Basic Methods of Natural Language Processing

Cristina Vertan

University of Hamburg • Informatics Department

Natural Language Systems Group

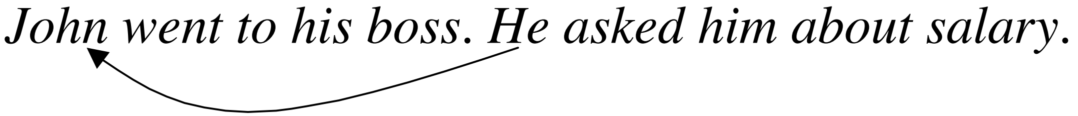
WWW: http://nats-www.informatik.uni-hamburg.de/~cri/

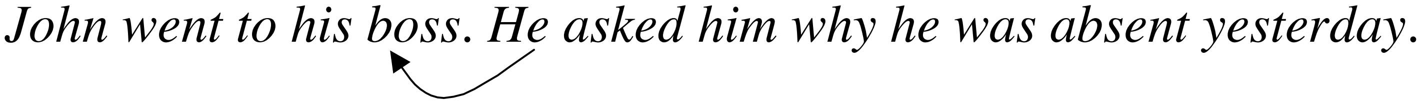
E-Mail: vertan@informatik.uni-hamburg.de

Typical features of NL

- Missing situational data
- Limited modal channels
- Limited technology
- New words/names
- Ambiguities at all levels of processing
- Non-deterministic processing

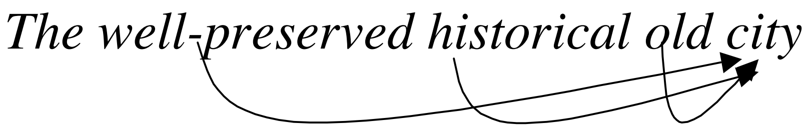
Typical Features of Natural Language -1-

- Unclear focus of analysis, esp. with spoken input (the whole text only?)
  
  – *John went to his boss. He asked him about salary.*

  
  – *John went to his boss. He asked him why he was absent yesterday.*
- Ambiguity on all levels
- Self-reference, meta language capacity (*I meant* ….)
- Valencies, i.e. syntactic/semantic co-occurrences of categories
  –*I give a party tonight.*
  –*I give you a present.*
- Multi-word lexemes and idioms with non-compositional meaning
  –*Give up, rain with cats and dogs*
- Hierarchical syntax in non-linear order
  –*The city which (I visited yesterday) was very interesting.*

Typical Features of Natural Language -2-

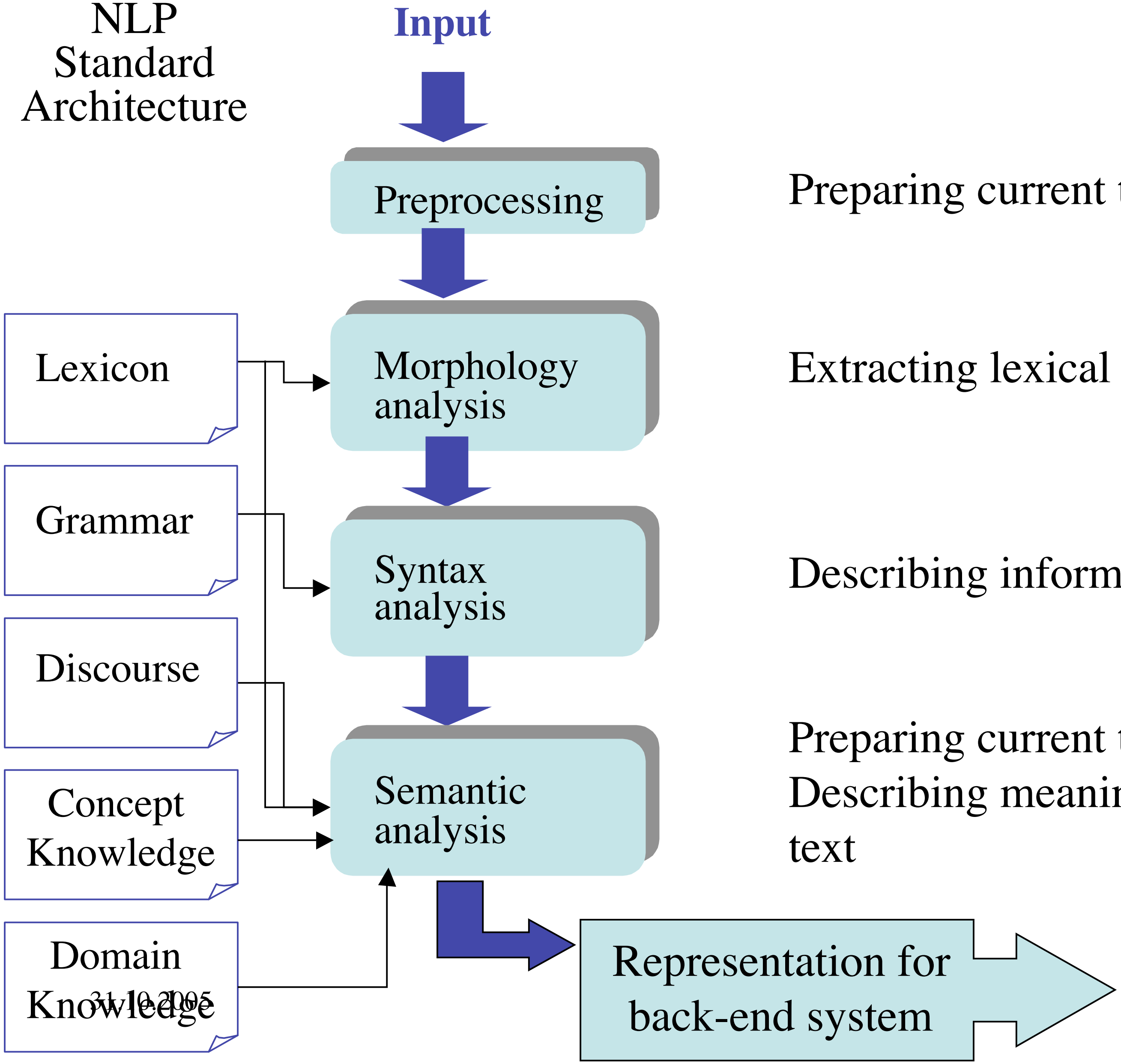
- Long distance dependencies



  - *The well-preserved historical old city*

- Discontinuous components (*ese …ahí (esp.), celui-la…..la-bàs*)
- Ellipses (*And this too*.)
- Paraphrases
- Coherence
- Understanding by word knowledge
  - *The dog attacked the man with black jacket*.

