Foundations of Computational Linguistics

man-machine communication in natural language

ROLAND HAUSSER
Computational Linguistics
Universität Erlangen Nürnberg
Germany

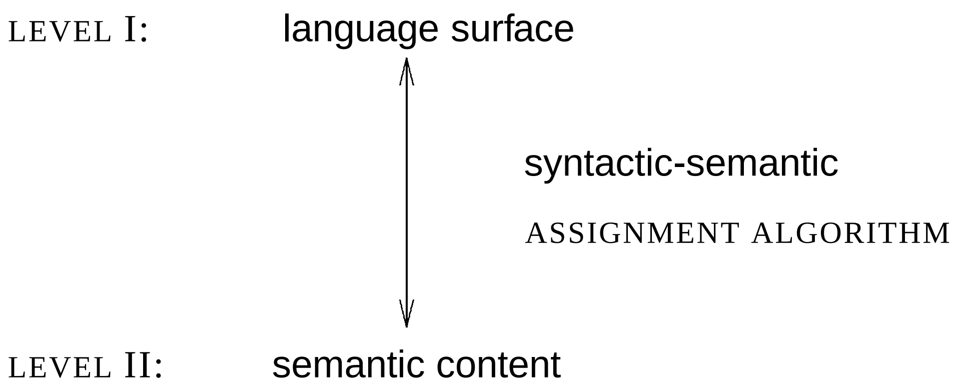
Part IV Semantics and Pragmatics

Part IV
Semantics and Pragmatics

19. Three system types of semantics

19.1 Basic structure of semantic interpretation

19.1.1 The 2-level structure of semantic interpretation



19.1.2 The function of semantic interpretation

For purposes of transmission and storage, semantic content is coded into surfaces of language (representation). When needed, the content may be decoded by analyzing the surface (reconstruction).

The expressive power of semantically interpreted languages resides in the fact that representing and reconstructing are realized *automatically*: a semantically interpreted language may be used correctly without the user having to be conscious of these procedures, or even having to know or understand their details.

19.2 Logical, programming, and natural languages

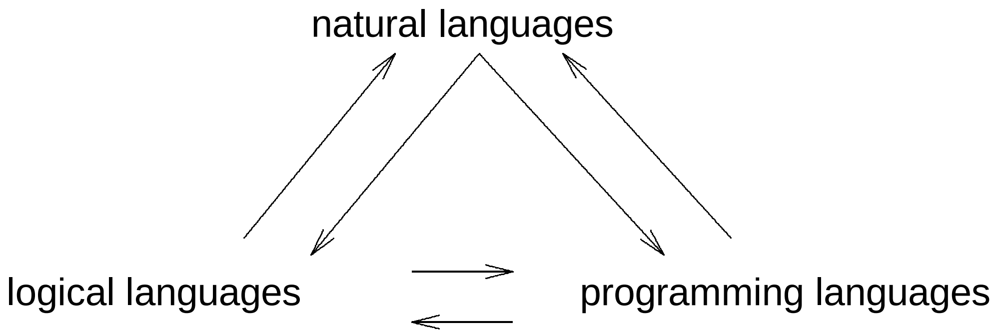
19.2.1 Three different types of semantic systems

1. *Logical languages*
   Designed to determine the truth value of arbitrary propositions relative to arbitrary models. The correlation between the two levels is based on *metalanguage definitions*.
2. *Programming languages*
   Designed to simplify the interaction with computers and the development of software. The correlation between the two levels is based on the *procedural execution* on an abstract machine, usually implemented electronically.
3. *Natural languages*
   Preexisting in the language community, they are analyzed syntactically by reconstructing the combinatorics of their surfaces. The associated semantic representations have to be deduced via the general principles of natural communication. The correlation between the two levels is based on *conventional association*.

19.2.2 Three types of semantic interpretation

	logical languages	**programming languages**	**natural languages**
LEVEL I:	propositions	commands	surfaces
	metalanguage definition	procedural execution	conventional association
LEVEL II:	*set theoretic model of the 'world'*	*operations on an abstract machine*	*literal meanings used by the speaker-hearer relative to the internal context*

19.2.3 Mapping relations between the three types of semantics

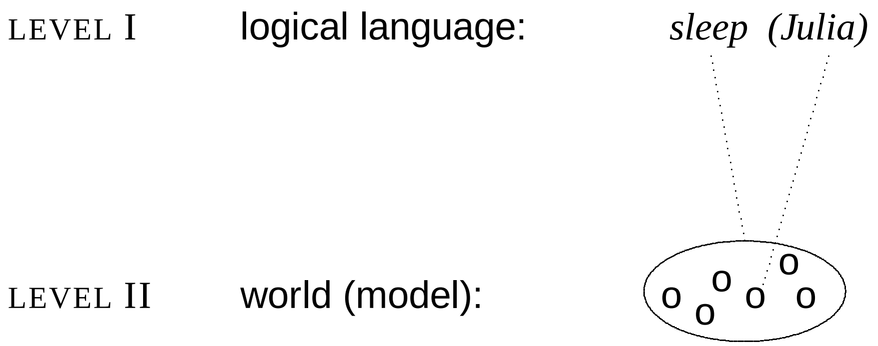


19.2.4 Characterizing the mapping relations: Replication, Reconstruction, Transfer, and Combination

- *Replication*
  Selected natural language phenomena are replicated in logical languages (N→L). Selected aspects of logical languages are replicated procedurally in programming languages like LISP and Prolog (L→P). The programming languages also replicate natural language concepts directly, e.g. 'command' (N→P).
- *Reconstruction*
  Theoretical linguistics attempts to reconstruct fragments of natural language in terms of logic (L→N). Computational linguistic aims at reconstructing natural languages by means of programming languages (P→N). One may also imagine a reconstruction of programming concepts in a new logical language (P→L).
- *Transfer*
  Computer science attempts to transfer methods and results of logical proof theory into the programming languages (L→P). Philosophy of language attempts to transfer the model-theoretic method to the semantic analysis of natural language (L→N).
- *Combination*
  Computational linguistics aims at modeling natural communication with the help of programming languages (P→N). Thereby methods and results of the logical languages play a role in both, the construction of programming languages (L→P) and the analysis of natural language (L→N). This requires a functional overall framework for combining the three types of language in a way that utilizes their different properties while avoiding redundancy as well as conflict.

19.3 Functioning of logical semantics

19.3.1 Interpretation of a proposition



19.3.2 Definition of a minimal logic

1. **Lexicon**

   Set of one-place predicates: {sleep, sing}
   Set of names: {Julia, Susanne}

2. **Model**
   A model $\mathcal{M}$ is a two-tuple (A, F), where A is a non-empty set of entities and F a denotation function (see 3).

3. **Possible Denotations**
   (a) If P_1 is a one-place predicate, then a possible denotation of P_1 relative to a model $\mathcal{M}$ is a subset of A. Formally, $F(P_1)\mathcal{M} \subseteq A$.
   (b) If α is a name, then the possible denotations of α relative to a model $\mathcal{M}$ are elements of A. Formally, $F(\alpha)\mathcal{M} \in A$.
   (c) If ϕ is a sentence, then the possible denotations of ϕ relative to a model $\mathcal{M}$ are the numbers 0 and 1, interpreted as the truth values 'true' and 'false.' Formally, $F(\phi)\mathcal{M} \in \{0,1\}$.

   Relative to a model $\mathcal{M}$ a sentence ϕ is a true sentence, if and only if the denotation ϕ in $\mathcal{M}$ is the value 1.

4. **Syntax**
   (a) If P_1 is a one-place predicate and α is a name, then $P_1(\alpha)$ is a sentence.
   (b) If ϕ is a sentence, then $\neg\phi$ is a sentence.
   (c) If ϕ is a sentence and ψ is a sentence, then ϕ & ψ is a sentence.

(d) If ϕ is a sentence and ψ is a sentence, then $\phi \vee \psi$ is a sentence.
(e) If ϕ is a sentence and ψ is a sentence, the $\phi \rightarrow \psi$ is a sentence.
(f) If ϕ is a sentence and ψ is a sentence, then $\phi = \psi$ is a sentence.

5. **Semantics**

(a) '$P_1(\alpha)$' is a true sentence relative to a model $\mathcal{M}$ if and only if the denotation of α in $\mathcal{M}$ is element of the denotation of P_1 in $\mathcal{M}$.
(b) '$\neg\ \phi$' is a true sentence relative to a model $\mathcal{M}$ if and only if the denotation of ϕ is 0 relative to $\mathcal{M}$.
(c) 'ϕ & ψ' is a true sentence relative to a model $\mathcal{M}$ if and only if the denotations of ϕ and of ψ are 1 relative to $\mathcal{M}$.
(d) '$\phi \vee \psi$' is a true sentence relative to a model $\mathcal{M}$ if and only if the denotation of ϕ or ψ is 1 relative to $\mathcal{M}$.
(e) '$\phi \rightarrow \psi$' is a true sentence relative to a model $\mathcal{M}$ if and only if the denotation of ϕ relative to $\mathcal{M}$ is 0 or the denotation of ψ is 1 relative to $\mathcal{M}$.
(f) '$\phi = \psi$' is a true sentence relative to a model $\mathcal{M}$ if- and only if the denotation of ϕ relative to $\mathcal{M}$ equals the denotation of ψ relative to $\mathcal{M}$.

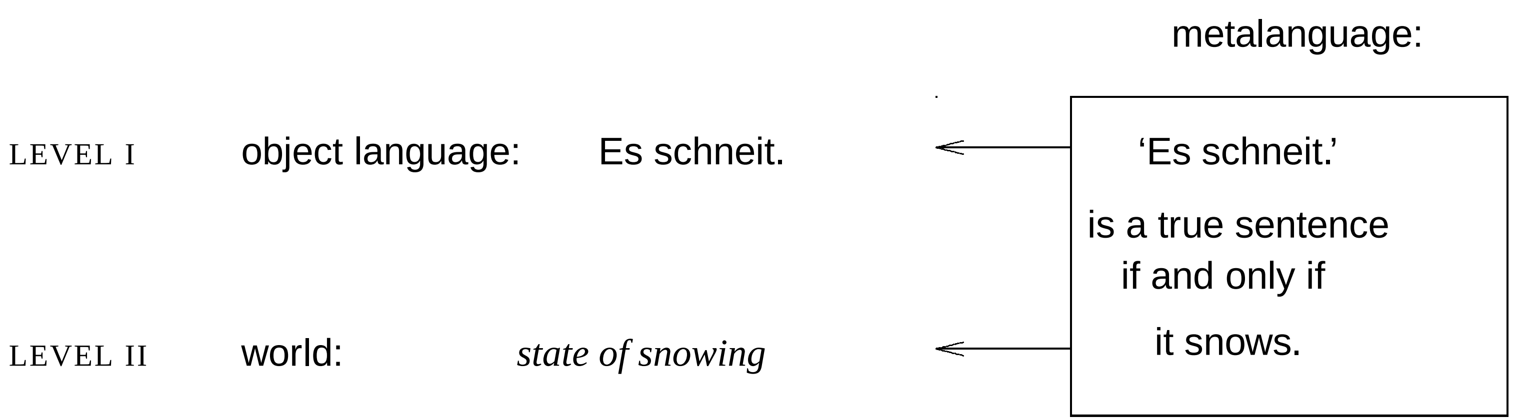
19.3.3 Schema of Tarski's T-condition

T: x is a true sentence if and only if p.

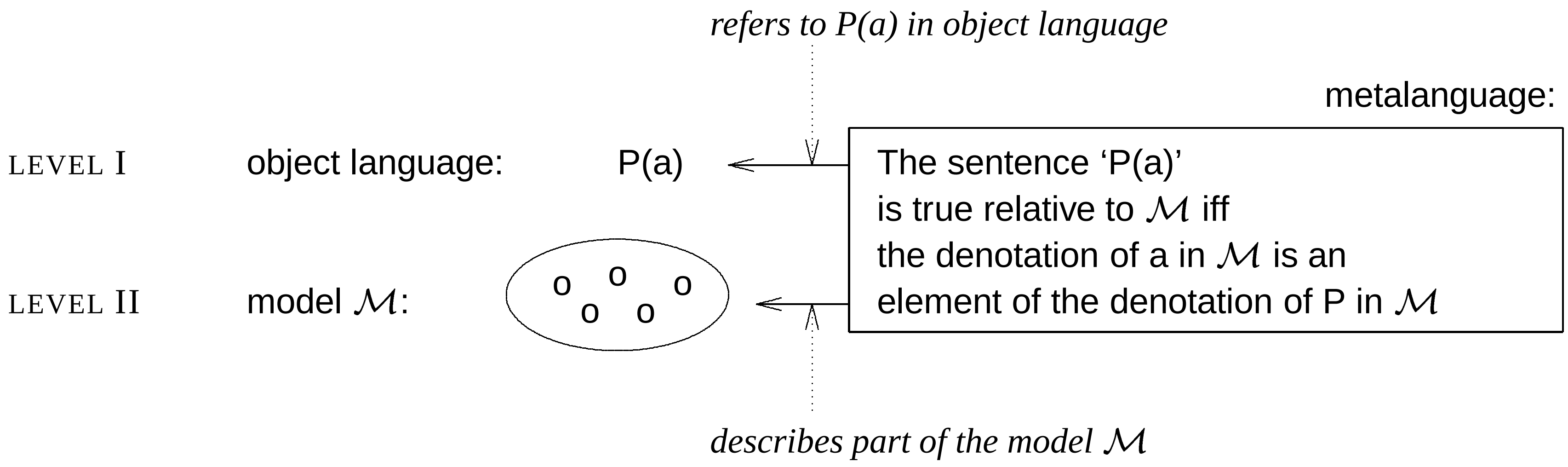
19.3.4 Instantiation of Tarski's T-condition

'Es schneit' is a true sentence if and only if it snows.

19.3.5 Relation between object and metalanguage



19.3.6 T-condition in a logical definition



19.3.7 The appeal to immediate obviousness in mathematics

En l’un les principes sont palpables mais éloignés de l’usage commun de sorte qu’on a peine à tourner late tête de ce côte-la, manque d’habitude : mais pour peu qu’on l’y tourne, on voit les principes à peine; et il faudrait avoir tout à fait l’esprit faux pour mal raisonner sur des principes si gros qu’il est presque impossible qu’ils échappent.

[In [the mathematical mind] the principles are obvious, but remote from ordinary use, such that one has difficulty to turn to them for lack of habit : but as soon as one turns to them, one can see the principles in full; and it would take a thoroughly unsound mind to reason falsely on the basis of principles which are so obvious that they can hardly be missed.]

B. PASCAL (1623 -1662), *Pensées*, 1951:340

19.4 Metalanguage-based versus procedural semantics

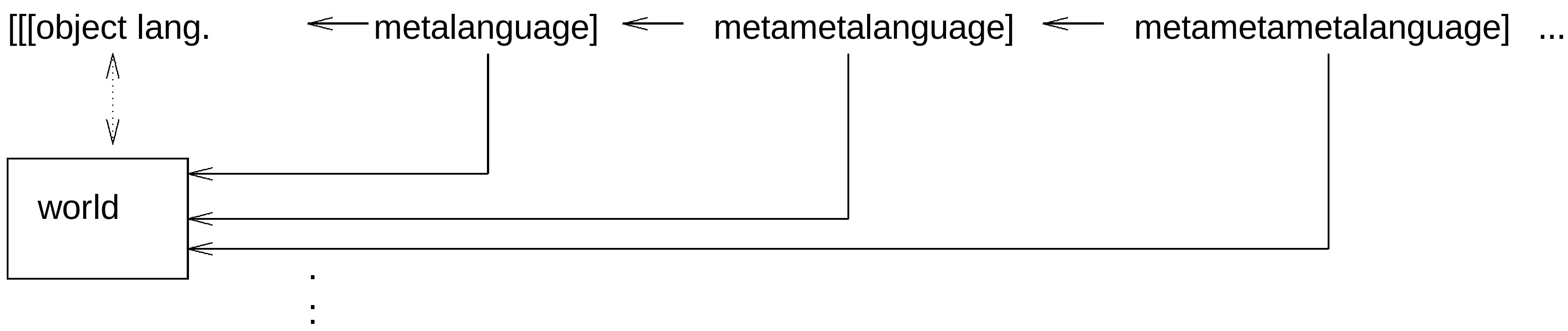
19.4.1 Example of a vacuous T-condition

‘A is red’ is a true sentence if and only if A is red.

19.4.2 Improved T-condition for red

‘A is red’ is a true sentence if and only if A refracts light in the electromagnetic frequency interval between α and β.

19.4.3 Hierarchy of metalanguages



19.4.4 Autonomy from the metalanguage

Autonomy from the metalanguage does not mean that computers would be limited to uninterpreted, purely syntactic deduction systems, but rather that Tarski's method of semantic interpretation is not the only one possible. Instead of assigning semantic representations to an object language by means of a metalanguage, computers use an operational method in which the notions of the programming language are realized automatically as machine operations.

19.4.5 Example of autonomy from metalanguage

There is no problem to provide an adequate metalanguage definition for the rules of basic addition, multiplication, etc. However, the road from such a metalanguage definition to a working calculator is quite long and in the end the calculator will function mechanically – without any reference to these metalanguage definitions and without any need to understand the metalanguage.

19.4.6 Programming logical systems

There exist many logical calculi which have not been and never will be realized as computer programs. The reason is that their metalanguage translations contain parts which may be considered immediately obvious by their designers (e.g. quantification over infinite sets of possible worlds in modal logic), but which are nevertheless unsuitable to be realized as empirically meaningful mechanical procedures.

19.5 Tarski's problem for natural language semantics

19.5.1 Logical semantics for natural language?

The attempt to set up a structural definition of the term 'true sentence' – applicable to colloquial language – is confronted with insuperable difficulties.

A. Tarski 1935, p. 164.

19.5.2 Tarski's proof

For the sake of greater perspicuity we shall use the symbol 'c' as a typological abbreviation of the expression 'the sentence printed on page 355 , line 8 from the bottom.' Consider now the following sentence:

c is not a true sentence

Having regard to the meaning of the symbol 'c', we can establish empirically:
(a) 'c is not a true sentence' is identical with c.
For the quotation-mark name of the sentence c we set up an explanation of type (2) [i.e. the T-condition 19.3.3]:
(b) 'c is not a true sentence' is a true sentence if and only if c is not a true sentence.
The premise (a) and (b) together at once give a contradiction:
c is a true sentence if and only if c is not a true sentence.

19.5.3 Inconsistent T-condition using Epimenides paradox

c	is	true	iff	c	is not	true
substitution based on abbreviation		*'true' from metalanguage T-condition*		*metalang. correlate to object language statement 'c'*		*metalang. correlate to object language word 'true'*

19.5.4 Three options for avoiding Tarski's contradiction in logical semantics

1. Forbidding the abbreviation and the substitution based on it. This possibility is rejected by Tarski because "no rational ground can be given why substitution should be forbidden in general."
2. Distinguishing between the truth predicate true^m of the metalanguage and true^o of the object language. On this approach

   c is true^m if and only if c is not true^o

   is not contradictory because true^m $\neq$ true^o.
3. This option, chosen by Tarski, consists in forbidding the use of truth predicates in the object language.

19.5.5 Reasons for the third option

If the goal is to characterize scientific theories like physics as true relations between logical propositions and states of affairs, then the vagueness and contradictions of the natural languages must be avoided – as formulated by G. Frege 1896:

> Der Grund, weshalb die Wortsprachen zu diesem Zweck [d.h. Schlüsse nur nach rein logischen Gesetzen zu ziehen] wenig geeignet sind, liegt nicht nur an der vorkommenden Vieldeutigkeit der Ausdrücke, sondern vor allem in dem Mangel fester Formen für das Schließen. Wörter wie >also<, >folglich<, >weil< deuten zwar darauf hin, daß geschlossen wird, sagen aber nichts über das Gesetz, nach dem geschlossen wird, und können ohne Sprachfehler auch gebraucht werden, wo gar kein logisch gerechtfertigter Schluß vorliegt.

> [The reason why the word languages are suited little for this purpose [i.e., draw inferences based on purely logical laws] is not only the existing ambiguity of the expressions, but mainly the lack of clear forms of inference. Even though words like ‘therefore,’ ‘consequently,’ ‘because’ indicate inferencing, they do not specify the rule on which the inference is based and they may be used without violating the well-formedness of the language even if there is no logically justified inference.]

The goal of characterizing scientific truth precludes reconstructing the object language as a natural language. Therefore there is no need for a truth predicate in the object language – which is in line with Tarski’s option.

19.5.6 Reasons against the third option

If the goal is to apply logical semantics to natural language, then the third option poses a serious problem. This is because natural language as the pretheoretical metalanguage *must* contain the words true and false. Therefore a logical semantic interpretation of a natural (object-)language in its entirety will unavoidably result in a contradiction.

19.5.7 Montague's choice: Ignoring the problem

> I reject the contention that an important theoretical difference exists between formal and natural languages. ... Like Donald Davidson I regard the construction of a theory of truth – or rather the more general notion of truth under an arbitrary interpretation – as the basic goal of serious syntax and semantics.
>
> R. Montague 1970

19.5.8 Davidson's choice: Suspending the problem

> Tarski's ... point is that we should have to reform natural language out of all recognition before we could apply formal semantic methods. If this is true, it is fatal to my project.
>
> D. Davidson 1967

20. Truth, meaning, and ontology

20.1 Analysis of meaning in logical semantics

20.1.1 The meaning principle of logical semantics

> If a speaker-hearer knows the meaning of a sentence, (s)he can say for any state of affairs whether the sentence is true or false with respect to it.

20.1.2 Existential generalization

The truth of a proposition F(a,b) implies that a exists and that b exists. For example, the sentence Julia kissed Richard is analyzed semantically as a *kiss*-relation between the entities Julia and Richard. If Julia kissed Richard is true, then it must be true that Julia exists and Richard exists.

