

# Ergativity and the Object-oriented Representation of Verb Meaning

Anton Benz  
Centre for General Linguistics  
Berlin/Germany  
benz@zas.gwz-berlin.dk

**Abstract:** There is an interesting parallelism between the representation of one-place and two-place operations in object-oriented programming and case marking in ergative languages. The object-oriented approach has proven to be highly successful in computational system design and analysis. One of its peculiarities is to define operations within the class of the objects which are their main arguments. More specifically, operations that correspond to intransitive verbs are encoded in the class which corresponds to the subject NP, and operations that correspond to transitive verbs are encoded in the class that corresponds to the NP of the direct object. In ergative languages these NPs are marked by absolutive case. We discuss a number of semantic phenomena related to ergativity that show that conceptual representation of word meaning follows the object-oriented paradigm.

## 1 Introduction

At the centre of our paper lies Keenan's (1984) observation that the meaning of intransitive verbs can be different for different subject arguments, and the meaning of transitive verbs can be different for different direct objects. Keenan supports his claim by the following examples (1984: p. 201):

- (1) a. *John / the horse is still running*  
b. *The car / the motor / my watch is still running*  
c. *The tap / my nose is still running*  
d. *My Fair Lady / the Braque exhibition is still running*
- (2) a. *John cut his arm / his foot*

- b. *John cut his nails / his hair / the lawn*
- c. *John cut his cake / the roast*
- d. *John cut a path (through the field) / a tunnel (through the mountain)*
- e. *John cuts his whiskey with water / his marijuana with tea*
- f. *The company cut production quota / prices*

The interpretation of “*is still running*” differs significantly from (1a) to (1d) but is constant for the same type of subject. In (2a) to (2f), we find the same variation with respect to the direct objects of “*cut*.” In (2a), the body part is lacerated but not cut through. In (2b), pieces of the cut entity are severed from the rest of it. In (2c), the cut entity is divided into several pieces. In (2d), the cutting clears a path through some material which would otherwise not be penetrable or only with considerable effort. In (2e), the whiskey and marijuana are mixed with some other substance. Finally, in (2f), the cutting means a reduction of production quota and prices.

Keenan (1984) discusses these examples in connection with the phenomenon of *ergativity*. He provides a list of properties that are shared by subjects of intransitive verbs and direct objects of transitive verbs. He maintains that the type of polysemy observed in (1) and (2) is bound to the subject of intransitive verbs, and the direct object of transitive verbs.

There is no obvious logical reason why this variation of verb meaning should depend universally on one argument role only, and why this role should be different for intransitive and transitive verbs. Although there are exceptions to this rule, we assume that it is basically correct. In this paper, we speculate that the explanation can be found in the principles of semantic representation which are part of the object-oriented paradigm in computer science.

A recurrent problem for which the object oriented approach proved to be very successful is the problem of *polymorphisms*, i.e., the phenomenon that operations show different behavior with different arguments. For example, the execution of the `print()` operation<sup>1</sup> can be quite different mathematically for different arguments. Printing a line involves different data and calculations than printing a square or a circle. Polymorphism is an obvious analogue to polysemy in verb semantics.

Object orientation was partly a reaction to the problems that the ever growing complexity of software imposed on the maintainability and re-usability of code, see, e.g., (Booch et al. 2007). These problems can be much reduced by a modular software design. This modularity is achieved by packing together data and the methods operating on them in one unit. These units are called *objects*. Each object combines attributes, i.e., data, and the methods which use these data.

In domain modeling, objects represent concrete entities. Methods represent the actions which can be performed by these entities. From another point of view, objects are instantiations of *classes* which define their general properties. In the following, we assume that classes correspond to the lexicon entries of common nouns and methods to verb meanings. This approach to lexical semantics has the interesting consequence that the meaning of verbs is defined within

---

<sup>1</sup>Sample expressions belonging to a programming language which does not need to be exactly specified in the present context are presented in typewriter font. In the example above the empty parentheses indicate that the `print`-method does not require any argument. In the object-oriented language Java the `draw` method used to render figures actually requires an object from the class `Graphics` as an argument. That object may be conceived as a pencil used to draw the figure at issue.

the lexical entries of their arguments. Furthermore, it provides a cognitive basis for the morphological case marking in ergative languages: the absolutive shows the addressee in which *class* to look up the meaning of the verb. If it is a transitive verb, the addressee has to look it up in the class representing the meaning of the direct object; if the verb is intransitive, the addressee has to look it up in the class representing the meaning of the subject. In this paper, we want to show the potential of the object-oriented approach by exploring semantic phenomena related to the ergative/absolutive distinction.

The paper is structured as follows: we first introduce the elements of object-oriented programming which play a role in our model of lexical semantics. We then discuss the phenomenon of polysemy in the context of the object-oriented lexicon. Finally, we show how this paradigm might provide a cognitive basis for the ergative/absolutive distinction.

## 2 The Object-Oriented Paradigm

What exactly the object-oriented paradigm makes up is a question which has no generally agreed upon answers. List of characteristics of the paradigm can vary considerably. The object-oriented paradigm resembles a bundle of related concepts and methodologies of which some core concepts can be identified (Armstrong 2006). We concentrate here on aspects which we think are especially interesting to lexical semantics. These are class and object, inheritance and polymorphism, and encapsulation and interfaces. For our paper, we can concentrate on the handling of polymorphisms (an operation may show different behavior depending on the type of its arguments) through subtyping (inheritance of properties).

A class is often defined as a set of objects that have a common structure and behavior (Booch et al. 2007). For example, we may have a class of triangles which are defined by three distinct points in the Euclidean plane, and which may be rotated or printed. In object-oriented languages, class definitions always divide into three parts: the class name, the class attributes, and the class operations. We can identify a class with an abstract object which contains specifications of name, attributes, and operations. In case of the `Triangle` class, let us assume for simplicity that its elements share one attribute, `position`, which is an array of three coordinates, which are again pairs of real numbers, and two operations. Graphically, we can represent the triangle class as in Figure 1. The top cell of the diagram contains the *name* of the class, the centre cell the list of *attributes*, and the bottom cell the list of *methods* or *operations*.