NLP Standard Architecture

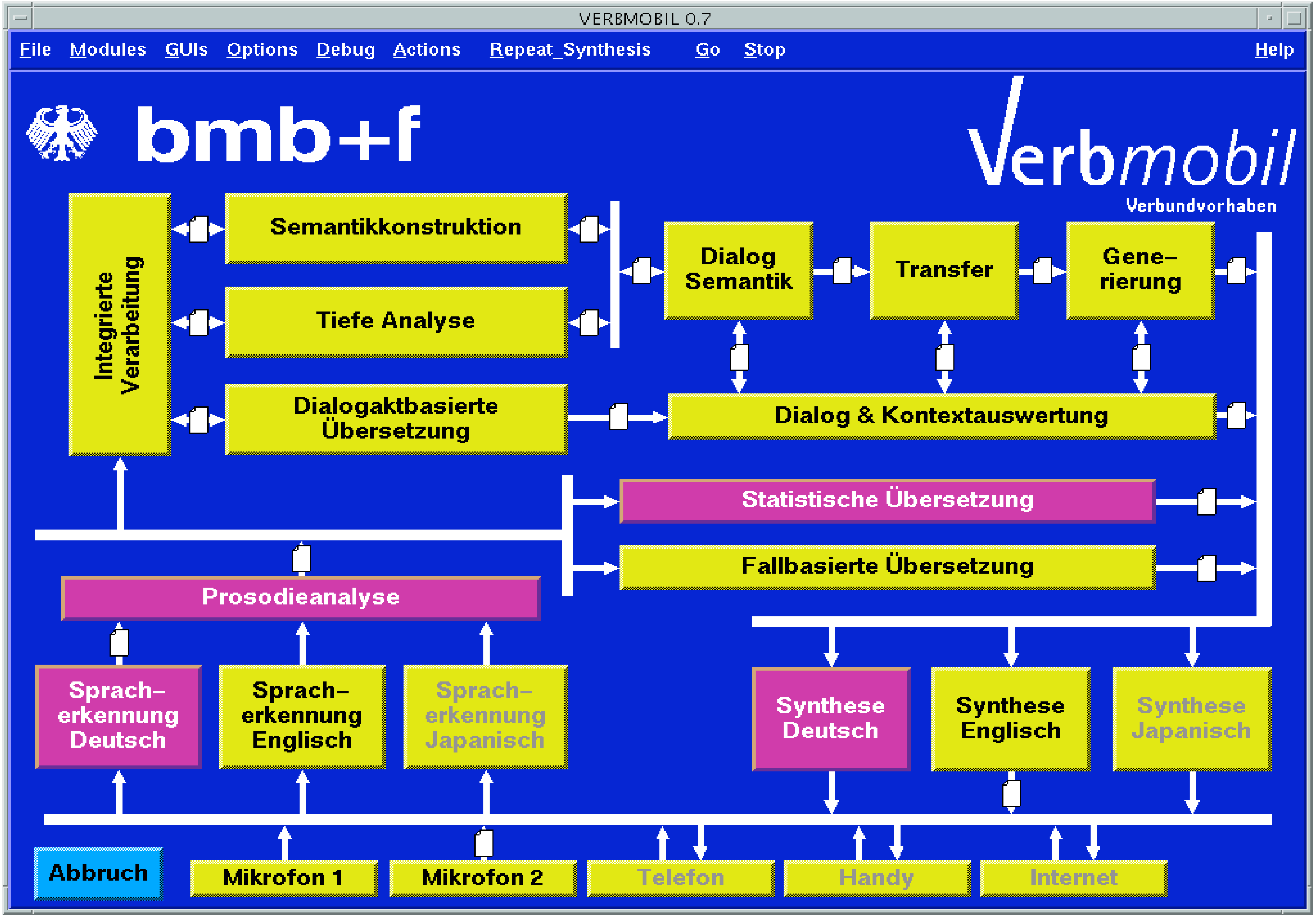


Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text


VERBMOBIL 0.7
File
Modules
GUIs
Options
Debug
Actions
Repeat_Synthesis
Go
Stop
Help
bmb+f
Verbmobil
Verbundvorhaben
Integrierte Verarbeitung
Semantikkonstruktion
Tiefe Analyse
Dialogaktbasierte Übersetzung
Dialog Semantik
Transfer
Gene-rierung
Dialog & Kontextauswertung
Statistische Übersetzung
Fallbasierte Übersetzung
Prosodieanalyse
Sprach-erkennung Deutsch
Sprach-erkennung Englisch
Sprach-erkennung Japanisch
Synthese Deutsch
Synthese Englisch
Synthese Japanisch
Abbruch
Mikrofon 1
Mikrofon 2
Telefon
Handy
Internet

Some Central Decisions of Analysis

- Which type of language is expected?
  - Spoken input may contain errors
  - Spoken language style (even in written transcripts) may have syntactic and semantic "errors"
  - Written language has no prosody, but is supposed to be correct
- Which type of output?
  - Table, representation expressions, slot filler, classification
- Which domain?
  - Technical, social, leisure,
- Which pragmatics?
  - Question answering, action control, information,

Result: GUS Semantic Slot Fillers

"I want to go to San Diego on May 28"

↓

```
(Client Declare
     (Case for want / e (Tense Present)
             Agent = Dialog.Client.Person
             Event = (Case for go (Tense Present)
                          Agent = Dialog.Client.Person
                          To-Place = (Case for City
                                                   Name = San Diego)
                          Date = (Case for Date
                                                   Month = May
                                                   Day = 28 ))))
```

Winograd

Result: DB-Interface Expression

"List the names of all suppliers, who deliver at least the parts that are delivered by supplier S2"

↓

```
SELECT          UNIQUE  S#
FROM            SP  SP X
WHERE           NOT EXISTS
                (SELECT *
                FROM SP SP Y
                WHERE S# = ´S2´
                AND             NOT EXISTS
                                (SELECT *
                                FROM SP
                                WHERE S# = SP X. S#
                                AND P# = SP Y. P#))
```

Result: Table of chemical reactions in SIE

REF	SCALE	PHASE	YIELD	TEMP
para.1	small	solid	77%	-78 to 20
TIME	ENERGY	APPARATUS	FEATURES	
	cooling		IR. NMR. MS	
REG. NO	FUNCTION	AMT.	AUTHOR ID	
78624-62-1	product	2.70 g	7a	
78624-61-0	reactant		6a	
13274-48-6	reactant	1.24 g	N-methyltriazolinedione	
	solvent	80 ml	pentane	
	solvent	40 ml	ethyl acetate	

