# Large Spoken Language Dialogue Systems: Verbmobil & SmartKom

Tilman Becker
DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
becker@dfki.de

http://verbmobil.dfki.de http://www.smartkom.org



# Overview

- Speech-to-speech translation: Verbmobil
- Multi-Modal Man-Machine Interaction: SmartKom
- Zooming in: Natural Language Generation



# Content

- **Overview of Verbmobil**
- **A walk through the system**
  - Acoustic Processing
  - Dialog Translation
  - Selection and Speech Synthesis
- **Technical issues**
- **Human Factors and Experiences**



# Overview of Verbmobil

## Challenges, Partners, and General Approaches

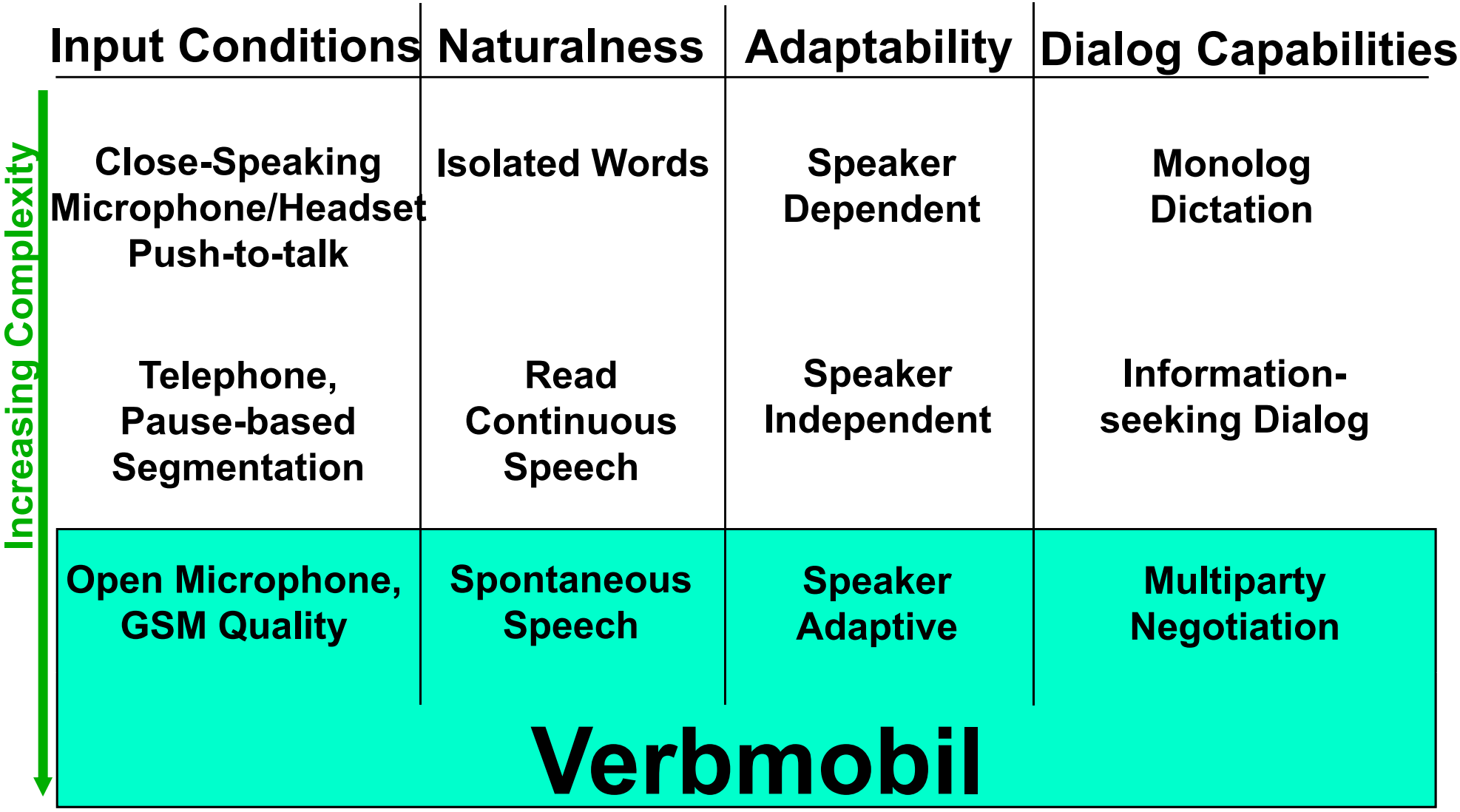


# What is Verbmobil?

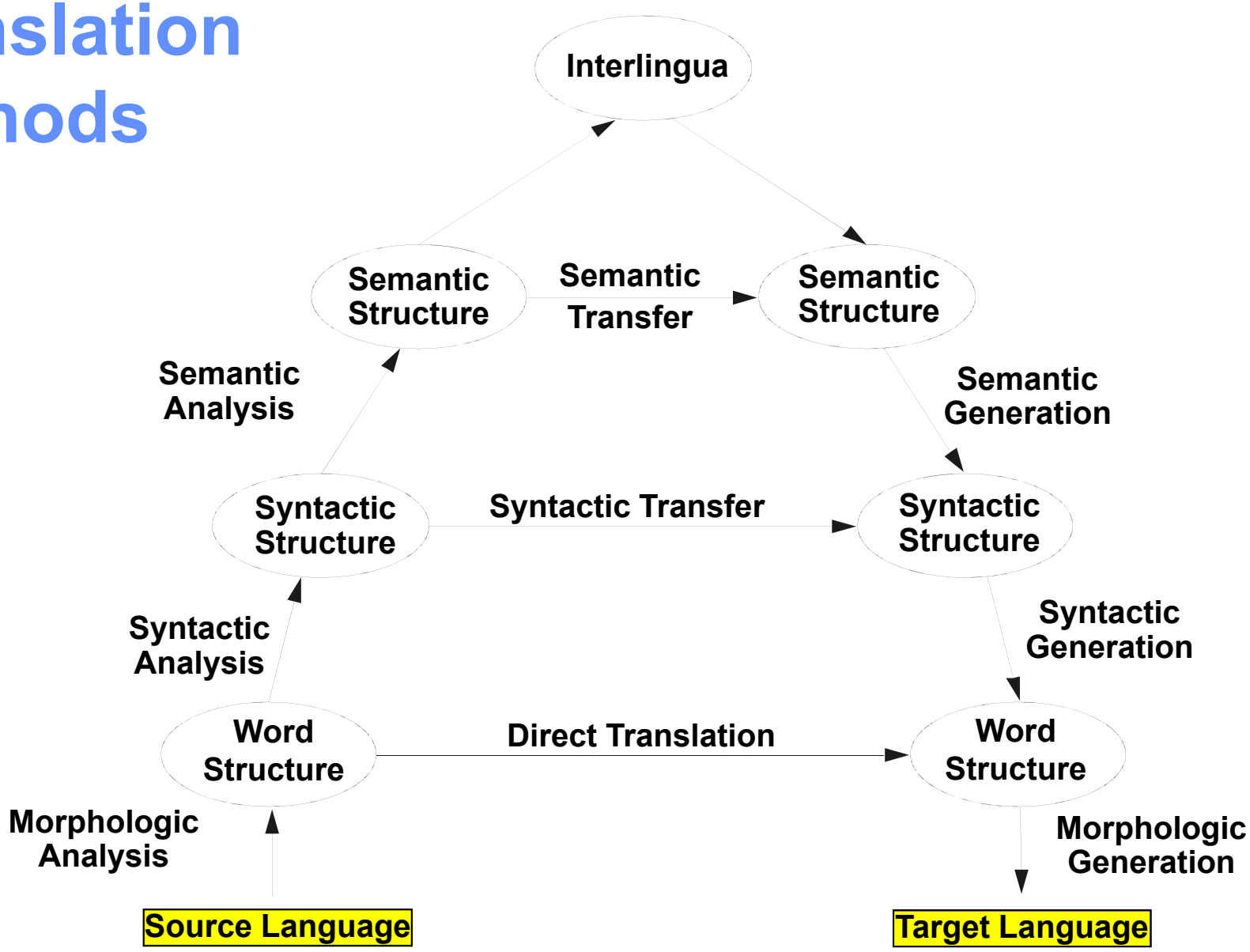
- Speech-to-speech translation system
- Robust processing of spontaneous dialogs
- Speaker independent (adaptive)
- Languages: English, German, Japanese
- Domains: Appointment scheduling, travel planning and hotel reservation, remote PC maintenance
- The system mediates between two humans, it does not play an active role
- There is no control of the ongoing dialog by the system