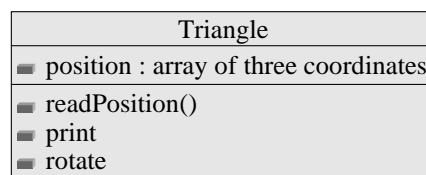


Figure 1: A Triangle class

This named, abstract collection of attributes and operations defines a specific class from which instantiations of concrete triangles can be derived. These concrete instantiations are called *objects*. An object is an entity which has a defined state, behavior and an identity (Booch et al.

2007). This means that an object `t` of type `Triangle` has to have a position, and the operations `readPosition()`, `print()`, and `rotate()` must be defined for it. Using a common notation, we can write this as follows: If `t` is of *type* `Triangle` (`t : Triangle`), then `t.position()` has a value, and `t.readPosition()`, `t.print()`, and `t.rotate()` are defined.<sup>2</sup> The obvious linguistic semantic analogues to *class* and *object* are *predicate* and domain *entity*.

The best known feature of object-orientation is probably inheritance. We are interested in this feature in connection with polymorphism. To start with, let us consider how a procedural program would handle the task of printing different geometrical objects, say triangles, circles, and squares. The data base entries for each of these figures may be very different. A triangle is defined by three points, but the circle may be defined by its centre and its radius, and the square by the coordinates of its lower left corner and the length of its sides. Hence, the actual implementation of the print procedure is different for each object, and it may be necessary to define a separate `drawFigure()` procedure for each geometrical figure. Hence, the print procedure may look as follows:

```
printImage (object) begin
  if object isa circle do drawCircle(object),
    else if object isa square do drawSquare(object),
      else if object isa triangle do drawTriangle(object),
        ...
      end;
end;
```

This phenomenon that one operation shows a different behavior depending on its arguments is called *polymorphism*.

What if it later turns out that other geometrical figures, say arbitrary quadrilaterals, have to be printed? In this case, first, new data types for quadrilaterals like trapezoids and parallelograms have to be defined. Second, the print procedure has to be extended by new if-else clauses for the `drawTrapezoid()`, `drawParallelogram()` etc. procedures. This means that the old code has to be changed at two different places, new data types have to be added and old procedures have to be rewritten. In an object-oriented approach, the print command is part of the operations included in the class definition of the geometrical figures. For example, the `printTriangle()` from our procedure would be the implementation of the print procedure in the `Triangle` class which we defined before. Hence, if a programmer has to extend his system by trapezoids and parallelograms, then he can just add new class definitions as in Figure 2.

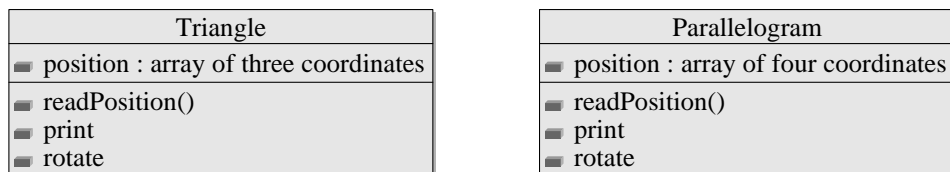


Figure 2: Two classes of the same type

<sup>2</sup>We assume here that `rotate()` rotates a figure by a certain fixed angle. Alternatively, we could assume a number argument degree of the number type `double` (double precision number): `rotate(float degree)`.

This means that the code for the old print commands remains untouched when adding new data types to the system. The advantage of the object-oriented approach lies in its easy extensibility. Seen from the outside, both programs, the procedural and the object-oriented, do the same things, they print geometrical figures, but the object-oriented approach leads to a much greater modularity of representation. This becomes important when we apply the object-oriented approach to the lexicon and natural language semantics. Similarly as the choice between an object-oriented and a procedural representation makes no difference for the actual behavior of operations, the choice of an object-oriented representation of the lexicon has no consequences in terms of truth conditions. One representation is translatable into the other. But it has consequences for language acquisition and the handling of polysemy.

So far, we have seen that the object-oriented approach reduces changes to the old code when extending it to a minimum. Furthermore, all changes are located in one single object, the definition of the class which represents the new data type. But still, the programmer has to remember which operations and attributes have to be defined for the new classes. Probably, he not only wants to print and rotate geometrical figures but also to shift them, stretch them, or fill them with color. In a procedural program, the modification has to be repeated for every procedure that may take an object of the new data type as argument. Hence, the programmer has to remember which procedures may operate on them. There is no code-internal means for memorizing this. In an object-oriented approach, this memorizing is done by the inheritance relation. If operations are applicable to all geometrical figures, then they are implemented by all objects representing them. Hence, they belong to the class which defines *geometrical figures*. The classes of triangles, squares, and circles are only subclasses of this larger class. Graphically, this is notated as in Figure 3. There, the name of the classes of geometrical figures and quadrilaterals are set in italics, which says that these classes are *abstract* classes. This means that they cannot immediately be instantiated by objects but only their subclasses.