20.1.3 Substitutivity of identicals

The premises F(b) and b = c, F(b) implies F(c). For example, if Richard = Prince of Burgundy, then the truth of the sentence Julia kissed Richard implies the truth of the sentence Julia kissed the Prince of Burgundy. This substitutivity of Richard and Prince of Burgundy *salva veritate*, i.e. preserving the truth-value, is based on the fact that these two different expressions denote the same object.

20.1.4 Valid and invalid instances of existential generalization

1) Julia finds a unicorn. $>$ A unicorn exists.
2) Julia seeks a unicorn. $\not>$ A unicorn exists.

The premises in these two examples have exactly the same syntactic structure, namely F(a,b). The only difference consists in the choice of the verb. Yet in (1) the truth of the premise implies the truth of the consequent, in accordance with the rule of existential generalization, while in (2) this implication does not hold.

20.1.5 First problem for extensional ontology

How a relation can be established between a subject and an object if the object does not exist. How can Julia seeks a unicorn be grammatically well-formed, meaningful, and even true under realistic circumstances?

20.1.6 Isolating the first problem

Part of the solution consisted in specifying certain environments in natural sentences in which the rule of existential generalization does not apply, e.g., in the scope of a verb like seek. These environments are called the *uneven* (Frege 1892), *opaque* (Quine 1960), or *intensional* (Montague 1974) *contexts*.

20.1.7 Second problem for extensional ontology

How should the difference in the meaning of different expressions for non-existing objects, such as square circle, unicorn, and Pegasus, be explained? This is necessary because of the second inference rule, the substitutivity of identicals.

For example, if we were to use the empty set as the referent of square circle, unicorn, and Pegasus in order to express that no real objects correspond to these terms, then the truth of Julia seeks a unicorn would imply the truth of Julia seeks Pegasus and Julia seeks the square circle because of the substitutivity of identicals.

20.1.8 Frege's solution to the second problem

Frege 1892 concluded from the non-equivalence of, e.g., Julia seeks a unicorn and Julia seeks a square circle that in addition to the real objects in the world there also exist natural language meanings, called *sense* (Sinn), which are independent of their referents *referents* (Bedeutung).

20.1.9 Ontological status of meaning (sense)

Frege attributed a similar form of existence to the meanings of natural language as to the numbers and their laws in mathematical realism. Mathematical realism proceeds on the assumption that the laws of mathematics exist even if no one knows about them; mathematicians *discover* laws which have extemporal validity. Frege supposed the meanings of natural language to exist in the same way, i.e., independently of whether there are speakers-hearers who have discovered them and use them more or less correctly.

20.2 Intension and extension

20.2.1 Examples of Carnap's *Intensions*

proposition: $\underbrace{I \times J}_{\textit{intension}} \rightarrow \underbrace{\{0,1\}}_{\textit{extension}}$

proper name: $\underbrace{I \times J}_{\textit{intension}} \rightarrow \underbrace{a \in A}_{\textit{extension}}$

1-pl. predicate: $\underbrace{I \times J}_{\textit{intension}} \rightarrow \underbrace{\{a1, a2, ..\} \subseteq A}_{\textit{extension}}$

20.2.2 Two approaches to meaning

Frege's theory: [+sense]

1. surface of expression
2. meaning (*sense*)
3. referent

Carnap's theory: [-sense]

1. surface of expression
2. function: index ⟶ 3. extension

20.3 Propositional attitudes

20.3.1 Two basic problems of logical semantics for natural language

- the Epimenides paradox and
- the problem of propositional attitudes.

20.3.2 Example of a propositional attitude

Suzanne believes that Cicero denounced Catiline.

20.3.3 Assumption of modal logic regarding proper names: rigid designators

According to the intuitions of modal logic, a proper name denotes the same individual in all possible worlds (rigid designator). For example, because Cicero and Tullius are names for one and the same person it holds necessarily (i.e, in all possible worlds) that Cicero = Tullius. Therefore, it follows necessarily from the truth of Cicero denounced Catiline that Tullius denounced Catiline.

20.3.4 Problem for propositional attitudes

Even though the referents of Cicero and Tullius are necessarily identical, this could be unknown to Suzanne. Therefore, a valid substitution *salva veritate* would require the truth of Suzanne believes that Cicero is Tullius.

Determining what an individual believes depends on what the individual chooses to report. Because it cannot be checked objectively whether this is true or not, individual 'belief-worlds' have justly been regarded as a prime example of what lies outside any scientific approach to truth.

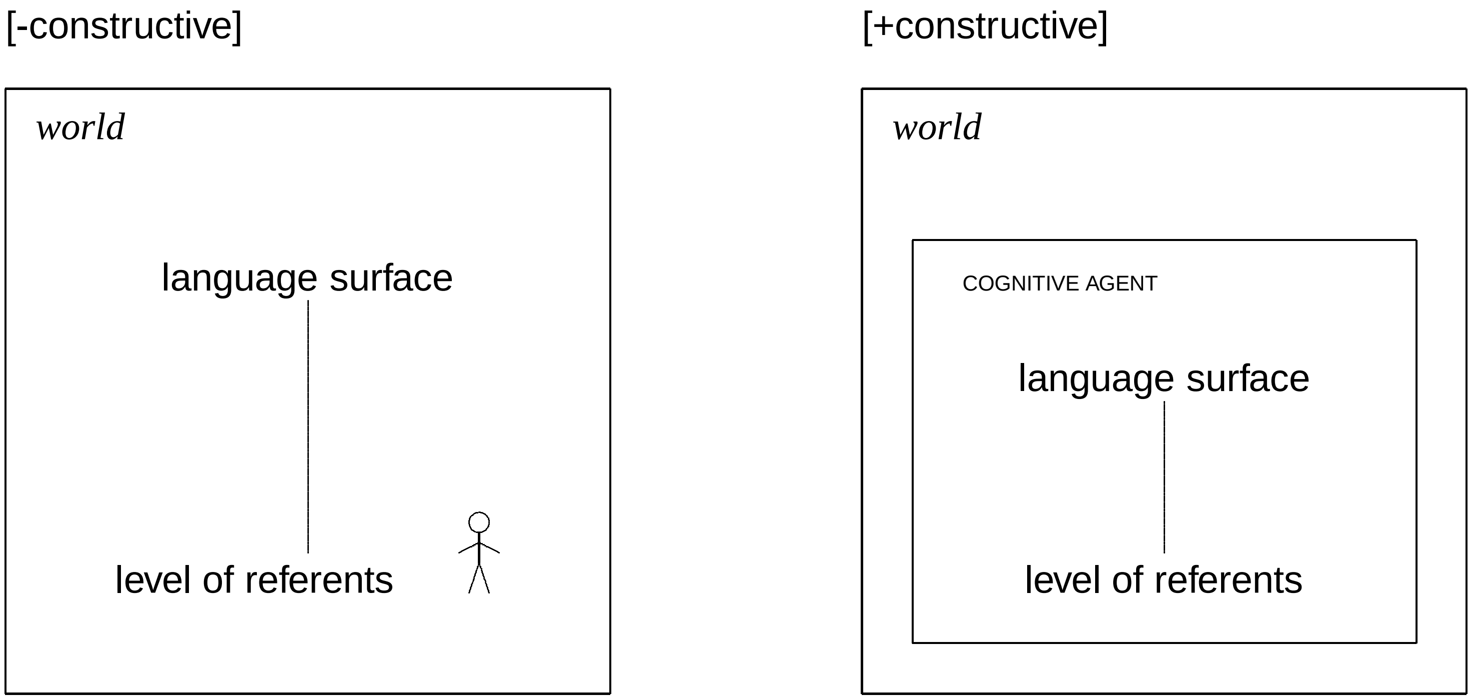
20.3.5 Fundamental question of logical semantics: Formulation I

Definition of truth (conditions) by means of meaning or
definition of meaning in terms of truth (conditions)?

20.3.6 Fundamental question of logical semantics: Formulation II

Is the speaker-hearer part of the model structure or
is the model structure part of the speaker-hearer?

20.3.7 Two ontological interpretations of model theory



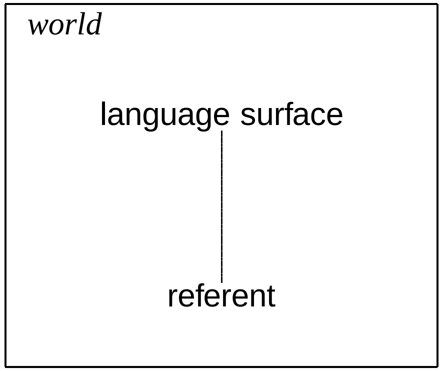
20.3.8 The most fundamental difference between [±constructive] ontologies

- Any system based on a [–constructive] ontology must have a metalanguage-based semantics.
- Any system based on a [+constructive] ontology must have a procedural semantics.

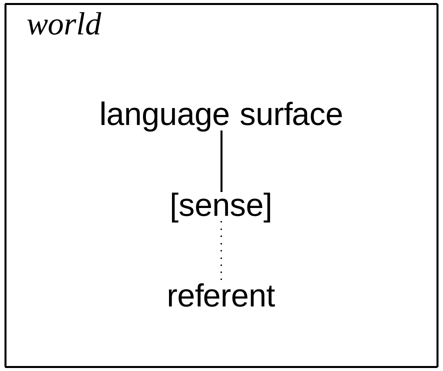
20.4 Four basic ontologies

20.4.1 Ontologies of semantic interpretation

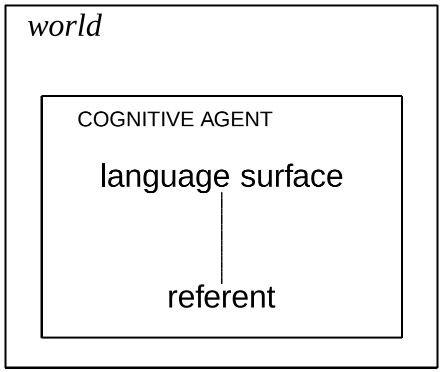
i [–sense, –constructive]
Russell, Carnap, Quine, Montague



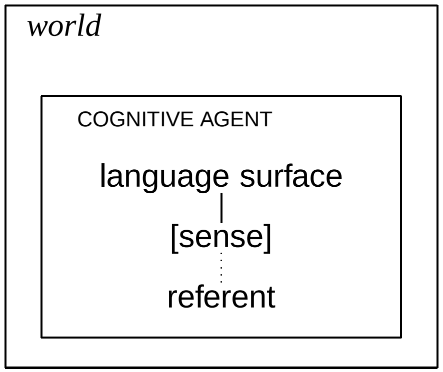
ii [+sense, –constructive]
Frege



iii [–sense, +constructive]
Newell & Simon, Winograd, Shank



iv [+sense, +constructive]
Anderson, CURIOUS, SLIM-machine



20.4.2 The [–sense,–constructive] ontology (i) of logical semantics

Concerned with a solid foundation for truth, logical semantics uses only referents which are considered to be real. Given its ontological foundations, logical semantics is in principle unsuitably for a complete analysis of natural language meaning.

20.4.3 The [+sense,–constructive] ontology (ii) of Frege

Attempt to analyze uneven (opaque, intensional) readings in natural language. As a theory of truth, any [-constructive], metalanguage-based semantics is necessarily incompatible with representing cognitive states.

20.4.4 The [–sense,+constructive] ontology (iii) of programming languages.

Designed to control electronic procedures via the commands of a programming language. A direct, fixed connection between language expressions and their referents prevents autonomous classification of new objects in principle. Therefore, [–sense, +constructive] systems are limited to closed worlds created by the programmer.

20.4.5 The [+sense,+constructive] ontology (iv) of the SLIM theory of language

The [+sense] property is the structural basis for matching of meaning_1 and the context of use, while the [+constructive] property allows the matching to occur inside the cognitive agent.

20.5 Sorites paradox and the treatment of vagueness

20.5.1 Sorites paradox or paradox of the heap

One grain of sand does not make a heap. Adding an additional grain still doesn't make a heap. If n grains do not form a heap, then adding another single grain will not make a heap either. However, if this process of adding a grain is continued long enough, there will eventually result a genuine heap.

20.5.2 Vagueness as motivation for non-bivalent logic

Sensitive students of language, especially psychologists and linguistic philosophers, have long been attuned to the fact that natural language concepts have vague boundaries and fuzzy edges and that, consequently, natural-language sentences will very often be neither true, nor false, nor nonsensical, but rather true to a certain extent and false to a certain extent, true in certain respects and false in other respects.

G. Lakoff 1972, p. 183

20.5.3 Future-contingent propositions as motivation for non-bivalent logic

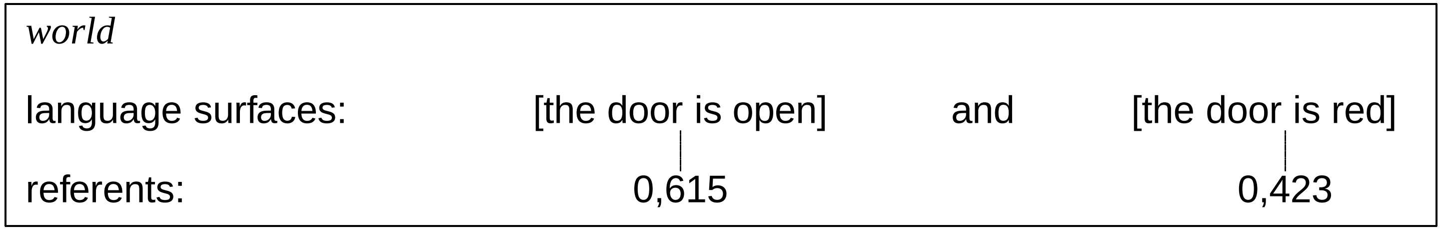
> Throughout the orthodox mainstream of the development of logic in the West, the prevailing view was that every proposition is either true or else false - although which of these is the case may well neither be *necessary* as regards the matter itself nor *determinable* as regards our knowledge of it. This thesis, now commonly called the "Law of Excluded Middle", was, however, already questioned in antiquity. In Chap. 9 of his treatise *On Interpretation (de interpretatione)*, Aristotle discussed the truth status of alternatives regarding "future-contingent" matters, whose occurrence – like that of the sea battle tomorrow – is not yet determinable by us and may indeed actually be undetermined.
>
> N. Rescher, 1969, p. 1

20.5.4 The basic problem of three-values logics and the many-valued logics

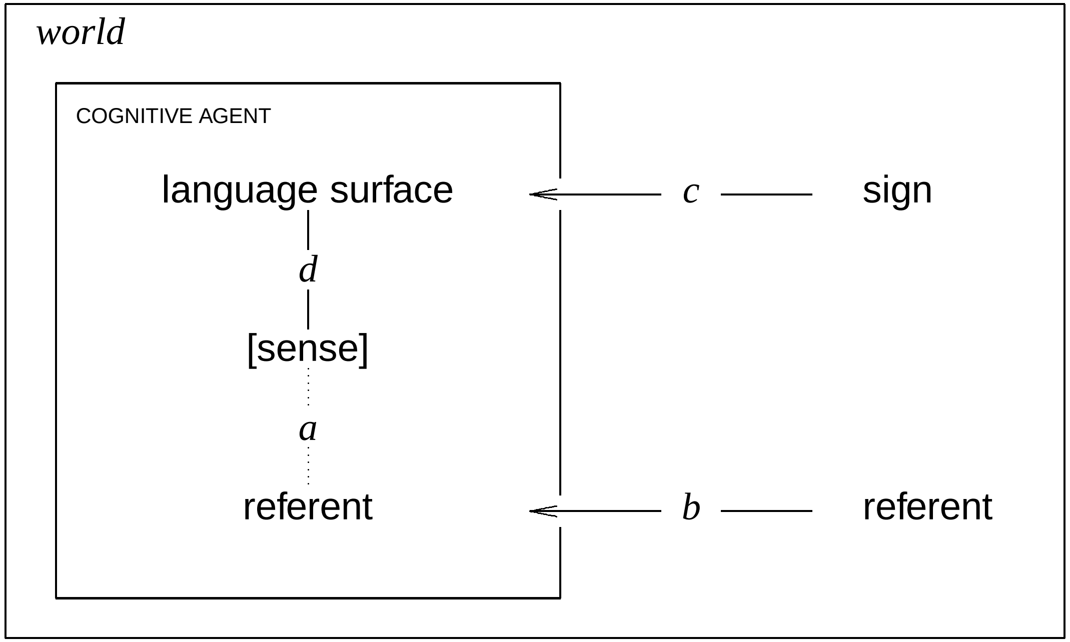
> Which truth-value should be assigned to complex propositions based on component propositions with non-bivalent truth-values?

For example: What should be the value of, e.g., A&B if A has the value 1 and B has the value #? Similarly in a many-valued system: if the component proposition A has the truth-value 0.615 and B has the value 0.423, what value should be assigned to A&B?

20.5.5 Vagueness in [-sense,-constructive] semantics

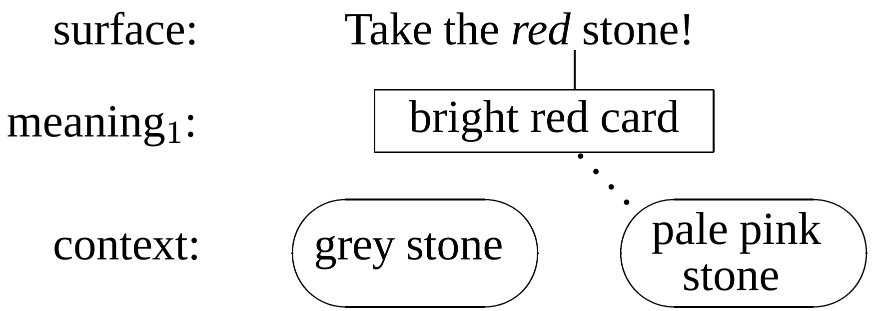


20.5.6 Vagueness in [+sense,+constructive] semantics

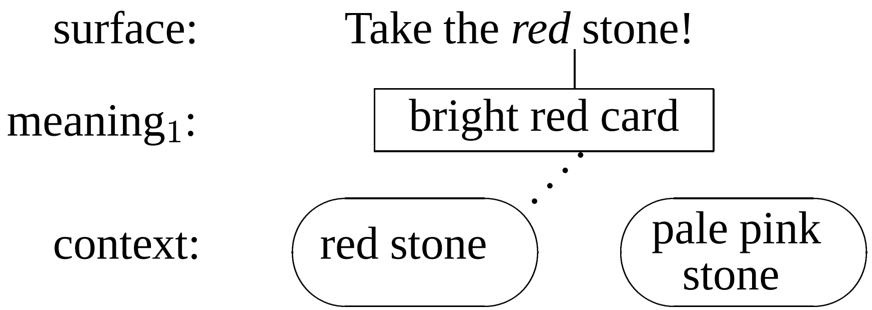


20.5.7 Why vagueness is not a property of language meaning

The hearer is faced with a context consisting of a grey stone and a pale pink stone. Responding to the utterance Take the red stone, the cooperative hearer will pick the pale pink stone. For simplicity, the meaning$_1$ of red is represented by a bright red card.



If the grey stone is replaced by a dark red one, the pale pink stone ceases to be the best match. Responding to Take the red stone, the cooperative hearer will not pick the pale pink stone, but the red one.



It is not the meaning$_1$ of red which changed, but the context.

21. Absolute and contingent propositions

21.1 Absolute and contingent truth

21.1.1 Notion of proposition in logic

Specialized use, representing sentences which do not require knowledge of the utterance situation for semantic interpretation. This use is problematic because it constitutes a hybrid between an *utterance* and an *expression*.

21.1.2 Absolute propositions

Express scientific or mathematical contents. These are special in that they make the interpretation largely independent from the usual role of the speaker. For example, in
In a right-angled triangle, it holds for the hypotenuse A and the cathetes B and C that $A^2 = B^2 + C^2$.
the circumstances of the utterance have no influence on interpretation and truth value.

21.1.3 Logical truth for absolute propositions

Logical truth is represented by the metalanguage words false and true referring to the set-theoretic objects $\emptyset$ und $\{\emptyset\}$, respectively. These serve as model-theoretic fix points into which the denotations of propositions are mapped by the metalanguage rules of interpretation.

21.1.4 Contingent propositions

Express everyday contents such as Your dog is doing well.
Can only be interpreted – and thereby evaluated with respect to their truth value – if relevant circumstances of utterance situation (STAR point) are known.