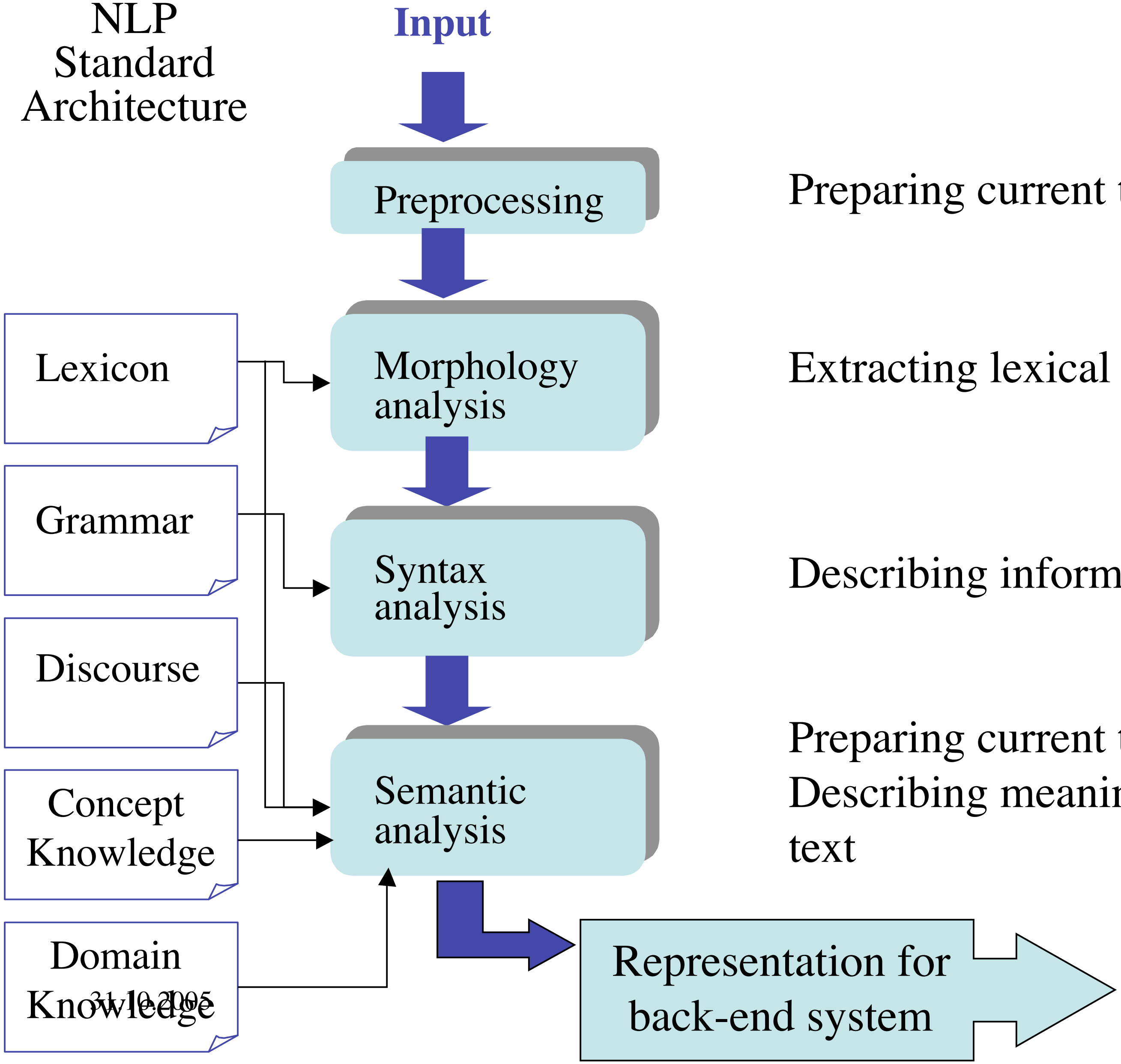
Result: Semantic Representation

"who is the man that leads the LOKI project?”

```
[   request (referent(_5747))
    presuppose (exists (_4340))   ]
    some (_4340)
[  unique (_4407)
   single (_4407)
   instance (_4407, person)
   propval (person,_4407,sex,male)
   [  some (_4725)
     [  unique (_5033)
        single (_5033)
        instance (_5033,project)
        propval (project,_5033,name,str
     (LOKI)) ]
```

```
      instance (_4725,leading)
         propval
   (leading,_4725,theta,_5033)
         propval
   (leading,_4725,alpha,_4407)
         topic (_4407)   ]     ]
   instance (_4340,identity)
   propval (identity,_4340,alpha,_4407)
   propval (identity,_4340,theta,_5747)
   topic (_4407)
   [   some (_5747)
     single (_5747)
     instance (_5747,person) ]
```

NLP Standard Architecture



Input

Preprocessing

Lexicon

Grammar

Discourse

Concept Knowledge

Domain Knowledge

Morphology analysis

Syntax analysis

Semantic analysis

Representation for back-end system

Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

First Step: Preprocessing

- Separate text from non-text (images, code, analyze tables ...)
- Lemmatization (Splitting)
- Normalize writing (e.g. Ablaut)

Lemmatizer

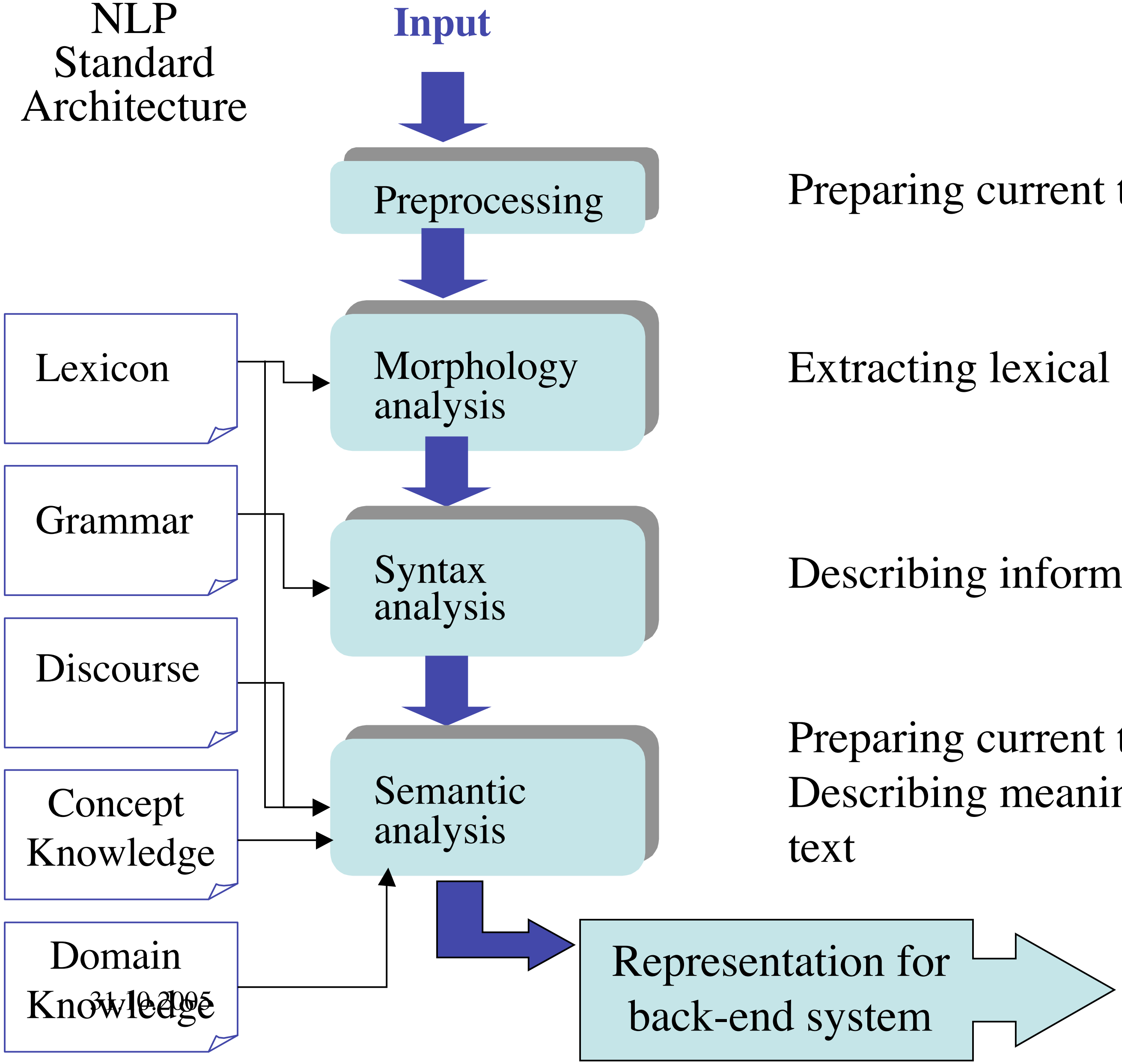
- Join separable suffixes (esp. in German)
  - "Er fing die Maus ein" ⇒ *einfangen*
  - *(*He caught the mouse in)*
- Separation of compounds

Tokenizer

- Block multiword terms and idioms,

- (Attach PoS PoS-Tagger)

NLP Standard Architecture



Input

Preprocessing

Morphology analysis

Syntax analysis

Semantic analysis

Representation for back-end system

Lexicon

Grammar

Discourse

Concept Knowledge

Domain Knowledge

Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

Second Step: Consulting the Lexicon

- Full form lexicon
  - Every form of a word is an entry in the Lexicon
  - Result: no morphological processes after tokenizing
  - For real life applications sometimes too large
  - Difficult for languages with strong composition

- Stem lexicon
  - Only stems are entries
  - Additional information about inflexion class
  - a morphological generator is necessary
  - Result: small resources

All further processes, except for names,
rely on lexicon information, at least on part of speech tags

Resources:
Lexicon

- **Dictionary**
  - ~~Pronunciation~~
  - ~~Definitions~~
  - ~~etymological information~~
  - ~~stylistic information~~
  - PoS (part of speech)
  - Few sub-classification features (usually gender, plr.)
  - Translation (in bi- or multilingual dictionaries)

- **Lexicon**
  - PoS (noun, Verb, etc.)
  - Sub-classification features (verb transitive/intransitive, Genus etc.)
  - Inflexion classes
  - semantic information (e.g. if a verb requires an alive Subject)
  - a link to translation equivalents in other lexicons, or a mark for lexical gap (in bi- or multilingual dictionaries)