# Challenges for Language Engineering

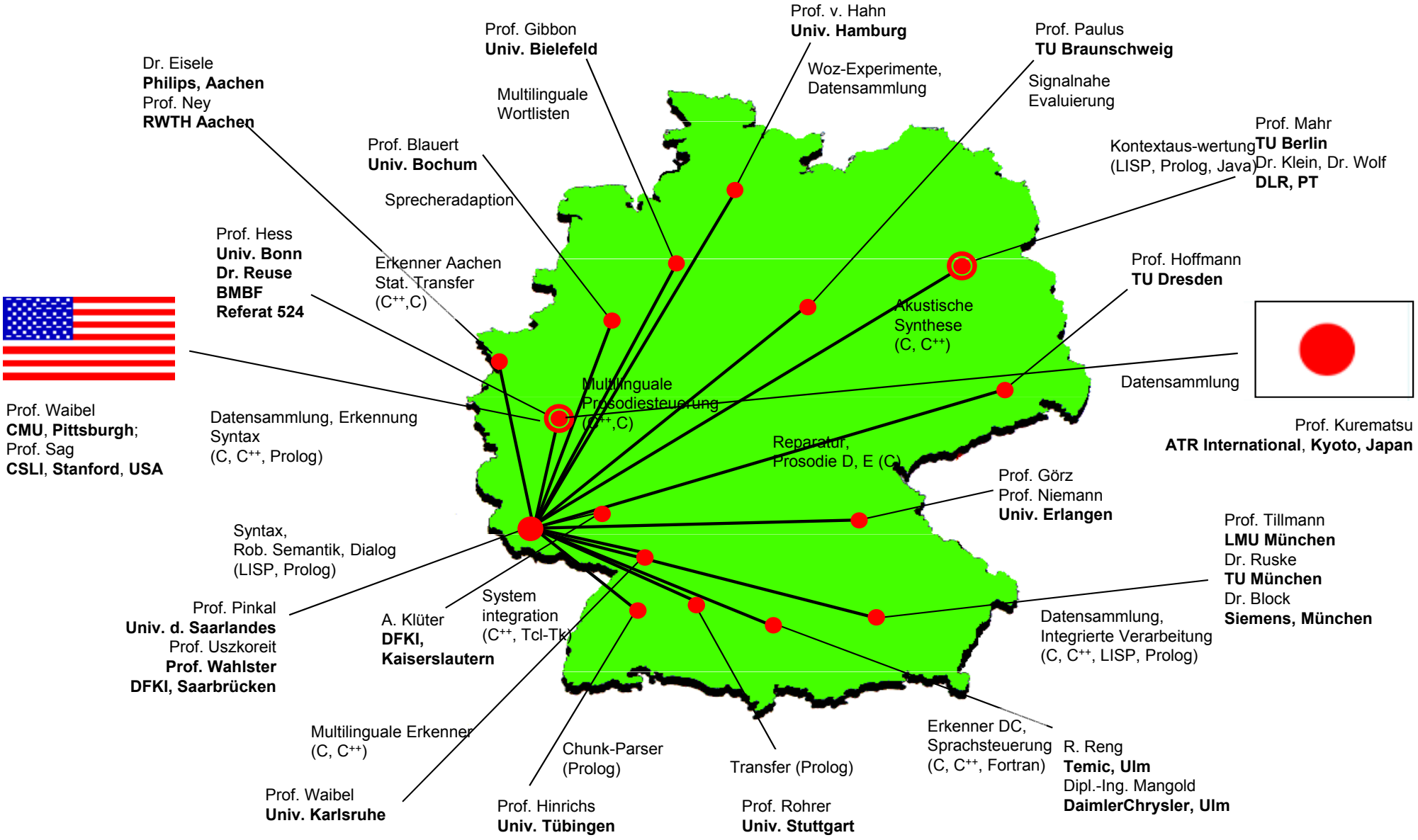


# Classification of Machine Translation Methods



# The Verbmobil Partners

# The Verbmobil Partners



# Facts About the Project

- **23 participating institutions (in Verbmobil II), from Germany and the USA**
- **Over 900 full-time employees and students involved over the whole duration**
- **Funded by the German Ministry for Education and Science and the participating companies:**

| | | |
|---|---|---|
| BMBF-Funding Phase I, 1.01.93 – 31.12.96 | 62.7 Mio. DM | 31.6 Mio € |
| BMBF-Funding Phase II, 1.01.97 - 30.9.2000 | 53.3 Mio. DM | 27 Mio € |
| Industrial investment I+II | 32.6 Mio. DM | 16.5 Mio € |
| Related industrial R & D activities | ca. 20 Mio. DM | ca. 10 Mio € |
| **Total** | 168.6 Mio. DM | 85.1 Mio € |

# Verbmobil – The Book

**There are over 600 refereed papers on the various aspects of and achievements in Verbmobil.**

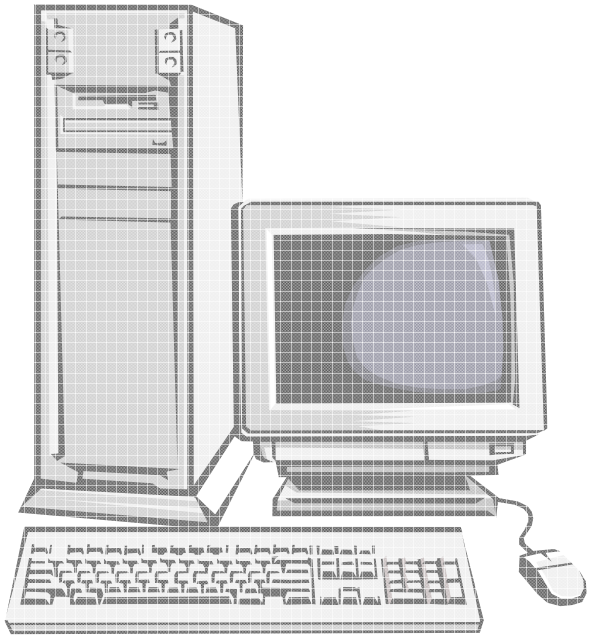
**Wolfgang Wahlster (ed.):**

***"Verbmobil: Foundations of Speech-to-Speech Translation"***

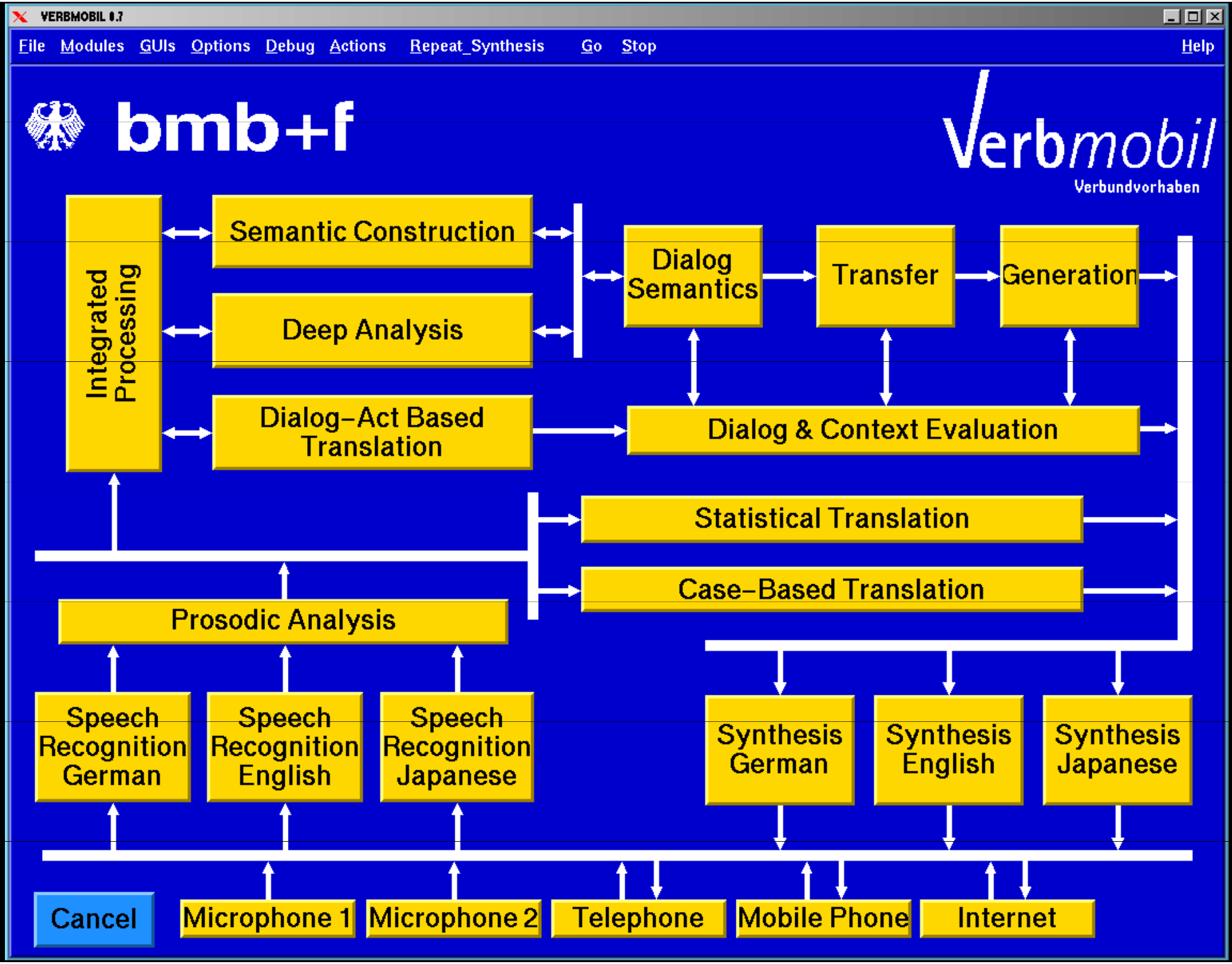

**Springer-Verlag Berlin Heidelberg New York. 679 Pages**

**ISBN 3-540-67783-6**

# Typical Verbmobil Hardware



- SUN Ultra-Sparc 80
- 4 processors (450 MHz)
- 2 GB main memory
- 8 GB swap
- no special signal processing hardware
- Desklab Gradient A/D converter or Sun internal audio device
- close-speaking cordless microphones


VERBMOBIL 0.7
File Modules GUIs Options Debug Actions Repeat_Synthesis Go Stop Help
bmb+f
Verbmobil
Verbundvorhaben
Integrated Processing
Semantic Construction
Deep Analysis
Dialog–Act Based Translation
Dialog Semantics
Transfer
Generation
Dialog & Context Evaluation
Statistical Translation
Case–Based Translation
Prosodic Analysis
Speech Recognition German
Speech Recognition English
Speech Recognition Japanese
Synthesis German
Synthesis English
Synthesis Japanese
Cancel
Microphone 1
Microphone 2
Telephone
Mobile Phone
Internet

# Walk Through the Verbmobil System

## Detailed Module Presentation and Demonstration



# Acoustic Processing



# Recording, Synthesizing and Synchronization

- Task:

  Providing a uniform interface to varying audio hardware; synchronizing in- and output
- Input:

  Audio data and system states
- Method:

  Introducing audio modules; Finite State Machine for synchronizing
- Result:

  Audio Data and Synchronization
- Benefit:

  Encapsulating audio hardware, “open microphone”, preventing out-of-sync or overlapping system output
- Responsible:

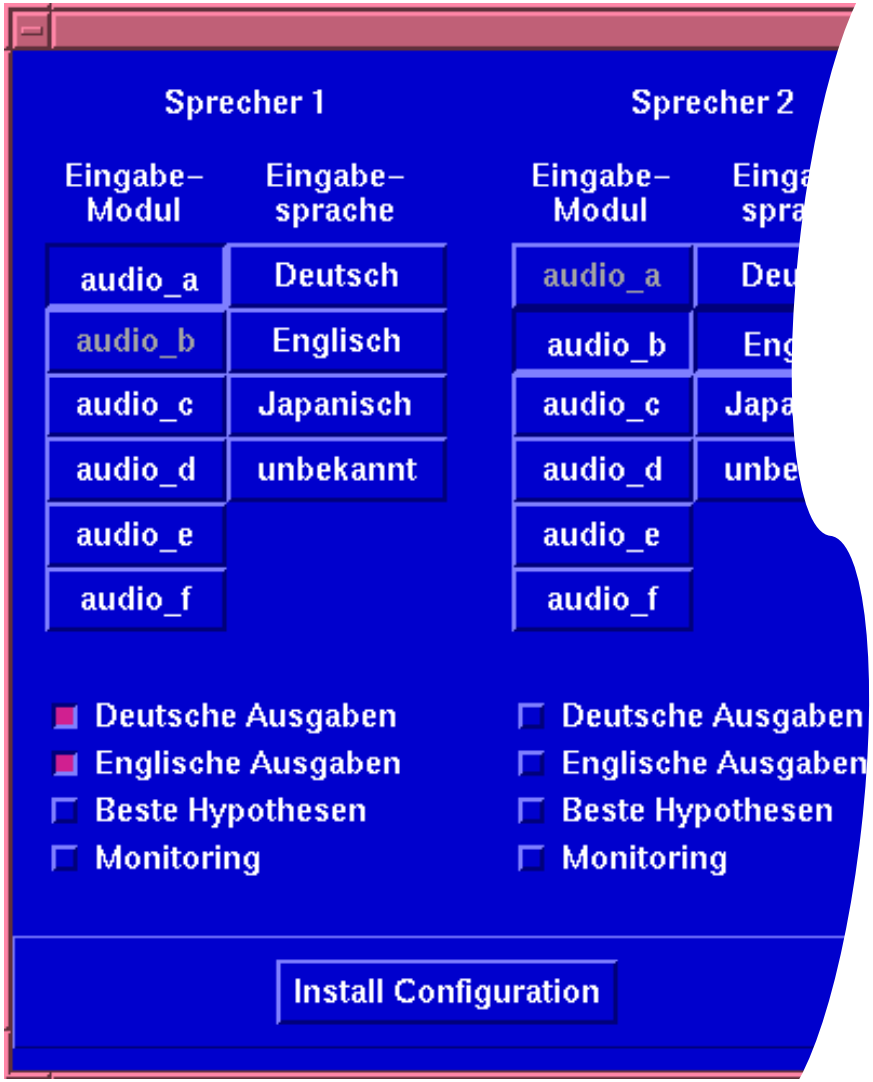
  DFKI, Kaiserslautern



# Audio Configuration

- **Configuration of the systems I/O behavior**
  - How many speakers?
  - For every (possible) speaker:
    - Input device (channel identification, speaker adaption)
    - Output device(s) (translation output, destination for man/machine dialogs)
    - Source language (or „unknown“)
  - Desired system output categories

- **Audio channel configuration**
  - Uniform configuration of heterogeneous audio hardware



# Recording Audio Data

- **Turn-based processing, barge-in available for voice commands**
- **Different audio quality:**
  - lab-quality close-speaking microphone (16kHz)
  - room microphone (16kHz)
  - telephone quality (8kHz)
  - GSM mobile (8kHz)

⇨ **Audio module concept**

  - provides a uniform interface of different hardware devices to the system
  - # of channels is only limited by hardware

- **Open Microphone Approach (essential for telephone translation service!)**
- **Input/output synchronization**
- **No cross-talk allowed**



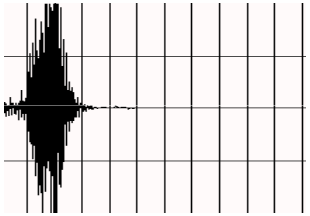
# Open Microphone Approach

Microphone open

Microphone 1

Speech input

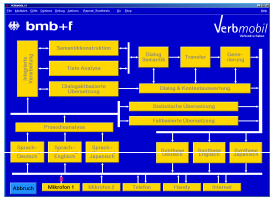
Pause detection



Microphone closed

Microphone 1

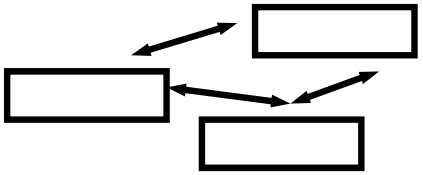
Translation




Speech output

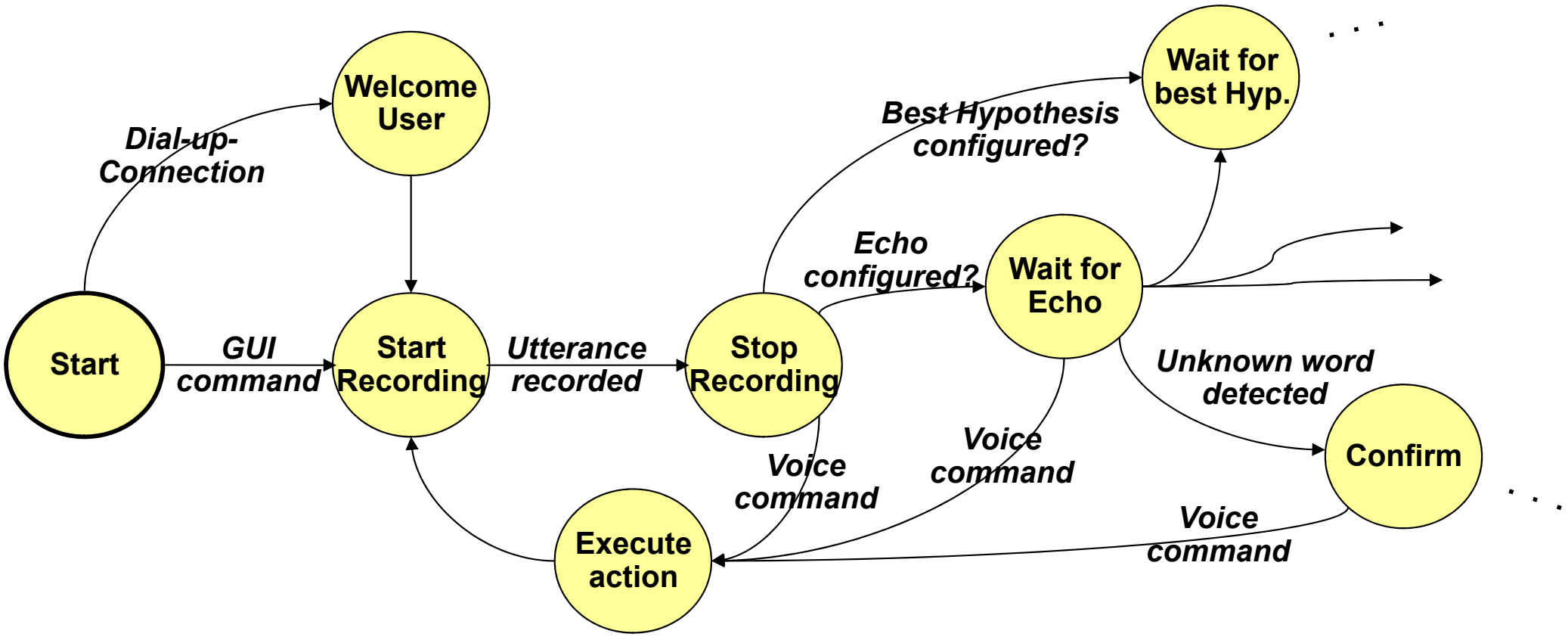
Microphone 1

Synchronization



# Synchronisation

- Synchronization controls the high-level System behavior
- Realized via Finite State Machine



# Recognizing Speech

- Task:

  Analyzing continuous spontaneous speech signals
- Input:

  Audio data
- Method:

  HMMs, class based language models, etc.
- Result:

  Word Hypotheses Graphs (WHG) and speech commands
- Benefit:

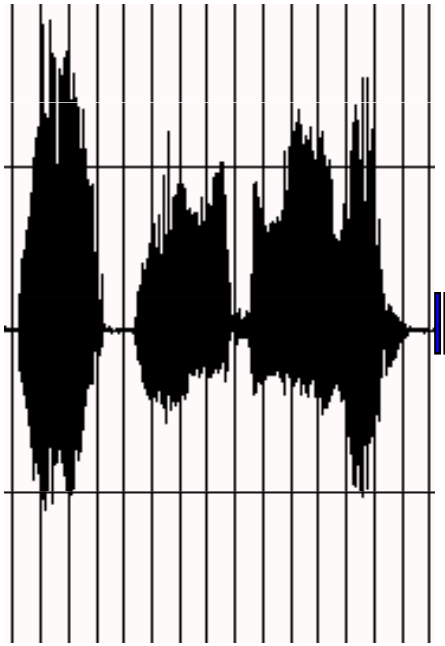
  Compact representation of hypotheses of what has been said
- Responsible:

  DaimlerChrysler AG
  University of Karlsruhe
  RWTH Aachen
  Philips GmbH (Language Models)