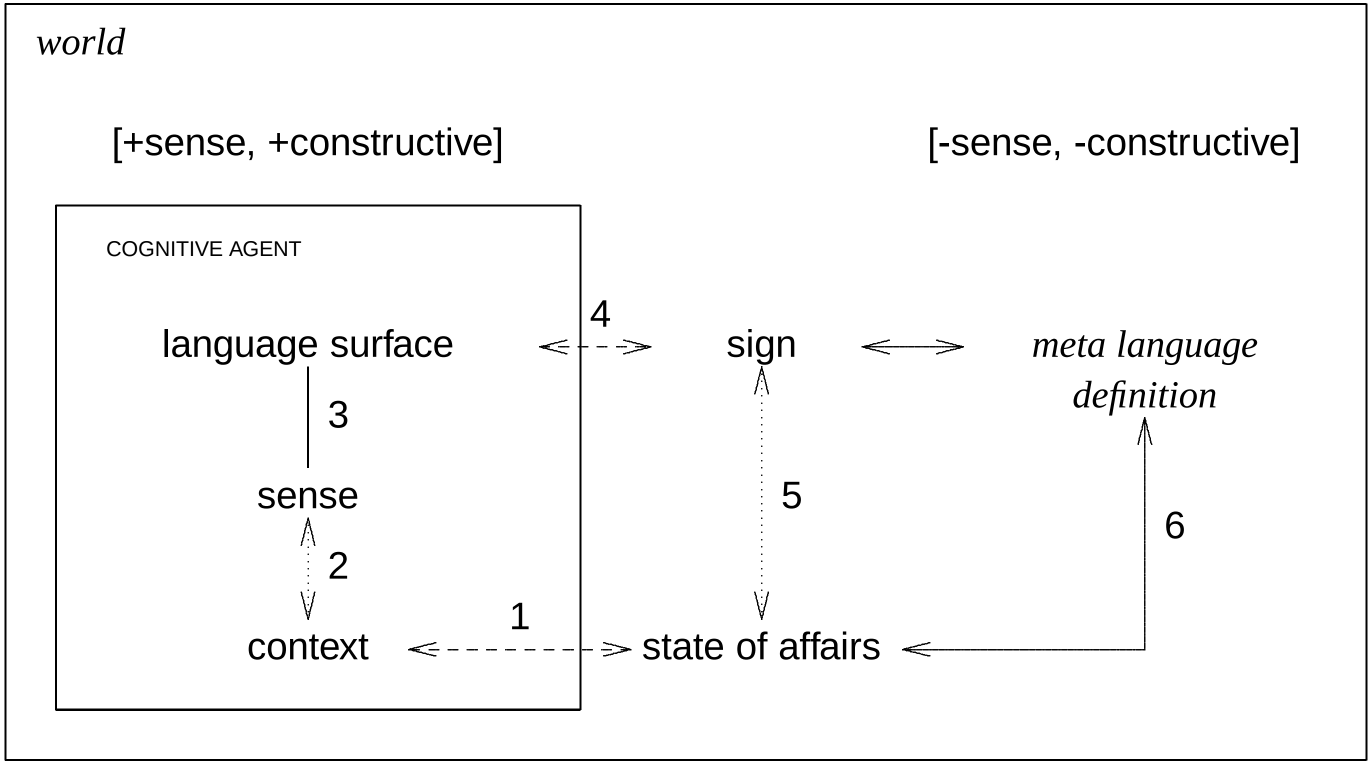
21.1.5 Natural truth for contingent proposition

Represented by the truth values true^c and false^c. A contingent proposition such as
The Persians have lost the battle
is true^c, if the speaker is an eye witness who is able to correctly judge and communicate the facts, or if there exists a properly functioning chain of communication between the speaker and a reliable eye witness.

21.1.6 Procedural definition of the natural truth values true^c and false^c

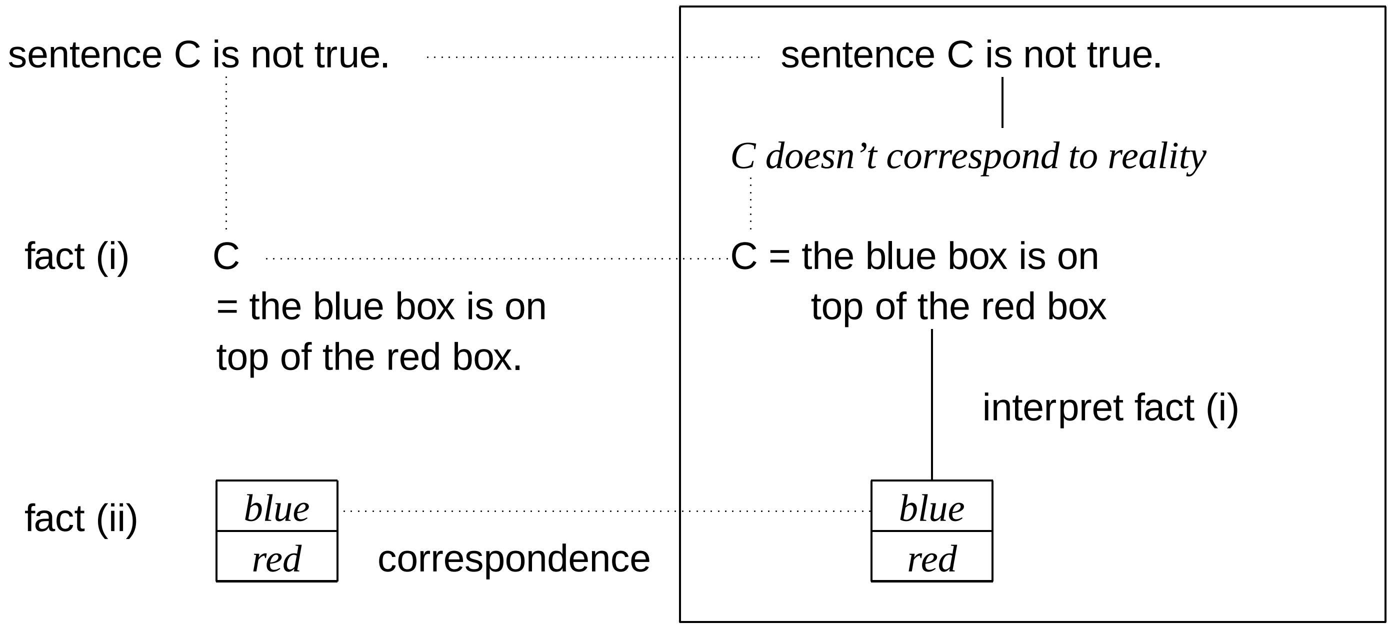
A proposition – or rather a statement – uttered by, e.g., a robot is evaluated as true^c, if all procedures contributing to communication work correctly. Otherwise it is evaluated as false^c.

21.1.7 Comparing natural and logical truth

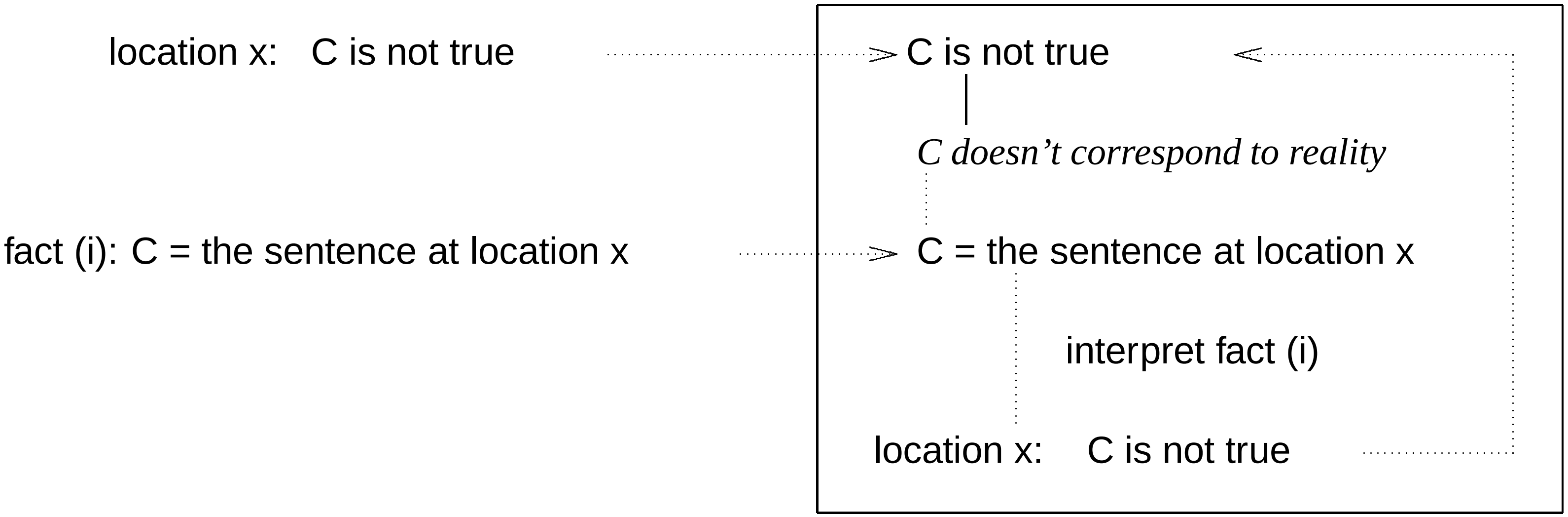


21.2 Epimenides in a [+sense,+constructive] system

21.2.1 Benign case of a language-based abbreviation



21.2.2 A [+constructive,+sense] reanalysis of the Epimenides paradox



21.2.3 How the [+constructive,+sense] reanalysis disarms the Epimenides paradox

- the words true^c and false^c may be part of the object language without causing a logical contradiction in its semantics, and
- the recursion caused by the Epimenides paradox can be recognized in the pragmatics and taken care of without adversely affecting the communicative functioning of the system.

21.2.4 Basis of the reanalysis of the Epimenides paradox

The distinction between (i) the logical truth values 1 and 0 from the T-condition and (ii) the natural truth values true^c and false^c from the object language replaces Tarski's logical contradiction

$a.$ C is 1 if and only if C is not 1

by the contingent statement

$b.$ C is 1 if and only if C is not true^c.

21.2.5 Why the reanalysis is not open to logical semantics

The procedural notion of natural truth – essential for avoiding Tarski's contradiction – can be neither motivated nor implemented outside a [+constructive,+sense] ontology.

21.3 Frege's principle as homomorphism

21.3.1 The communicative function of natural syntax

is the composition of semantic representations by means of composing the associated surfaces. Montague formalized this structural correlation between syntax and semantics mathematically as a *homomorphism*.

21.3.2 Intuitive notion of a homomorphism

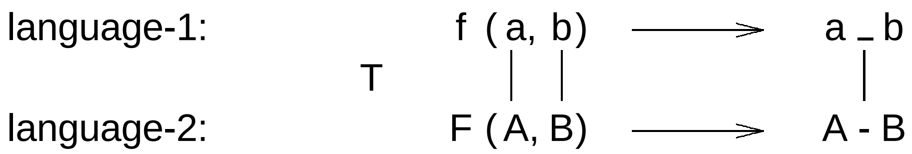
A structural object so is homomorphic to another structural object SO, if for each basic element of so there is a (not necessarily basic) counterpart in SO, and for each relation between elements in so there is a corresponding relation between corresponding elements in SO.

21.3.3 Homomorphism as a relation between two (uninterpreted) languages

Language-2 is homomorphic to language-1 if there is a function T which

- assigns to each word of category a in language-1 a corresponding expression of category A in language-2, and
- assigns to each n-place composition f in language-1 a corresponding n-place composition F in language-2, such that
- T(f(a,b)) = F((T(a))(T(b)))

21.3.4 Schematic representation of Montague's homomorphism



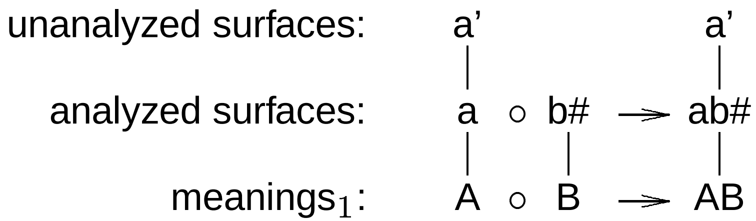
21.3.5 Syntactic composition with homomorphic semantics

21.3.6 Why the homomorphism condition by itself is not sufficient as a formalization of Frege's principle

Frege's principle is defined for *analyzed* surfaces, whereas natural language communication is based on *unanalyzed* surfaces. The problem is that the transition from unanalyzed to analyzed surfaces (interpretation) and vice versa (production) has been misused to enrich the levels of the analyzed surface and/or the meaning$_1$ by means of zero elements or identity mappings.

21.3.7 Use of zero element (illegal)

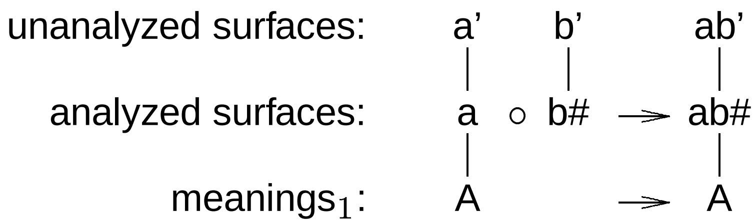
1. Smuggling in during interpretation (↓) – Filtering out during production (↑)



Postulated whenever the unanalyzed surface does not contain what the grammar theory would like to find.

Peter drank DET# wine YOU# help me!

2. Filtering out during interpretation (↓) – Smuggling in during production (↑)



Postulated whenever the surface contains something which the grammar theory would not like to find.

Peter believes THAT# Jim is tired.
mixed: DET# wine WAS# ordered BY# Peter
mixed: Peter promised Jim TO# Peter# sleep
mixed: Peter persuaded Jim TO# Jim# sleep.

21.3.8 Use of identity mapping (illegal)

1. Filtering out during production (↑) – Smuggling in during interpretation (↓)

unanalyzed surfaces:	a’		b’		a’
analyzed surfaces:	a	∘	b	⟶	ab#
meanings_1:	A	∘	B	⟶	AB

2. Smuggling in during production (↑) – Filtering out during interpretation (↓)

unanalyzed surfaces:	a’		b’		ab’
analyzed surfaces:	a	∘	b	⟶	ab#
meanings_1:	A	∘	B	⟶	A

21.3.9 Surface compositionality II (SC-II principle)

A semantically interpreted grammar is surface compositional if and only if

- the syntax is restricted to the composition of concrete word forms (i.e. no zero elements and no identity mappings),
- the semantics is homomorphic to the syntax, and
- objects and operations on the level of semantics which correspond to the syntax in accordance with the homomorphism condition may not be realized by zero elements or identity mappings.

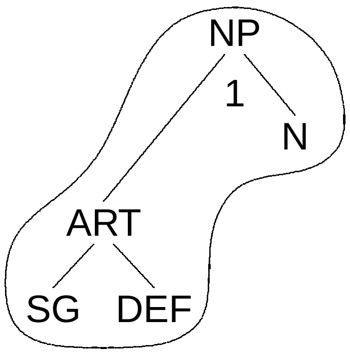
21.4 Time-linear syntax with homomorphic semantics

21.4.1 Time-linear build-up of semantic hierarchies

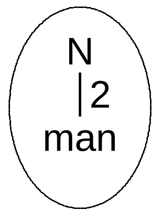
- Step 1: *Translation of word forms into component hierarchies*
  Each word form is mapped into a semantic component hierarchy (tree). The structure of the tree is determined by the syntactic category of the word form.
- Step 2: *Left-associative combination of component hierarchies*
  For each combination of the left-associative syntax there is defined a corresponding combination of component hierarchies on the level of the semantics.

21.4.2 Derivation of component hierarchies from word forms

the
(SN′ SNP) →



man
(SN) →



21.4.3 Time-linear composition with homomorphic semantics

21.4.4 Why 21.4.3 is not a constituent structure

A constituent structure analysis would proceed on the assumption that gave is semantically closer to the woman and the book than to the man.

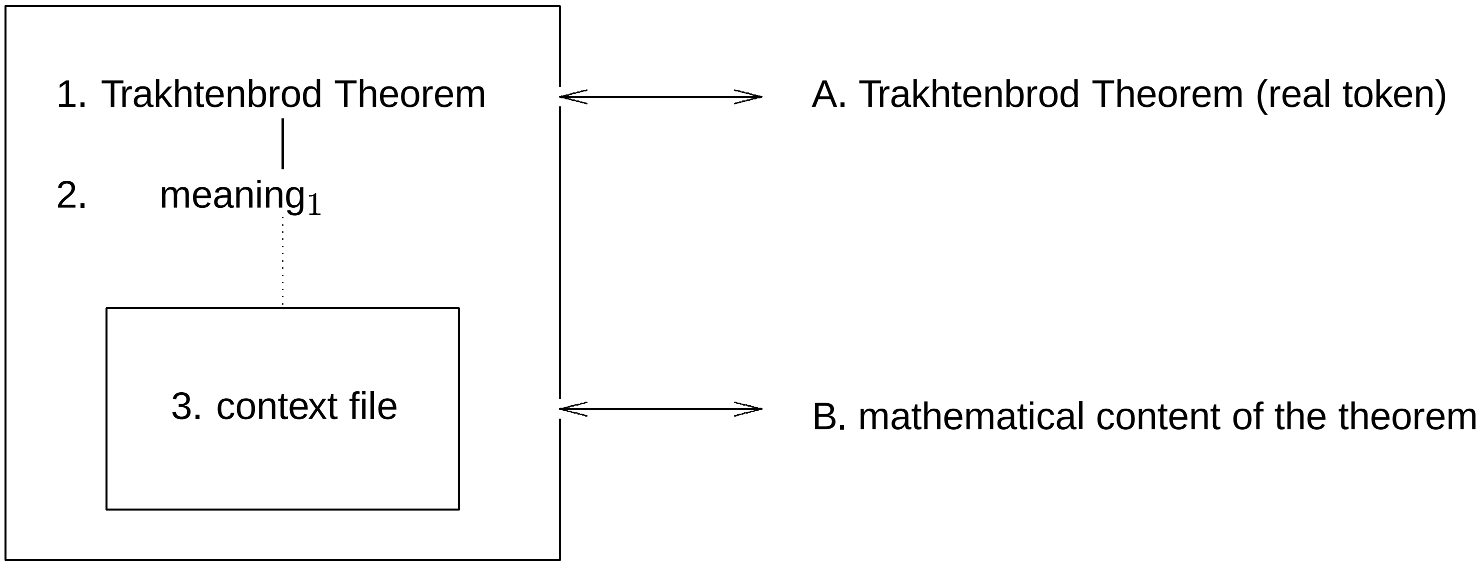
21.5 Complexity of natural language semantics

21.5.1 Low complexity of syntactic system may be pushed sky high by semantic interpretation

(a) π | 3.14159265...

(b) 1:3 | 1' :' 3' = 0.333...

21.5.2 Interpretation of 'Trakhtenbrod Theorem' within SLIM theory



21.5.3 CoNSem hypothesis (Complexity of Natural language Semantics)

The interpretation of a natural language syntax within the C-LAGs is empirically adequate only if there is a finite constant k such that

- it holds for each elementary word form in the syntax that the associated semantic representation consists of at most k elements, and
- it holds for each elementary composition in the syntax that the associated semantic composition increases the number of elements introduced by the two semantic input expressions by maximally k elements in the output.

This means that the semantic interpretation of syntactically analyzed input of length n consists of maximally $(2n - 1) \cdot k$ elements.

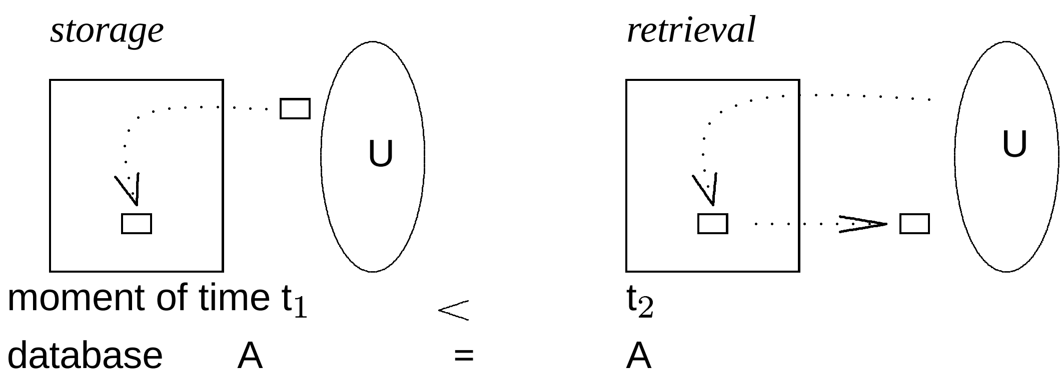
21.5.4 Illustration of CoNSem hypothesis with k = 5

a	·	b	⟶	ab	·	c	⟶	abc
|		|		|		|		|
[1–5]	&	[6–10]		[1–15]	&	[16–20]		[1–25]

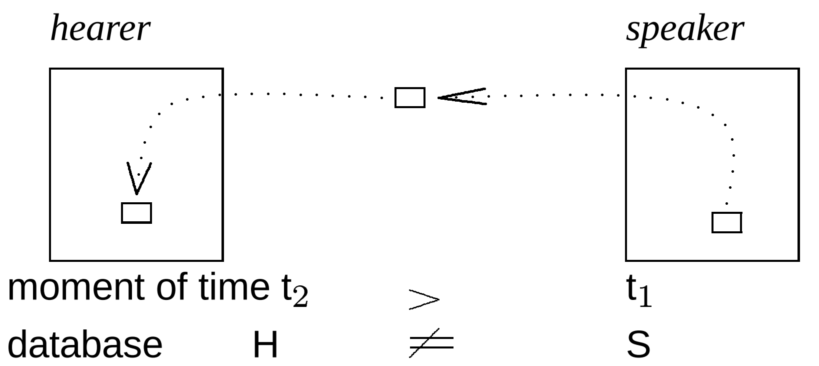
22. Database semantics

22.1 Database metaphor of natural communication

22.1.1 Interaction with a conventional database



22.1.2 Interaction between speaker and hearer



22.1.3 DB interaction and NL communication

- ENTITIES INVOLVED

  Database interaction: takes place between two different entities, the user and the database.

  NL communication: takes place between two similar and equal cognitive agents, the speaker and the hearer.

- ORIGIN OF CONTROL

  Database interaction: operations of input and output are controlled by the user.

  NL communication: there is no user. Instead, the cognitive agents control each other by alternating in the speaker- and the hearer-mode (*turn taking*).

- METHOD OF CONTROL

  Database interaction: user controls the operations of the database with a programming language the commands of which are executed as electronic procedures.

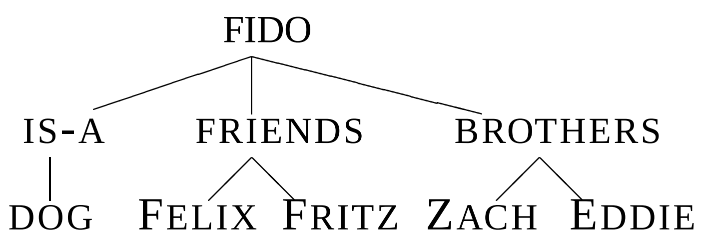
  NL communication: speaker controls language production as an autonomous agent, coding the parameters of the utterance situation into the output expressions. The hearer's interpretation is controlled by the incoming language expression.