Resources: Lexicon

Representation and Encoding - 1 -

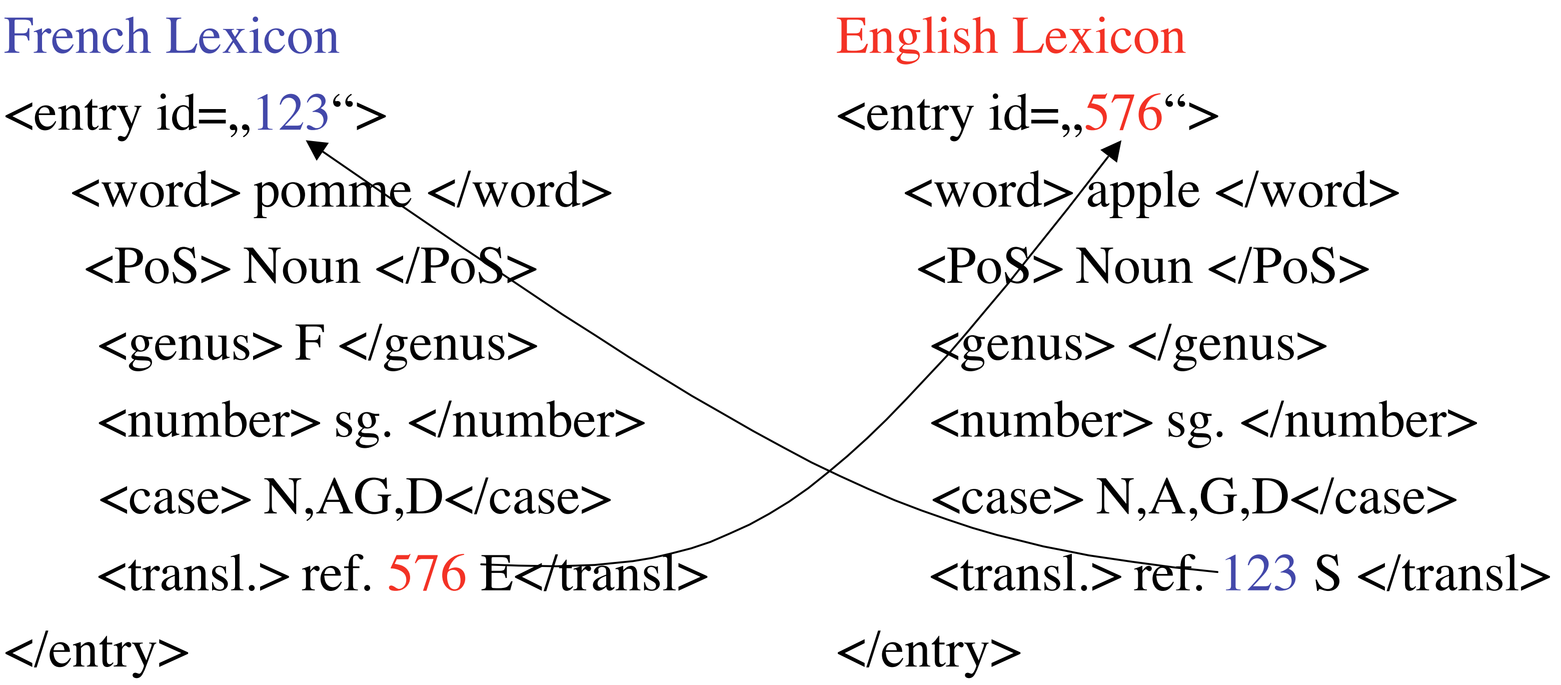
- There is a huge number of lexicon formats according to:
  - the encoded linguistic information (which features, in which order)
  - the encoding schema (distributed lexicon, delimiters between linguistic categories or entries, pointer to entries in other lexicons)
- The lexicon design is an extremely time consuming process, therefore „re-usability“ has high priority on the agenda of lexicon developers,,
- Several standard models have been proposed (PAROLE / SIMPLE, MILE) as well as standard encoding schemas based on XML (SALT, OLIF)

Resources: Lexicon

Representation and Encoding - 2 -

- Most existing standard models are very complicated because they intend to cover a large spectrum of linguistic features but
  - they still did not succeed to model all linguistic phenomena of the European languages,
  - Existing lexicons, which do not follow these standards, cannot be re-used.
  - MANAGELEX (Univ. Hamburg) is a tool for reformatting, editing, developing and merging of lexicons (under development)

Bilingual (Multilingual) Lexicon Example



French Lexicon

```
<entry id=„123“>
   <word> pomme </word>
    <PoS> Noun </PoS>
     <genus> F </genus>
     <number> sg. </number>
     <case> N,AG,D</case>
     <transl.> ref. 576 E</transl>
</entry>
```

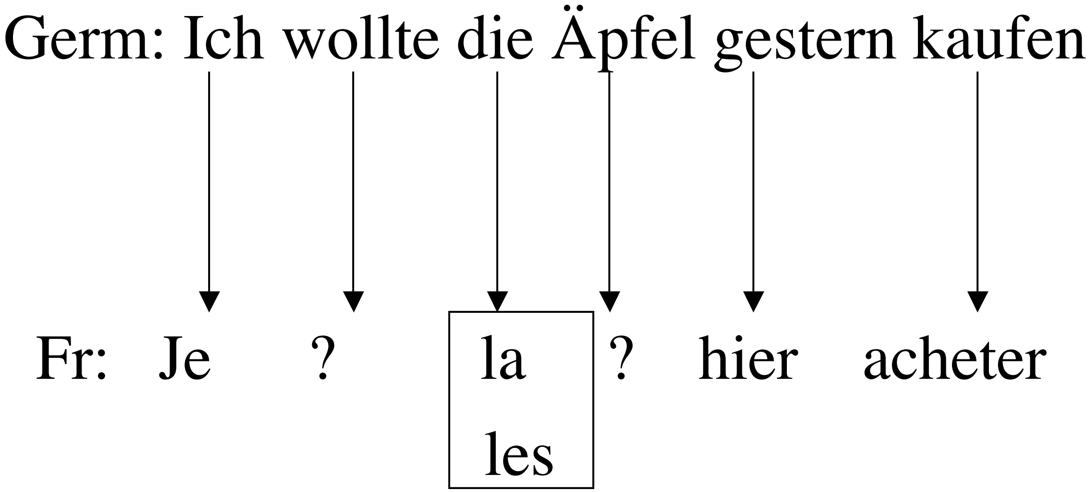
English Lexicon

```
<entry id=„576“>
   <word> apple </word>
    <PoS> Noun </PoS>
     <genus> </genus>
     <number> sg. </number>
     <case> N,A,G,D</case>
     <transl.> ref. 123 S </transl>
</entry>
```

Thesauri

- Are a particular form of lexicons and contain fixed expressions (and their translations)
- Expressions contained in such thesauri are replaced from the very beginning (in particular by their translations), and are no longer object of syntactic or semantic interpretations
- E.g.:
  - *United States = Statele Unite*
  - *Civil law = Cod Civil*
- Sometimes abbreviations are also part of thesauri:
- E.g.:
  - *Dvs.= Dumneavoastr_ =You (politeness)*
- Thesauri are domain specific

Morphological analysis -1-



Germ: Ich wollte die Äpfel gestern kaufen

Fr: Je ? la les ? hier acheter

Informations about:

Inflection: Apfel (sg. masc) inflection class N23

wollen (present)

Declination: Äpfel (acusative, pl.)

Conjugation: Ich wollte, etc.

Stem lexicon

Ich = Je

die = la, les

gestern = hier

kaufen = acheter

Morphology-representation

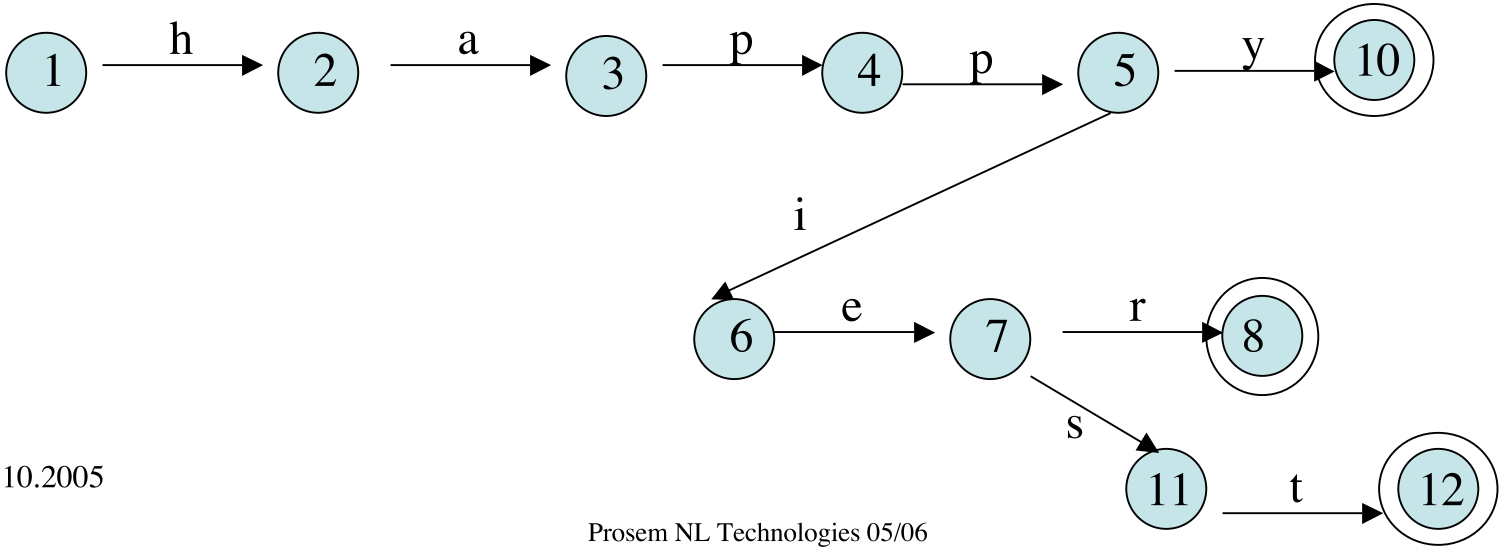
- Rules:

(lex=V, cat=v,+finite, person=3rd, number=sing, tense=pres) ↔ V+s

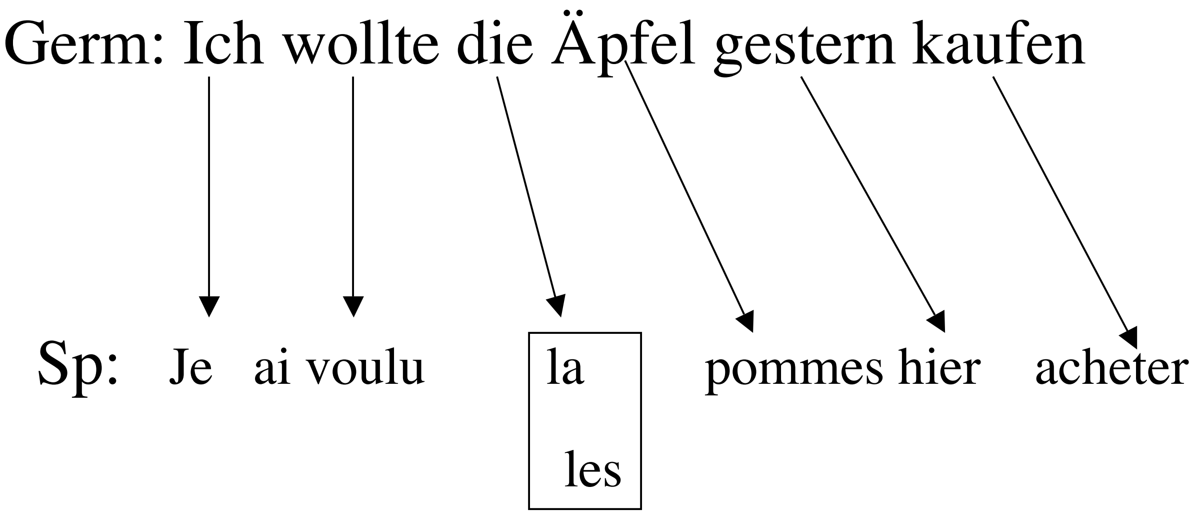
Exception

(lex=be, cat=v,+finite, person=3rd, number=sing, tense=pres) ↔ is

- Finite State Transduers (FST)



Morphological analysis -2-



Germ: Ich wollte die Äpfel gestern kaufen

Sp: Je ai voulu la les pommes hier acheter

Full-form lexicon

Langenscheidts Universal-Wörterbuch

French

Ich = Je

wollte = voulu

die = la, les

Äpfel = pommes

gestern = hier

kaufen = acheter

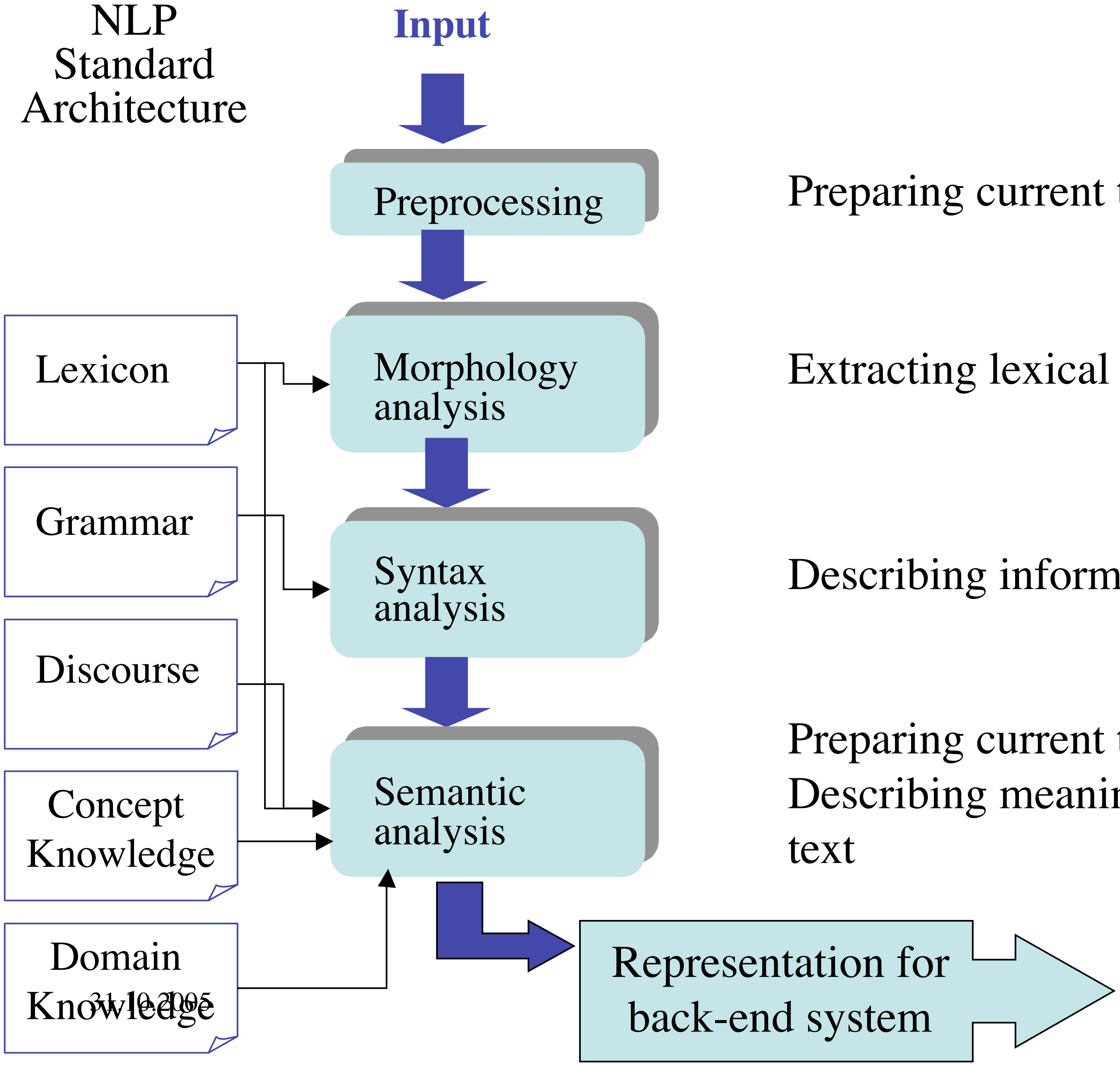
Limitations of morphological analysis - 1-

- From the previous example: after the morphological analysis the translation would be:
  - *J ‘ai voulu la/les pommes hier acheter*
- 2 Problems:
  - no correct word-order
  - Ambiguity when translating „die“
- The word-order can be solved by introducing transfer rules: e.g. the verb has to be moved from the last position (according to the German order) near the auxiliary (according to the French order). But not all such changes can be defined by rules.