# General Speech Recognition Task

## Audio Signal



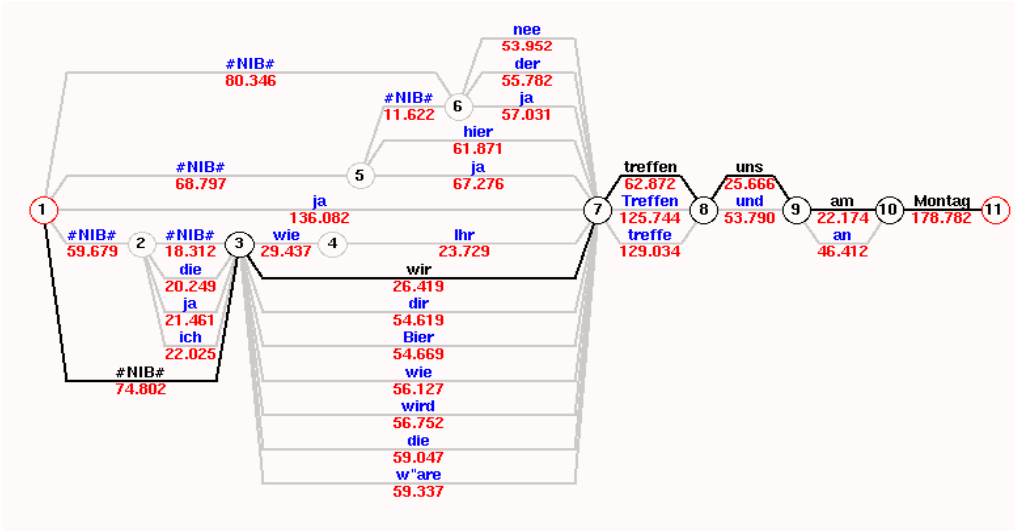
## Recognizers

German

English

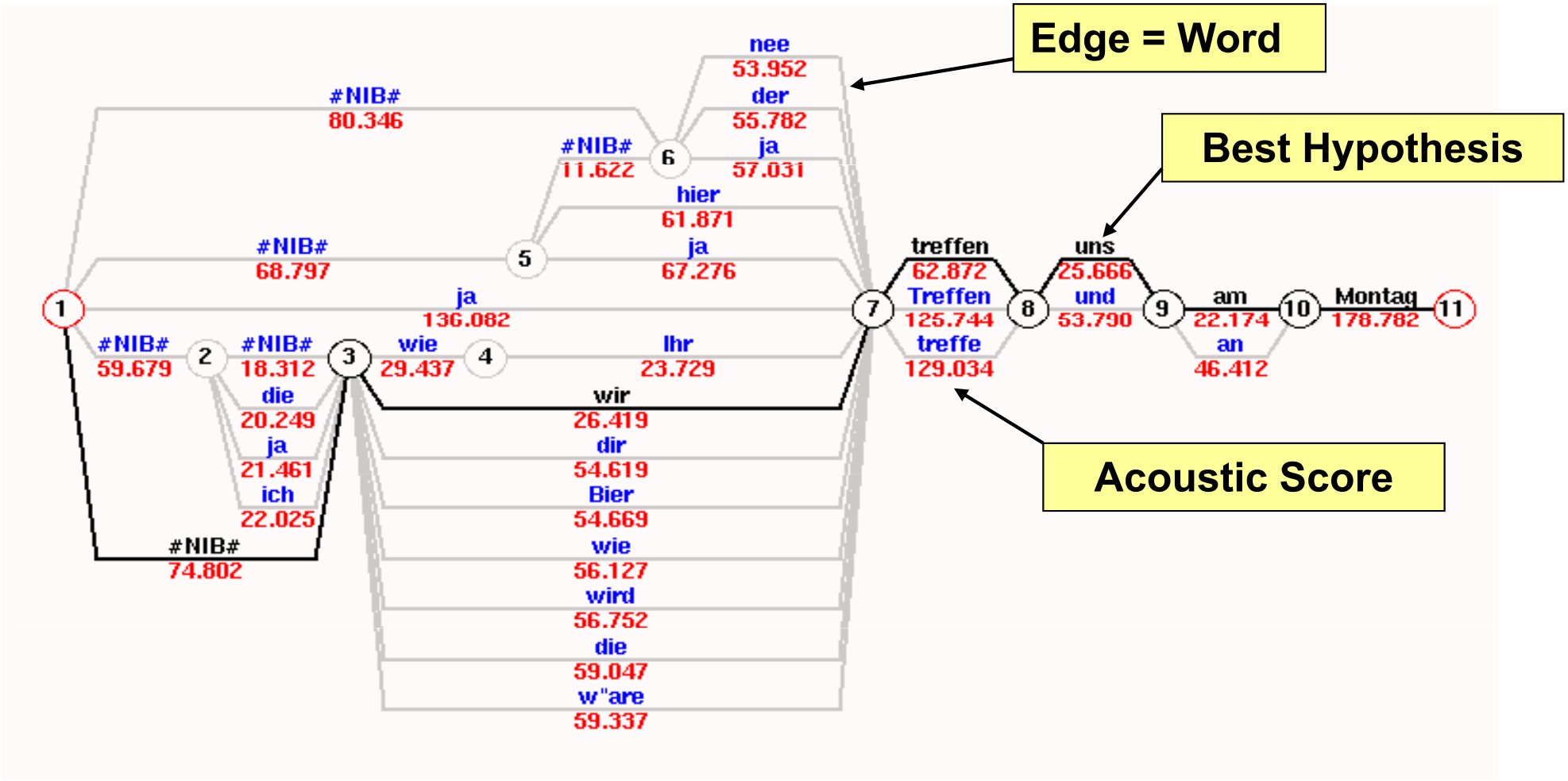
Japanese

## Word Hypotheses Graph

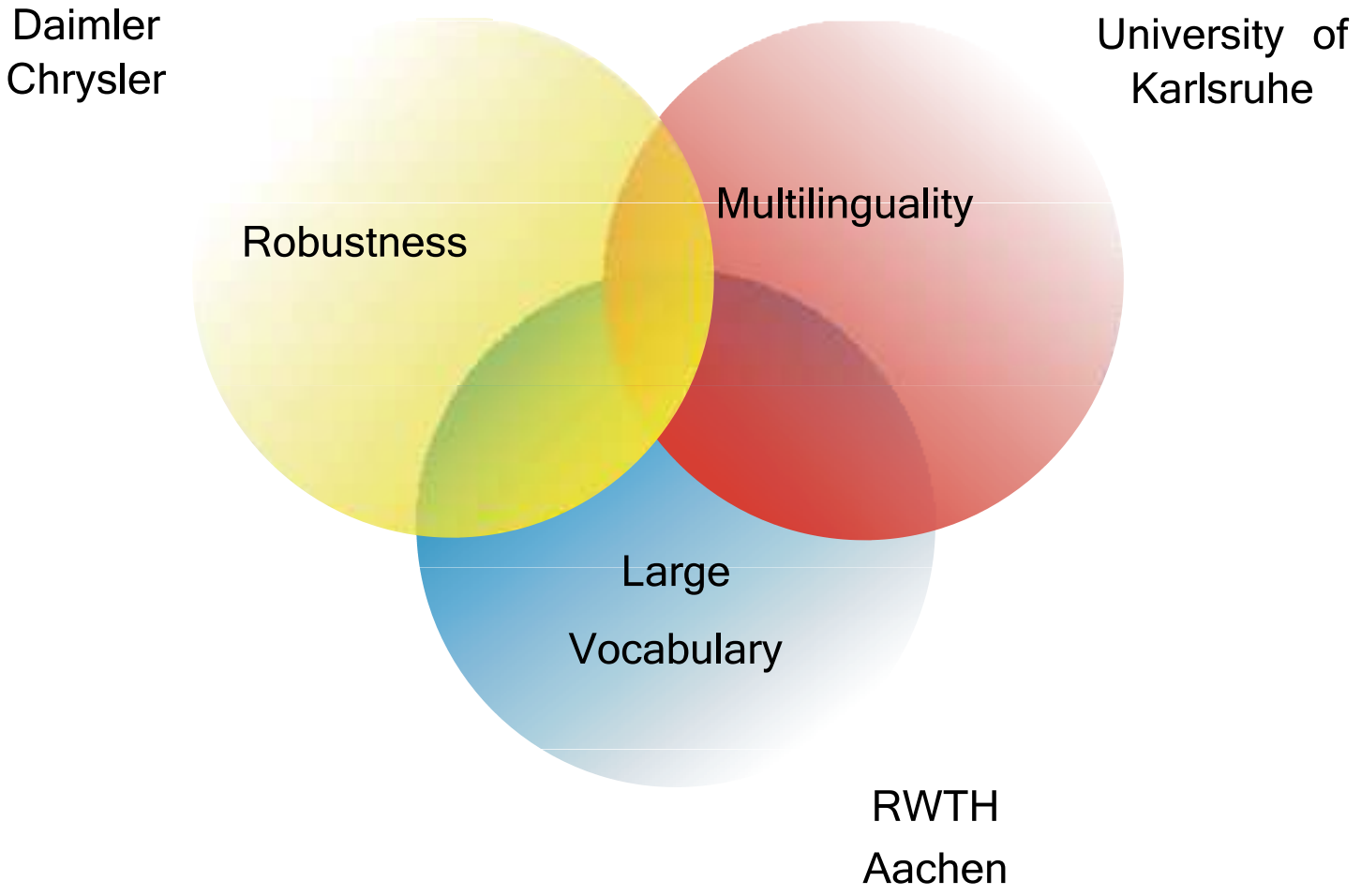


# Word Hypotheses Graphs (WHGs)

**WHGs realize the interface between acoustic and linguistic processing**



# Focuses of Speech Recognition in Verbmobil

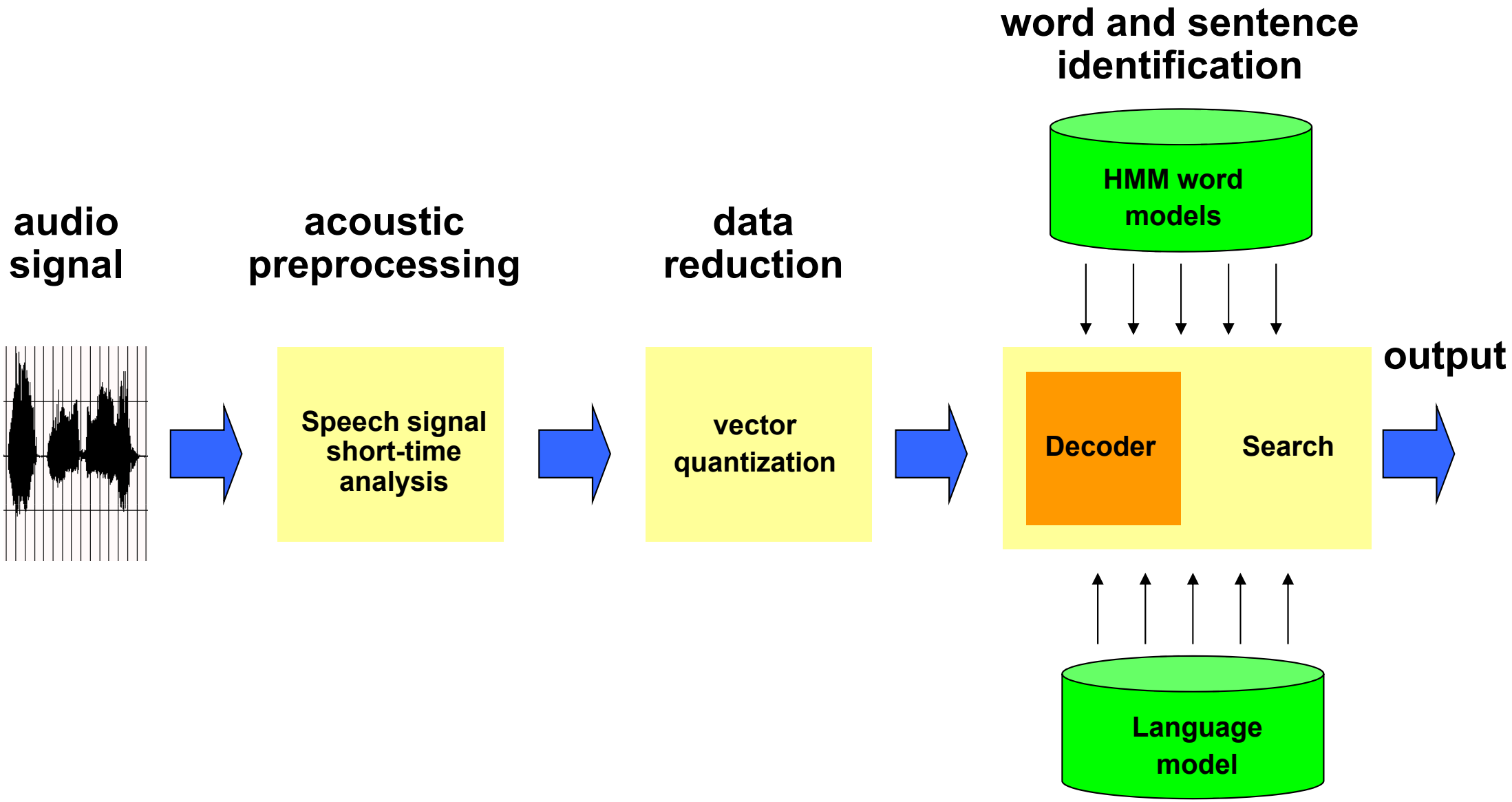


# Nine Available Recognizer Modules

- **DaimlerChrysler**
  - German, 16 kHz, speaker adaptive, approx. 10000 words
  - German, 8 kHz, telephone/GSM quality, speaker adaptive, approx. 10000 words
  - English, 8 kHz, telephone/GSM quality, speaker adaptive, approx. 7000 words
- **University of Karlsruhe**
  - German, 16 kHz, speaker adaptive, approx. 10000 words
  - English, 16 kHz, speaker adaptive, approx. 7000 words
  - Japanese, 16 kHz, speaker adaptive, approx. 2600 words
  - Language Identification Component (German, English, Japanese)
- **RWTH Aachen**
  - German, 16 kHz, speaker adaptive, approx. 10000 words
  - German, 16 kHz, speaker dependent, approx. 30000 words



# Principal Recognizer Architecture



# The Speech Recognition Task

- **Some Highlights of the Verbmobil Recognizers:**
  - Speaker adaptive recognition:
    - Start speaker independent
    - Recognition results enhance during the dialog
  - Capable of dividing speech and noise input using garbage models
  - Segmentation of speech input allows incremental processing
  - Word class based language models and recognition allow flexible vocabulary extension
  - Online vocabulary extension through unknown word detection (names, towns, street names, …)
  - Integrated continuous und speech command recognition

**… and many more**

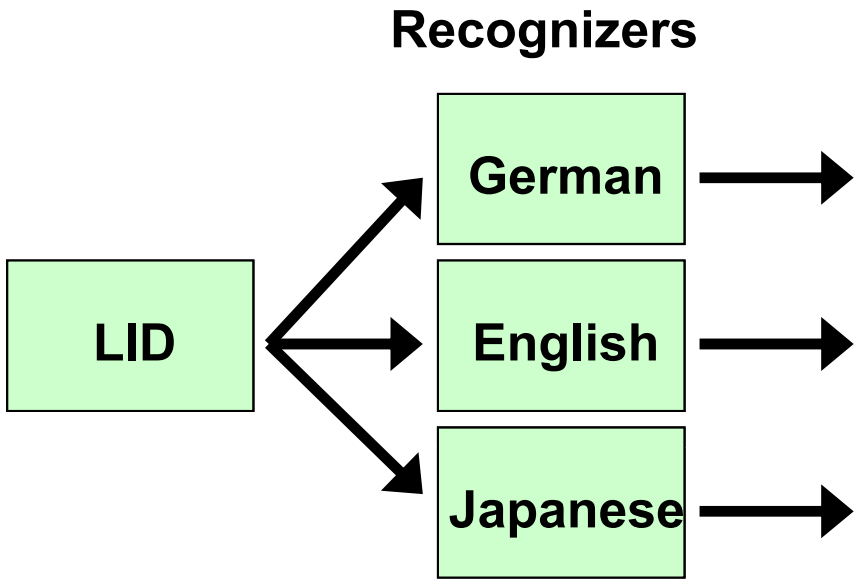