- TEMPORAL ORDER

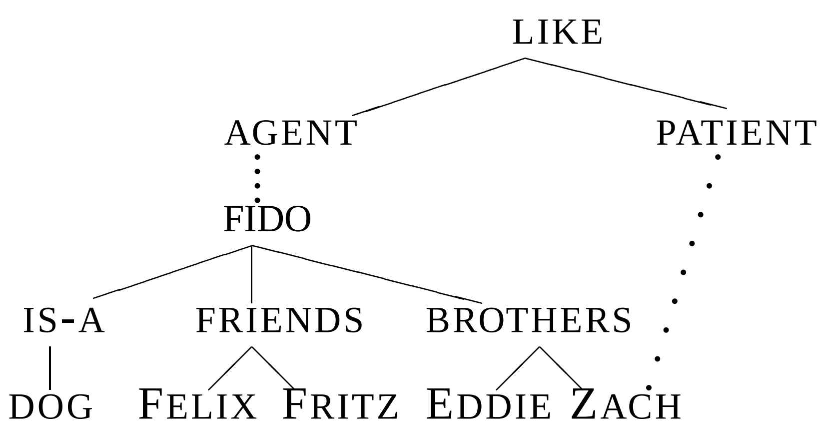
  Database interaction: output (database as 'speaker') occurs necessarily *after* the input (database as 'hearer').

  NL communication: production (output procedure of the speaker) occurs necessarily *before* interpretation (input procedure of the hearer).

22.1.4 Sketch of a simple subcontext



22.1.5 Pragmatic interpretation of 22.1.1



22.2 Descriptive aporia and embarrassment of riches

22.2.1 Model-theoretic definition of a context

Let $\mathcal{MS}$ be a model structure (A,I,J,$\leq$,F), where A, I, J are sets, $\leq$ is a simple ordering on J, and F is a denotation function.

A, I, J, and F have the following definition:

$A = \{a_0, a_1, a_2, a_3, a_4\}$, $I = \{i_1\}$, $J = \{j_1\}$

F(fido')(i_1, j_1) = a_0

F(felix')(i_1, j_1) = a_1

F(fritz')(i_1, j_1) = a_2

F(zach')(i_1, j_1) = a_3

F(eddie')(i_1, j_1) = a_4

F(dog')(i_1, j_1) = $\{a_0\}$

F(fido-friends')(i_1, j_1) = $\{a_1, a_2\}$

F(fido-brothers')(i_1, j_1) = $\{a_3, a_4\}$

22.2.2 Extending the hearer context to the meaning of a new sentence such as Fido likes Zach

Requires automatic addition of 'F(like)(i_1, j_1) = $\{(a_0,a_3)\}$' to 22.2.1

22.2.3 Creating a *frame*

```
(make-frame
  fido
    (is-a (value dog))
    (friends (value felix fritz))
    (brothers (value zach eddie))
)
```

22.2.4 Definition of 22.4.2 as a *frame*

```
(fido
  (is-a (value dog))
  (friends (value felix fritz))
  (brothers (value zach eddie))
)
```

22.2.5 Retrieving information

```
(get-values ’FIDO ’FRIENDS)
(FELIX FRITZ)
```

22.2.6 Extending the hearer context to Fido likes Zach

Requires deriving

```
(fido
 (like (value Zach)
)
```

and automatically adding the part

```
(like (value Zach)
```

as a new slot into 22.2.4.

22.3 Propositions as sets of coindexed proplets

22.3.1 Proposition 3.4.2 as a set of proplets (preliminary format)

Type:
- M-concept: field
- role: argument

Token:
- I-concept$_{loc}$: x1
- functor: contain
- prn: 23
- id: 7

Type:
- M-concept: contain
- role: functor

Token:
- I-concept$_{loc}$: x2
- argument 1: field
- argument 2: triangle
- prn: 23
- epr: 23 and 24

Type:
- M-concept: triangle
- role: argument

Token:
- I-concept$_{loc}$: x3
- functor: contain
- prn: 23
- id: 8

Type:
- M-concept: field
- role: argument

Token:
- I-concept$_{loc}$: x4
- functor: contain
- prn: 24
- id: 7

Type:
- M-concept: contain
- role: functor

Token:
- I-concept$_{loc}$: x5
- argument 1: field
- argument 2: square
- prn: 24
- epr: 23 and 24

Type:
- M-concept: square
- role: argument

Token:
- I-concept$_{loc}$: x6
- functor: contain
- prn: 24
- id: 9

22.4 Proplets in a classic database

22.4.1 Types of databases

classic: record based
non-classic: based on the principle of slot and filler

22.4.2 Types of classic databases

Relational database, hierarchical database, network database

22.4.3 Relations between proplet features

type $\leftrightarrow$ token
token $\leftrightarrow$ prn
prn $\leftrightarrow$ epr
token $\leftrightarrow$ id
functor $\leftrightarrow$ argument
modifier $\leftrightarrow$ modified

22.4.4 Propositions 3.4.2 as a word bank

TYPES	SIMPLIFIED PROPLETS	
[M-concept: contain role: functor]	[I-concept$_{loc}$: x2 argument 1:field argument 2:triangle prn: 23 epr: 23 and 24]	[I-concept$_{loc}$: x5 argument 1:field argument 2:square prn: 24 epr: 23 and 24]
[M-concept: field role: argument]	[I-concept$_{loc}$: x1 functor: contain prn: 23 id: 7]	[I-concept$_{loc}$: x4 functor: contain prn: 24 id:7]
[M-concept: square role: argument]	[I-concept$_{loc}$: x6 functor: contain prn: 24 id: 9]	
[M-concept: triangle role: argument]	[I-concept$_{loc}$: x3 functor: contain prn: 23 id: 8]	

22.4.5 Example of a network database

owner records	*member records*			
Comp.Sci.	Riedle	Schmidt	Stoll	...
Mathematics	Müller	Barth	Jacobs	...
Physics	Weber	Meier	Miele	...

22.4.6 Types of continuations

intrapropositional:
from argument to functor, functor to argument, from modifier to modified and vice versa

extrapropositional:
epr from verb to verb, id from noun to noun

22.5 Example of a word bank

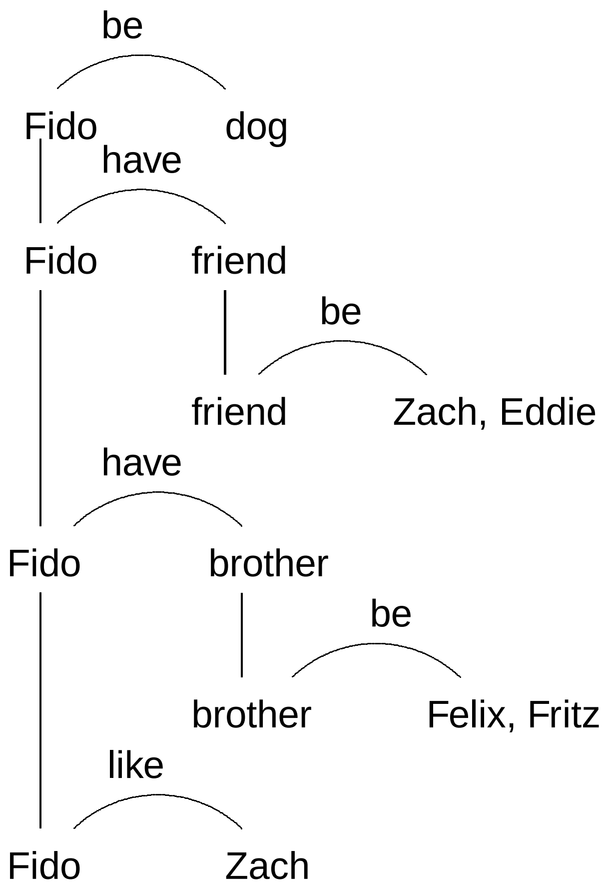
22.5.1 Propositional presentation of subcontext 22.1.4

1. Fido is a dog.
2. Fido has friends.
3. The friends are Zach and Eddie.
4. Fido has brothers.
5. The brothers are Felix and Fritz.

---

6. Fido likes Zach.

22.5.2 Graphical presentation of the propositions in 22.5.1



22.5.3 Subcontext 22.1.1 as a word bank

TYPES	PROPLETS		
M-concept: be role: functor	I-concept$_{loc}$: x1 arg1: Fido arg2: dog prn: 1 epr: 1 and 2	I-concept$_{loc}$: x2 arg1: friend arg2: Zach, Eddie prn: 3 epr: 2 and 3 3 and 4	I-concept$_{loc}$: x3 arg1: brother arg2: Felix, Fritz prn: 5 epr: 4 and 5 5 and 6
M-concept: brother role: argument	I-concept$_{loc}$: x4 functor: have prn: 4 id:	I-concept$_{loc}$: x5 functor: be prn: 5 id:	
M-concept: dog role: argument	I-concept$_{loc}$: x6 functor: be prn: 4 id:		
M-concept: Eddie role: argument	I-concept$_{loc}$: x7 functor: be prn: 3 id: 3		

[M-concept: Felix role: argument]	[I-concept$_{loc}$: x8 functor: be prn: 5 id: 4]				
[M-concept: Fritz role: argument]	[I-concept$_{loc}$: x9 functor: be prn: 5 id: 5]				
[M-concept: Fido role: argument]	[I-con.$_{loc}$: x10 functor: be prn: 1 id: 1]	[I-con.$_{loc}$: x11 functor: have prn: 2 id: 1]	[I-con.$_{loc}$: x12 functor: have prn: 4 id: 1]	[I-con.$_{loc}$: x13 functor: like prn: 6 id: 1]	&
[M-concept: friend role: argument]	[I-concept$_{loc}$: x14 functor: have prn: 2 id:]	[I-concept$_{loc}$: x15 functor: be prn: 3 id:]			
[M-concept: have role: functor]	[I-concept$_{loc}$: x16 arg1: Fido arg2: friend prn: 2 epr: 1 and 2 2 and 3]	[I-concept$_{loc}$: x17 arg1: Fido arg2: brother prn: 4 epr: 3 and 4 4 and 5]			

$$\begin{bmatrix}\text{M-concept: like}\\ \text{role: functor}\end{bmatrix} \qquad \begin{bmatrix}\text{I-concept}_{loc}\text{: x18}\\ \text{arg1: Fido}\\ \text{arg2: Zach}\\ \text{prn: 6}\\ \text{epr: 5 and 6}\end{bmatrix} \&$$

$$\begin{bmatrix}\text{M-concept: Zach}\\ \text{role: argument}\end{bmatrix} \qquad \begin{bmatrix}\text{I-concept}_{loc}\text{: x19}\\ \text{functor: be}\\ \text{prn: 3}\\ \text{id: 2}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: x20}\\ \text{functor: like}\\ \text{prn: 6}\\ \text{id: 2}\end{bmatrix} \&$$

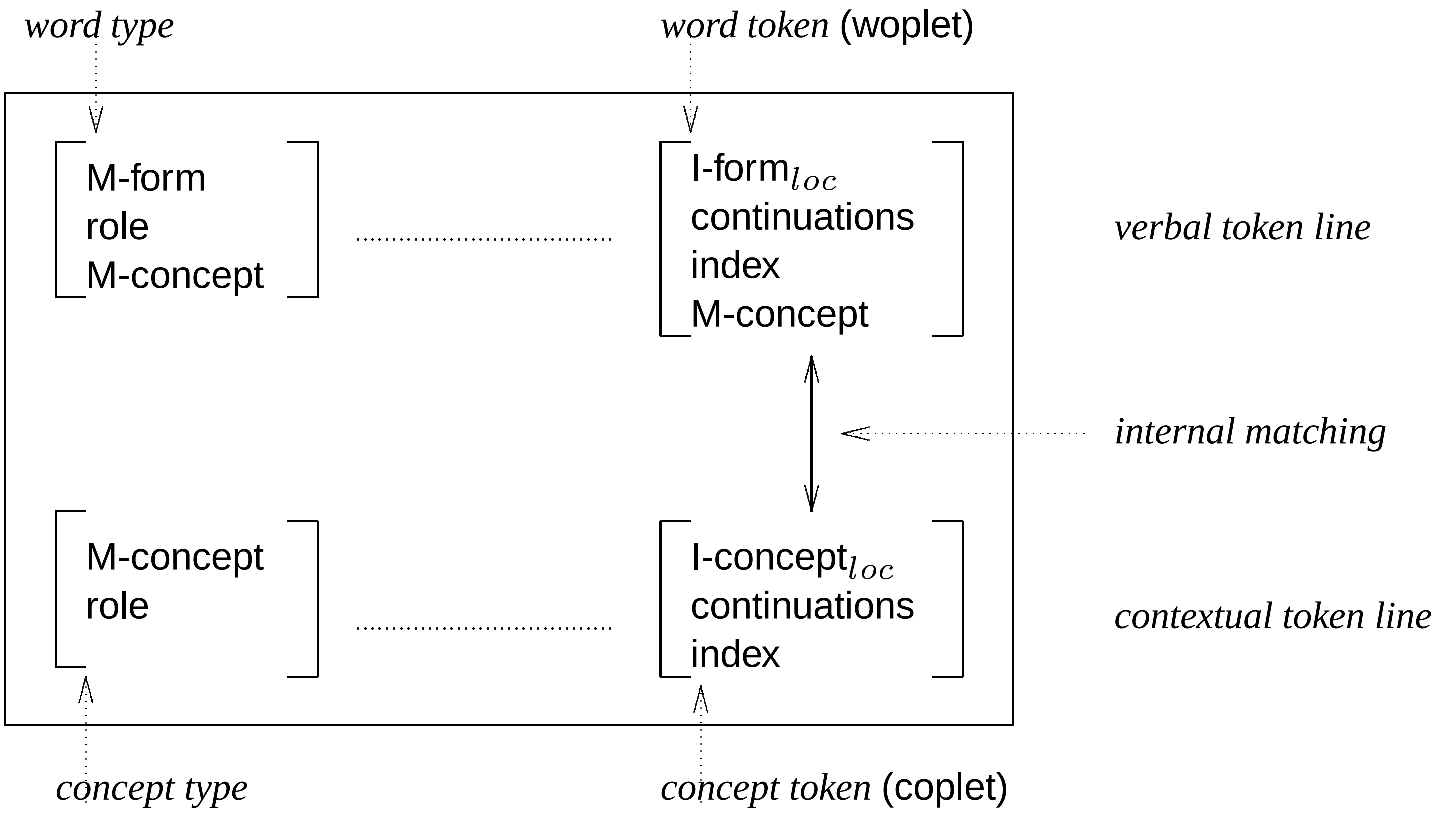
22.5.4 Semantic representation of proposition 6

TYPES	PROPLETS
[M-concept: Fido role: argument]	[I-concept$_{loc}$: x13 functor: like prn: 6 id: ?]
[M-concept: like role: functor]	[I-concept$_{loc}$: x18 arg1: Fido arg2: Zach prn: 6 epr: ?]
[M-concept: Zach role: argument]	[I-concept$_{loc}$: x20 functor: like prn: 6 id: ?]

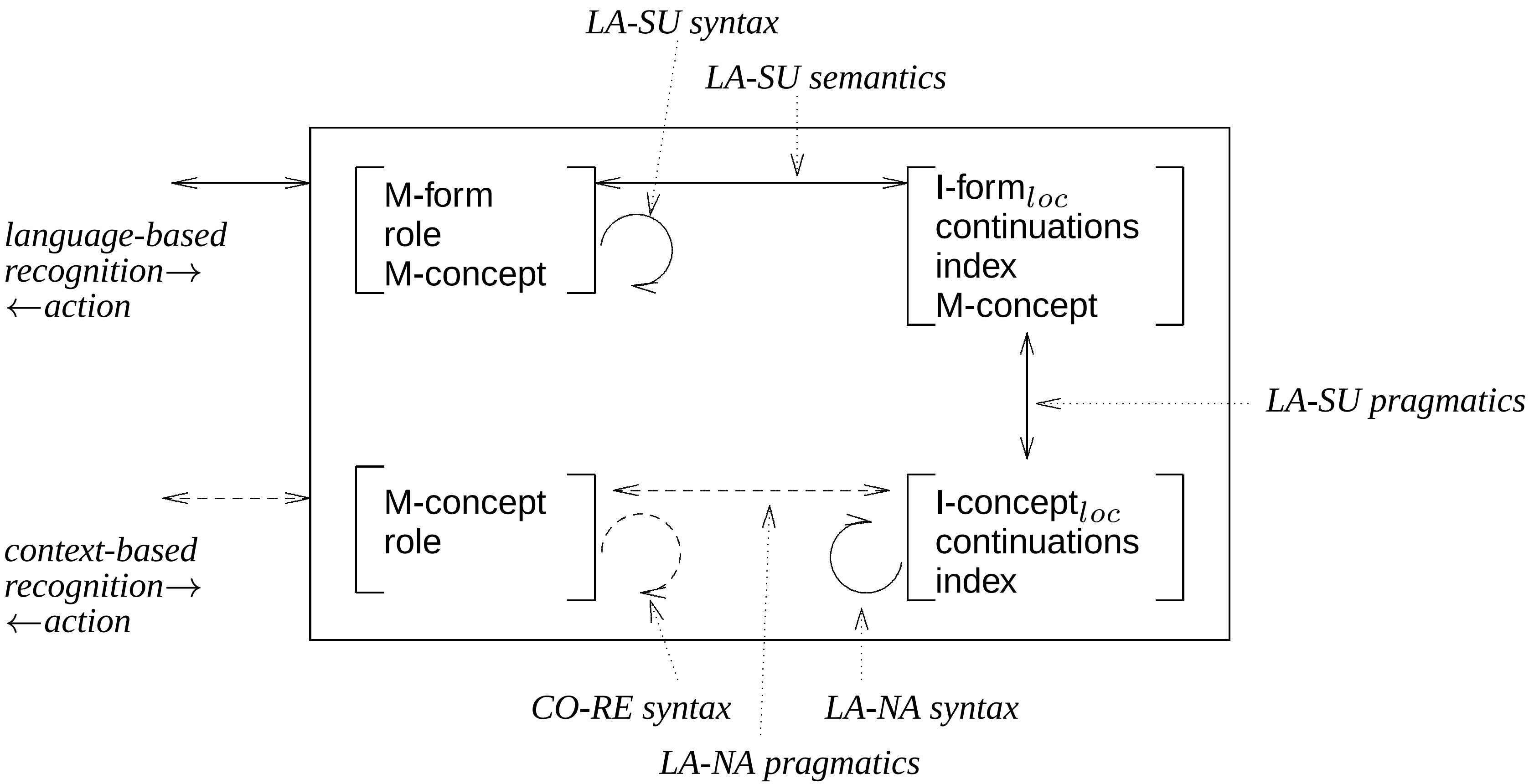
23. SLIM machine in the hearer mode

23.1 External connections and motor algorithms

23.1.1 Static structures of the SLIM machine

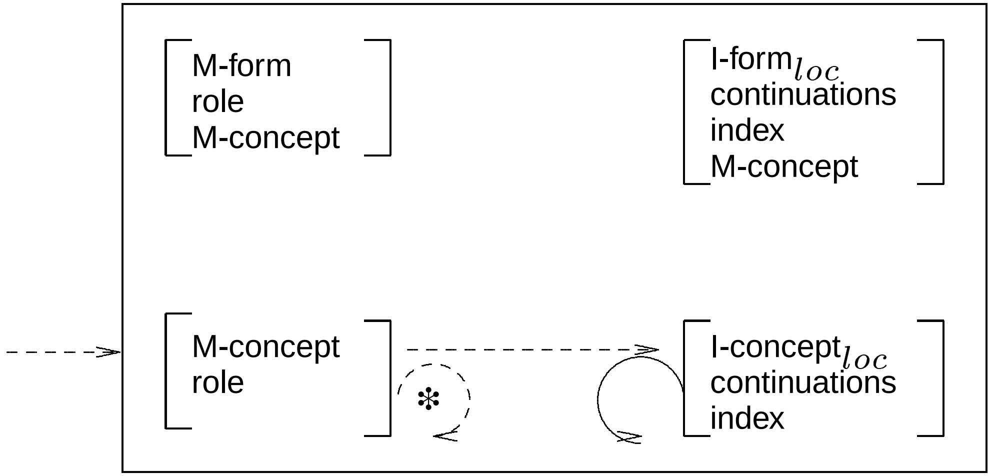


23.1.2 External connections and motor algorithms of the SLIM machine

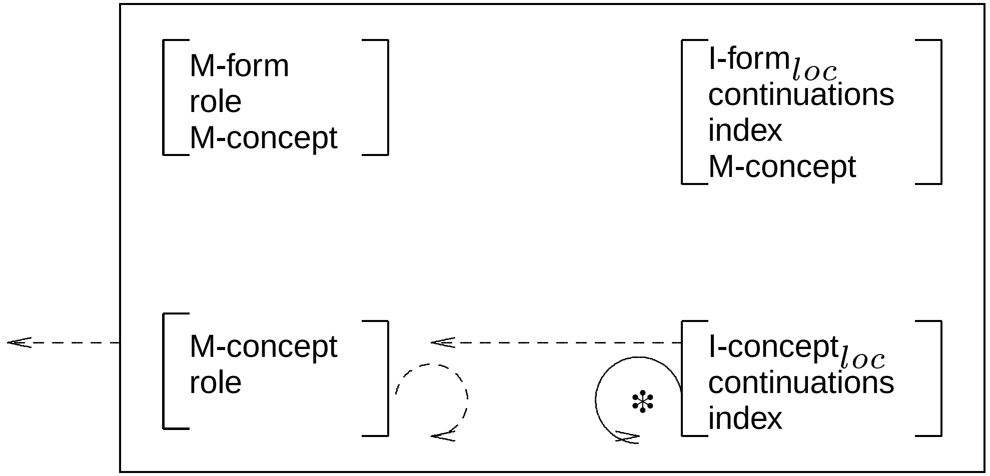


23.2 Ten SLIM states of cognition

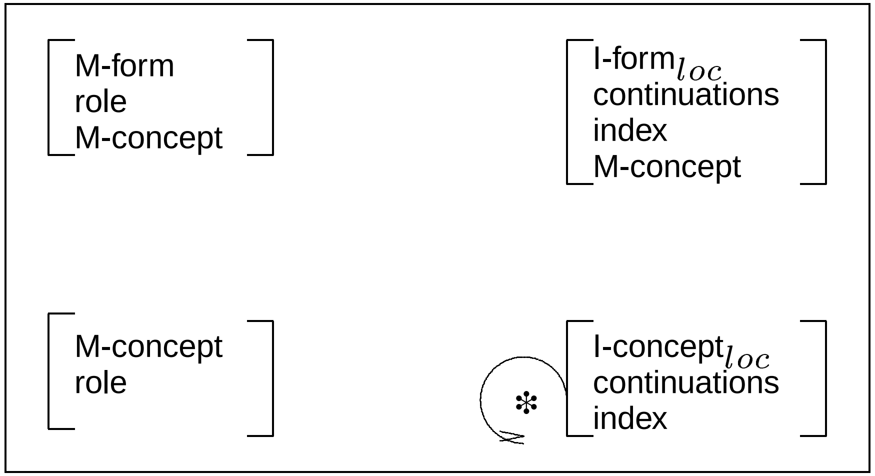
23.2.1 SLIM 1: Recognition (contextual)