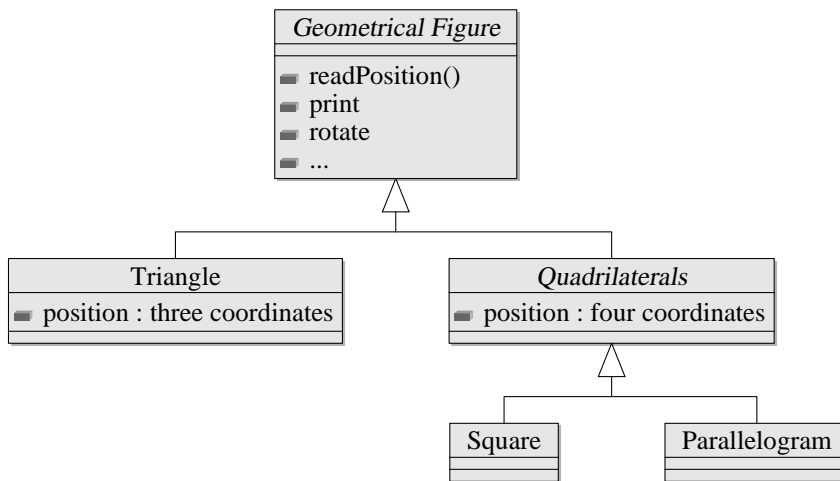


Figure 3: Sub-classes and Super-classes in Inheritance Relation

Inheritance means that all subclasses share the attributes and operations of their super-classes. In our example, instances of triangles, squares, parallelograms all have to define the *print*,

rotate and other operations, and all squares and parallelograms have to provide four coordinates for their position. If the programmer now introduces a new class of geometrical figures, e.g., the class of trapezoids, he can introduce them as a new subclass to the super-class of quadrilaterals. This will automatically tell him which attributes and operations he has to implement for trapezoids. Hence, from the perspective of maintainability and extendibility of programs, the object-oriented approach helps to minimize changes, to bundle them together in one place, and to memorize which attributes and operations have to be defined.

The obvious analogues to the inheritance relation between super-class and sub-class are the semantic relations of *hypernymy* and *hyponymy*. The class hierarchy defined by inheritance is closely related to the notion of an ontology which is defined by an *isa* relation. In an *ontology*, the typical definition of a class contains attributes and constraints which must be satisfied by each instantiation of the class. For example, triangles can be defined as triples of points in the Euclidean plane such that no straight line which connects two of the points also contains the third point. An equilateral triangle is a triangle for which the distance between all points is equal. Hence, if we predicate of an entity  $\mathbf{t}$  that it is of type equilateral triangle, then it must satisfy the condition for triangle and the condition for equilaterality.

The inheritance relation in the object oriented class hierarchy is also an *isa* relation but with an important difference with respect to operations. If we say that an object  $\mathbf{t}$  is an instantiation of the triangle class as defined before, then it follows that there must be an array  $\vec{a}$  of Euclidean points such that  $\vec{a}$  contains the coordinates of  $\mathbf{t}$ . Logically, this means that:  $\exists \mathbf{t} \text{ triangle}(\mathbf{t})$  implies  $\exists \vec{a} [\text{array-of-EucP}(\vec{a}) \wedge \text{coordinates-of}(\mathbf{t}, \vec{a})]$ . The same does not hold for operations. If  $\text{op}$  is an operation defined in the class or super-class of triangle, e.g., the print operation, then the existence of an instantiation does not entail the existence of an execution of  $\text{op}$ . This is important as operations are the natural analogues to verb meanings. This also marks an important difference to previous applications of inheritance to lexical semantics, as we will see soon.

What we have considered so far is *single* inheritance only. In addition to single inheritance, object-oriented programming also knows *multiple* inheritance. This is typically used for modeling the behavior of a class in different contexts. Multiple inheritance found significant and wide spread applications in lexical semantics.<sup>3</sup> Most important in the context of systematic polysemy is the application of *multiple* inheritance for representing context dependent meaning differences in the nominal domain. An example is the difference between the meanings of *newspaper* in ‘*John got angry at the newspaper and spilled coffee over it.*’ The first occurrence refers to the content of the newspaper, the second to the physical object.

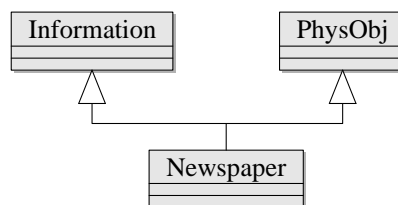


Figure 4: The Newspaper class and multiple inheritance

<sup>3</sup>See the collection of (Briscoe et al. 1993), and (Pustejovsky 1995)

Pustejovsky (1995) introduced *dotted types* in order to handle such example. If the dotted type **Info-PhysObj** is assigned to *newspaper*, then this means that in some context *newspaper* can fill an argument position which requires an argument of type **Info**, in another context it can fill an argument position which requires an argument of type **PhysObj**, and in yet another context it can fill a position in which both aspects are interpreted. An example for the latter case is the sentence “*John wrote the letter*” where *letter* is also of the type **Info-PhysObj**, and where John’s writing produced both the letter as a physical object and its content. Dotted types can be interpreted as a type theoretic equivalent to multiple-inheritance.

Our explanation of the phenomenon observed in the introductory examples (1) and (2) will only make use of single inheritance. In Pustejovsky’s generative lexicon, they are cases for *co-composition*, which is a generative mechanism which is in principle independent of dotted types. The meaning differentiations in (1) and (2) are not an effect of dotted types. Even in a sentence as “*John spilled coffee at the newspaper*” both aspects of the newspaper, physical object and information, are present. It is only the verb which selects one of them. This means that the existential quantifier  $\exists x$  newspaper( $x$ ) the logical form of the sentence allows us to infer the existence of a physical object and the information contained in it but the predicate spilled-at(*coffee*,  $y$ ) takes as argument only the physical substance  $y$  of  $x$ . In (1) and (2), however, the different meanings are not different co-existing aspects of one entity or one event which are selected by the verb but exists independently of each other.

We next have a closer look at the dependency between meaning variations and argument positions of verbs.

