. the action of opening a window is intended to change a particular aspect of the external world (the window will move from closed to open). R-actions are those executed covertly, for example when two people are going out and person A tells B "It's cold outside" (communicative i-action). Then person B listens and interprets the message (r-action). Following this, person B will take a jacket on the way out (material i-action) (Goldkuhl, 2001). Among i-actions and r-actions we have indefinite and predefined actions.

Indefinite actions are those performed by humans. We call them indefinite since it is not certain how they will be performed by the actor. The same action can vary from actor to actor. When two employees are ordered to clean a shelf, they will both do it but not in the same way, one can do it better or faster than the other one. Indefinite actions can be either r-actions or i-actions. On the other hand we have pre-defined actions which are performed by artifacts. These actions will always be performed in the same way following previously programmed instructions (Goldkuhl, 2005). Pre-defined actions are i-actions, since they are intended to change an aspect of the external world. Among indefinite and pre-defined actions we find both communicative and material actions.

Both material and communicative actions aim at changing an aspect of the world surrounding the actor or artifact, but communicative actions have at least two phases where actor A first performs a communicative action that is directed towards another actor or artifact B. In the second phase B (if A was successful) executes the action that A desired. Although material, the last action can sometimes be performed without an initial communicative action.

Material and communicative actions within organizations form patterns. Although human beings perform them, we can also say that the organization acts. An organizational action has human origins and purposes and is done through humans, by humans or by artifacts that act on behalf of the organization (Goldkuhl et al., 2001). We will consider actions as organizational if they constitute an interaction between two or more actors or agents of the organization within an organizational context. We can say that a worker at a clothes factory using a sewing machine to manufacture clothes is performing an organizational action. He is acting to perform an organizational objective (to produce clothes). But for instance a man on a farm that goes to the forest to chop wood using an axe, although using an artifact to perform the action, is not performing an organizational action since there is no organizational purpose if he merely burns the wood to warm up his house.

When performing actions by means or with the help of IT systems we can distinguish between three different types of actions: interactive, automatic and consequential actions. Interactive actions are supported by and performed through IS and they consist of one or more elementary interactions. Elementary interactions (e-actions) consist of 3 phases: a user action, an IT system action and a user interpretation (Goldkuhl, 2001). Let us take the example of an online bank transfer done by the user online. The user will initially introduce his username and password to access the bank system (phase 1), after this the IT system will check in the database if the information is correct and if it is it will grant access to the user and display a welcome screen (phase 2). The welcome screen is interpreted, and the user now knows that he can start his transaction. This is the end of the elementary interaction. Later on the user inputs the data to make the bank transfer, such as account number, amount to be transferred, etc. (phase 1 of a second e-interaction), and so on.

Automatic actions are performed by IT systems that produce messages for the actors or other systems. They are done entirely without human intervention. Let us take the banking system again: After logging on, a message pops up telling the customer that the due date for the credit card payment is very close. The system will execute this operation by itself and present it to the user.

Consequential actions are those performed as a consequence of a message. Taking the bank example again, when the customer sees that his payment is due he might proceed to execute the payment, or he might decide not to do it and wait for the final day.

18.2.1.4 Agents

Agents are a special type of object. They are created by actors, and perform actions to help them complete their tasks. They can be seen as servants of actors, but they have a level of communicative capabilities that allow them to act as communicative mediators, and they are also capable of creating signs for the actors or other agents to interpret. The difference between agents and actors lies in the fact that actors can perform social actions while agents cannot (Rose and Jones, 2004).

IT systems can perform as agents. They can be seen as static artifacts, automated artifacts or dynamic artifacts (Goldkuhl and Ågerfalk, 2005). In all three cases the common denominator is communication. Communication is seen as a kind of action that IT systems can perform and by doing so they become communication mediators. IT systems as well as actors have the capability to create signs and to process them. Actors can also interpret them (Goldkuhl, 2001). The relation between the signs and their interpreters/processors is called pragmatics. Within IS pragmatics, actions are divided into those that occur within the sign transfer and consequential actions that are performed in response to the transferred sign (Goldkuhl and Ågerfalk, 2000).

18.2.2 The Resulting Meta-model

As a result of the contemplations in the previous sections we have developed a meta-model (see Fig. 18.3) that covers the most important aspects of Socio-Instrumental Pragmatism as outlined in Section 18.2. Technically the meta-model takes the form of a UML class diagram with generalization/specialization and association.