# Language Identification

- **Features**
  - ID on 3 seconds speech signal (maximum)
  - Real time factor 0.5
  - Speaker independent
  - Unknown audio channel
  - Using language model know-how

- **Flexible Architecture:**

  **LID can be combined with any speech recognizer**



# Prosodic Processing

- Task:

  Recognizing prosodic phenomena (accents, sentence mood) and boundaries
- Input:

  WHG and speech signal
- Method:

  Neural networks and statistical classifiers
- Result:

  WHG annotated with accent and boundary information
- Benefit:

  Provides prosodic information needed for correct translation of spontaneous speech
- Responsible:

  Universität Erlangen-Nürnberg



# Prosody in Speech Communication

**Prosody can help to disambiguate**

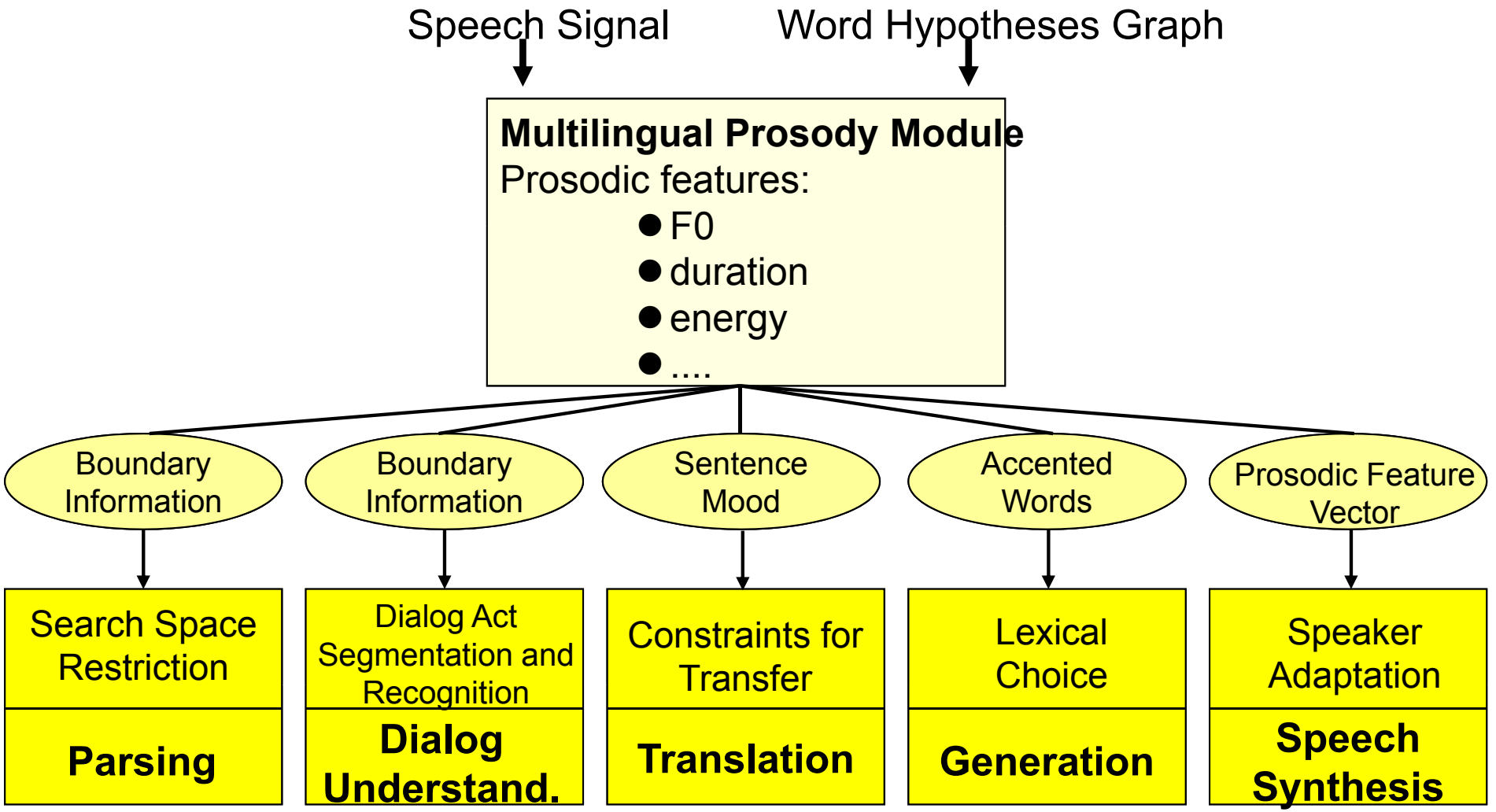
- lexical and phrasal accent
- phrasing (chunks of speech)
- sentence mood
- emotion, attitude, foreign accent

**Parameters represented by Features**

- F0 (fundamental frequency)
- Energy
- Duration
- Speech tempo
- Pause



# Prosody in Verbmobil



# What Linguistic Analysis Really Needs

- **Syntactic Boundaries**
  *He saw* ? *the man* ? *with the telescope* **Prosody cannot help**

- **Dialog Act Boundaries**
  *No, I have no time at all on Thursday.* **D**
  *But how about on Friday?*
  **Dialog acts are pragmatic units that chunk the input into units which can be processed alone.**

- **Prosodic Syntactic Boundaries**
  *Of course* ? *not* ? *on Saturday*
  **Syntactic boundaries that correlate to the acoustic-phonetic reality; help during analysis within one chunk/dialog act. Important in spontaneous speech with elliptical utterances.**



# Extraction of Prosodic Features

- computed for each word
- from basic prosodic features and segmental information
- over different time contexts
- modeling of FO:

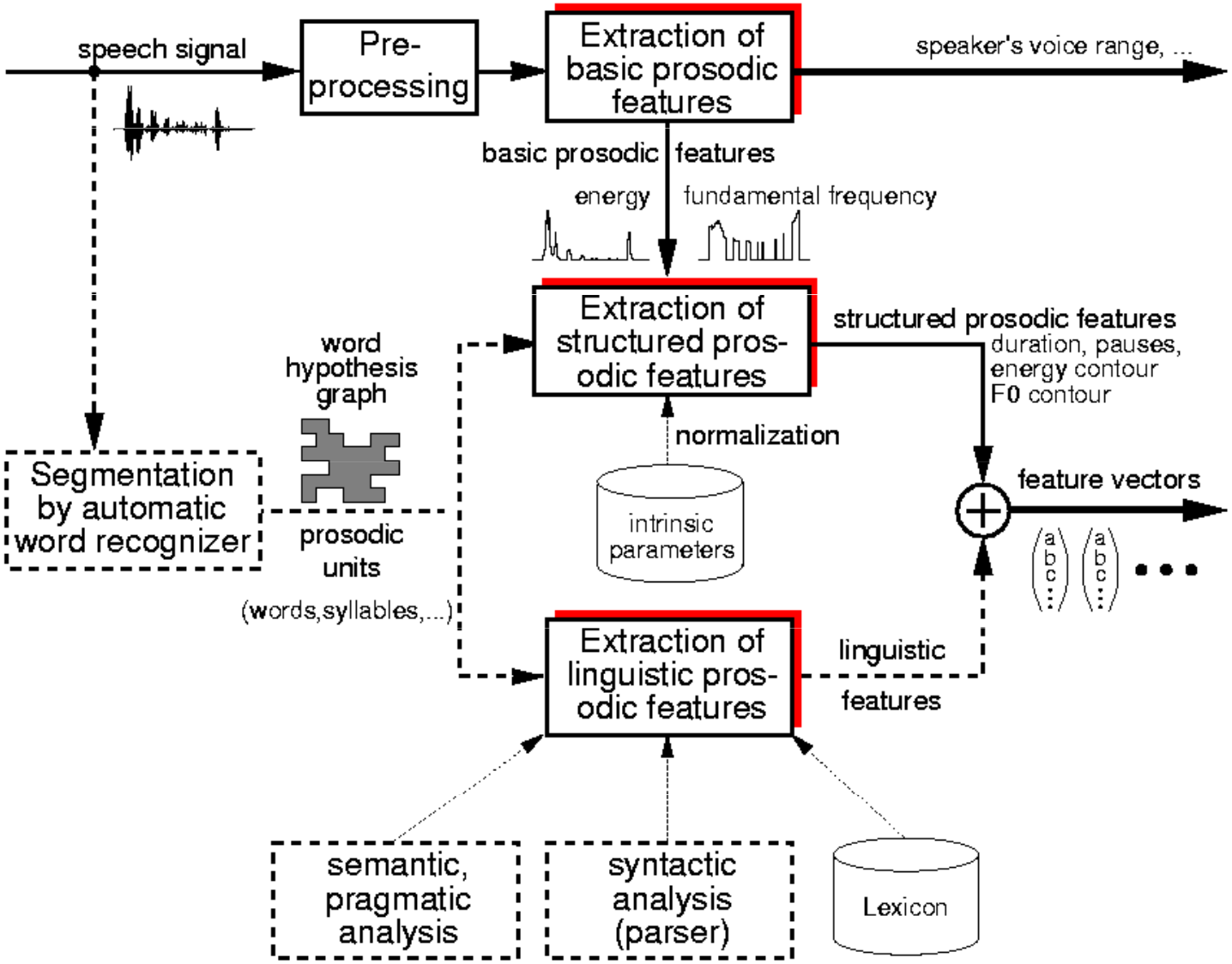
  linear regression coefficient, regression error, mean, median, minimum, maximum, onset, offset and their temporal locations
- modeling of energy--contour

  mean, median, maximum, max-pos, regression coefficient, ...

  and phoneme intrinsic normalizations



# Extraction of Prosodic Features



# Prosodic Classification in Verbmobil

- five classes of boundaries: default, particles, phrases, clauses, sentences
- sentence mood: question vs. non-questions
- phrase accent: disambiguation of particles
- Computed by NN-classifiers and Language Models
- Language Models trained on a corpus annotated with syntactic prosodic boundaries and dialog act boundaries

# An Example

I am calling about the trip to Hanover on the seventh and eighth of March

```
...
2       3       I       50.284023       34      46      (ID r3485) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.82 0.18)  (F 0.92 0
...
3       9       am      24.803406       47      52      (ID r3489) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.84 0.16)  (F 0.81 0
3       10      am      32.151409       47      54      (ID r3490) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.88 0.12)  (F 0.37 0
...
9       11      going   142.015503      53      91      (ID r3504) (PR (S 0.94 0.00 0.05 0.00 0.00)  (A 0.14 0.86)  (F 0.10 0
10      11      calling 131.019409      55      91      (ID r3505) (PR (S 0.39 0.01 0.32 0.27 0.01)  (A 0.07 0.93)  (F 0.13 0
11      12      about   125.144707      92      124     (ID r3506) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.22 0.78)  (F 0.92 0
12      13      the     40.895718       125     136     (ID r3507) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.90 0.10)  (F 1.00 0
12      13      that    42.615807       125     136     (ID r3508) (PR (S 0.80 0.00 0.07 0.00 0.12)  (A 0.84 0.16)  (F 1.00 0
13      14      trip    106.785835      137     167     (ID r3509) (PR (S 0.10 0.00 0.80 0.10 0.00)  (A 0.24 0.76)  (F 0.03 0
14      15      to      69.326729       168     188     (ID r3510) (PR (S 0.86 0.02 0.08 0.02 0.02)  (A 0.85 0.15)  (F 1.00 0
15      16      Hanover 245.755707      189     261     (ID r3511) (PR (S 0.02 0.14 0.43 0.01 0.40)  (A 0.01 0.99)  (F 0.04 0
...
16      18      and     69.891464       266     284     (ID r3514) (PR (S 0.57 0.08 0.11 0.23 0.02)  (A 0.87 0.13)  (F 0.95 0
17      18      on      75.358749       264     280     (ID r3515) (PR (S 0.92 0.03 0.01 0.03 0.00)  (A 0.87 0.13)  (F 0.62 0
18      19      the     37.180725       285     295     (ID r3516) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.94 0.06)  (F 0.98 0
19      20      seventh 184.631897      296     350     (ID r3517) (PR (S 0.06 0.10 0.31 0.00 0.53)  (A 0.07 0.93)  (F 0.11 0
20      21      and     44.750828       356     369     (ID r3518) (PR (S 0.99 0.00 0.01 0.00 0.00)  (A 0.85 0.15)  (F 0.15 0
21      22      the     42.576515       370     376     (ID r3520) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.95 0.05)  (F 1.00 0
22      23      eighth  134.293030      381     420     (ID r3521) (PR (S 0.00 0.00 0.99 0.00 0.01)  (A 0.24 0.76)  (F 0.38 0
23      24      of      62.543167       425     443     (ID r3522) (PR (S 1.00 0.00 0.00 0.00 0.00)  (A 0.74 0.26)  (F 1.00 0
24      25      March   204.886185      444     497     (ID r3523) (PR (S 0.02 0.63 0.03 0.02 0.30)  (A 0.04 0.96)  (F 0.03 0
```

# Repair of Self-Corrections

- Task:

  Detecting and repairing self-corrections
- Input:

  WHGs
- Method:

  Stochastic models
- Result:

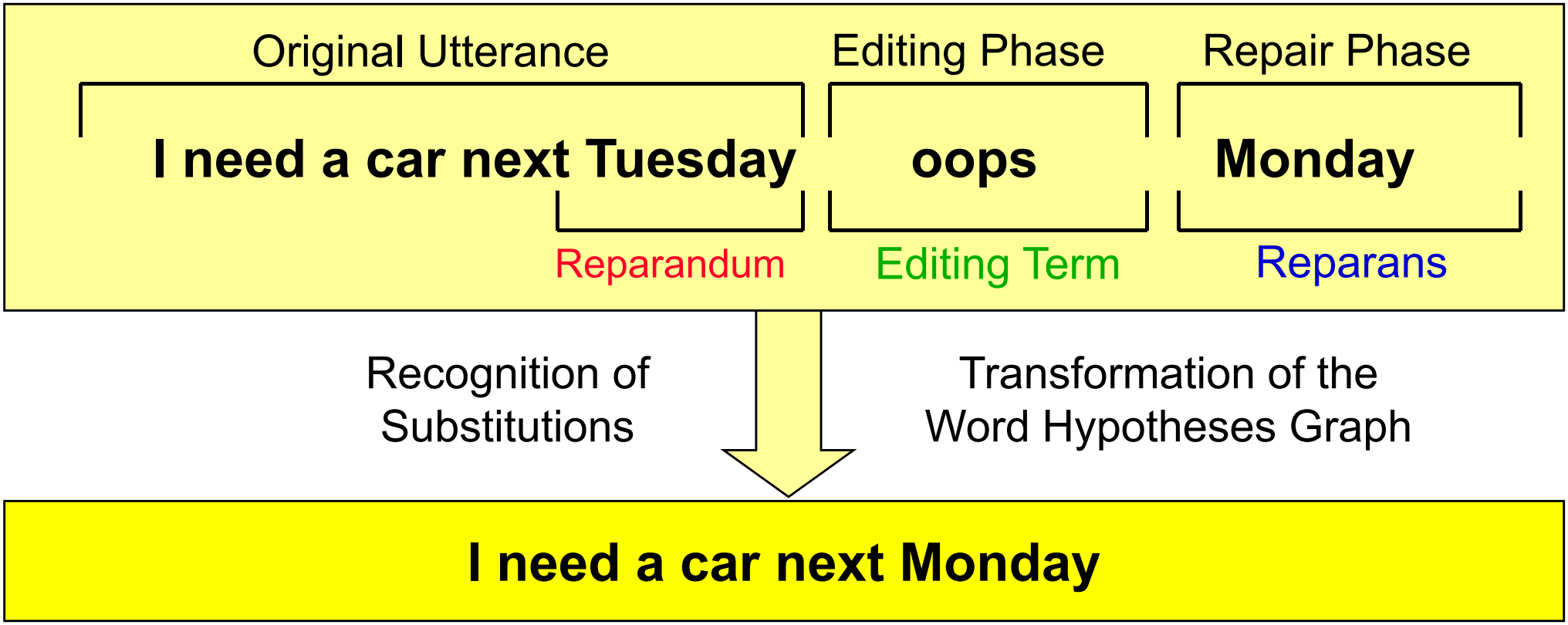
  Enriched WHGs, including additional repaired hypotheses
- Benefit:

  Enabling Verbmobil to repair self-corrections of spontaneous speech input
- Responsible:

  Universität Erlangen-Nürnberg



# The Understanding of Spontaneous Speech Repairs



# Facts about Repairs in the Verbmobil Corpus

- 21% of all turns in the Verbmobil corpus (79 562 turns ) contain at least one self correction
- The syntactic category is preserved in most cases