### 3 Polysemy and Argument Roles

Polymorphism refers to the phenomenon that operations show different behavior with different arguments. Polysemy means the phenomenon that one word has several meanings. As mentioned in the introduction, Keenan (1984) observes that there is a universal regularity to be found in the languages of the world according to which the meaning of intransitive verbs depends on the subject argument, and the meaning of transitive verbs depends on the direct object. Examples for the first rule, we have seen in (1), and for the latter rule in (2). The following German/English examples confirm this observation:

- (3) a. *Hans schlägt Peter*  
*John beats Peter*  
 b. *Hans schlägt den Esel*  
*John beats the donkey*  
 c. *Hans schlägt die Trommel*  
*John beats the drum*  
 d. *Hans schlägt die Schlagsahne*  
*John beats the cream*

In (3a), a situation in which Hans uses his hands for beating Peter is expected, whereas in (3b) we expect him to use an instrument as, e.g., a stick. In (3c) and (3d), we also expect him to use some instrument but of different type. In (3a) and (3b), it is Hans’ goal to hurt the beaten; in

(3c) his goal is to produce rhythmic sounds, and in (3d) to make the cream fluffy. The following examples show that, if the arguments are of the same type, the meaning does not change:

- (4) a. *Hans schlägt Peter / sein Kind / den Schaffner / die Großmutter*  
*John beats Peter / his child / the conductor / grandmother*  
 b. *Hans schlägt den Esel / das Pferd / das Kamel / die Kuh*  
*John beats the donkey / the horse / the camel / the cow*  
 c. *Hans schlägt die Trommel / die Pauke / das Becken*  
*John beats the drum / the timpani / the cymbal*  
 d. *Hans schlägt die Schlagsahne / das Eiweiß / den Honig*  
*John beats the cream / the egg clear / the honey*

Keenan (1984) notes that subjects of transitive verbs have no significant influence on verb meaning. He provides the following examples:

- (5) a. *John / the machine / the company / the storm cut the lawn*  
 b. *John / the machine / the company / the storm cut the cake*  
 c. *John / the machine / the company / the storm cut a path (through the field)*  
 d. *John / the machine / the company / the storm cut production quota / prices*

This is also confirmed by the following German/English examples:

- (6) a. *Hans / der Mähdrescher / der Affe schlägt den Mann*  
*John / the combine harvester / the monkey beats the man*  
 b. *Hans / der Mähdrescher / der Affe schlägt den Esel*  
*John / the combine harvester / the monkey beats the donkey*  
 c. *Hans / der Mähdrescher / der Affe schlägt die Trommel*  
*John / the combine harvester / the monkey beats the drum*  
 d. *Hans / der Mähdrescher / der Affe schlägt die Schlagsahne*  
*John / the combine harvester / the monkey beats the cream*

There may be conflicts with selectional restrictions; e.g., a combine harvester has no arms to beat a man, and it is also not possible to use it for beating cream. This makes it difficult to interpret the respective sentences, but these violations of selectional restrictions do not create new meanings.

If we assume that the lexicon is organized along object-oriented lines, it follows immediately that the meaning of verbs can only depend on one argument, namely the argument in the class of which they are defined. When reading a verb, the addressee therefore must know by which argument it is defined. We call the position in which the defining argument has to be found the *defining core role* of the verb.

As in a standard lexicon, we can assume that each verb has a lexical entry on its own. But this entry only needs to provide the information necessary for identifying the defining core role. This entails that the specification of optional arguments is not necessary. If the verb has one obligatory argument only, then this argument must show the defining core role. If it has two or more, then there is an option. In line with Keenan's Observation, indexObjectsKeenan's Observation we can



hypothesize that the class of the affected object defines the meaning of the verb. For transitive verbs, the affected object typically is the direct object, whereas the subject is typically filled by the entity which initiates the action which affects the direct object. This leads to the following working hypothesis about verbs with an agentive argument:

1. Transitive verbs (*cut, hit, see, move to, carry*) correspond to procedures where the initiating role is different from the defining core role.
2. Intransitive verbs (*grow, turn, walk, sleep, look around*) correspond to procedures where the initiating role is the same as the defining core role.

The initiating and the defining core role do not need to be disjoint. If a process only manipulates or reads the data that belong to the initiating object, then both roles, the initiating and the defining, are filled by the same object. This is the case for intransitive verbs, but also for reflexive transitive sentences:

- (7) a. *The woman shaves her husband*
- b. *The husband shaves himself*
- c. *The husband shaves*

In all three cases, the default interpretation of the shaving event is the same: the husband gets his facial hair removed by use of a razor. This meaning depends on the argument which received the shaving, as can be seen from the sentence *The woman shaved herself* in which *shave* has a quite different interpretation.

We use Dixon's (1994) terminology, and distinguish between the argument roles S for subjects of intransitive verbs, A for subjects of transitive verbs, and O for direct objects of transitive verbs. Keenan notices that arguments in S and O relation are commonly *patients*. This means that they are affected by the event denoted by the sentence as in the following examples:

- (8) a. *The car exploded / broke down*
- b. *The milk evaporated / spilled*
- c. *Fred's argument collapsed / fell apart*
- d. *John blew up the car*
- e. *John spilled the milk*
- f. *John destroyed Fred's argument*

In contrast, arguments in A relation rarely are patients. Instead, they are typically *agents, instruments*, or *experiencers*.<sup>4</sup>

Although there is a strong statistical bias, this regularity is not a hard universal constraint as the following examples show:

- (9) a. *John jumped*
- b. *John bores*

---

<sup>4</sup>Keenan also discusses the case of direct objects which are assigned the role of *themes* with verbs of motion, i.e., they are objects which move from one location to another location as in *Bill threw the log into the pool*.

- c. *The bridge crosses the river*
- d. *The police surrounded the house*

In (9a) the subject is an *agent*, and in (9b) something we might call a *cause*; in (9c) and (9d) the direct object is assigned *locative* role.

Our hypothesis about the existence of a unique defining core role and its connection to S and O relations carries no implications about the thematic roles assigned to these relations. In so far, it is in full agreement with the observation that there are only statistical preferences for the kind of thematic role but no hard universal constraints. However, the defining core role is not independent of the thematic properties of the arguments as the following variations of examples (9c) and (9d) show (Keenan 1984: footnote 4):

- (10) a. *The soldiers cross the river*  
*The bridge crosses the river*
- b. *The police surrounded the house*  
*The moat surrounded the house*

The meaning of *cross* and *surround* varies with the subjects in these examples. In contrast to previous examples, the direct object provides a *location*, i.e., it does not denote an object which is affected by an event or state. This seems to indicate that meaning variation depends on the type of thematic role which is assigned to an argument rather than its syntactic role.