18.3 Enterprise Ontology

Enterprise Ontology has its roots in the ontological frameworks of Bunge and Wittgenstein (Bunge, 1977; Wittgenstein, 2001). According to it a world can be in any of a number of states. The ability to move from one state to another is called transition. An occurrence of a transition is an event. An event is caused by an act. Worlds consist of two kinds of objects, stata and facta. A statum is a thing that exists in all states of the world independent of any act. A factum is the result of an act that brings it about. Stata can be declared (i.e. original) or derived (from other stata). Existence laws determine the possible states of the world by specifying which combinations of stata are allowed in a state (the state space).

A factum is associated with an event (i.e. the occurrence of a transition). Ocurrence laws determine the possible orders of transitions by specifying which sequences of transitions are allowed (the transition space). An ontology is defined by specifying the state and transition spaces. It is done with the help of a language, the World Ontology Specification Language. It uses a graphical notation borrowed from conceptual modeling, i.e. ORM (Halpin, 2001).

18.3.1 World Ontology Specification Language

A statum can be declared intensionally via its properties, or extensionally by enumerating the instances that belong to the statum type. The intension is an *n*-ary predicate that is denoted as a rectangle with *n* horizontal fields with an inscribed placeholder each. Underneath we write a predicative sentence detailing the role of each placeholder. On top we write the name of the statum type in bold lower-case letters. The extension is a set denoted by a rounded rectangle and inscribed with the name of the statum type in capital letters. Intensionally defined statum types can be entified (extensionalized) by surrounding them with a rounded rectangle.