Limitations of morphological analysis -2-

- Lexical Ambiguity:
  - Categorial ambiguity: the same word can belong to more than one PoS E.g. *last* (engl.):
    - *Verb*: The show lasts 2 hours
    - *Adjective*: last time
    - *Adverb*: He is the last
  - Homography and Polysemy (the same word has more meanings) e.g. Bank (engl.) capital (sp.)
  - Translation ambiguity: e.g. the English *leg* can be translated in Spanish with *pierna (human), pata (animal, table), pie(chair), etapa (of a journey)*
- Structural ambiguity : *la pommes* or *les pommes*, or complicated syntactical problems

NLP
Standard
Architecture



Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

Pattern matching - 1 -

- Pattern = a syntactic frame for lexical-semantic equivalence classes.
- Patterns describe frequent expressions of a language

Wo is president of Germany ?

31.10.2005

Pattern matching - 2 -

- Patterns specify
  - A fixed word order
  - Places for variable words
  - A clause can be processed, if there is (at least) one matching pattern
- Patterns can be filled by categorical, syntactic or semantic constraints

  " _any word_" / "_any noun_" / "_any noun+sing_" / "_any noun with attr +living_" / "any noun ⊃ class MUSHROOMS_"
- For one sentence more than one pattern may be applied

Problems: syntactically or semantically deviant sentences can be accepted, if the equivalence classes are not defined tightly:

- z. B: Who is the pencils of football league ?

Maintenance problems with large number of (partly overlapping) patterns

➡ Patterns do not deliver a symbolic description

Basic Syntactic Decisions

• **System type**	Parser&Grammar	vs	separate grammar modules
• **Result of parser**	Full	vs	selective
• **Type of grammar**	Dependency	vs	constituency
• **Formalism**	any sort		
• **Architecture**	parallel	vs	sequential
• **Start point**	top-down	vs	bottom-up
• **Rule application**	deterministic	vs	non-deterministic
• **Rule choice**	first guess	vs	informed choice
• **Strategy**	breadth first	vs	depth first
• **Scope**	word by word	vs	phrase by phrase
• **Ambiguity handling**	until success	vs	exhaustive

Syntactic analysis

- The output of the morphological analysis is parsed according to the chosen grammar
- i.e: parsing of a sequence of PoS symbols (retrieved by the morphological analysis)
  - the correct order of the PoS is proved
  - Iteratively a structural description is written into a data structure
  - `<Structure= Art + N + .....>`
- Very often a part of the input is abandoned for the moment because substructures have to be analyzed first.
- e.g *The books, which we bought yesterday are very interesting*

```
    <Art + N+ <Colon + sub-clause...> + V +Mod+Adj
                <Rel +Pron +V+Adv>
```

Resources: Grammars

- Grammars define the conditions of well-formed expressions in a language (syntax)
- Describe three basic relationships in sentences:
  – *sequence of words* (in English adjectives normally precede the nouns that they modify, whereas for e.g. in Spanish they normally follow it)
  – *categories*: e.g. a noun phrase may consist of a determiner and a noun or a determiner, an adjective and a noun.
  – *dependency* i.e. relations between categories: prepositions determine the case of the nouns which depend on them: e.g. „*mit*“ (germ.) „*con*“ (sp.) always require dative

Resources:Grammars

Grammar types

- Two basic types of grammatical representations are in common use, dependency grammars or constituency grammars.
- Sequence is optionally indicated in both types of representations
- Dependency relations are represented
  - in a dependency grammar by a word tree starting with the verb
  - In a constituency grammar by a tree of constituents
- Categories are explicitly represented in a phrase structure tree, in s dependency grammar categories are subtrees