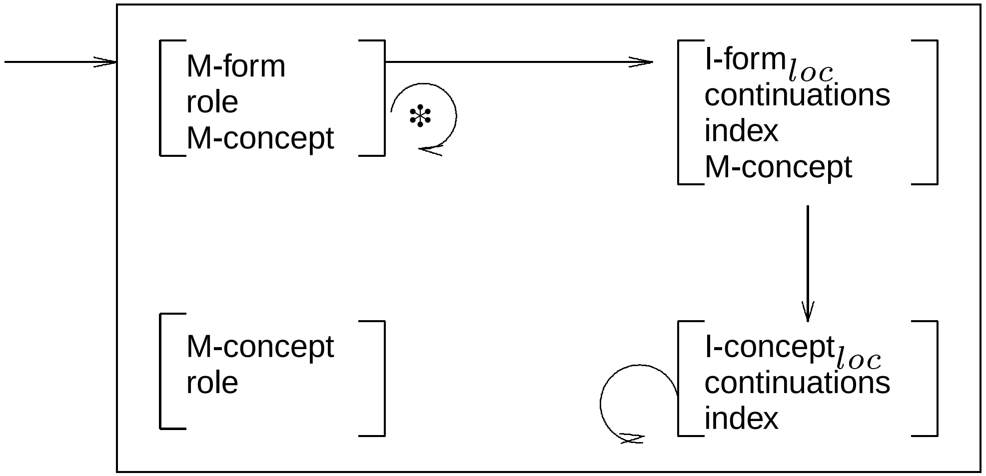
23.2.2 SLIM 2: Action (contextual)



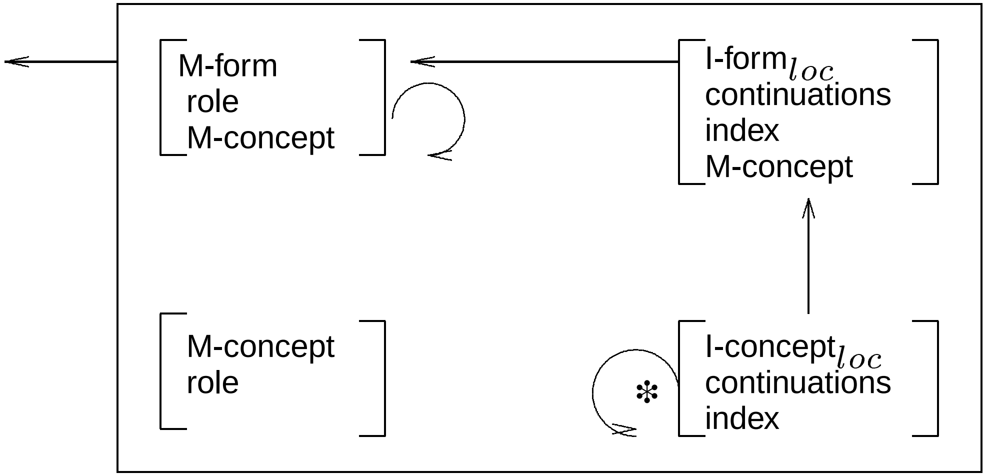
23.2.3 SLIM 3: Inference (contextual)



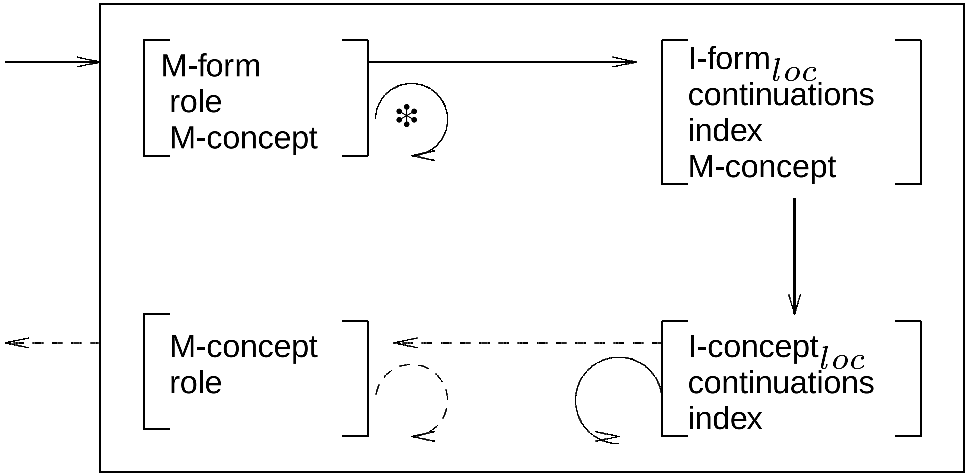
23.2.4 SLIM 4: Interpretation of language (mediated reference)



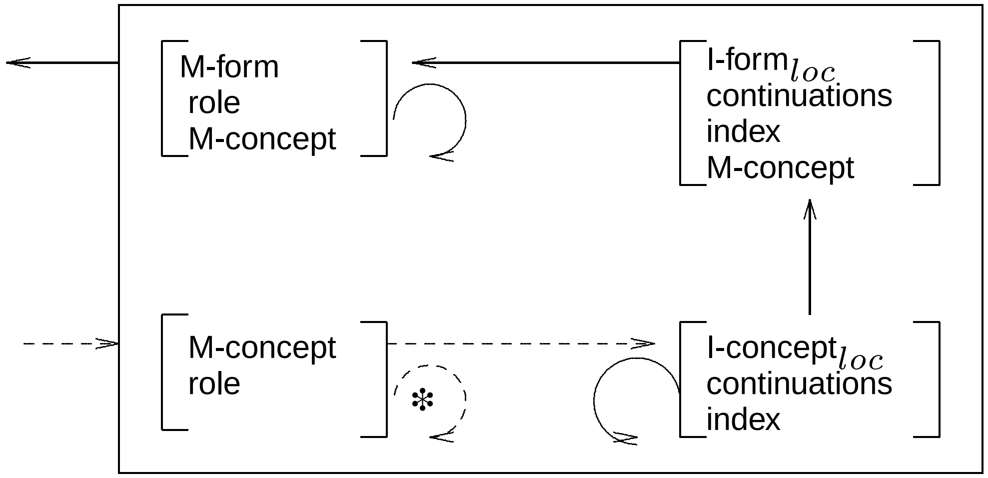
23.2.5 SLIM 5: Production of language (mediated reference)



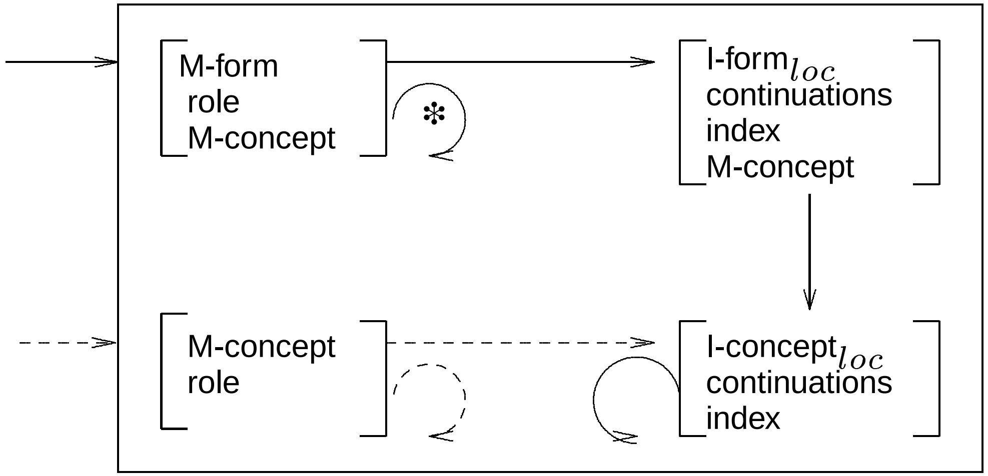
23.2.6 SLIM 6: Language-controlled action (immediate reference)



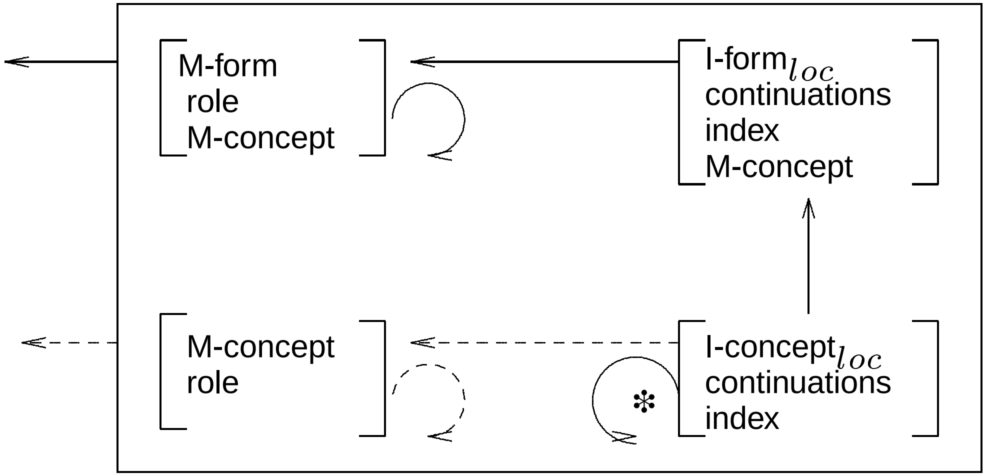
23.2.7 SLIM 7: Commented recognition (immediate reference)



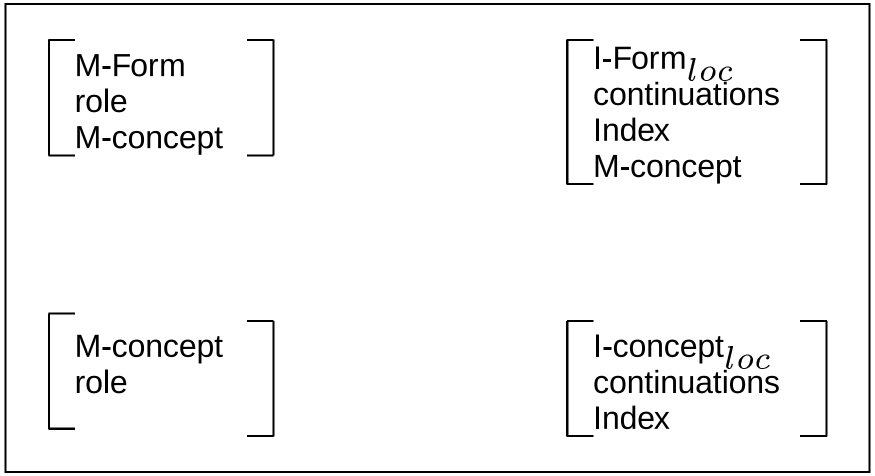
23.2.8 SLIM 8: Language-controlled recognition (immediate reference)



23.2.9 SLIM 9: Commented action (immediate reference)



23.2.10 SLIM 10: Cognitive stillstand



23.2.11 Notions grounded in the ten SLIM states

Context-based cognition is represented by SLIM 1 to SLIM 3,

Language-based cognition is represented by SLIM 4 and SLIM 5,

Simultaneous context- and language-based cognition is represented by SLIM 6 to SLIM 9.

Context-based cognition distinguishes between *recognition* (SLIM 1), *action* (SLIM 2), and *inferencing* (SLIM 3).

Language-based cognition distinguishes between the *hearer mode* (SLIM 4, SLIM 6, SLIM 8), and the *speaker mode* (SLIM 5, SLIM 7, SLIM 9).

Mediated reference (SLIM 6 to SLIM 9) is distinguished from *immediate* reference (SLIM 4, SLIM 5).

In *immediate reference*, language-based *control* (SLIM 6, SLIM 8) is distinguished from context-based *commenting* (SLIM 7, SLIM 9).

23.3 Semantic interpretation of LA-SU syntax

23.3.1 Representing inflectional variants in a word type

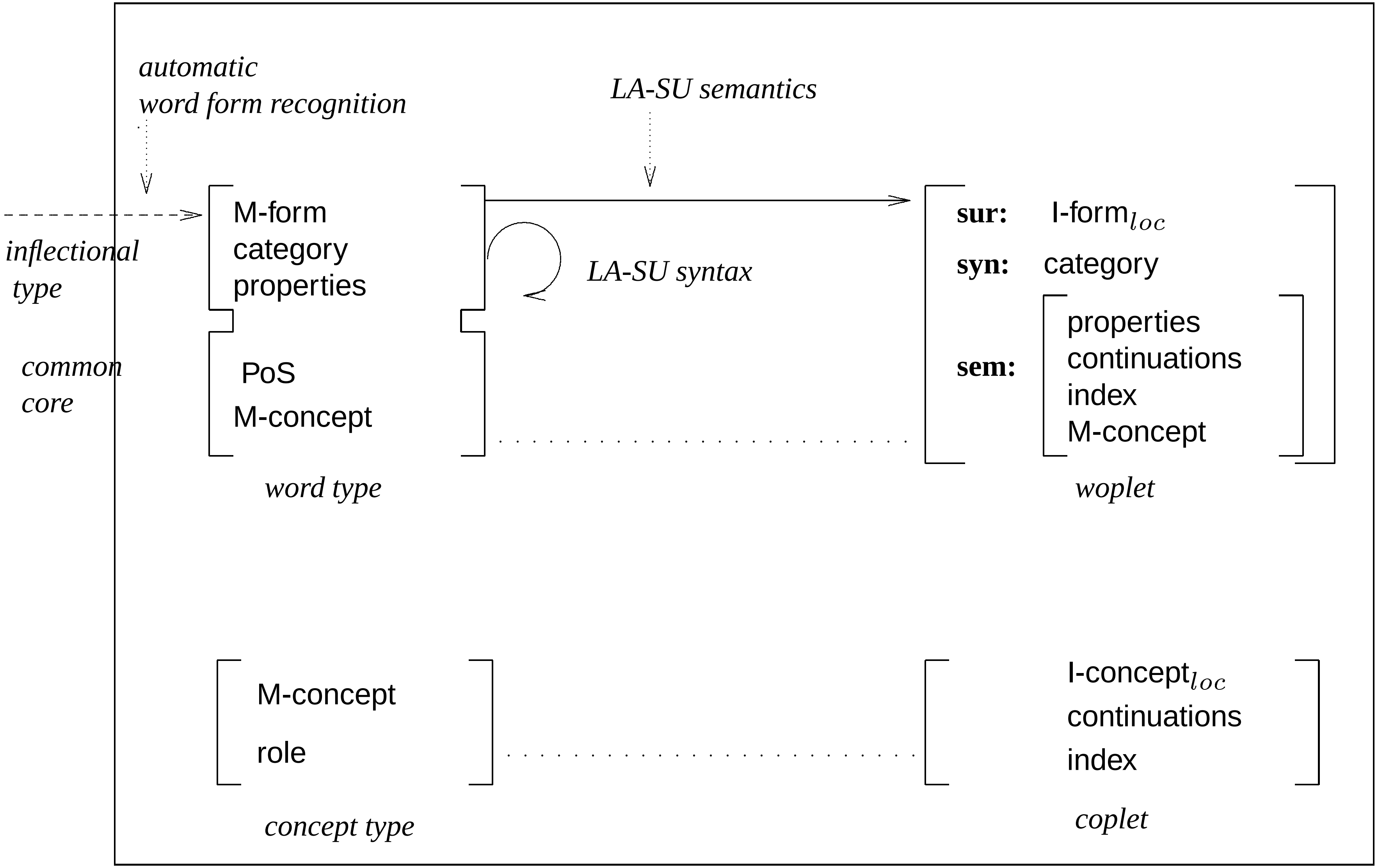
M-form: **square** category: (SN) property: singular	M-form: **square's** category: (NG) property: singular	M-form: **squares** category: (PN) property: plural	M-form: **squares'** category: (NG) property: plural

PoS: noun

M-concept: 3.3.2

common core

23.3.2 Word form recognition and derivation of a woplet



23.3.3 Nominal, verbal, and adjectival woplet structures

nominal woplet

- sur:
- syn:
- sem:
  - properties:
  - cont.:
    - MOD:
    - VERB:
  - index:
    - prn:
    - id:
  - M-concept:

verbal woplet

- sur:
- syn:
- sem:
  - properties:
  - cont..:
    - MOD:
    - NP:
  - index:
    - prn:
    - epr:
  - M-concept:

adjectival woplet

- sur:
- syn:
- sem:
  - properties:
  - cont.:
    - NP:
    - VERB:
  - index:
    - prn:
  - M-concept:

23.3.4 Schema of semantically interpreted LA-SU rule

rule:

syn: $\langle$ss-pattern$\rangle$ $\langle$nw-pattern$\rangle$ $\Longrightarrow$ $\langle$ss$'$-pattern$\rangle$

sem: semantic operations

input: output:

$$\begin{bmatrix}\text{sur:}\\ \text{syn: } \langle a\rangle\\ \text{sem:} b\end{bmatrix}_1 \ldots \begin{bmatrix}\text{sur: } m\\ \text{syn: } \langle c\rangle\\ \text{sem: } d\end{bmatrix}_i + \begin{bmatrix}\text{sur: } n\\ \text{syn: } \langle e\rangle\\ \text{sem: } f\end{bmatrix}_{i+1} \quad \begin{bmatrix}\text{sur:}\\ \text{syn: } \langle a\rangle\\ \text{sem: } b\end{bmatrix}_1 \ldots \begin{bmatrix}\text{sur: } m+n\\ \text{syn: } \langle g\rangle\\ \text{sem: } h\end{bmatrix}_{i+1}$$

23.3.5 The six basic operations of the LA-SU semantics

1. copy$_{ss}$: include the woplets of the sentence start in the result.
2. copy$_{nw}$: include the woplet of the next word in the result.
3. n_1.x –[a]→ n_2.y: copy the values of the source feature x in n_1 additively into the goal feature y in n_2, whereby n_1 and n_2 may be the woplets of the sentence start or the next word.
4. n_1.x –[e]→ n_2.y: copy the values of the source feature x in n_1 exclusively into the goal feature y in n_2, whereby the value of y must be NIL (empty value).
5. n_1.x –[r]→ n_2.①: substitute all occurrences of the variable ① in n_2 simultaneously with the value of the source feature x in n_1.
6. n.x –[m]→ n.x: mark the first value of the source feature x in n, whereby the value of x must be a list.