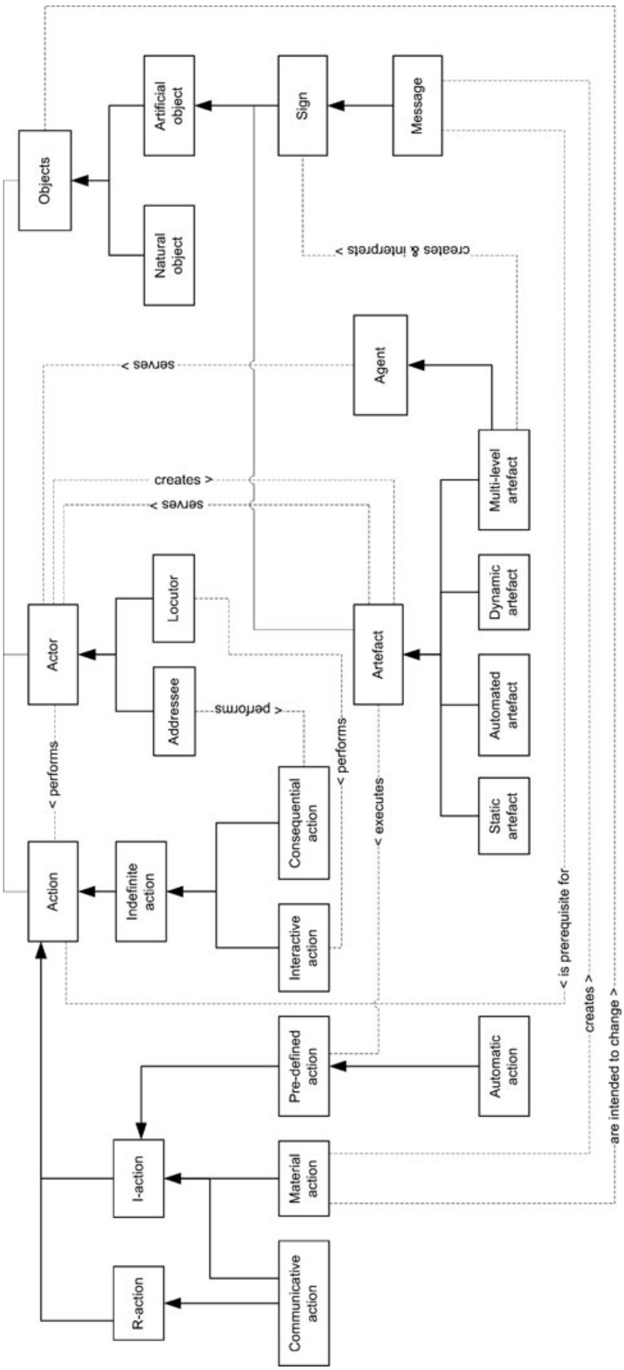

Fig. 18.3 Meta-model of SIP

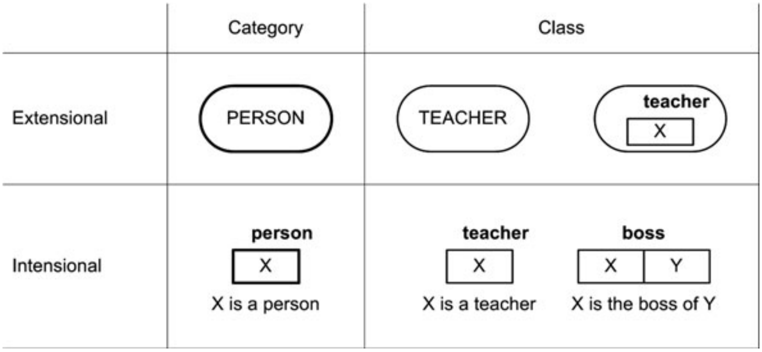

Fig. 18.4 Declaration of statum types

Categories are primal statum types in the sense of ultimate sortals according to (Guizzardi et al., 2004), i.e. they do not rely on other stata for their definition. They are denoted by a thick borderline around the respective shape. Classes are defined by establishing a reference to another class or a category. They are surrounded by a normal line. Figure 18.4 shows the different ways of declaring statum types.

We have 4 different types of existence laws that can be applied to restrict the space of allowed states: Reference laws, dependency laws, unicity laws and exclusion laws (see Fig. 18.5). With the help of a reference law we can define the domain of a placeholder in a statum type. In the example a lecturer must always be a teacher (but not every teacher is a lecturer). Dependency laws imply a mutual reference, e.g. every employee has a boss and everyone with a boss is an employee. A unicity law excludes that a statum can participate in multiple instances of the same predicate, for example, an employee cannot have two bosses. An exclusion law prevents a statum that is engaged in one relation to participate in another, e.g. a CEO does not have

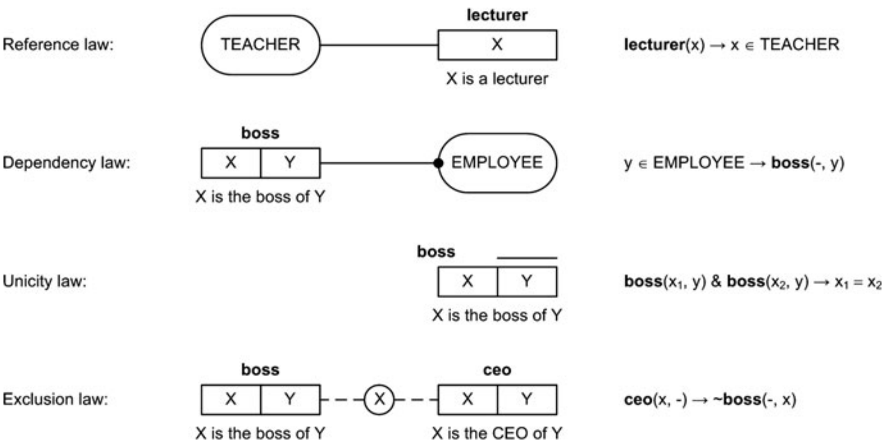

Fig. 18.5 Existence laws for stata

a boss as denoted by the exclusion relation between the boss and CEO objects in Fig. 18.5 (an encircled symbol *X* on a dashed line).

Statum types can also be derived from others. The shape of the derived type is in that case hatched (i.e. the shape's background area is filled with diagonal lines). Derivation rules include aggregation, generalization and specialization. Aggregation is the same as entification. Specialization and generalization are not complementary. For instance, *student* can be seen as a specialization of the category PERSON, but PERSON is not a generalization of *student*, *teacher* and other classes as PERSON is a primal type. The class VEHICLE, on the other hand, is a generalization of HORSE CART, BICYCLE, CAR, and so on. The latter are not specializations of the former as the first mode of transport that was developed was a horse cart and not a vehicle. Only later, when additional transport modes had been devised, did it make sense to generalize on their common ability to transport human beings and to address them collectively as vehicles.

Factum types are denoted by a diamond inscribed with a placeholder. Above it a predicative sentence specifies the transition that marks the occurrence of the factum (see Fig. 18.6, above the line). Occurrence laws indicate the restrictions on the transition space. They can prescribe a certain sequence, e.g. an order must have been placed before it can be shipped, or forbid it, e.g. an order that has already been shipped cannot be cancelled anymore (see Fig. 18.6, below the line).