Keenan acknowledges that meaning of verbs can vary with general properties such as animacy and agenthood of their A argument, but he argues that the differences in examples like (10) are of a more systematic nature than the *ad hoc* meaning differences generated by different direct objects. However, if we assume that the defining core role is always the direct object, then these examples pose a problem which cannot be remedied so easily. Let us discuss some possible explanations.

A first solution is to assume that a hidden ambiguity of the direct object is responsible for the deviations exhibited by the examples in (10). The event type in (10a) and (10b) changes from accomplishment to state, whereas it always remains the same in (1) and (2). If meaning is inherited and modified from a super-class, then the latter regularity is to be expected, and we may speculate that, e.g., *the river* in (10a) figures as an *obstacle* in the first sentence, and as a *landmark* in the second. Hence, *the river* shows here a similar behavior as dotted objects do, and we could explain the examples in (10) by assuming that *river* is of type **obstacle·landmark**, and *cross* is defined one time in the super-class *obstacle*, and the other time in the super-class *landmark*. Against this analysis stand examples like “*The plane / the bird / the balloon crossed the river*” in which *the river* does not figure as an obstacle, the event type is still that of accomplishment, and the subject moves over the river along a path through air. The meaning differentiations clearly depend on the semantic properties of the subject, and not on a hidden polysemy of the direct object.

Another solution is to assume that the defining core role is associated with a thematic role and not with a syntactic role. The defining core role cannot simply be identified with the patient or theme as the subject of intransitive verbs may take on all kinds of thematic roles. But as intransitive verbs have only one argument, they may be left out of consideration. However, there is an argument connected to selectional restrictions which tells against this solution.

Let us reconsider the examples in (4). What the direct objects have in common is the fact that they receive a blow from some object, but there is no common super-class for which the verb *schlagen* (*beat*) is defined. For example in (4c), *schlagen* can only be applied to drum like music instruments. If they are not drum like, selectional restrictions are violated as in “*Hans schlägt die Flöte*” (*John beats the flute*). Therefore, the four types of direct objects in (4) cannot be subsumed, e.g., under the general super-class of physical objects. Hence, the defining core role cannot be identified with a general super-class. It can be argued now that thematic roles do not define classes in the object-oriented sense, but have to be identified with a pair consisting of a class together with an abstract active operation. For example, an instance of the *patient* role may be defined as an instance of the physical object class which is affected by some force, a prototypical *agent* an animated being which performs some action, an *experiencer* an animated being which experiences some sensation, a *result* some entity which is produced by an action, etc. In this sense, all direct objects in (4) can be said to be *patients*. But the subjects in (10) fill different thematic roles, and even for *schlagen* we can find examples as *Die Stadt schlug Münzen mit dem Portrait des Kaisers* (*The city struck (beat) coins showing the portrait of the emperor*) in which the direct object is a *result* and not a *patient*. A well known example with the same alternation of the thematic role of its direct object is *bake a cake* vs. *bake a potato* (Pustejovsky 1995). It would be possible to define roles even more abstractly, but then their function becomes indistinguishable from signaling the syntactic role of the argument. Hence, we conclude that the defining core role is always identified with a syntactic role, that this syntactic role is the role of the direct object for transitive verbs but can be the subject role in exceptional cases. The exceptional cases arise with direct objects which are not affected by the event or state denoted by the sentence. Hence, the defining core role cannot be identified with a specific thematic role but it can be excluded from some thematic roles.

## 4 Ergativity

It is a tempting idea that the object-oriented principles for system design are connected to the ergative/absolute case marking systems. Speakers want to be understood, hearers to understand. If the speaker says that *A man killed a lion*, he does not want that the addressee understands this as *A lion killed a man*. This means that the speaker has to make clear which noun phrase fills which argument position of the verb. Languages offer a variety of solutions to this problem, morphological case marking is one of them. We already introduced Dixon’s (1994) notation for the argument roles of subjects of intransitive verbs S, for subjects of transitive verbs A, and for direct objects of transitive verbs O. Every language has to make clear which NP in a transitive sentence is in A relation and which is in O relation. According to Dixon, there are three principal possibilities to do this: case marking a NP by affixation, by means of adpositions and particle, and by cross-referencing from the verb.

Among the successful case marking systems, there are the following two solutions: Either mark the O-role, and let the other roles be unmarked, or mark the A-role, and let S- and O-role be unmarked. If a language employs the former solution, then it has *nominative/accusative* case marking, if the latter, it has *ergative/absolute* case marking, with *accusative* and *ergative* being the marked forms. From a purely functional perspective, this may be all that there is to

say about the difference between ergative/absolutive and nominative/accusative case marking systems. Both are efficient in the sense that they uniquely identify the NPs in S, A, and O relation:

- *Nominative/accusative case marking*: NPs in S and A role are treated the same. NPs in O role are marked differently from those in S, A position. The case of S and A is called *nominative*, the case of O is called *accusative*.
- *Ergative/absolutive case marking*: NPs in S and O role are treated the same. NPs in A role are marked differently from those in S, O position. The case of S and O is called *absolutive*, the case of A is called *ergative*.

The following examples are from German which shows a nominative/accusative distinction:

- (11) a. Der Mann lächelt  
 'the.NOM man smiles-3rd.Sing.PRES'  
 'The man smiles'
- b. Der Mann füttert den Hund  
 'the.NOM man feed-3rd.Sing.PRES the.ACC dog'  
 'The man feeds his dog'

The determiner of the NP in S and A relation shows nominative marking (*der*), the determiner of the NP in O relation shows accusative marking (*den*). The subject is cross-referenced from the verb (-*t*, 3rd per. sing.).