Resources: Grammars

- Example -

S→ NP VP

NP→N

NP→AdjP NP

NP→Det NP

AdjP→Mod Adj

VP→VNP

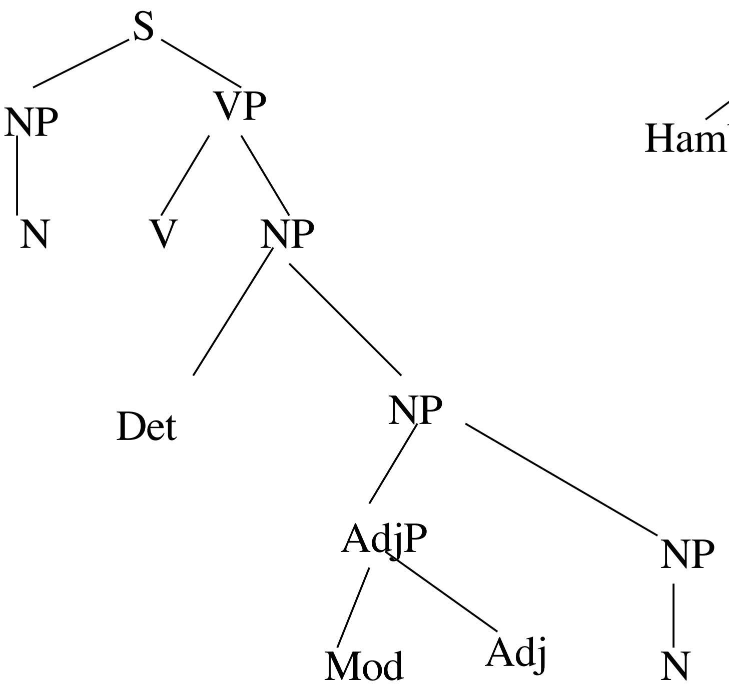
N→Hamburg, Stadt

V→ ist

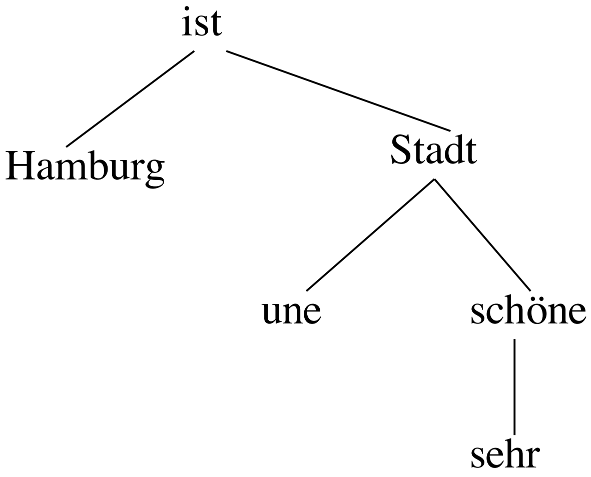
Det →eine

Mod→ sehr

Adj→schöne



Constituent structure



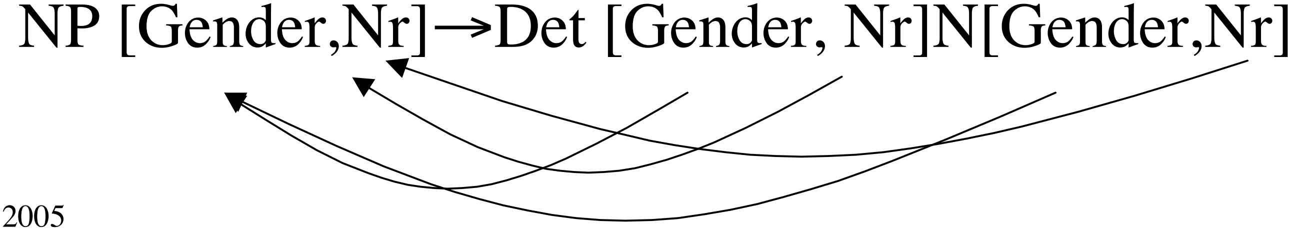
Dependency structure

Hamburg ist eine sehr schöne Stadt

Resources: Grammars

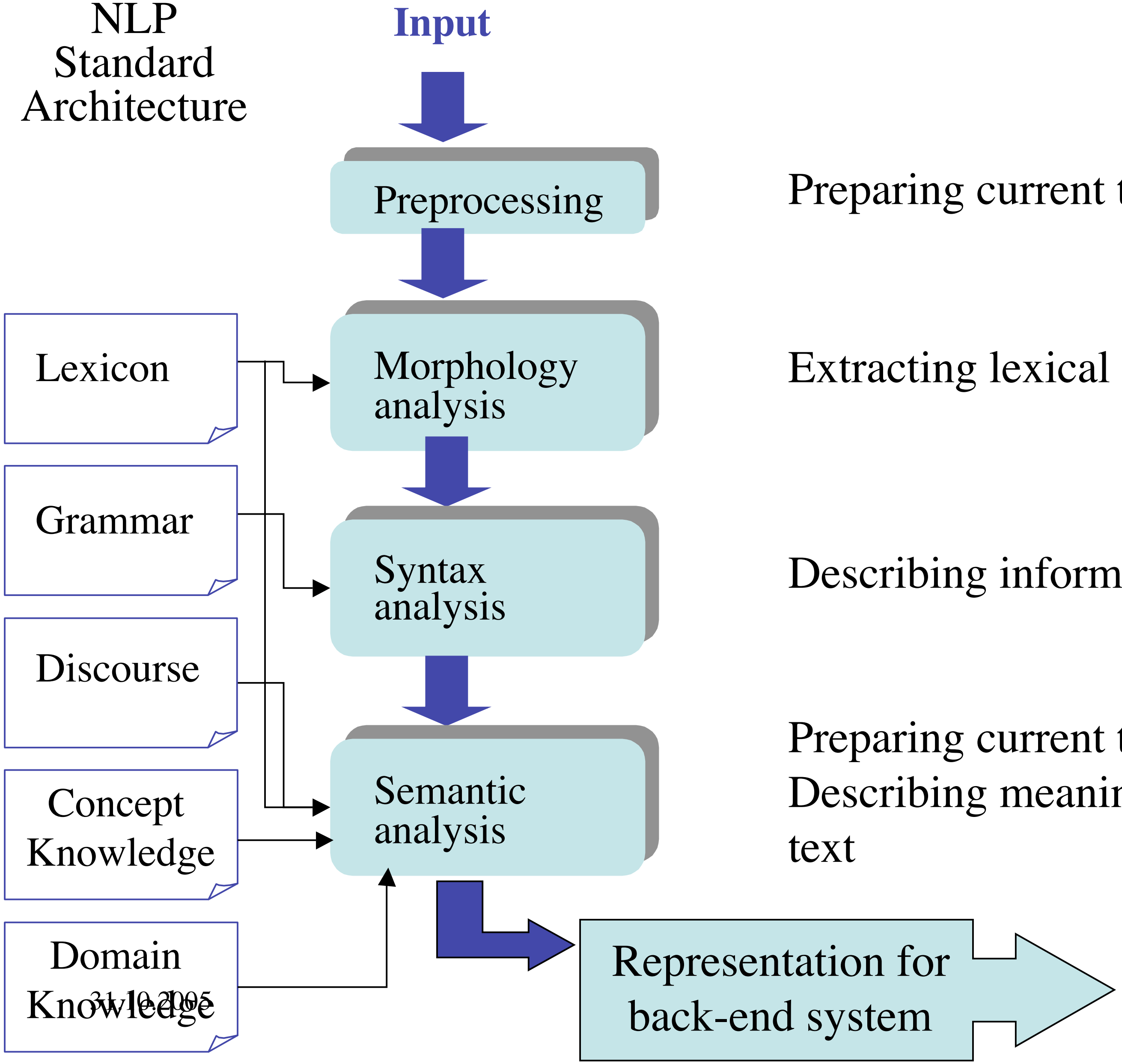
Feature -based representations

- To shift grammatical features to higher nodes or inherit features from them
- Linguistic features are represented as attribute-value-pairs
- Additionally, rules for combining features must be specified (e.g.for correspondence)
- Features can be inserted both in constituent and dependency structures
- E.g



NP [Gender,Nr]→Det [Gender, Nr]N[Gender,Nr]

NLP
Standard
Architecture



Input

Preprocessing

Lexicon

Grammar

Discourse

Concept Knowledge

Domain Knowledge

Morphology analysis

Syntax analysis

Semantic analysis

Representation for back-end system

Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

Basic Semantic Decisions

- Semantic inside the parser? ⇒ semantic parser / case grammar
- Separation of construction, resolution and evaluation? ⇒ Multi-phase processing vs extraction
- Frame oriented processing or compositional treatment
- Domain knowledge in semantics? ⇒ Reference semantics
- Conceptual knowledge separated from facts? ⇒ interaction
- User specific interpretation? ⇒ Partner model
- Time-dependent? ⇒ time logic

Semantic Strategies

- Elementary: Key word spotting
- Basic: Syntax looks only for semantic slot fillers
- Technical solution: The parser delivers already a semantic structure by looking for semantic roles and dependencies only
- Standard: Lexical semantic entries are amalgamated with parsing result ← **Semantic case grammar**

- Advanced: A full logical representation of the proposition and presupposition is built up

Semantic Roles /Deep Case Semantics

Based on lexical semantics and syntax, sentence semantics delivers the semantic **potential** of an utterance

Often used: Semantic Roles (Deep Cases):

- Actor
- Instrument
- Object, etc.

specify "persons and items" of a sentence,

E.g., in an action:

An <Actor> moves an <Object> from a <Location1> to a <Location2> along a <Path> for a <Beneficiary>.

Grammarians propose up to 25 roles. Specific domains may have a very limited number of roles (e.g. weather reports)

Semantic Resolution

Define the current meaning by

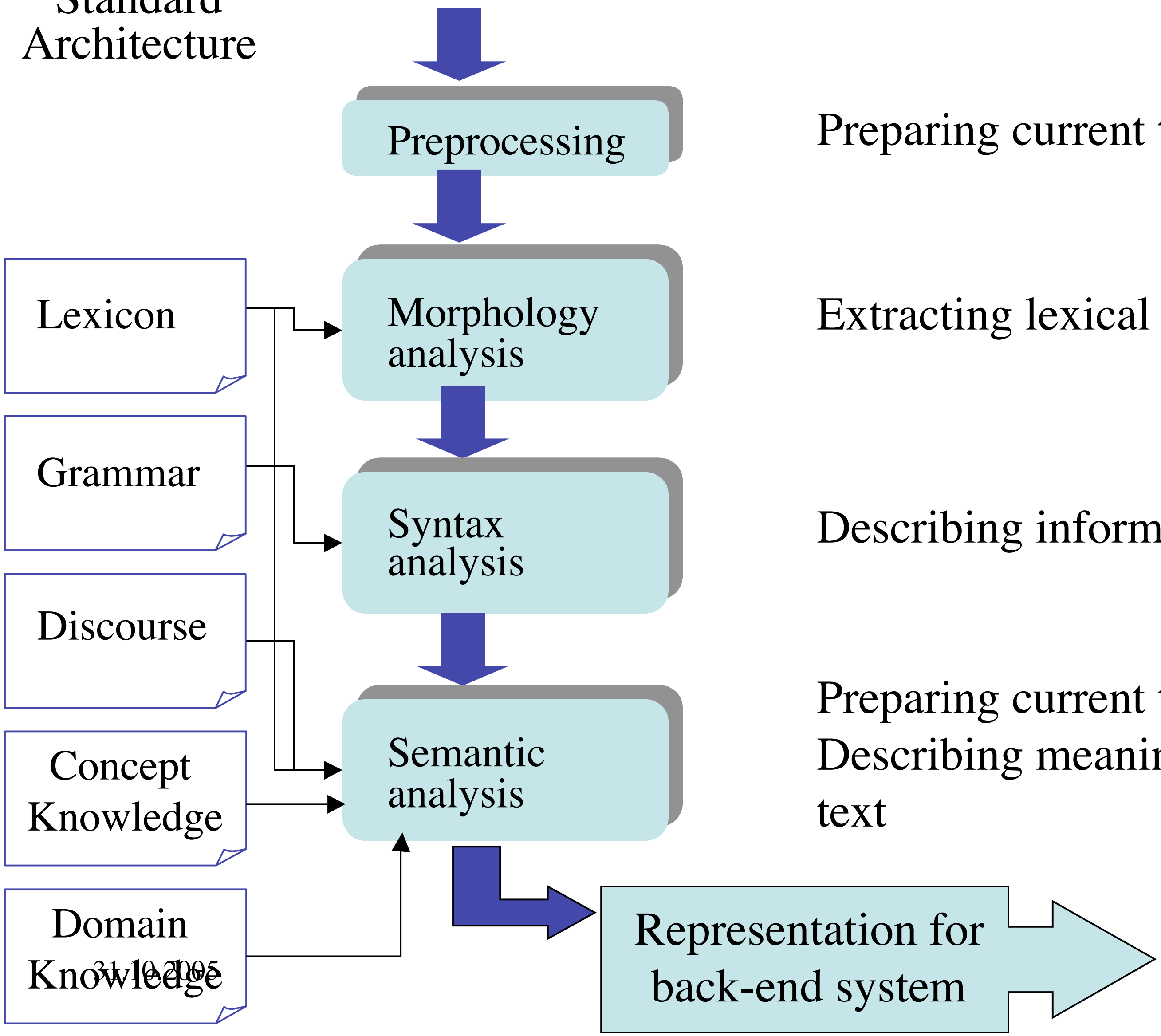
- Using contextual knowledge:
  - Determination of current values
  - Disambiguation