18.3.2 The Axioms of Enterprise Ontology

The World Ontology Specification Language allows for the specification of almost any world (or domain). In order for the resulting ontology to be a business ontology a number of axioms must be satisfied. They are the operation axiom, transaction axiom, composition axiom and distinction axiom.

18.3.2.1 The Operation Axiom

The operation axiom claims that business action takes place in two worlds: the coordination world (or C-world) and the production world (or P-world). P-acts can be material (transport, manufacturing, storage) or immaterial (e.g. decisions). C-acts

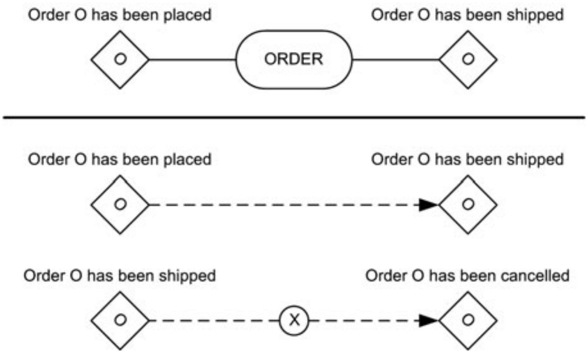

Fig. 18.6 Declaration of facta and occurrence laws

Fig. 18.7 Worlds of enterprise ontology

are used to coordinate, i.e. manage, the productive behavior. C-acts are communicative acts that allow us to establish and follow up commitments. Both types of acts lead to respective facts that in turn are evaluated by the actors to determine further acts. Actors in roles are minimal units of authority and responsibility. Figure 18.7 depicts the implications of the operation axiom.

18.3.2.2 The Transaction Axiom

The transaction axiom states that coordination acts are performed according to patterns. The standard pattern assumes that both parties go on confirming each other's acts. It is shown in Fig. 18.8. If the parties disagree they move to a different layer, the discussion layer to resolve the conflict. If this fails they can move to the discourse layer where general norms and values can be addressed. There are also additional patterns that deal with cancellation (Dietz, 2006).

Here we will only look at the standard pattern. In it two parties, called initiator and executor, try to establish and follow up a commitment that concerns the execution of a P-act by the executor. The transaction is divided into three phases, order, execution and result. The order phase is an actagenic (i.e. action generating)

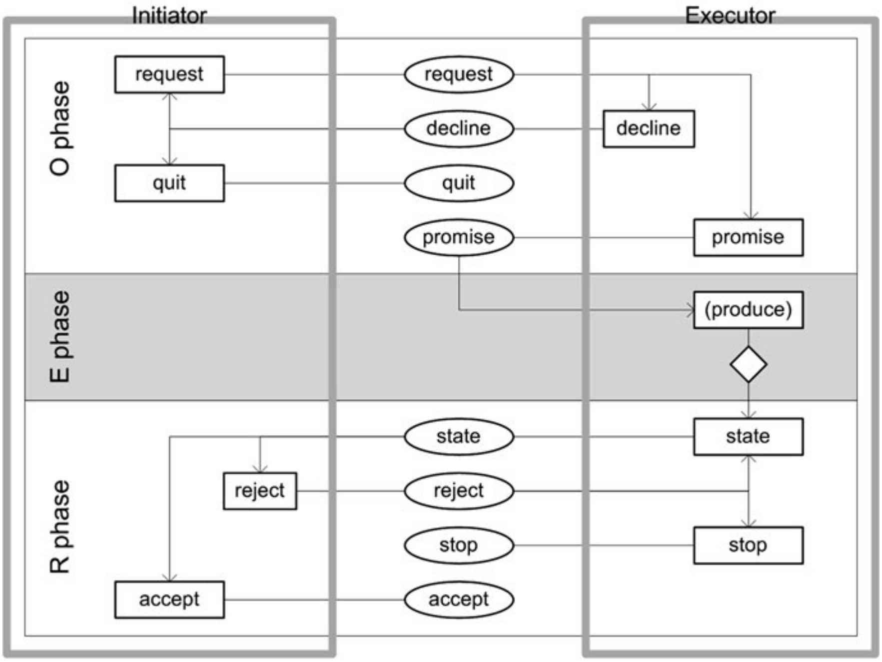

Fig. 18.8 Interaction pattern of a transaction according to (Dietz and Habing, 2004)

conversation that results in a commitment of the executor to perform the P-act (produce). In the second phase the P-act is actually done and a corresponding factum is established ("P-act has been executed"). The result phase is a factagenic conversation that aims at reaching an agreement on this factum to assess whether the commitment has been met.