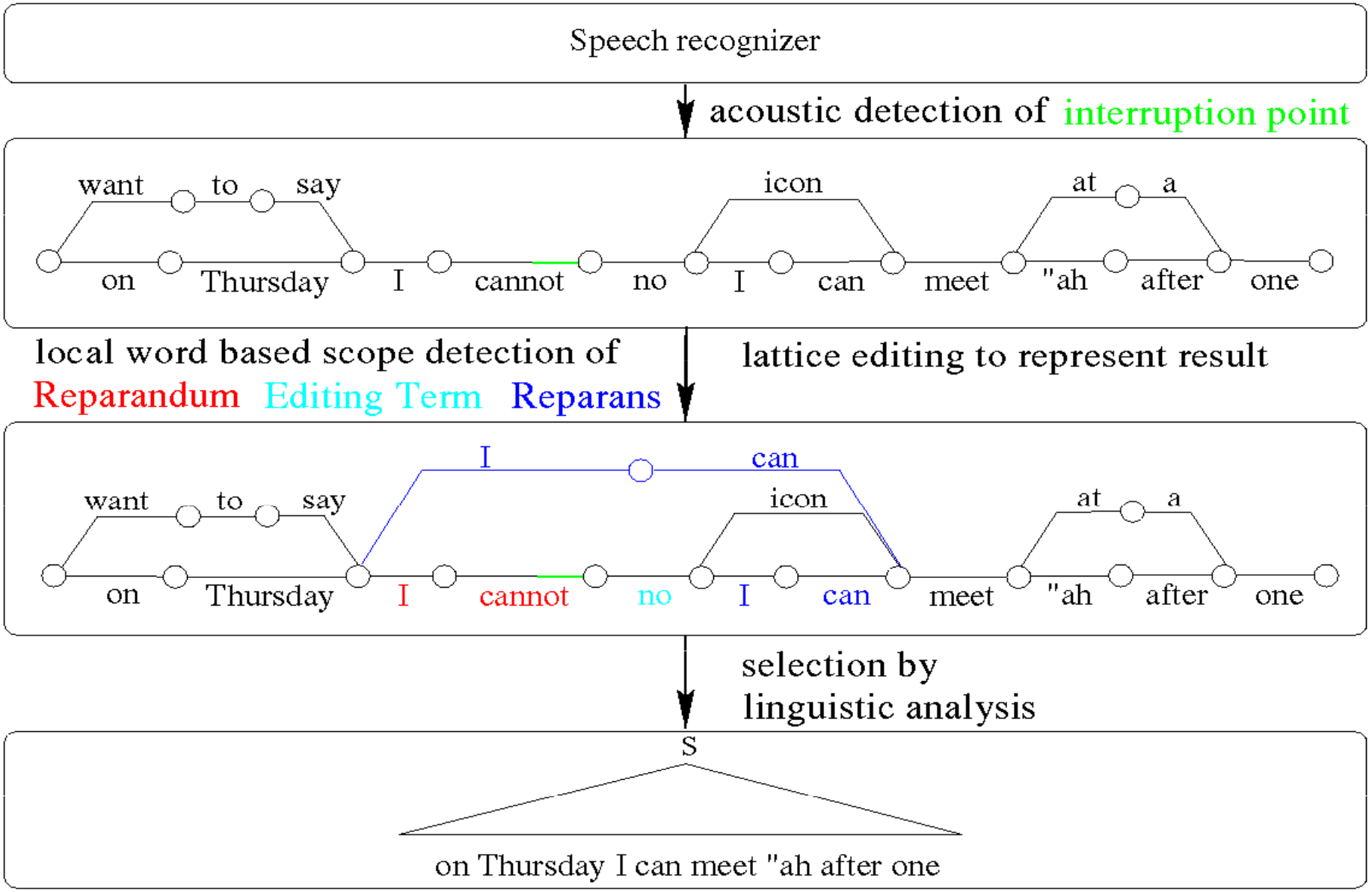
  (For example: Out of a sample of 266 verb replacements, 224 are again mapped to verbs)
- Repairs take place in a restricted context

  (in 98% the reparandum consists of  less than 5 words)
- Repair sequences underlie certain regularities



# Architecture of Repair Processing

***"On Thursday I cannot no I can meet äh after one"***



# Scopus Detection

- **The editing term (ET) is given by the prosody**
- **Wanted: Beginning (RB) and end (RE) of the Repair**
- **Search the best replacement of a word order on the left hand side of ET through a word order on the right hand side of ET**

$\Rightarrow$ **rate the possible replacements**

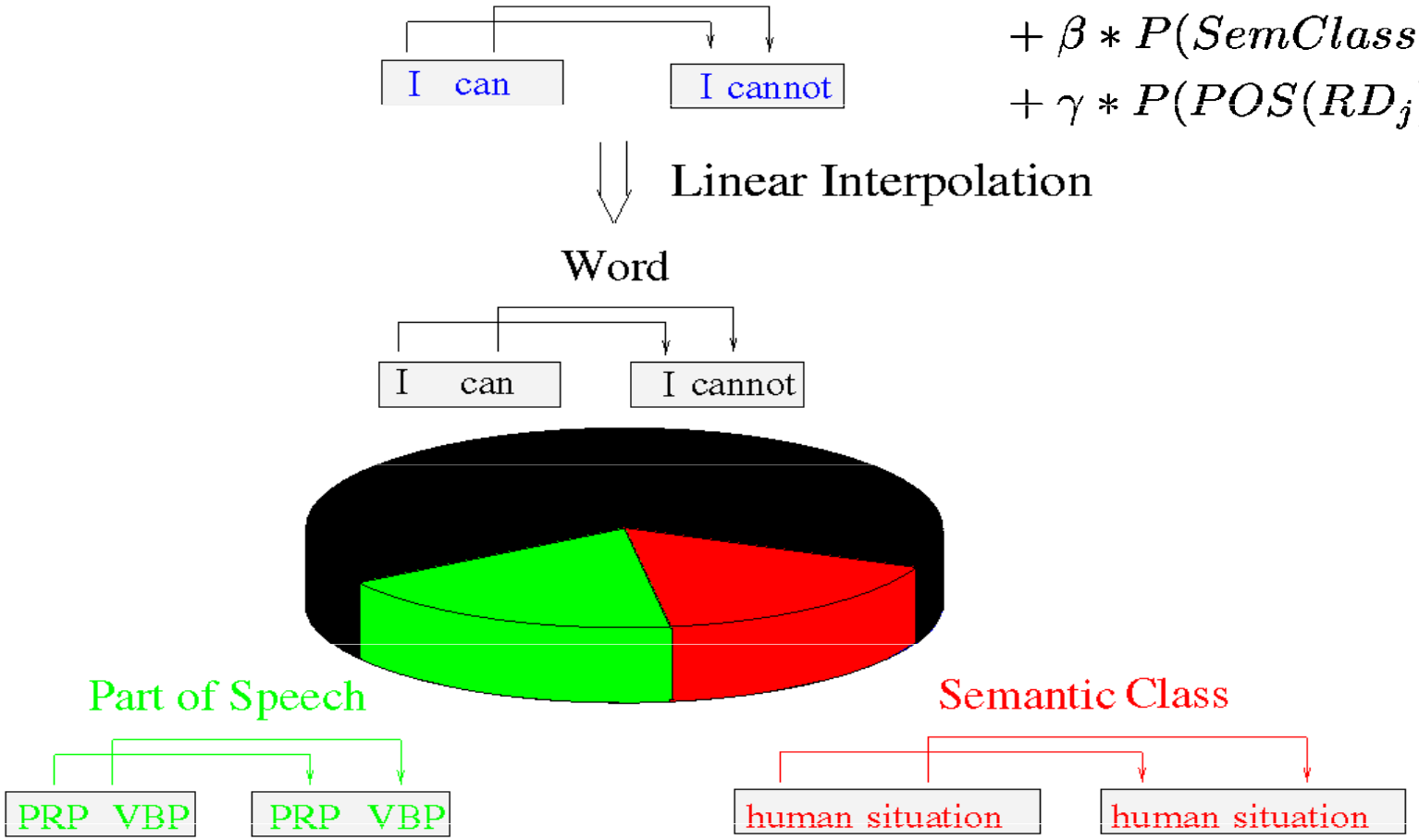
**search space is limited through looking at 4 words before and after ET**

- ***Choose the best rated replacement over a certain threshold***



# Repair Detection and Word Smoothing

$$P_r(RD_j|RS_{a_j}) =$$
$$\alpha * P(Word(RD_j)|Word(RS_{a_j}))$$
$$+ \beta * P(SemClass(RD_j)|SemClass(RS_{a_j}))$$
$$+ \gamma * P(POS(RD_j)|POS(RS_{a_j}))$$



# Dialog Translation



# Multiple Approaches

- **Mono-cultural approaches are dangerous**
  - humans vs. viruses 🡻 diversity
  - Microsoft vs. ILOVEYOU and copycats 🡻 alternative software solutions
- **Some sources of errors in a speech translation system**
  - external
    - spontaneous speech: not well formed, hesitations, repairs
    - bad acoustic conditions
    - human dialog behavior
  - internal
    - knowledge gaps in modules
    - software errors
    - probabilistic processing

❑ **Use multiple engines, varying approaches on various stages of processing**



# Multiple Approaches in Verbmobil

- **Exclusive alternatives: three different 16 kHz German speech recognizers with various capabilities**
- **Competing approaches:**
  - three parsers: HPSG, Chunk, Statistical
  - five translation tracks: case-based, dialog-act based, statistical, substring-based, linguistic (deep) semantic translation
- **Needed: selection and combination of results from competing tracks**
  - parsers: combination of partial analyses in the semantic processing modules
  - translation: preselection module



# Multiple Translation Tracks - Approaches and Advantages

- **Case-based:**
  - Approach: uses examples from the aligned bilingual Verbmobil corpus
  - Advantage: good translation if input matches example in corpus
- **Dialog-act based:**
  - Approach: extract core intention (dialog act) and content
  - Advantage: robust wrt. recognition errors
- **Statistical**
  - Approach: use statistical language and translation models
  - Advantage: guaranteed translation with high approximate correctness
- **Substring- based**
  - Approach: combines statistical word alignment with precomputation of translation ”chunks” and contextual clustering
  - Advantage: guaranteed translation with high approximate correctness
- **Linguistic (deep) semantic translation**
  - Approach: “classic” approach using semantic transfer
  - Advantage: high quality translation in case of success