An example of a language with ergative/absolutive distinction is Dyrbal, spoken in Queensland, Australia. The NP in S and O relation shows absolutive case, the NP in A relation shows ergative marking (-*ŋgu*) (Dixon 1994: p. 10):

- (12) a. ŋuma banaga-n<sup>y</sup>u  
 'father.ABS return-NONFUT'  
 'father returned'
- b. yabu banaga-n<sup>y</sup>u  
 'mother.ABS return-NONFUT'  
 'mother returned'
- c. ŋuma yabu-ŋgu bura-n  
 'father.ABS mother-ERG see-NONFUT'  
 'mother saw father'
- d. yabu ŋuma-ŋgu bura-n  
 'mother.ABS father-ERG see-NONFUT'  
 'father saw mother'

In general, nominative and absolutive are the unmarked cases, accusative and ergative the marked ones. Efficiency of linguistic code can be increased by, e.g., taking features like animacy or gender into account. In German for example, only a direct object with male grammatical gender is morphologically marked as accusative (*den*). If a direct object's grammatical gender is female

or neuter, its accusative form is identical to the nominative form. This can be explained by the fact that female or neuter direct objects are very frequent, whereas male objects are infrequent. Hence, the strife for efficient encoding of information should favor case marking that takes into account these different frequencies.<sup>5</sup>

The examples we presented all show *morphological* ergativity. The term *ergativity* is also used for syntactic regularities connected to the S, A, and O relation.<sup>6</sup> As we are only interested in semantic patterns which are related to ergativity, i.e., semantic patterns which treat NPs in S and O relation in the same way, we leave syntactic ergativity out of consideration.

The picture is made more complicated by the fact that in one and the same languages we may find both nominative and ergative patterns. Many ergative languages show a *split* between an ergative and an accusative pattern according to some grammatical category. There are the following possibilities (Dixon 1994: Ch. 4):

1. A split conditioned by the semantic nature of the verb. There may be a split in the class of intransitive verbs depending on, e.g., whether the described eventuality is an activity or done voluntarily.
2. A split conditioned by the semantic nature of the NPs.
3. A split conditioned by tense / aspect / mood.
4. A split between main and subordinated clauses.

The split conditioned by tense / aspect / mood distinguishes often between past / perfect and present / future. According to (Trask 1979), the languages with a split in the NPs have a different historic origin than languages with a split according to tense / aspect / mood. In languages of the latter type, the effect of an ergative seems to be similar to a perfective emphasizing the result of an event. According to (Trask 1979), this split is restricted to languages which do not have an auxiliary for *have*.

We have already noticed that explicit morphological marking of accusative case may be conditioned by the grammatical gender of an NP. Since (Silverstein 1976) it is well known that the distribution of ergative marking in a language may be conditioned by an animacy scale. In the following scale, the NPs to the left are more likely to follow an accusative pattern, the NPs to the right more likely to follow an ergative pattern. Where exactly the split occurs is language dependent.

1st pers. pronouns	2nd pers. pronouns	Demonstratives 3rd pers. pronouns	Proper nouns	Common nouns human animate inanimate
-----------------------	-----------------------	---	-----------------	---

Trask (1979) claims that this type of split only occurs in languages in which the ergative originated from a reanalysis of a passive construction. Another obvious theoretical explanation

<sup>5</sup>This is the basis for evolutionary and diachronic learning models for differential object marking as proposed by (Jäger 2004, 2007).

<sup>6</sup>See for example (Dixon 1994: p. 13).

of the origin of this type of split is to assume a nominative/accusative system at the beginning for which explicit accusative case marking gets restricted to animate objects. This leads to an ambiguity if inanimate NPs occur in both A and S relation. This situation can be disambiguated by marking the inanimate subject, which leads to an ergative language with a split along the animacy scale.

There is an obvious motivation for the ergative/absolutive case distinction if we assume that the mental lexicon is organized along object-oriented lines. In this case, the addressee not only has to know which NP fills which argument position, but also which noun phrase defines the meaning of the verb. In an ergative/absolutive case marking system, this noun phrase is always the absolutive (un)marked NP. The existence of splits in the case marking system of a language does not pose any problem to this explanation as the objective of indicating the defining core role may only be one among other competing objectives.

This short overview of grammatical phenomena related to ergativity can only mention the most important features. All of them can be explained along purely functional lines, i.e., by the fact that languages have to solve the task of assigning NPs to argument roles and do it in an efficient way. Hence, our hypothesis that ergative/absolutive case marking is related to object-oriented principles of organizing the lexicon cannot be directly tested. What is of interest to us are, therefore, not the morpho-syntactic patterns associated with ergativity but the semantic regularities observed by Keenan (1984). We turn to these regularities in the next section.

## 5 Correlates of the Ergative/Absolutive Distinction

Keenan (1984) lists a number of grammatical phenomena that link the S role together with the O role. We are interested whether these regularities support the theses that there exist defining core roles, and that these roles are the S and the O role. Keenan divides the phenomena into three groups:

1. ***Bondedness to the verb***: under this heading, Keenan lists existence dependencies, semantic polysemy, selectional restrictions and verbal classifiers, and noun incorporation.
2. ***Thematic role properties***: Keenan notes that NPs in S and O relation are often assigned the thematic roles of *patient* or *theme*.
3. ***Control phenomena***: under this heading he addresses expressions such as adjectives and infinitival phrases predicating something about the arguments of the verb. For example, he notes that adjectives within a predicate are normally denoting a property of the object in S or O relation.

We have already discussed the thematic role properties and found that the defining core role cannot be identified with a thematic role but that some thematic roles are excluded from being the defining core role. We are now going to discuss the other regularities one by one with relation to our hypothesis that each verb has exactly one defining core role. We are especially interested in putative counter-examples.