The order phase starts with a request from the initiator to perform a certain P-act. Should the executor decline it, the initiator can decide to quit or to make another request, possibly after having convinced the executor on the discussion layer to consider his original request again, or after having realized the inappropriateness of his request and formulated a new one. If the executor accepts the request it becomes a promise, i.e. a binding commitment. The executor will then perform the P-act (execution phase) and enter the result phase by stating that he has done the P-act as agreed. The initiator can now reject this statement upon which the executor can stop (unsuccessful termination) or make another statement. If the initiator accepts the statement the whole transaction is terminated successfully.

18.3.2.3 The Composition Axiom

The composition axiom claims that each transaction is either activated externally (e.g. by a customer), or it is embedded in another transaction, or it is self-activated. In the first case the transaction is started by some person or organization in the environment, i.e. that lies outside the borders of the organization in question. Embedded transactions take into account that products and services typically have a hierarchical structure which must be mirrored by the transactions that produce them. Finally, a transaction can activate itself. This takes care of repeated activities, e.g. periodical transactions.

18.3.2.4 The Distinction Axiom

The distinction axiom states that all human activities, and thereby also business activities, can be divided into three categories: forma, informa and performa. Forma concerns the "outer appearance". In the C-world this corresponds to passing on or receiving information. Informa is concerned with the content that is "in the form". In the C-world it comprises the expression or interpretation of thoughts. Performa is related to the creation of new things with the help of the form, i.e. "through the form". Regarding the C-world this could involve establishing and following up commitments (creation of commitments and facts). These levels apply also to the P-world.

18.4 Conclusion

We have introduced two different approaches to a domain ontology for business that mark the opposite ends of a plane spanned by scope and elaboration. Socio-Instrumental Pragmatism makes a weak ontological commitment that can be applied to any business context and even beyond that. But in order to do this the concept system, the taxonomy and the relations have to be refined, and axioms or rules need to be introduced. Enterprise Ontology, on the other hand, makes a strong ontological commitment which might be in conflict with a particular view on a business. Its scope is therefore limited but in an appropriate context it can provide a strong support.

But in order for business ontologies to be really applicable they must support activities in a multitude of interrelated domains. No single ontology can achieve that. We therefore need mechanisms to connect related ontologies in an effective way. This implies that ontology X may import or derive concepts and relations from another ontology Y. But it also means that it must be possible to integrate ontologies from different but related domains to a new one that covers all these domains and resolves the conflicts that exist between them (see Section 18.1.2). It is the combination of the bottom-up and top-down approach that can bridge the gap between a general ontology for the business domain and the application-specific ontology.

References

Anderson, P. 1999. Complexity theory and organization science. *Organization Science* 10(3): 216–232.

Austin, J.L. 1975. *How to do things with words*. Cambridge, MA: Harvard University Press.

Bertolazzi, P., C. Krusich, M. Missikoff. 2001. An approach to the definition of a core enterprise ontology: CEO. Paper presented at the International Workshop on Open Enterprise Solutions: Systems, Experiences, and Organizations (OES-SEO 2001), Rome, 14–15 Sep 2001.

Bugaite, D., and O. Vasilecas. 2005. Framework on application domain ontology transformation into set of business rules. Paper presented at the International Conference on Computer Systems and Technologies – CompSysTech 2005.

Bunge, M. 1977. *Ontology I: The Furniture of the World*. Dordrecht: Reidel.

Carmichael, D.J., J. Kay, and B. Kummerfeld. 2004. Personal ontologies for feature selection in intelligent environment visualisations. In *Artificial intelligence in mobile systems 2004*, eds. J. Baus, C. Kray, and R. Porzel, 44–51. Saarbrücken, Germany: Universität des Saarlandes.

Corbett, D. 2003. Comparing and merging ontologies: A concept type hierarchy approach. In *Foundations of intelligent systems*, eds. N. Zhong, Z.W. Ras, S. Tsumoto, and E. Suzuki. Proceedings of the 14th International Symposium, ISMIS 2003. Maebashi City, Japan, 28–31 Oct 2003, 75–82. Berlin: Springer.