Discourse Semantics

- Using domain and every-day knowledge:
  - Identification of referred objects
  - domain data and state of affairs
  - Determination of relevant utterances
  - User specific inferences

Reference Semantics

NLP
Standard
Architecture



Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

Resources: World Knowledge and Domain Knowledge

- Concepts and relations between concepts are represented in an ontology (semantic feature hierarchy)
- The lexemes (lexicon entries) in different languages can be mapped onto this ontology (details in knowledge-based MT)
- Sensors or time dependent expressions describe the state of affair (speaker t_1 = Antonio = "I")
- Usually this information is language independent.

Ambiguities on all levels

The central difference between formal and natural languages is the ambiguity. E.g.,

- Speech ambiguity "*Lead a ship*" *vs.* " *leadership*"
  "*peak*" *vs.* "*peek*"
- Lexical ambiguity *“Drive to the bank, please!”*
- Syntactic ambiguity *“I saw the Grand Canon flying to New York”*
- Pragmatic ambiguity *“Can I print some reports?”*
- Referential ambiguity *“He took some papers out of the envelopes and send them to his boss*

Discourse Coherence

So far, the scope of analysis was the sentence. However, many syntactic and semantic structures are super-segmental, especially in spoken language and in spoken style.

Important tasks:	Anaphora resolution	*”Take the cake from the fridge and eat it*"
	Cataphorics	"*John did the following:*
	Ellipses	"*and those for 2005?* "

Pronouns can often be resolved (replaced by their antecedent) by searching for an adjacent noun in the previous sentence, which has the same grammatical and semantic features (role restrictions of the verb)

Cataphora-phenomens are much more difficult, but they are rare.

Ellipses can be completed by testing unification with previous sentences

Resources:
Discourse memory

Minimal case:

- List of before-mentioned objects with gender information (for easy pronoun resolution)

Best case

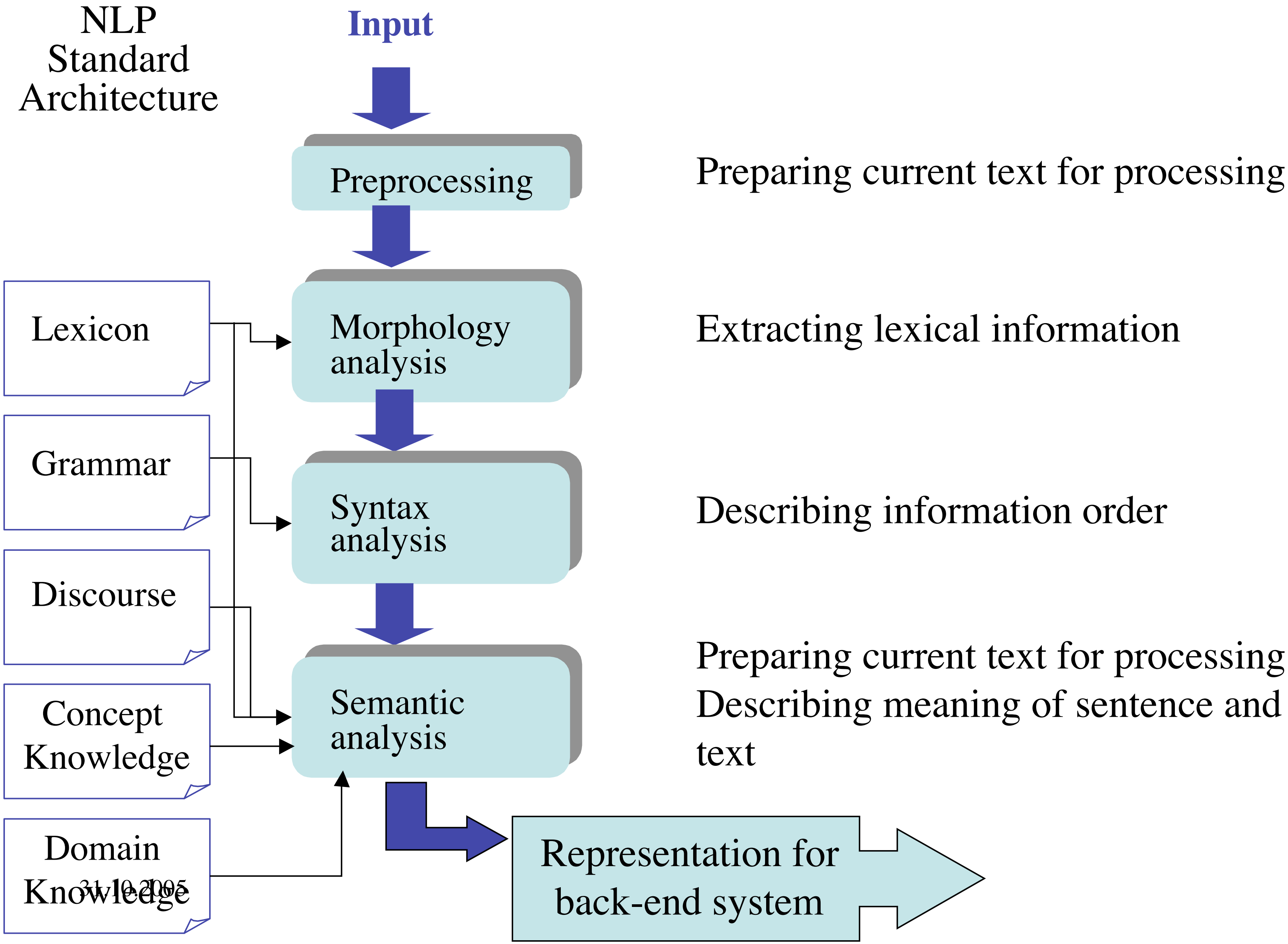
- List of before-mentioned objects with gender information and semantic features (for elaborated pronoun resolution)
- Memory of syntactic structures (for ellipses reconstruction)
- Memory of propositions (for ellipses reconstruction)
- Memory of speech acts (for pragmatically adequate reactions)

Pragmatics reasoning

- Pragmatics is the linguistic field, which describes relations between language and action (planning). E.g.:
- Whenever I say "*can you do X?*" I assume, that
    - ➢ it is an order, not a yes/no-question
    - ➢ I want X
    - ➢ X can be done
    - ➢ I want it to be done immediately
    - ➢ I expect a rejection in case of disagreement
    - ➢ I am responsible for X, etc

All pragmatic presuppositions and implications must be fulfilled in the discourse. This is especially important with legal and economic translation. Counterexample:

~~"Can you ship these trains?" → "yes, we do it immediately!"~~



NLP Standard Architecture

Preparing current text for processing

Extracting lexical information

Describing information order

Preparing current text for processing
Describing meaning of sentence and text

Generation

- Elementary solution in simple domains: Output patterns with the result values:
- "*Who is the president of the UN* ⇒
- "*Kofi Annan*"
- Better solution: The syntax for the answer is extracted from the question Processes:
  - Consider topicalization
  - Adjust word order
  - Omit trivial parts to be not too repetitive
- Best solution: Evaluation renders propositions (predicate-argument structures) and adequate grammar rules allow for free generation.

Thank you!