23.3.6 Comparison of additive and exclusive copying

Additive: nw.y —[a]→ ss.x

copy$_{ss}$

$[\text{x: } a]_1 \; [\text{x:}]_2 \; + \; [\text{y: } b]_3 \Longrightarrow \; [\star \text{ x: } a \; b]_1 \; [\star \text{ x: } b]_2$

Exclusive: nw.y —[e]→ ss.x

copy$_{ss}$

$[\text{x: } a]_1 \; [\text{x:}]_2 \; + \; [\text{y: } b]_3 \Longrightarrow \; [\text{x: } a]_1 \; [\star \text{ x: } b]_2$

23.4 Example of syntactic-semantic derivation (*LA-E4*)

23.4.1 The man gave Mary a flower because he loves her.

23.4.2 *LA-E4* for adverbial subclauses of English

LX = LX of *LA-E3* plus {(slowly (ADP) *), (because (# ADP) *)}
Variable definitions = those of *LA-E3* plus *mn* ϵ {*np* ∪ {V, VI}}

$ST_S =_{def}$ { [(x) { 1 DET+ADJ, 2 DET+N, 3 NOM+FV, 4 AUX+MAIN, 5 STRT-SBCL}] }

DET+ADJ:	(*n* x) (ADJ)	⇒	(*n* x)	{ 6 DET+ADJ, 7 DET+N}
DET+N:	(*n* x) (*n*)	⇒	(x)	{ 8 NOM+FV, 9 FV+MAIN, 10 AUX+NFV, 11 ADD-ADP, 12 IP}
NOM+FV:	(*np* # *x*) (*np*′ y V)	⇒	(y # x)	
	(*np*) (*np*′ x V)	⇒	(x V)	{13 FV+MAIN, 14 AUX+NFV, 15 ADD-ADP, 16 IP}
FV+MAIN:	(*np*′ # x) (y *np*)	⇒	(y x)	
	(*np*′ x # y) (z *np*)	⇒	(z x # y)	
	(*np*′ x V) (y *np*)	⇒	(y x V)	{17 DET+ADJ, 18 DET+N, 19 FV+MAIN, 20 IP}
AUX+NFV:	(*aux* # x V) (*aux*)	⇒	(x V)	
	(*aux* # x V) (*y aux*)	⇒	(y # x V)	

	(*aux* V) (x *aux*)	⇒	(x V)	{21 FV+MAIN, 22 IP}
AUX+MAIN:	(*np aux* V) (x *np'*)	⇒	(x *aux* VI)	{23 AUX+NFV, 24 DET+ADJ, 25 DET+N}
ADD-ADP:	(x ADP) (*mn* y)	⇒	(x *mn* y)	
	(*mn* y) (x ADP)	⇒	(x *mn* y)	{26 STRT-SBCL, 27 NOM+FV, 28 FV+MAIN}
STRT-SBCL:	(# x) (y *np*)	⇒	(y *np* # x)	{29 DET+ADJ, 30 DET+N, 31 NOM+FV, 32 ADD-ADP}
IP:	(*vt*) (*vt* x)	⇒	(x)	{}

$ST_F =_{def}$ { [(V) rp_{ip}], [(VI) rp_{ip}]}

23.4.3 SYNTACTICO-SEMANTIC ANALYSIS OF field contains triangle

combination step 1:

syntax: field (N) + contains (N A V) ⟹ field contains (A V)

semantics:

np:	field
FUNC:	
prn:	23

verb:	contain
ARG:	
prn:	23

⟹

np:	field
FUNC:	contain
prn:	23

verb:	contain
ARG:	field
prn:	23

combination step 2:

syntax: field contains (A V) + triangle (A)

semantics:

np:	field
FUNC:	contain
prn:	23

verb:	contain
ARG:	field
prn:	23

np:	triangle
FUNC:	
prn:	23

syntax: ⟹ field contains triangle (V)

semantics: ⟹

np:	field
FUNC:	contain
prn:	23

verb:	contain
ARG:	field triangle
prn:	23

np:	triangle
FUNC:	contain
prn:	23

23.4.4 The man gave Mary a flower because he loves her.

23.4.5 Applying DET+N to *the* + *man*

$$
\begin{array}{llll}
\text{syn: } \langle \text{n x} \rangle & \langle \text{n} \rangle & \Longrightarrow & \langle \text{x} \rangle \\
\text{sem:} & & \multicolumn{2}{l}{\text{nw.M} \;\text{—}\boxed{\text{r}}\!\longrightarrow \text{ss.①}} \\
 & & \multicolumn{2}{l}{\text{copy}_{ss}}
\end{array}
$$

$$
\begin{bmatrix}
\text{sur: the} \\
\text{syn: } \langle \text{SN}' \ \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg def} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } +1 \end{bmatrix} \\
\text{M: ①}
\end{bmatrix}
\end{bmatrix}_1
+
\begin{bmatrix}
\text{sur: man} \\
\text{syn: } \langle \text{SN} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P:} \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } = \\ \text{id: } = \end{bmatrix} \\
\text{M: } \textit{man}
\end{bmatrix}
\end{bmatrix}_2
\Longrightarrow
\begin{bmatrix}
\text{sur: the man} \\
\text{syn: } \star \ \langle \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg def} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } \star \ 1 \end{bmatrix} \\
\text{M: } \star \ \textit{man}
\end{bmatrix}
\end{bmatrix}_1
$$

23.4.6 Applying NOM+FV to *the man* + *gave*

syn: $\langle$np$\rangle$ $\langle$np$'$ x V$\rangle$ $\Longrightarrow$ $\langle$x V$\rangle$

sem: nw.M —[e]→ ss.VERB

ss.M —[a]→ nw.NP

copy$_{ss}$ copy$_{nw}$

$$
\begin{bmatrix}
\text{sur: the man} \\
\text{syn: } \langle \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg def} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 1 \end{bmatrix} \\
\text{M: } \textit{man}
\end{bmatrix}
\end{bmatrix}_1
+
\begin{bmatrix}
\text{sur: gave} \\
\text{syn: } \langle \text{N}' \ \text{D}' \ \text{A}' \ \text{V} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: past tense} \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } = \\ \text{epr:} \end{bmatrix} \\
\text{M: } \textit{give}
\end{bmatrix}
\end{bmatrix}_3
\Longrightarrow
\begin{bmatrix}
\text{sur:} \\
\text{syn: } \langle \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg def} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: } \star \text{ give} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 1 \end{bmatrix} \\
\text{M: } \textit{man}
\end{bmatrix}
\end{bmatrix}_1
\begin{bmatrix}
\text{sur: the man gave} \\
\text{syn: } \star \ \langle \text{D}' \ \text{A}' \ \text{V} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{past tense} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \star \ \langle \text{man} \rangle \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \star \ \langle 1 \rangle \\ \text{epr:} \end{bmatrix} \\
\text{M: } \textit{give}
\end{bmatrix}
\end{bmatrix}_3
$$

23.4.7 Applying FV+MAIN to *the man gave + Mary*

syn: ⟨np′ x V⟩ ⟨y np⟩ ⟹ ⟨y x V⟩

sem: nw.M —[a]→ ss.NP

ss.M —[e]→ nw.VERB

copy_{ss} copy_{nw}

$$\begin{bmatrix} \text{sur: the man gave} \\ \text{syn: } \star \langle \text{D}' \ \text{A}' \ \text{V} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{past tense} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \langle \text{man} \rangle \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{epr:} \end{bmatrix} \\ \text{M: } \textit{give} \end{bmatrix} \end{bmatrix}_3 + \begin{bmatrix} \text{sur: Mary} \\ \text{syn: } \langle \text{SNP} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{sg name} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } = \\ \text{id: } +1 \end{bmatrix} \\ \text{M: } \textit{Mary} \end{bmatrix} \end{bmatrix}_4$$

$$\Longrightarrow \begin{bmatrix} \text{sur: the man gave Mary} \\ \text{syn: } \star \langle \text{A}' \ \text{V} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{past tense} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \langle \text{man}, \star \text{ Mary} \rangle \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{epr:} \end{bmatrix} \\ \text{M: } \textit{give} \end{bmatrix} \end{bmatrix}_3 \begin{bmatrix} \text{sur:} \\ \text{syn: } \langle \text{SNP} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{sg name} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: } \star \text{ give} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \star \langle 1 \rangle \\ \text{id: } \star \ 2 \end{bmatrix} \\ \text{M: } \textit{Mary} \end{bmatrix} \end{bmatrix}_4$$

23.4.8 Applying FV+MAIN to *the man gave Mary + a*

syn: ⟨np′ x V⟩ ⟨y np⟩ ⟹ ⟨y x V⟩

sem:
nw.M — [a] → ss.NP
ss.M — [e] → nw.VERB
$copy_{ss}$ $copy_{nw}$

$$\left[\begin{array}{l}\text{sur: the man gave Mary}\\ \text{syn: } \langle A' \; V\rangle\\ \text{sem: } \left[\begin{array}{l}\text{P: } \langle \text{past tense}\rangle\\ \text{C: } \left[\begin{array}{l}\text{MOD:}\\ \text{NP: } \langle \text{man, Mary}\rangle\end{array}\right]\\ \text{I: } \left[\begin{array}{l}\text{prn: } \langle 1\rangle\\ \text{epr:}\end{array}\right]\\ \text{M: } \textit{give}\end{array}\right]\end{array}\right]_3 + \left[\begin{array}{l}\text{sur: a}\\ \text{syn: } \langle SN' \; SNP\rangle\\ \text{sem: } \left[\begin{array}{l}\text{P: } \langle \text{sg indef}\rangle\\ \text{C: } \left[\begin{array}{l}\text{MOD:}\\ \text{VERB:}\end{array}\right]\\ \text{I: } \left[\begin{array}{l}\text{prn: } =\\ \text{id: } +1\end{array}\right]\\ \text{M: } ①\end{array}\right]\end{array}\right]_5$$

$$\Longrightarrow \left[\begin{array}{l}\text{sur: the man gave Mary a}\\ \text{syn: } \star\ \langle SN' \; V\rangle\\ \text{sem: } \left[\begin{array}{l}\text{P: } \langle \text{past tense}\rangle\\ \text{C: } \left[\begin{array}{l}\text{MOD:}\\ \text{NP: } \langle \text{man, Mary, } \star\ ①\rangle\end{array}\right]\\ \text{I: } \left[\begin{array}{l}\text{prn: } \langle 1\rangle\\ \text{epr:}\end{array}\right]\\ \text{M: } \textit{give}\end{array}\right]\end{array}\right]_3 \left[\begin{array}{l}\text{sur:}\\ \text{syn: } \langle SN' \; SNP\rangle\\ \text{sem: } \left[\begin{array}{l}\text{P: } \langle \text{sg indef}\rangle\\ \text{C: } \left[\begin{array}{l}\text{MOD:}\\ \text{VERB: } \star\ \text{give}\end{array}\right]\\ \text{I: } \left[\begin{array}{l}\text{prn: } \langle 1\rangle\\ \text{id: } \star\ 3\end{array}\right]\\ \text{M: } ①\end{array}\right]\end{array}\right]_5$$

23.4.9 Applying DET+N to *The man gave Mary a + flower*

syn: ⟨n x⟩ ⟨n⟩ ⟹ ⟨x⟩

sem: nw.M —[r]→ ss.①
copy$_{ss}$

$$
\begin{bmatrix}
\text{sur: the man gave Mary a} \\
\text{syn: } \langle \text{SN}' \ \text{V} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{past tense} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \langle \text{man, Mary, ①} \rangle \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{epr:} \end{bmatrix} \\
\text{M: } \textit{give}
\end{bmatrix}
\end{bmatrix}_3
\begin{bmatrix}
\text{sur:} \\
\text{syn: } \langle \text{SN}' \ \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg indef} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: give} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 3 \end{bmatrix} \\
\text{M: ①}
\end{bmatrix}
\end{bmatrix}_5
+
\begin{bmatrix}
\text{sur: flower} \\
\text{syn: } \langle \text{SN} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P:} \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } = \\ \text{id: } = \end{bmatrix} \\
\text{M: } \textit{flower}
\end{bmatrix}
\end{bmatrix}_6
$$

$$
\Longrightarrow
\begin{bmatrix}
\text{sur: the man gave Mary a flower} \\
\text{syn: } \langle \text{V} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: past tense} \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \langle \text{man, Mary, } \star \text{ flower} \rangle \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{epr:} \end{bmatrix} \\
\text{M: } \textit{give}
\end{bmatrix}
\end{bmatrix}_3
\begin{bmatrix}
\text{sur:} \\
\text{syn: } \langle \text{SNP} \rangle \\
\text{sem: } \begin{bmatrix}
\text{P: } \langle \text{sg indef} \rangle \\
\text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: give} \end{bmatrix} \\
\text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 3 \end{bmatrix} \\
\text{M: } \star \ \textit{flower}
\end{bmatrix}
\end{bmatrix}_5
$$

23.4.10 Applying ADD-ADP to *The man gave Mary a flower + because*

syn: ⟨mn y⟩ ⟨x ADP⟩ $\Longrightarrow$ ⟨x mn y⟩

sem:

ss.prn — [e] → nw.p1
nw.prn — [e] → nw.p2
nw.epr — [a] → ss.epr
copy_{ss} copy_{nw}

[sur: the man gave Mary a flower
syn: ⟨V⟩
sem: [P: ⟨past tense⟩
C: [MOD:
NP: ⟨man, Mary, flower⟩]
I: [prn: ⟨1⟩
epr:]
M: *give*]]$_3$

+

[sur: because
syn: ⟨# ADV⟩
sem: [P:
C: [MOD:
NP:]
I: [prn: ⊕
epr: [p1:
con: *because*
p2:]]
M:]]$_7$

$\Longrightarrow$

[sur:
syn: ⟨V⟩
sem: [P: ⟨past tense⟩
C: [MOD:
NP: ⟨man, Mary, flower⟩]
I: [prn: ⟨1⟩
epr: ⋆ [p1: 1
con: *because*
p2: 2]]
M: *give*]]$_3$

[sur: the ... flower because
syn: ⋆ ⟨# V⟩
sem: [P:
C: [MOD:
NP:]
I: [prn: ⋆ ⟨2, 1⟩
epr: [p1: ⋆ 1
con: *because*
p2: ⋆ 2]]
M:]]$_7$

23.4.11 The epr feature structure introduced by the The conjunction because

$$
\left[\begin{array}{ll} \text{epr:} & \left[\begin{array}{l} \text{p1:} \\ \text{con: } \textit{because} \\ \text{p2:} \end{array}\right] \end{array}\right]
$$

23.4.12 Applying START-SUBCL to *The man gave Mary a flower because + he*

$$
\begin{array}{llll}
\text{syn: } \langle \# \text{ x} \rangle & \langle \text{y np} \rangle & \Longrightarrow & \langle \text{y np } \# \text{ x} \rangle \\
\text{sem:} & & \text{copy}_{ss} \text{ copy}_{nw} &
\end{array}
$$

$$
\left[\begin{array}{ll}
\text{sur: the man gave M. a f. because} & \\
\text{syn: } \langle \# \text{ V} \rangle & \\
\text{sem:} & \left[\begin{array}{ll} \text{P:} & \\ \text{C:} & \left[\begin{array}{l} \text{MOD:} \\ \text{NP:} \end{array}\right] \\ \text{I:} & \left[\begin{array}{l} \text{prn: } \langle 2, 1 \rangle \\ \text{epr: } 1 \textit{ bec } 2 \end{array}\right] \\ \text{M:} & \end{array}\right]
\end{array}\right]_7
+
\left[\begin{array}{ll}
\text{sur: he} & \\
\text{syn: } \langle \text{SNP} \rangle & \\
\text{sem:} & \left[\begin{array}{ll} \text{P: } \langle \text{nom sg} \rangle & \\ \text{C:} & \left[\begin{array}{l} \text{MOD:} \\ \text{VERB:} \end{array}\right] \\ \text{I:} & \left[\begin{array}{l} \text{prn: } = \\ \text{id: } +1 \end{array}\right] \\ \text{M: } \textit{pro-1} & \end{array}\right]
\end{array}\right]_8
$$

$$
\Longrightarrow
\left[\begin{array}{ll}
\text{sur: the man gave M. a f. because he} & \\
\text{syn: } \star \langle \text{SNP } \# \text{ V} \rangle & \\
\text{sem:} & \left[\begin{array}{ll} \text{P:} & \\ \text{C:} & \left[\begin{array}{l} \text{MOD:} \\ \text{NP:} \end{array}\right] \\ \text{I:} & \left[\begin{array}{l} \text{prn: } \langle 2, 1 \rangle \\ \text{epr: } 1 \textit{ bec } 2 \end{array}\right] \\ \text{M:} & \end{array}\right]
\end{array}\right]_7
\left[\begin{array}{ll}
\text{sur:} & \\
\text{syn: } \langle \text{SNP} \rangle & \\
\text{sem:} & \left[\begin{array}{ll} \text{P: } \langle \text{nom sg} \rangle & \\ \text{C:} & \left[\begin{array}{l} \text{MOD:} \\ \text{VERB:} \end{array}\right] \\ \text{I:} & \left[\begin{array}{l} \text{prn: } \star \langle 2, 1 \rangle \\ \text{id: } \star\, 1 \end{array}\right] \\ \text{M: } \textit{pro-1} & \end{array}\right]
\end{array}\right]_8
$$

23.4.13 Application of NOM+FV to *The man g. M. a f. because he + loves*

syn: ⟨np # x⟩ ⟨np′ y V⟩ ⟹ ⟨y # x⟩

sem:

ss.M — [a] → nw.NP
nw.M — [e] → ss.VERB
nw.M — [e] → ss.M
nw.NP — [e] → ss.NP
nw.P — [e] → ss.P
copy_{ss}

[sur: The . . . because he
syn: ⟨SNP # V⟩
sem: [P:
C: [MOD: NP:]
I: [prn: ⟨2, 1⟩ epr: 1 *bec* 2]
M:]]$_7$

[sur:
syn: ⟨SNP⟩
sem: [P: ⟨nom sg⟩
C: [MOD: VERB:]
I: [prn: ⟨2, 1⟩ id: 1]
M: *pro-1*]]$_8$

\+

[sur: loves
syn: ⟨S3′ A′ V⟩
sem: [P: ⟨present tense⟩
C: [MOD: NP:]
I: [prn: = epr:]
M: *love*]]$_9$

⟹

[sur: The . . . because he loves
syn: ⟨ ⋆ A′ # V⟩
sem: [P: ⋆ present tense
C: [MOD: NP: ⋆ pro-1]
I: [prn: ⟨2, 1⟩ epr: 1 *bec* 2]
M: ⋆ *love*]]$_7$

[sur:
syn: ⟨SNP⟩
sem: [P: ⟨nom sg⟩
C: [MOD: VERB: ⋆ love]
I: [prn: ⟨2, 1⟩ id: 1]
M: *pro-1*]]$_8$

23.4.14 Application of FV+MAIN to *The m. g. M. a f. because he loves + her*

syn: ⟨np′ # x⟩ ⟨y np⟩ ⟹ ⟨y x⟩

sem:

nw.np — [a] → ss.NP

ss.verb — [e] → nw.VERB

ss.prn — [m] → ss.prn

copy$_{ss}$ copy$_{nw}$

$$\begin{bmatrix} \text{sur: The ... because he loves} \\ \text{syn: } \langle \text{A}' \ \# \ \text{V} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: present tense} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: pro-1} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 2, 1 \rangle \\ \text{epr: 1 } \textit{bec} \text{ 2} \end{bmatrix} \\ \text{M: } \textit{love} \end{bmatrix} \end{bmatrix}_7 + \begin{bmatrix} \text{sur: her} \\ \text{syn: } \langle \text{SNP} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{obl sg} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB:} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: =} \\ \text{id: +1} \end{bmatrix} \\ \text{M: } \textit{pro-2} \end{bmatrix} \end{bmatrix}_{10}$$

$$\Longrightarrow \begin{bmatrix} \text{sur: The ... because he loves her} \\ \text{syn: } \langle \star \ \text{V} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: present tense} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: pro-1} \star \text{pro-2} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \star \langle 2\text{-}, 1 \rangle \\ \text{epr: 1 } \textit{bec} \text{ 2} \end{bmatrix} \\ \text{M: } \textit{love} \end{bmatrix} \end{bmatrix}_7 \begin{bmatrix} \text{sur:} \\ \text{syn: } \langle \text{SNP} \rangle \\ \text{sem: } \begin{bmatrix} \text{P: } \langle \text{obl sg} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: } \star \text{ love} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \star \langle 2\text{-}, 1 \rangle \\ \text{id: } \star \text{ 2} \end{bmatrix} \\ \text{M: } \textit{pro-2} \end{bmatrix} \end{bmatrix}_{10}$$

23.4.15 Proposition number of embedded subclause

the man,		gave her a flower.
prn: ⟨1⟩	because he loves Mary	prn: ⟨2-, 1⟩
	prn: ⟨2, 1⟩	

23.5 From SLIM semantics to SLIM pragmatics

23.5.1 SLIM semantic representation of example 23.4.1

the man

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{SNP} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{sg def} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: give} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 1 \end{bmatrix} \\ \text{M: } \textit{man} \end{bmatrix} \end{bmatrix}_1$$

gave

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{V} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{past tense} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \langle \text{man, Mary, flower} \rangle \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{epr: } 1 \textit{ bec } 2 \end{bmatrix} \\ \text{M: } \textit{give} \end{bmatrix} \end{bmatrix}_3$$

Mary

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{SNP} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{sg name} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: give} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 2 \end{bmatrix} \\ \text{M: } \textit{Mary} \end{bmatrix} \end{bmatrix}_4$$

a flower

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{SNP} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{sg indef} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: give} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 1 \rangle \\ \text{id: } 3 \end{bmatrix} \\ \text{M: } \textit{flower} \end{bmatrix} \end{bmatrix}_5$$

because loves

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{V} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{3sg, present tense} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: pro-1, pro-2} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 2-,1 \rangle \\ \text{epr: } 1\ \textit{bec}\ 2 \end{bmatrix} \\ \text{M: } \textit{love} \end{bmatrix} \end{bmatrix}_7$$

he

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{SNP} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{nom sg} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: love} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 2\text{-}, 1 \rangle \\ \text{id: } 1 \end{bmatrix} \\ \text{M: } \textit{pro-1} \end{bmatrix} \end{bmatrix}_8$$

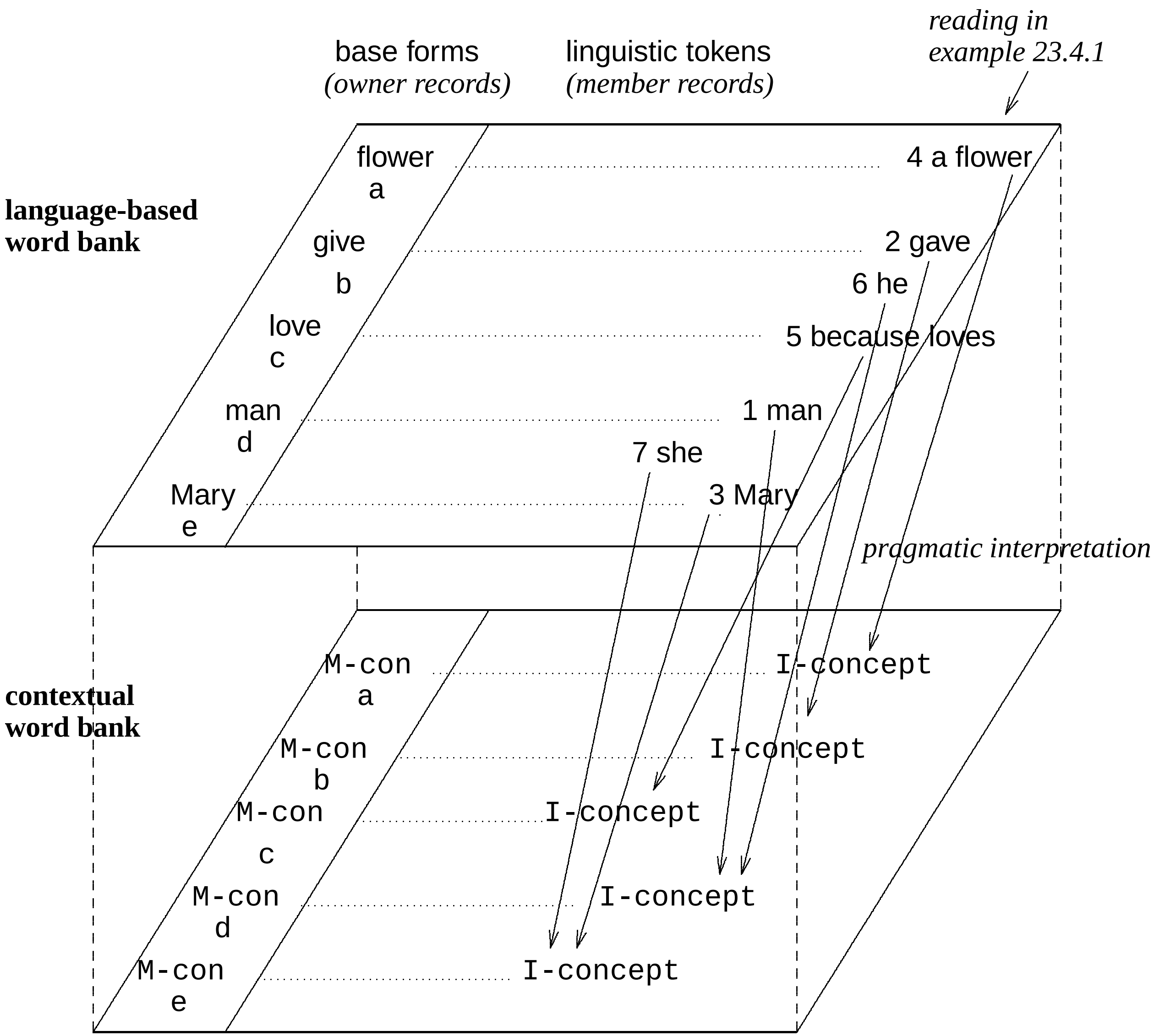
her

$$\begin{bmatrix} \text{sur:} & \\ \text{syn:} & \langle \text{SNP} \rangle \\ \text{sem:} & \begin{bmatrix} \text{P: } \langle \text{obl sg} \rangle \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: love} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \langle 2\text{-},1 \rangle \\ \text{id: } 2 \end{bmatrix} \\ \text{M: } \textit{pro-2} \end{bmatrix} \end{bmatrix}_9$$