Davern, M. 1997. Social networks and economic sociology. A proposed research agenda for a more complete social science. *American Journal of Economics and Sociology* 56(3):287–301.

Dieng, R., and S. Hug. 1998. Comparison of <<personal ontologies>> represented through conceptual graphs. In ECAI 98. 13th European Conference on Artificial Intelligence, ed. H. Prade, 341–345. New York, NY: Wiley.

Dietz, J.L.G. 2006. *Enterprise ontology: Theory and methodology*. Heidelberg: Springer.

Dietz, J.L.G., and N. Habing. 2004. A meta ontology for organizations. In *On the move to meaningful internet systems 2004: OTM 2004 workshops*, eds. R. Meersman, Z. Tari, A. Corsaro, P. Herrero, M. S. Pérez, M. Radenkovic, V. Robles, C. Santoro, A. Albani, K. Turowski, M. Jarrar, A. Gangemi, E. Duval, P. Spyns, and A. Palinginis, Proceedings of the OTM Confederated International Workshops and Posters, GADA, JTRES, MIOS, WORM, WOSE, PhDS, and INTEROP 2004, Agia Napa, Cyprus, 25–29 Oct 2004, vol. 3292, 533–543. Berlin: Springer.

Domingue, J. 1998. Tadzebao and WebOnto: Discussing, browsing, and editing on the web. In eds. B. Gaines, and M. Musen. Proceedings of the 11th Knowledge Acquisition for Knowledge-Based Systems Workshop, 18–23 Apr, Canada: Banff.

Embley, D., B. Kurtz, and S. Woodfield. 1992. *Object oriented system analysis: A model-driven approach*. Upper Saddle River, NJ: Yourdon Press.

Farquhar, A., R. Fikes, and J. Rice. 1997. The Ontolingua server: Tools for collaborative ontology construction. *International Journal of Human Computer Studies* 46:707–728.

Flood, R.L. 2005. Unleashing the "open system" metaphor. *Systemic Practice and Action Research* 1(3):313–318.

Fox, M.S., and M. Gruninger. 1998. Enterprise modeling. *AI Magazine* 19(3):109–121.

Gareth, M. 1997. *Images of organization*. London: Sage.

Giddens, A. 1986. *The constitution of society. Outline of the theory of structuration*. Berkeley, CA: University of California Press.

Goldkuhl, G. 2001. Communicative vs material actions: Instrumentality, sociality and comprehensibility. Paper presented at the 6th International Workshop on the Language Action Perspective (LAP2001), Montreal.

Goldkuhl, G. 2002. Anchoring scientific abstractions – Ontological and linguistic determination following socio-instrumental pragmatism. Paper presented at the European Conference on Research Methods in Business and Management (ECRM 2002), Reading, 29–30 Apr 2002.

Goldkuhl, G. 2005. Socio-instrumental pragmatism: A theoretical synthesis for pragmatic conceptualisation in information systems. Paper presented at the 3rd International Conference on Action in Language, Organisations and Information Systems (ALOIS), University of Limerick.

Goldkuhl, G., and P. Ågerfalk. 2000. Actability: A way to understand information systems pragmatics. Paper presented at the 3rd International Workshop on Organisational Semiotics, Stafford, 4 July 2000.

Goldkuhl, G., and P. Ågerfalk. 2005. IT artefacts as socio-pragmatic instruments: Reconciling the pragmatic, semiotic, and technical. *International Journal of Technology and Human Interaction* 1(3):29–43.

Goldkuhl, G., and M. Lind. 2004. Developing e-interactions – A framework for business capabilities and exchanges. Paper presented at the 12th European Conference on Information Systems, Turku, Finland, 14–16 June 2004.

Goldkuhl, G., A. Röstlinger, and E. Braf. 2001. Organisations as practice systems – integrating knowledge, signs, artefacts and action. Paper presented at the Organisational Semiotics, IFIP 8.1 Conference, Montreal.

Gordijn, J. 2004. E-business value modelling using the e3-value ontology. In *Value creation form e-business models*, ed. W.L. Curry, 98–127, Chapter 5. Oxford: Elsevier Butterworth-Heinemann.

Guarino, N. 1998. Formal ontology and information systems. In Proceedings of the 1st International Conference on Formal Ontology in Information Systems (FOIS'98), 3–15. Amsterdam: IOS Press.