## 5.1 Bondedness to the verb

There is a large class of verbs which refer to events which involve the coming into existence of an object. These verbs may be intransitive (*A crowd gathered around John*) as well as transitive:

- (13) a. *A student lit a fire in the basement*  
 b. *He committed a crime/made a mistake*

As Keenan notes, the object coming into existence is always the object appearing in O relation.<sup>7</sup> Given our assumption that for each verb there must be a defining core role, it is not surprising that this core role is the role of the object created by an activity. What is involved in creating an entity much more depends on the type of entity created than on the creator. For our argument it is important that we again find that the meaning differs with different direct objects but not with different subjects:

- (14) a. *John painted a picture / the wall*  
 b. *John / God / the machine painted a picture*  
 c. *John / the machine painted the wall*

Whether an object comes into existence depends on the type of direct object: painting a picture creates a picture but not painting a wall.

There are, however, examples in which the verb meaning appears to depend on the subject:

- (15) a. *John painted a wall*  
 b. *Rembrandt painted a wall*

Sentence (15b) will normally be understood as meaning that Rembrandt painted a *picture of a wall*. This putative counter-example can be explained if we assume that the literal content of the sentence is pragmatically enriched to *Rembrandt painted a picture of a wall*. We only have to assume that the specific interpretation of the verb is calculated after enrichment.<sup>8</sup> In this case, the meaning difference is caused by a difference in the types of direct objects.

Under the heading of *multiple senses*, Keenan discusses the meaning variation which we find in our core examples (1) and (2), and which we called *Keenan's Observation*.

In (10), we have already seen examples which seem to contradict his observation. There are also other putative counter-examples which seem to show that the subject in A role changes the meaning of a verb (Benz 2006):

- (16) a. *The monk killed the patient*  
 b. *The doctor killed the patient*  
 c. *The psychologist killed the patient*  
 d. *The sadist killed the patient*

<sup>7</sup>We concentrate on transitive verbs in this section as they provide the crucial data.

<sup>8</sup>*Pragmatic enrichment* here refers to pragmatic strengthening of meanings as postulated by relevance theorists, specifically to *free enrichment*; see, e.g., (Carston 2004).

In all examples, we find some slight difference in interpretation. In the second, third and fourth example, we would expect that the killing is indirect and in some way connected to the profession of the subject and the role of the victim. In *The doctor/the psychologist/the pharmacist killed the butterfly* a much more direct killing could be expected. In (16d) the expected killing may again be more direct but in addition also extended and torturous — in contrast to (16a). These differences do not affect the semantics of the event type; they are rather related to different modes of the killing, e.g., they may carry special expectations about the instrument (*pharmacist*), and are different connotations rather than different meanings of the sentence.

Another property which is common to S and O relation but not to A relation is the existence of rather strict selectional restrictions. Keenan mentions, e.g., the case of the ambitransitive verb *spill*: The objects which can spill or which can be spilled by someone seem to be restricted to liquids and finely granular objects like coffee grounds. In contrast, he notes that he knows of no verb which would impose a similar restriction to the A role. Furthermore, things that shatter or can be shattered must have a special physical property. Keenan claims that there seem to be no transitive verbs that impose restrictions of the same kind on their subjects (A).

We have already discussed the *Schlagen*-examples in (4). We have seen that there is no common super-class which would be identical to the set of all possible direct objects. We found, for example, a restriction to drum like music instruments (4c), or to human beings in (4a). The different types of direct objects in (4) can, in particular, not be subsumed, e.g., under the general super-class of physical objects, as there are physical objects which cannot appear in direct object position of *schlagen*. Hence, we find also in this example an instance of very fine-grained selectional restrictions in the defining core role.

The assumption that the meaning of verbs is defined in the class of its defining core role poses no restrictions on the fine-grainedness of the selectional restrictions for this role. It is not necessary that the defining classes combine to a single super-class, or that the defining classes are high up in the class hierarchy. The further down in the class hierarchy they are, the more fine-grained the restriction on the argument type will be. In contrast, the restrictions for the other arguments must be provided together with the definition of the verb's meaning. Once the defining core role is filled, the general selectional restriction and the theta role of the remaining argument must be defined. Hence, variation of thematic roles, if it exists, should always be determined by the class of the direct object.

A similar pattern as with selectional restrictions can be found in the restrictions imposed by *object* classifiers. These classifiers are affixes to the verb root and require that certain arguments satisfy specific semantic constraints. In Navajo, e.g., there exists a class of six affixes; if they are attached to a verb, then they indicate that an argument is *a round solid object, a long slender object, a wool-like mass, a mud-like mass, etc.*<sup>9</sup> The available data seem to indicate that these restrictions only apply to NPs in S and A relation but not to those in S relation. Both phenomena lend strong support to the assumption that verb meaning is defined in the class denoted by the NP in S or O relation.

The final regularity which Keenan discusses under the heading of bondedness to the verb is noun incorporation. Noun incorporation is a process by which a noun is incorporated in a verbal

---

<sup>9</sup>See (Keenan 1984: Sec. A.3) for references.



structure as, e.g., in German *staubsaugen* (*dust-sucking*, i.e., *vacuum cleaning*). Keenan uses a survey by (Mardirussian 1975) which has only examples for the incorporation of nouns from S or from O relation but no nouns that come from A position. This phenomenon is also in line with the assumption that verb meaning is defined in the arguments in S and O relation.