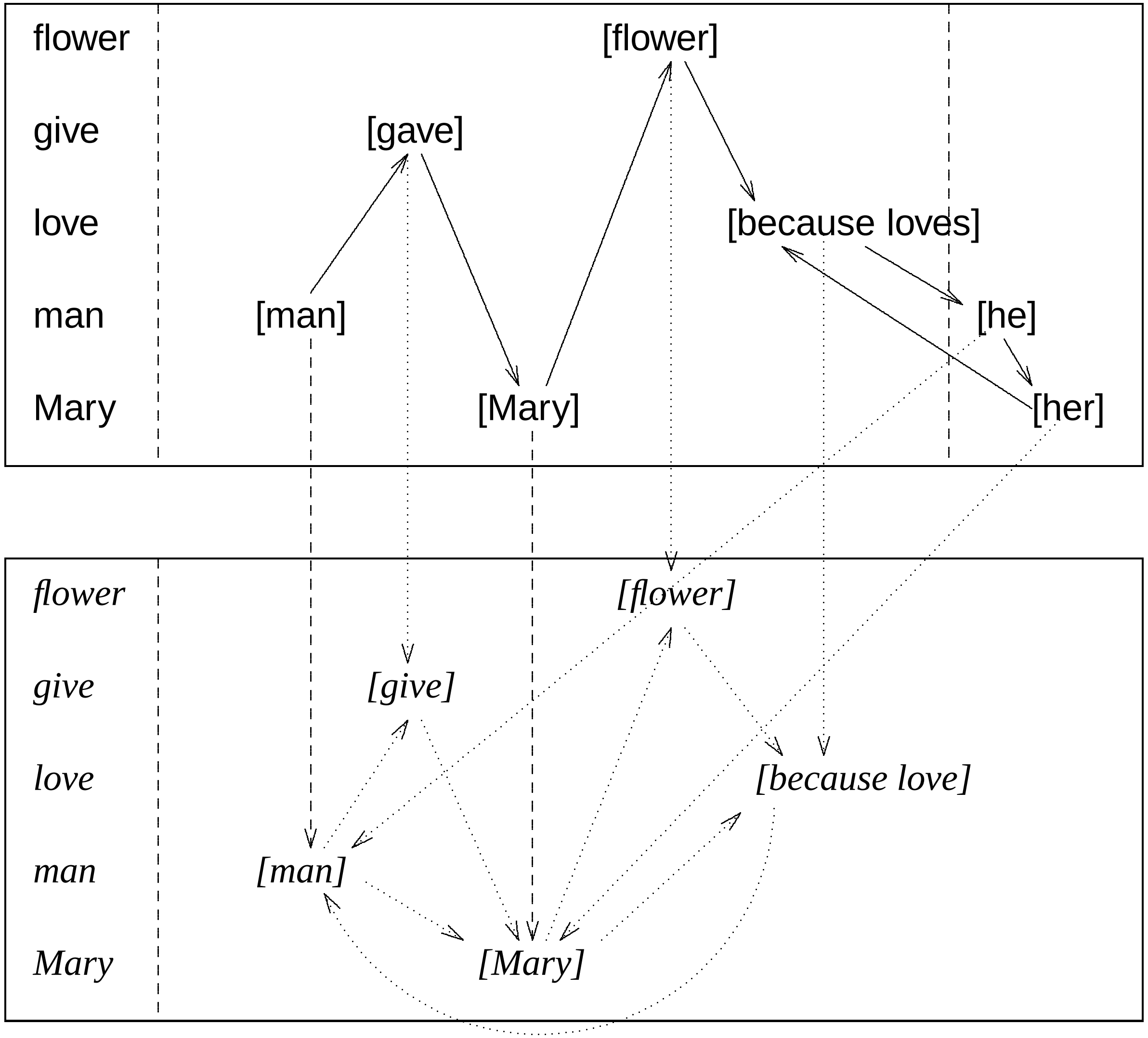
23.5.2 Components of meaning$_1$

- Compositional semantics (sentence semantics)
  1. Decomposition of input into elementary propositions.
  2. Functor-argument structure within an elementary proposition.
  3. Extrapropositional relations among elementary propositions.
- Lexical semantics (word semantics)
  1. Properties and M-concepts of woplets.
  2. Extrapropositional relations between word types by means of *absolute propositions*.

23.5.3 Embedding 23.5.1 into the contextual word bank



23.5.4 Contextual reconstruction of language information



24. SLIM machine in the speaker mode

24.1 Subcontext as concatenated propositions

24.1.1 Immediate vs. mediated subcontexts

In immediate subcontexts, the coherence of the content follows directly from the coherence of the external world which they reflect, i.e., the temporal and spatial sequence of events, the part-whole relations of objects, etc. In contrast, mediated subcontexts have the special property that the elements familiar from direct recognition may be reordered and reconnected by the author at will.

24.1.2 Comparing coherence and inchorence, Example I

The representation of a swimmer standing at the pool side, diving into the water, and disappearing with a splash is coherent. In contrast, a representation in which a pair of feet appears in the foaming water and a swimmer flies feet first into the air landing on the pool side, would be incoherent – unless it is specified in addition that the representation happens to be, e.g., a backward running movie.

24.1.3 Comparing coherence and inchorence, Example II

A representation of people talking with each other would be coherent. In contrast, a similar representation of a deer conversing with a skunk in English would be incoherent – unless it is specified in addition that the representation happens to be fictional.

24.1.4 Mediated subcontexts reflecting the coherence of the external world

world $\rightarrow$ speaker context $\rightarrow$ language $\rightarrow$ hearer context $\rightarrow$ world

24.1.5 A sequence of propositions forming a subcontext

1. Peter leaves the house. 2. Peter crosses the street. 3. Peter enters a restaurant. 4. Peter orders a salad. 5. Peter eats the salad. 6. Peter pays the salad. 7. Peter leaves the restaurant. 8. Peter crosses the street. 9. Peter enters the house.

24.1.6 Equivalent representation of 24.1.1 as a word bank

CONCEPT TYPES: COPLETS:

$$\begin{bmatrix}\text{M-concept: cross}\\ \text{role: T-verb}\end{bmatrix}\quad
\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{cross}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, street}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 2}\\ \text{epr: } \begin{bmatrix}\text{2 then 3}\\ \text{1 then 2}\end{bmatrix}\end{bmatrix}\end{bmatrix}\quad
\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{cross}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, street}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 8}\\ \text{epr: } \begin{bmatrix}\text{8 then 9}\\ \text{7 then 8}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: eat}\\ \text{role: T-verb}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{eat}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 5}\\ \text{epr: } \begin{bmatrix}\text{5 then 6}\\ \text{4 then 5}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: enter}\\ \text{role: T-verb}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{enter}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, restaurant}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 3}\\ \text{epr: } \begin{bmatrix}\text{3 then 4}\\ \text{2 then 3}\end{bmatrix}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{enter}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, house}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 9}\\ \text{epr: } \begin{bmatrix}\\ \text{8 then 9}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: house}\\ \text{role: noun}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{house}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: leave}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 1}\\ \text{id: 2}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{house}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: enter}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 9}\\ \text{id: 2}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: leave}\\ \text{role: T-verb}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{leave}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, house}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 1}\\ \text{epr: } \begin{bmatrix}\text{1 then 2}\\ \end{bmatrix}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{leave}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, restaurant}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 7}\\ \text{epr: } \begin{bmatrix}\text{7 then 8}\\ \text{6 then 7}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: order}\\ \text{role: T-verb}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{order}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 4}\\ \text{epr: } \begin{bmatrix}\text{4 then 5}\\ \text{3 then 4}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: pay}\\ \text{role: T-verb}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{pay}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 6}\\ \text{epr: } \begin{bmatrix}\text{6 then 7}\\ \text{5 then 6}\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: Peter}\\ \text{role: name}\end{bmatrix}$$

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{VERB: leave}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 1}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MD:}\\ \text{VB: cross}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 2}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MD:}\\ \text{VB: enter}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 3}\\ \text{id: 1}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{VERB: order}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 4}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{VERB: eat}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 5}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{VERB: pay}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 6}\\ \text{id: 1}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{VERB: leave}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 7}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MD:}\\ \text{VB: cross}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 8}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: }\begin{bmatrix}\text{MD:}\\ \text{VB: enter}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 9}\\ \text{id: 1}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: restaurant}\\ \text{role: noun}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{restaurant}\\ \text{P: A sg indef}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: enter}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 3}\\ \text{id: 4}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{restaurant}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: leave}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 7}\\ \text{id: 4}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: salad}\\ \text{role: noun}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{salad}\\ \text{P: A sg indef}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: order}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 4}\\ \text{id: 5}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{salad}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: eat}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 5}\\ \text{id: 5}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{salad}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: pay}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 6}\\ \text{id: 5}\end{bmatrix}\end{bmatrix}$$

$$\begin{bmatrix}\text{M-concept: street}\\ \text{role: noun}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{street}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: cross}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 2}\\ \text{id: 3}\end{bmatrix}\end{bmatrix} \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{street}\\ \text{P: A sg def}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: cross}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 8}\\ \text{id: 3}\end{bmatrix}\end{bmatrix}$$

24.2 Tracking principles of LA-navigation

24.2.1 Step 1 of a LA-NA rule application

$$\text{rule}_{1+2\Rightarrow 2}: \quad \overset{\text{START}}{\begin{bmatrix} \text{m1: } a \\ \text{M2: } b \\ \text{prn: } c \end{bmatrix}} \quad \overset{\text{NEXT}}{\begin{bmatrix} \text{m2: } b \\ \text{M1: } x\ a\ y \\ \text{prn: } c \end{bmatrix}} \quad \Longrightarrow \quad \overset{\text{NEW START}}{\begin{bmatrix} \text{m2: } b \\ \\ \\ \end{bmatrix}} \quad \text{rule package}_{1+2\Rightarrow 2}$$

$$\begin{array}{c}\text{coplets} \\ \text{of the} \\ \text{word bank}\end{array} \quad \begin{bmatrix} \dots \\ \text{m1: } c1 \\ \dots \\ \text{M2: } c2 \\ \dots \\ \text{prn: } c3 \\ \dots \end{bmatrix}_1$$

24.2.2 Step 2 of an LA-NA rule application

$$
\begin{array}{lccccc}
 & \text{START} & \text{NEXT} & & \text{NEW START} & \\
\text{rule}_{1+2\Rightarrow 2}: &
\begin{bmatrix} \text{m1: } a \\ \text{M2: } b \\ \text{prn: } c \end{bmatrix} &
\begin{bmatrix} \text{m2: } b \\ \text{M1: } x\ a\ y \\ \text{prn: } c \end{bmatrix} &
\Longrightarrow &
\begin{bmatrix} \text{m2: } b \\ \\ \\ \end{bmatrix} & \text{rule package}_{1+2\Rightarrow 2}
\end{array}
$$

$$
\begin{array}{lccc}
\begin{array}{c}\text{coplets} \\ \text{of the} \\ \text{word bank}\end{array} &
\begin{bmatrix} \dots \\ \text{m1: } c1 \\ \dots \\ \text{M2: } c2 \\ \dots \\ \text{prn: } c3 \\ \dots \end{bmatrix}_1 &
+ &
\begin{bmatrix} \dots \\ \text{m2: } c2 \\ \dots \\ \text{M1: } ..c1.. \\ \dots \\ \text{prn: } c3 \\ \dots \end{bmatrix}_2
\end{array}
$$

24.2.3 Step 3 of a LA-NA rule application

$$
\begin{array}{llllll}
 & \text{START} & \text{NEXT} & & \text{NEW START} & \\
\text{rule}_{1+2\Rightarrow 2}: &
\begin{bmatrix} \text{m1: } a \\ \text{M2: } b \\ \text{prn: } c \end{bmatrix} &
\begin{bmatrix} \text{m2: } b \\ \text{M1: } x\ a\ y \\ \text{prn: } c \end{bmatrix} &
\Longrightarrow &
\begin{bmatrix} \text{m2: } b \\ \\ \\ \end{bmatrix} &
\text{rule package}_{1+2\Rightarrow 2}
\end{array}
$$

$$
\begin{array}{llllll}
\begin{array}{c}\text{coplets} \\ \text{of the} \\ \text{word bank}\end{array} &
\begin{bmatrix} \ldots \\ \text{m1: } c1 \\ \ldots \\ \text{M2: } c2 \\ \ldots \\ \text{prn: } c3 \\ \ldots \end{bmatrix}_1 &
+ &
\begin{bmatrix} \ldots \\ \text{m2: } c2 \\ \ldots \\ \text{M1: } ..c1.. \\ \ldots \\ \text{prn: } c3 \\ \ldots \end{bmatrix}_2 &
\Longrightarrow &
\begin{bmatrix} \ldots \\ \text{m2: } c2 \\ \ldots \\ \text{M1: } ..c1.. \\ \ldots \\ \text{prn: } c3 \\ \ldots \end{bmatrix}_2
\end{array}
$$

24.2.4 Tracking principles of LA-navigation

1. *Completeness*
   Within an elementary proposition those coplets are preferred which have not yet been traversed during the current navigation.
2. *Uniqueness*
   If several START or NEXT coplets are available, no more than one of each are selected whereby the choice may be at random or – if activated – based on a specific navigation pattern.
3. *Recency*
   In extrapropositional navigations, propositions which have been least recently traversed are preferred.
4. *Frequency*
   When entering a new subcontext, the navigation prefers paths most frequently traversed in previous navigations.

24.2.5 Definition of universal LA-NA syntax

ST$_S$: {([M-np: a] {1 V+NP1, 2 V+NP2})}

$$\text{V+NP1:}\quad \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\ b\ y\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-np: } b\\ \text{VERB: } a\\ \text{prn: } m\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \\ \\ \end{bmatrix} \quad \{3\text{ V+NP1, 4 V+NP2, 5 V+epr}\}$$

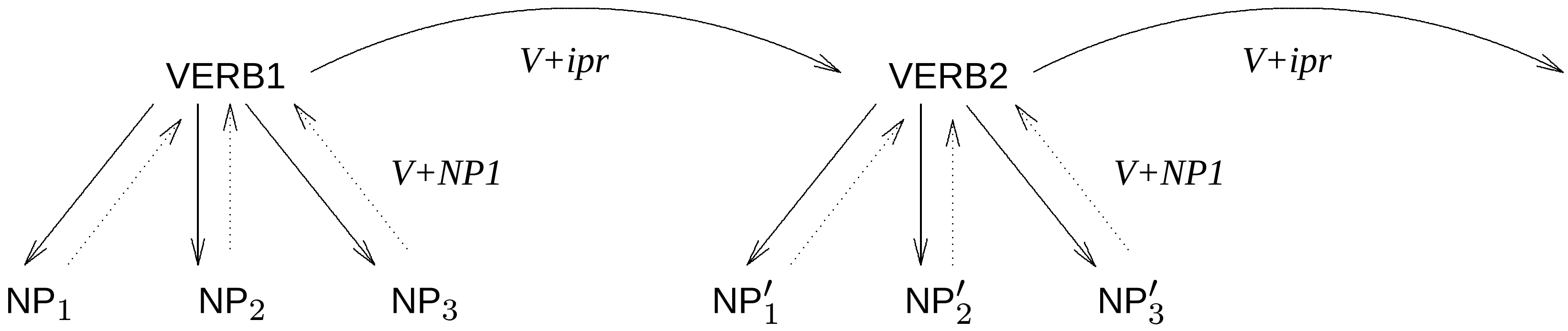
$$\text{V+NP2:}\quad \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\ b\ y\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-np: } b\\ \text{VERB: } a\\ \text{prn: } m\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-np: } b\\ \\ \\ \end{bmatrix} \quad \{6\text{ NP+id}\}$$

$$\text{V+epr:}\quad \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\\ \text{prn: } m\\ \text{epr: } m\ C\ n\end{bmatrix} \begin{bmatrix}\text{M-verb: } b\\ \text{NP: } y\\ \text{prn: } n\\ \text{epr: } m\ C\ n\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } b\\ \\ \\ \end{bmatrix} \quad \{7\text{ V+NP1, 8 V+NP2}\}$$

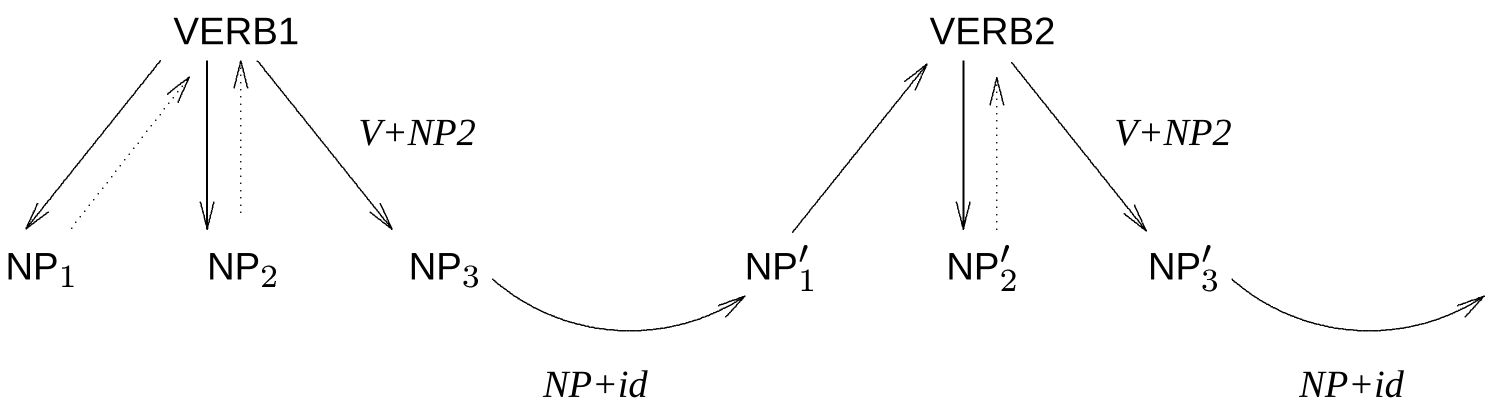
$$\text{NP+id:}\quad \begin{bmatrix}\text{M-np: } a\\ \text{VERB: } b\\ \text{prn: } k\\ \text{id: } m\end{bmatrix} \begin{bmatrix}\text{M-np: } a\\ \text{VERB: } c\\ \text{prn: } l\\ \text{id: } m\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } c\\ \text{NP: } x\ a\ y\\ \text{prn: } l\\ \\ \end{bmatrix} \quad \{9\text{ V+NP1 10 V+NP2}\}$$

ST$_F$: { ([M-verb: x] rp $_{\text{V+NP1}}$)}

24.2.6 Extrapropositional epr-navigation



24.2.7 Extrapropositional id-navigation



24.2.8 First Application of V+NP1 in the word bank 24.1.2

$$
\text{V+NP1: } \begin{bmatrix} \text{M-verb: } a \\ \text{NP: } x\ b\ y \\ \text{prn: } c \end{bmatrix} \quad \begin{bmatrix} \text{M-np: } b \\ \text{VERB: } a \\ \text{prn: } c \end{bmatrix} \Longrightarrow \quad \begin{bmatrix} \text{M-verb: } a \\ \\ \\ \end{bmatrix} \ \{3 \text{ V+NP1, } 4 \text{ V+NP2, } 5 \text{ V+epr}\}
$$

$$
\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{eat} \\ \text{P: indicative} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: Peter, salad} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{epr: } \begin{bmatrix} \text{5 then 6} \\ \text{4 then 5} \end{bmatrix} \end{bmatrix} \end{bmatrix}
\quad
\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{salad} \\ \text{P: A sg def} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: eat} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{id: 2} \end{bmatrix} \end{bmatrix}
\quad
\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{eat} \\ \text{P: indicative} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: Peter,salad} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{epr: } \begin{bmatrix} \text{5 then 6} \\ \text{4 then 5} \end{bmatrix} \end{bmatrix} \end{bmatrix}
$$