Guizzardi, G., G. Wagner, N. Guarino, and M.V. Sinderen. 2004. An ontologically well-founded profile for UML conceptual models. In Proceedings of the 16th International Conference on Advanced Information Systems Engineering, CAiSE 2004, vol. 3084, 112–126, Berlin: Springer.

Haase, P., A. Hotho, L. Schmidt-Thieme, and Y. Sure. 2005. Collaborative and usage-driven evolution of personal ontologies. In eds. A. Gómez-Pérez, and J. Euzenat. Proceedings of the 2nd European Semantic Web Conference, Heraklion, Greece, 2005 Vol. 3532, 486–499, Heidelberg: Springer.

Habermas, J. 1984. *The theory of communicative action 1 – reason and the rationalization of society*. Boston, MA: Beacon Press.

Halpin, T.A. 2001. *Information modeling and relational databases*. San Francisco, CA: Morgan Kaufmann.

Heflin, J., and J. Hendler. 2000. Dynamic ontologies on the web. In Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-2000), 443–449, Menlo Park, CA: AAAI/MIT Press.

Huhns, M.N., and L.M. Stephens. 1999. Personal ontologies. *IEEE Internet Computing* 3(5): 85–87.

Jensen, M.C., and W.H. Meckling. 1976. Theory of the firm: managerial behavior, agency costs and ownership structure. *Journal of Financial Economics* 3:305–360.

Kendall, J.E., and K.E. Kendall. 1993. Metaphors and methodologies: Living beyond the systems machine. *MIS Quarterly* 17(2):149–171.

Klein, B., R. Crawford, and A. Alchian. 1978. Vertical integration, appropriable rents, and the competitive contracting process. *Journal of Law and Economics* 21:297–326.

Law, J. 1992. Notes on the theory of the actor-network: Ordering, strategy and heterogeneity. *Systemic Practice and Action Research* 5(4):379–393.

Luhmann, N. 1990. The autopoiesis of social systems. In *Essays on self-reference*, ed. N. Luhmann, 1–21. New York, NY: Columbia University Press.

OMG. 2005. Unified modeling language: Superstructure. Version 2.0, August 2005, Needham, MA: OMG. Retrieved, June 20 2006, from http://www.uml.org/.

OMG. 2006. Unified modeling language: Infrastructure. Version 2.0, March 2006, Needham, MA: OMG. Retrieved, June 20 2006, from http://www.uml.org/.

Osterwalder, A. 2004. The business model ontology – a proposition in a design science approach. Ph.D. thesis, University of Lausanne, Ecole des Hautes Etudes Commerciales HEC: 173.

Rose, J., and M. Jones. 2004. The double dance of agency: A socio-theoretic account of how machines and humans interact. Paper presented at the ALOIS Workshop: Action in Language, Organisations and Information Systems, Linköping.

Ross, S. 1973. The economic theory of agency: The principal's problem. *American Economic Review* 63(2):134–139.

Scott, A. 1997. Modernity's machine metaphor. *The British Journal Of Sociology* 48(4):561–575.

Searle, J.R. 1997. *Speech acts – An essay in the philosophy of language*. Cambridge, MA: Cambridge University Press.

Searle, J.R. 1999. *Expression and meaning. Studies in the theory of speech acts*. Cambridge, MA: Cambridge University Press.

Senge, P.M. 1990. *The fifth discipline. The art and practice of the learning organization*. New York, NY: Doubleday.

Uschold, M., M. King, S. Moralee, and Y. Zorgios. 1998. The enterprise ontology. *Knowledge Engineering Review* 13(1):31–89.

Verharen, E. 1997. *A language-action perspective on the design of cooperative information agents*. Tilburg: Katholieke Universiteit Brabant.

Walsham, G. 1997. Actor-network theory: Current status and future prospects. In *Information systems and qualitative research*, eds. A.S. Lee, J.Liebenau, and J.I. Degross. London: Chapman & Hall.

Williamson, O.E. 1981. The modern corporation: Origins, evolution, attributes. *Journal of Economic Literature* 19:1537–1568.

Williamson, O.E. 1983. *Markets and hierarchies*. New York, NY: Free Press.

Williamson, O.E. 1998. *The economic institutions of capitalism*. New York, NY: Free Press.

Wittgenstein, L. 2001. *Tractatus logico-philosophicus*. London: Routledge.