## 5.2 Control phenomena

As mentioned before, Keenan considers here expressions such as adjectives and infinitival phrases which occur as arguments of a verb and predicate some property of another NP argument. Examples containing adjectives and one subject NP are:

- (17) a. *John looks smart / angry / lazy*  
 b. *The meeting sounds noisy / calm*  
 c. *The milk turned sour / green*  
 d. *The door flew open*

The following examples also contain a direct object:

- (18) a. *John saw Bill angry*  
 b. *John considers Bill smart*  
 c. *John packed the meat raw*  
 d. *The sun turned the milk sour*  
 e. *John drove Bill insane*

Keenan notes that the adjective either predicate about arguments in S or O relation but not about those in A relation. This is in line with our hypothesis. It is to be expected that if the direct object defines the specific meaning of the verb, it also defines the specific meaning of the adjective which occurs in an argument position. There are also examples in which the meaning varies with the argument in S and O position:

- (19) a. *John turned sour*  
 b. *The milk turned sour*  
 c. *Supposedly safe investments turned sour*  
 d. *Relations turn sour*  
 e. *John considers Bill smart*  
 f. *John considers the plan smart*

However, it is not obvious whether these examples really support our claim. The examples in (19c) and (19d) are metaphorical, and the *smart* in (19e) may simply refer to a special aspect of Bill namely his ideas. The meaning of *sour* in (19c) is not lexicalized, hence, this example should better be explained by some productive pragmatic mechanism.

Keenan's examples of infinitive phrases exhibit the same pattern as the examples with adjectives. The following may serve as illustration:

- (20) a. *John needs to arrive on time*

- b. *John started to study law*
- c. *John asked to leave the room*
- d. *John asked Bill to leave early*
- e. *John obliged Bill to review the proposal*
- f. *The incident caused Bill to lock his door at night*

From these examples Keenan infers that the infinitive phrase predicates something about either the argument in S or O relation. Again, these examples are not convincing. For example, (20c) does not fit into this pattern, and the examples with transitive verbs report speech acts. If a commissive speech act is reported then the infinitive phrase predicates a future act of the A argument, as in *John promised Bill to leave early*. It is not a marginal exception, as Keenan indicates, but well in line with the predictions of speech act theory.

We may conclude that the examples under the heading of *bondedness to the verb* are the most interesting to our hypothesis, and the only ones which really allow it to be tested against the data. The most problematic examples are in (10), repeated as (21):

- (21) a. *The soldiers cross the river*  
       *The bridge crosses the river*
- b. *The police surrounded the house*  
       *The moat surrounded the house*

All other examples are in accordance with the assumption that the defining core role is either the S or the O role but never the A role.

## 6 Conclusion

We started out with Keenan's Observation that intransitive verbs can receive different interpretations for different subjects, and that transitive verbs can receive different interpretations for different objects but not for different subjects. The main objective of this paper was to point out an interesting motivation for this regularity in the representation of one-place and two-place operations in object-oriented programming. Operations that correspond to intransitive verbs are encoded in the class which corresponds to the subject NP, and operations that correspond to transitive verbs are encoded in the class that corresponds to the NP of the direct object. This led us to the hypothesis that ergative case marking is conceptually connected with the object-oriented structure of the lexicon. We discussed a number of semantic phenomena related to ergativity following (Keenan 1984) which show that the semantic meaning of the verb is determined by subjects of intransitive, or by objects of transitive verbs.

We can summarize the discussion by the following hypotheses:

1. For each verb there exists exactly one defining core role. This means:
  - a) If we fix all other NPs, then replacing the NP in the defining core role may lead to a different interpretation of the verb.
  - b) If we fix the NP in the defining core role, then replacing other NP arguments will not change the interpretation of the verb. (*Keenan's Observation*).

2. The selectional restrictions on the arguments of the defining core role can be arbitrarily fine-grained. Moreover,
  - a) Semantic restrictions on the core role can be identified with the set of classes in which the verb is defined.
  - b) Possible selectional restrictions on the other arguments are determined by the class of the argument in the defining core role.
3. A verb cannot be interpreted if the object in defining core role is not known. More specifically:
  - a) If the NP in the non-core role is omitted, then an interpretation is still possible.
  - b) If the NP in the defining core role is omitted, then the argument type must be inferable from context.

## Bibliography

- Armstrong, Deborah J. (2006): "The quarks of object-oriented development." *Communications of the ACM* 49(2): 123–128.
- Benz, Anton (2006): "Partial blocking and associative learning." *Linguistics and Philosophy* 29: 587–615.
- Booch, Grady, Maksimchuk, Robert A., Engle, Michael W., Young, Bobbi, Conallen, Jim & Houston, Kelli A. (2007): *Object-Oriented Analysis and Design with Applications*. Amsterdam: Addison-Wesley Longman.
- Briscoe, Ted, de Paiva, Valeria & Copestake, Ann (eds.) (1993): *Inheritance, Defaults, and the Lexicon*. Cambridge: Cambridge University Press.
- Carston, Robyn (2004): "Relevance theory and the saying/implicating distinction." In: Lawrence Horn & Gregory Warden (eds.) *The Handbook of Pragmatics*, Oxford: Blackwell Publishing. 632–656.
- Dixon, Robert Malcolm Ward (1994): *Ergativity*, volume 69 of *Cambridge Studies in Linguistics*. Cambridge: Cambridge University Press.
- Jäger, Gerhard (2004): "Learning constraint subhierarchies: The bidirectional gradual learning algorithm." In: Reinhard Blutner & Henk Zeevat (eds.) *Optimality Theory and Pragmatics*, Basingstoke: Palgrave Macmillan. 251–287.
- Jäger, Gerhard (2007): "Evolutionary game theory and typology. A case study." *Language* 83: 74–109.
- Keenan, Edward J. (1984): "Semantic correlates of the ergative/absolute distinction." *Linguistics* 22: 197–223.

Mardirussian, Galust (1975): “Noun incorporation in universal grammar.” *Chicago Linguistic Society Bulletin* 11: 383–389.

Pustejovsky, James (1995): *The Generative Lexicon*. Cambridge, MA: The MIT Press.

Silverstein, Michael (1976): “Hierarchy of features and ergativity.” In: Robert Malcolm Ward Dixon (ed.) *Grammatical Categories in Australian Languages*, Canberra: Australian Institute of Aboriginal Studies, volume 22 of *Linguistic Series*. 112–171.

Trask, Robert Lawrence (1979): “On the origins of ergativity.” In: Frans Planck (ed.) *Ergativity: Towards a Theory of Grammatical Relations*, London: Academic Press. 385–406.