24.2.9 Second application of V+NP1 in the word bank 24.1.2

$$\text{V+NP1:} \begin{bmatrix} \text{M-verb: } a \\ \text{NP: } x\ b\ y \\ \text{prn: } c \end{bmatrix} \quad \begin{bmatrix} \text{np: } b \\ \text{VERB: } a \\ \text{prn: } c \end{bmatrix} \Longrightarrow \quad \begin{bmatrix} \text{M-verb: } a \\ \\ \\ \end{bmatrix} \{\text{3 V+NP1, 4 V+NP2, 5 V+epr}\}$$

$$\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{eat} \\ \text{P: indicative} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: Peter, salad} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{epr: } \begin{bmatrix} \text{5 then 6} \\ \text{4 then 5} \end{bmatrix} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{Peter} \\ \text{P: Nom} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{VERB: eat} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{id: 1} \end{bmatrix} \end{bmatrix} \quad \begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{eat} \\ \text{P: indicative} \\ \text{C: } \begin{bmatrix} \text{MOD:} \\ \text{NP: Peter, salad} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: 5} \\ \text{epr: } \begin{bmatrix} \text{5 then 6} \\ \text{4 then 5} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

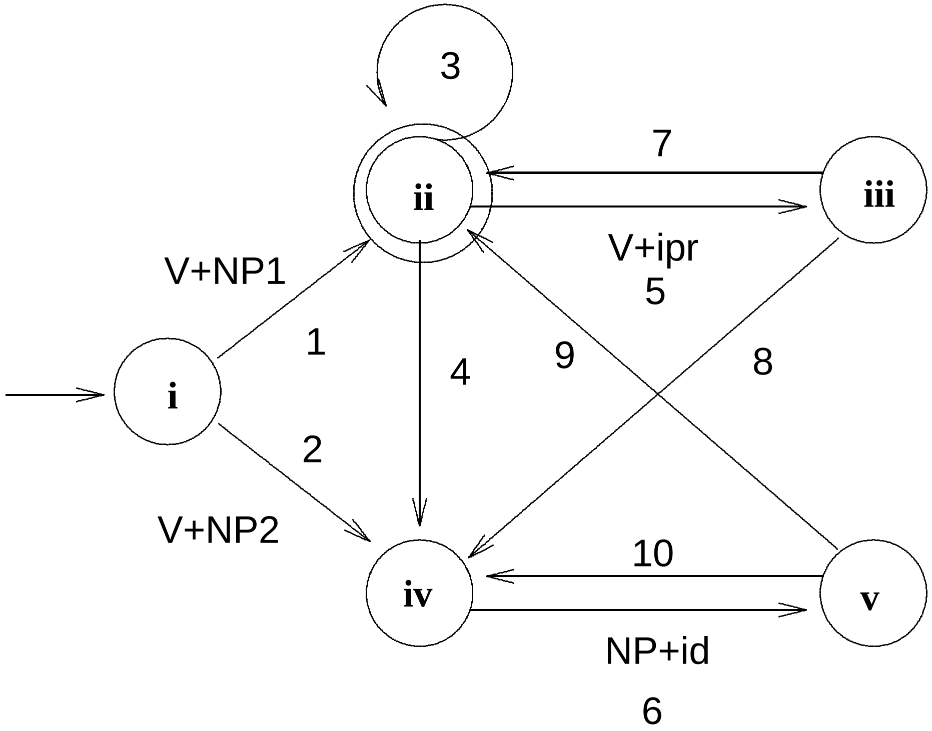
24.2.10 Application of V+epr in the word bank 24.1.2

$$
\text{V+epr:}\quad
\begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\\ \text{prn: } m\\ \text{epr: } m\ C\ n\end{bmatrix}
\quad
\begin{bmatrix}\text{M-verb: } b\\ \text{NP: } y\\ \text{prn: } n\\ \text{epr: } m\ C\ n\end{bmatrix}
\quad\Longrightarrow\quad
\begin{bmatrix}\text{M-verb: } b\\ \\ \\ \\ \end{bmatrix}
\quad \{7 \text{ V+NP1}, 8 \text{ V+NP2}\}
$$

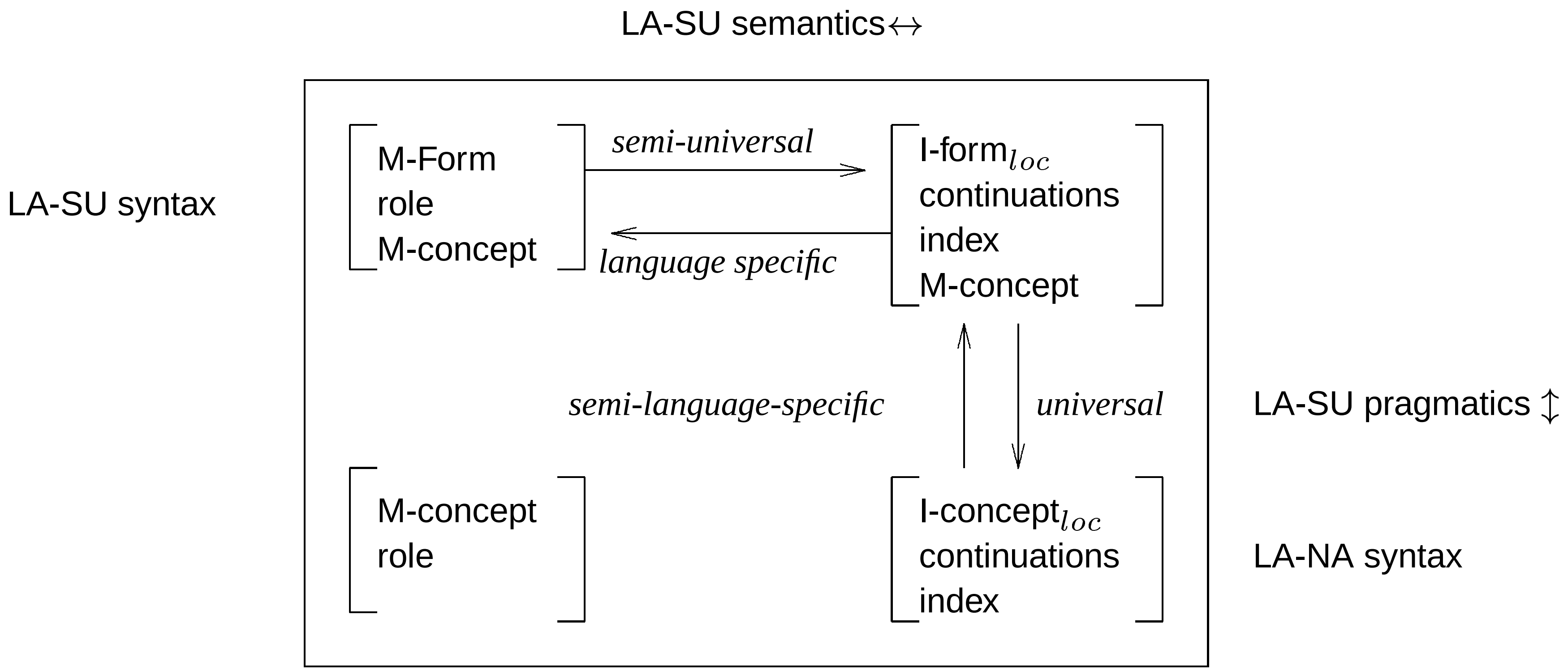
$$
\begin{bmatrix}
\text{I-concept}_{loc}\text{: } \textit{eat}\\
\text{P: indicative}\\
\text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\
\text{I: } \begin{bmatrix}\text{prn: 5}\\ \text{epr: } \begin{bmatrix}\text{5 then 6}\\ \text{4 then 5}\end{bmatrix}\end{bmatrix}
\end{bmatrix}
\quad
\begin{bmatrix}
\text{I-concept}_{loc}\text{: } \textit{pay}\\
\text{P: indicative}\\
\text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\
\text{I: } \begin{bmatrix}\text{prn: 6}\\ \text{epr: } \begin{bmatrix}\text{6 then 7}\\ \text{5 then 6}\end{bmatrix}\end{bmatrix}
\end{bmatrix}
\quad
\begin{bmatrix}
\text{I-concept}_{loc}\text{: } \textit{pay}\\
\text{P: indicative}\\
\text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, salad}\end{bmatrix}\\
\text{I: } \begin{bmatrix}\text{prn: 6}\\ \text{epr: } \begin{bmatrix}\text{6 then 7}\\ \text{5 then 6}\end{bmatrix}\end{bmatrix}
\end{bmatrix}
$$

24.3 Interpreting autonomous LA-navigation with language

24.3.1 The finite state back bone of LA-NA



24.3.2 Universality and language specificity in a SLIM machine



24.3.3 Realization principles of the basic word orders

VSO languages

```
                                 realization
V+NP                                  ↑
buffer: [Verb] + [NP1]  ⟹  [Verb] [NP1]

                                               realization
V+NP                                                ↑
buffer: *[Verb] [NP1] + [NP2]  ⟹  *[Verb] [NP1] [NP2]

                                                             realization
V+NP                                                           1↑     2↑
buffer: *[Verb] *[NP1] [NP2] + [NP3]  ⟹  *[Verb] *[NP1] [NP2] [NP3]
```

SVO languages

realization
V+NP ↑
buffer: [Verb] + [NP_1] $\Longrightarrow$ [Verb] [NP_1]

realization
V+NP ↑
buffer: *[NP_1] [Verb] + [NP_2] $\Longrightarrow$ *[NP_1] [Verb] [NP_2]

realization
V+NP 1↑ 2↑
buffer: *[NP_1] *[Verb] [NP_2] + [NP_3] $\Longrightarrow$ *[NP_1] *[Verb] [NP_2] [NP_3]

SOV languages

```
                                        realization
V+NP                                        ↑
buffer: [Verb] + [NP1]  ⟹  [Verb] [NP1]

                                                        realization
V+NP                                                        ↑
buffer: *[NP1] [Verb] + [NP2]  ⟹  *[NP1] [Verb] [NP2]

                                                                realization
V+NP                                                            2↑     1↑
buffer: *[NP1] *[NP2] [Verb] + [NP3]  ⟹  *[NP1] *[NP2] [Verb][NP3]
```

24.4 Subordinating navigation

24.4.1 epr-concatenation

Peter leaves the house. Then he crosses the street.
Peter crosses the street. Before that he leaves the house.

24.4.2 id-concatenation

Peter orders a salad. The salad is eaten by Peter.

24.4.3 epr-subordination (adverbial clauses)

Before Peter crosses the street, he leaves the house.
Peter, before he crosses the street, leaves the house.
Peter leaves, before he crosses the street, the house.
Peter leaves the house, before he crosses the street.
After Peter leaves the house, he crosses the street.
Peter, after he leaves the house, crosses the street.
Peter crosses, after he leaves the house, the street.
Peter crosses the street, after he leaves the house.

24.4.4 id-subordination (relative clause)

Peter, who leaves the house, crosses the street.

24.4.5 Applying NP+id in the word bank 24.1.2

$$\text{NP+id:}\quad \begin{bmatrix}\text{M-np: } a\\ \text{VERB: } b\\ \text{prn: } k\\ \text{id: } m\end{bmatrix} \quad \begin{bmatrix}\text{M-np: } a\\ \text{VERB: } c\\ \text{prn: } l\\ \text{id: } m\end{bmatrix} \quad \Longrightarrow \quad \begin{bmatrix}\text{M-verb: } c\\ \text{NP: } x\ a\ y\\ \text{prn: } l\\ \text{epr:}\end{bmatrix} \quad \{\text{9 V+NP1, 10 V+NP2}\}$$

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: cross}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 2}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{Peter}\\ \text{P: Nom}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{VERB: leave}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 1}\\ \text{id: 1}\end{bmatrix}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{leave}\\ \text{P: indicative}\\ \text{C: } \begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, house}\end{bmatrix}\\ \text{I: } \begin{bmatrix}\text{prn: 1}\\ \text{epr: } [\,]\end{bmatrix}\end{bmatrix}$$

24.4.6 Adnominal embedding navigation (preverbal)

Peter, who leaves the house, crosses the street.

▼*cross*	*Peter*	NP+id:		*street*
prn:2	prn:2	*leave*	*house* ▲	prn: 2
	id: 1	prn:1	prn:1	id: 2
			id:3	

24.4.7 Word order of adnominal embedding in German

Peter, der das Haus verlassen hat, überquert die Straße.
Peter, who the house left-has, crosses the street.

realization
↑
V+NP2
buffer: [Verb] + [NP_1] $\Longrightarrow$ [Verb]▼ [NP_1]

realization
↑
NP+id
buffer: *[NP_1] [Verb] ▼ + [NP1′] $\Longrightarrow$ *[NP_1] PRO [Verb]▼ [Verb′]

realization
2↑ 1↑
V+NP1
buffer: *[NP_1] *[PRO] [Verb]▼ [Verb′] + [NP2′] $\Longrightarrow$ *[NP_1] *[PRO] [Verb]▲ [Verb′] [NP2′]

V+NP1
realization
1↑ 2↑
buffer: *[NP_1] *[PRO] *[NP2′] *[Verb′] [Verb]▲ + [NP_2]
$\Longrightarrow$ *[NP_1] *[PRO] *[NP2′] *[Verb′] [Verb]▲ [NP_2]

24.4.8 Application of V+epr in the word bank 24.1.2

$$\text{V+epr: }\begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\\ \text{prn: } n\\ \text{epr: } m\ C\ n\end{bmatrix}\quad\begin{bmatrix}\text{M-verb: } b\\ \text{NP: } y\\ \text{prn: } m\\ \text{epr: } m\ C\ n\end{bmatrix}\quad\Longrightarrow\quad\begin{bmatrix}\text{M-verb: } b\\ \\ \\ \\ \end{bmatrix}\ \{7\ \text{V+NP1},\ 8\ \text{V+NP2}\}$$

▼

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{cross}\\ \text{P: indicative}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, street}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 2}\\ \text{epr: }\begin{bmatrix}\text{1 then 2}\\ \text{2 then 3}\end{bmatrix}\end{bmatrix}\end{bmatrix}\quad\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{leave}\\ \text{P: indicative}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, house}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 1}\\ \text{epr: }\begin{bmatrix}\text{1 then 2}\\ \\ \end{bmatrix}\end{bmatrix}\end{bmatrix}\quad\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{leave}\\ \text{P: indicative}\\ \text{C: }\begin{bmatrix}\text{MOD:}\\ \text{NP: Peter, house}\end{bmatrix}\\ \text{I: }\begin{bmatrix}\text{prn: 1}\\ \text{epr: }\begin{bmatrix}\text{1 then 2}\\ \\ \end{bmatrix}\end{bmatrix}\end{bmatrix}$$

24.4.9 Different realizations of conjunctions

	temporal	causal	modal
coordinating forward:	P1. Then P2.	P1. Therefore P2.	P1. Thus P2.
coordinating backward:	P2. Earlier P1.		
subordinating forward:	p1, before P2, p1.	p1, for which reason P2, p1.	p1, as P2, p1
subordinating backward:	p2, after P1, p2.	p2, because P1, p2.	

24.4.10 Adverbial embedding navigation

Peter crossed, after he left the house, the street.

▼*cross*	*Peter*	V+epr			*street*
prn:2	prn:2	*leave*	*Peter*	*house* ▲	prn: 2
(2 then 3)	id: 1	prn: 1	prn:1	prn:1	id: 3
(1 then 2)		(1 then 2)	id:1	id:2	

24.4.11 Word order of adverbial embedding in German

Peter überquert, nachdem er das Haus verlassen hat, die Straße.

(*Peter crosses, after he the house left-has, the street.*)

realization

V+NP1 ↑

buffer: [Verb] + [NP$_1$] $\Longrightarrow$ [Verb] [NP$_1$]

realization

V+epr 1↑ 2↑

buffer: *[NP$_1$] [Verb] + [Verb$'$] $\Longrightarrow$ *[NP$_1$] [Verb]▼ [CNJ] [Verb$'$]

realization

V+NP1 1↑

buffer: *[NP$_1$] *[Verb]▼ *[CNJ] [Verb$'$] + [NP1$'$] $\Longrightarrow$ *[NP$_1$] *[Verb]▼ *[CNJ] [Verb$'$] [NP1$'$]

V+NP1 realization

buffer: *[NP$_1$] *[Verb]▼ *[CNJ] *[NP1$'$] [Verb$'$] + [NP2$'$] 2↑ 1↑

$\Longrightarrow$ *[NP$_1$] *[Verb]▲ *[CNJ] *[NP1$'$] [Verb$'$] [NP2$'$]

V+NP1 realization

buffer: *[NP$_1$] *[Verb]▲ *[CNJ] *[NP1$'$] *[NP2$'$] *[Verb$'$] + [NP$_2$] 1↑

$\Longrightarrow$ *[NP$_1$] *[Verb]▲ *[CNJ] *[NP1$'$] *[NP2$'$] *[Verb$'$] [NP$_2$]

24.4.12 Multiple center embeddings in German

Peter, der den Salat, den er gegessen hatte, bezahlt hatte, verließ das Restaurant.
(*Peter, who the salad, which he paid-had, eaten-had, left the restaurant.*)

▼*leave*	*Peter*	NP+id:				*restaurant*
prn:7	prn:7	▼*pay*	*salad* ▲			▲ prn: 7
	id: 1	prn:6	prn:6	▼*eat*	*Peter* ▲	id: 4
			id:5	prn: 5	id:1	

24.5 LA-search and LA-inference

24.5.1 Basic types of questions in natural language

Wh-question	**Yes/no-question**
Who entered the restaurant?	Did Peter enter the restaurant?

24.5.2 Search coplets of the two basic types of queries

Wh-question

$$\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{enter} \\ \text{E:} \\ \text{F: } \begin{bmatrix} \text{MOD:} \\ \text{NP: } \sigma\text{-1, restaurant} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \sigma\text{-}\textit{2} \end{bmatrix} \end{bmatrix}$$

Yes/no-question

$$\begin{bmatrix} \text{I-concept}_{loc}\text{: } \textit{enter} \\ \text{E:} \\ \text{F: } \begin{bmatrix} \text{MOD:} \\ \text{NP: Peter, restaurant} \end{bmatrix} \\ \text{I: } \begin{bmatrix} \text{prn: } \sigma\text{-}\textit{2} \end{bmatrix} \end{bmatrix}$$

24.5.3 LA-Q1 (WH-questions)

ST_S: {([a]{1 r_1, 2 r_2})}

$$\text{r}_1: \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } y\ \sigma\ z\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } y\ \sigma\ z\\ \text{prn: } m-1\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } y\ \sigma\ z\\ \text{prn: } m-1\end{bmatrix} \{3\ \text{r}_1\ 4\ \text{r}_2\}$$

$$\text{r}_2: \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } y\ \sigma\ z\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } y\ \sigma\ z\\ \text{prn: } m-1\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } y\ \sigma\ z\\ \text{prn: } m-1\end{bmatrix} \{5\ \text{r}_3\}$$

$$\text{r}_3: \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } y\ \sigma\ z\\ \text{prn: } n\end{bmatrix} \begin{bmatrix}\text{M-np: } \sigma\\ \text{VERB: } a\\ \text{prn: } n\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-np: } \sigma\\ \text{VERB: } a\\ \text{prn: } n\end{bmatrix} \{\ \}$$

ST_F: {([M-np: σ] rp_3)}

24.5.4 LA-Q2 (yes/no-questions)

ST_S: {([a]{1 r_1, 2 r_2})}

$$r_1: \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } x\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } x\\ \text{prn: } m-1\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } x\\ \text{prn: } m-1\end{bmatrix} \{3\ r_1\ 4\ r_2\}$$

$$r_2: \begin{bmatrix}\text{M-verb: } a\\ \neg\text{NP: } x\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\\ \text{prn: } m-1\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \text{NP: } x\\ \text{prn: } m-1\end{bmatrix} \{\ \}$$

ST_F: {([verb: a] rp_1) ([verb: a] rp_2)}

24.5.5 Inference schemata of propositional calculus

1. $\dfrac{A, B}{\vdash\ A\&B}$ 2. $\dfrac{A \vee B, \neg A}{\vdash\ B}$ 3. $\dfrac{A \rightarrow B, A}{\vdash\ B}$ 4. $\dfrac{A \rightarrow B, \neg B}{\vdash\ \neg A}$

5. $\dfrac{A\&B}{\vdash\ A}$ 6. $\dfrac{A}{\vdash\ A \vee B}$ 7. $\dfrac{\neg A}{\vdash\ A \rightarrow B}$ 8. $\dfrac{\neg\neg A}{\vdash\ A}$

24.5.6 LA-rule for the propositional inference of conjunction

$$\text{inf1:}\quad \begin{bmatrix}\text{M-verb: } a\\ \text{prn: } m\end{bmatrix} \begin{bmatrix}\text{M-verb: } b\\ \text{prn: } n\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a\\ \text{prn: } m\\ \text{epr: } m\ and\ n\end{bmatrix} \begin{bmatrix}\text{M-verb: } b\\ \text{prn: } n\\ \text{epr: } m\ and\ n\end{bmatrix}$$

24.5.7 Coplets of an absolute proposition

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{be} \\ \text{NP: dog, animal} \\ \text{prn: abs327}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{dog} \\ \text{VERB: be} \\ \text{prn: abs327}\end{bmatrix} \quad \begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{animal} \\ \text{VERB: be} \\ \text{prn: abs327}\end{bmatrix}$$

24.5.8 Coplet of an episodic proposition

$$\begin{bmatrix}\text{I-concept}_{loc}\text{: } \textit{see} \\ \text{NP: Peter, dog} \\ \text{prn: 969}\end{bmatrix}$$

24.5.9 Inference rule inf2 for absolute propositions

$$\text{inf2:} \quad \begin{bmatrix}\text{M-verb: } a \\ \text{NP: } x\ b\ y \\ \text{prn: } n\end{bmatrix} \begin{bmatrix}\text{M-verb: } be \\ \text{NP: } b\ c \\ \text{prn: } abs\end{bmatrix} \Longrightarrow \begin{bmatrix}\text{M-verb: } a \\ \text{NP: } x\ c\ y \\ \text{prn: } n\end{bmatrix}$$