# Example Based Translation

- Task:

  Providing a translation based on translation templates and partial linguistic analysis
- Input:

  WHGs or best Hypothesis
- Method:

  Definite Clause Grammar (DCG), graph matching algorithms
- Result:

  Translation and a confidence value
- Benefit:

  Improving Verbmobils translation capabilities through an additional translation path
- Responsible:

  DFKI, Kaiserslautern

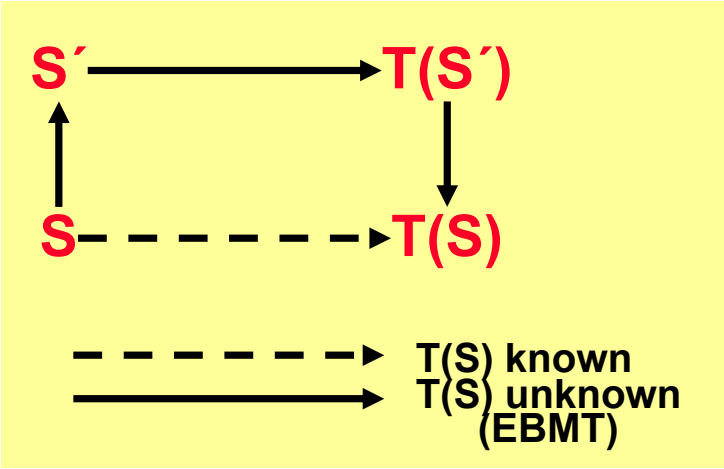


# The Case Based Approach

- **Training is based on Verbmobil‘s bilangual corpus**

```
E: I am on vacation, on the sixth and the seventh.
D: ich bin am sechsten und siebten verreist.
```

- **Principle: Look up an example in the example storage that matches the input sentence best, use it’s translation as output**



# Generalization in Example Based Machine Translation (EBMT)

- **Handicap of this naive approach: inadequate coverage**

```
S :    I am not free on Friday.
S':    I am not free on Monday.
T(S'): am Montag habe ich keine Zeit.
```

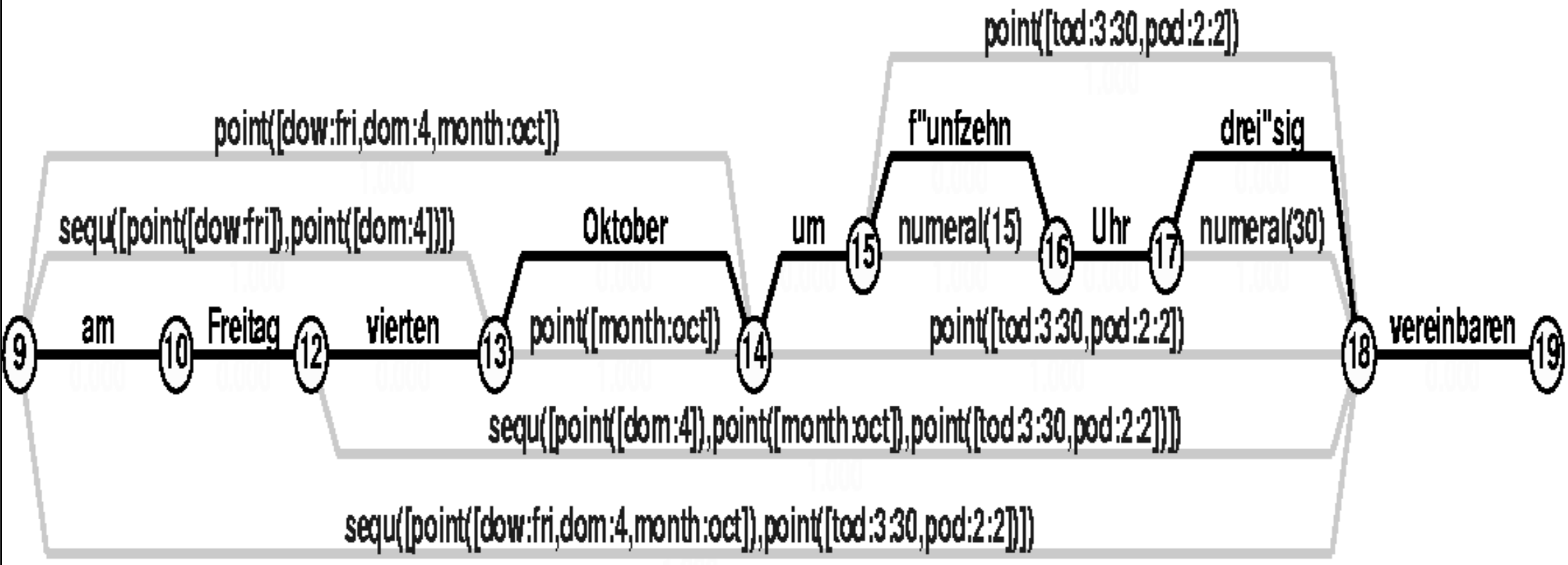
- **Solution: partial generalization (analysis and generation)**

```
E: I am not free <Temp>.
D: <Temp> habe ich keine Zeit.
```

- **Automatic generalization approach:**
  - The grammar automatically generalizes the corpus (offline)
  - The runtime module generalizes incoming input (online)
  - Match generalized input sentence with generalized corpus example
  - Result: instantiated corpus translation



# Generalization of WHGs



# Example Based Translation – Some More Features

- **Generalization grammar for temporals, names, locations (region, town, country), institutions**
- **Fast and robust WHG search:**
  - WHG packing
  - Optimal alignment for fast corpus search
  - Search space pruning
  - Search space caching
  - Any time capable
- **Adequate confidence value for selection**



# Dialog-Act Based Translation

- Task:

  Robustly provide a translation of core intentions and contents of the domain
- Input:

  Prosodically annotated best hypothesis (flat WHG)
- Method:

  Statistical dialog-act classifier and Finite State Transducers
- Result:

  Translation and a confidence value, additionally content descriptions for the dialog module
- Benefit:

  Robust translation and content extraction even when the recognition is erroneous
- Responsible:

  DFKI, Saarbrücken



# Dialog Acts

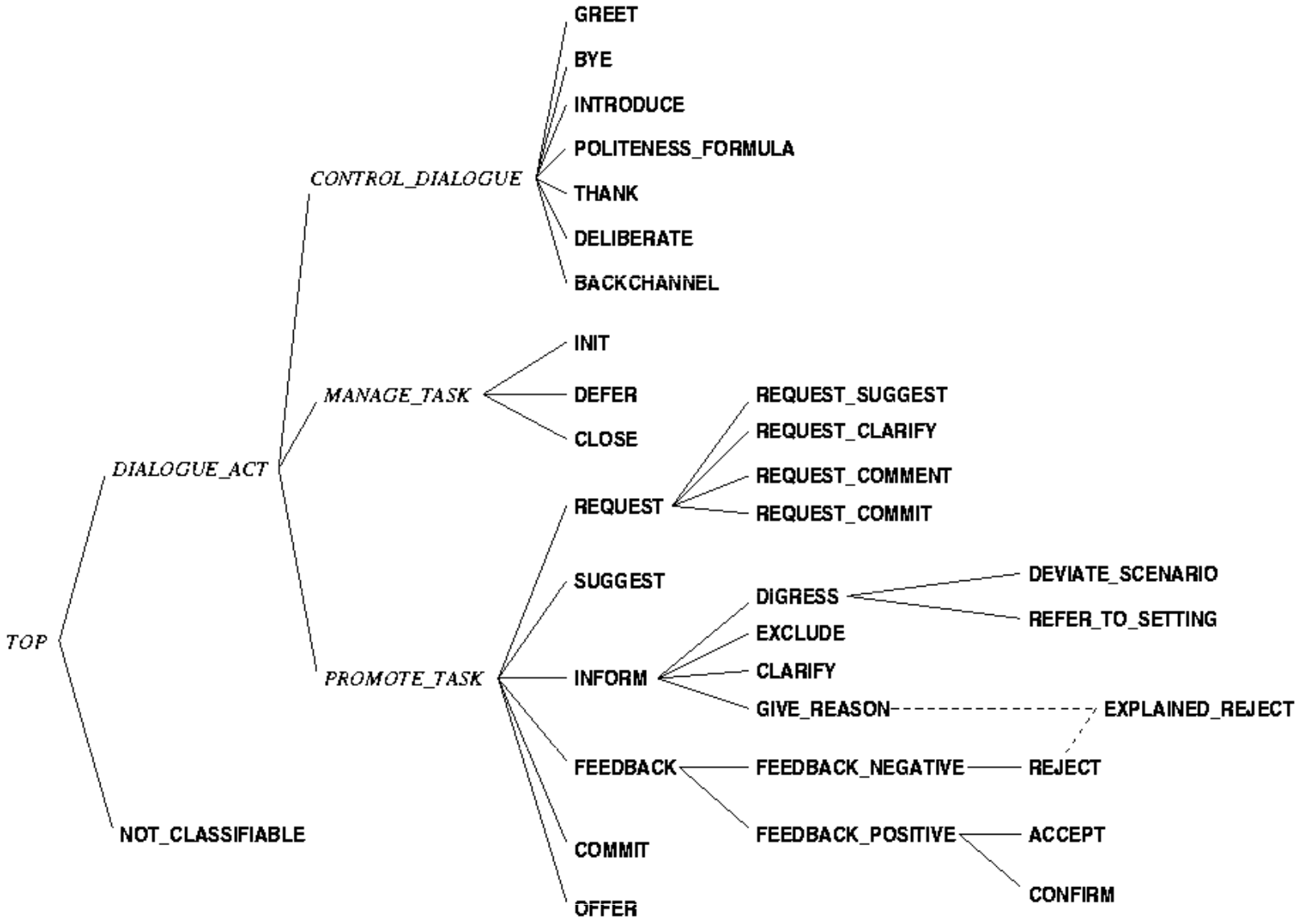
- Describe the core intention of an utterance
- 32 acts defined in a hierarchy, 19 used in processing
- 21 CD-ROMs with 1505 dialogs (German, English, Japanese) annotated with dialog acts for training and test purposes
- Computation uses bigram language models

$$D = \underset{D}{\operatorname{argmax}} P(w \mid D) \cdot P(D)$$

- Probabilities estimated from the annotated corpus
- Leave-One-Out test results for approx. 1000 German, English and Japanese dialogs: Recall 72.48 % (27185 of 37505), Precision 69.90 %

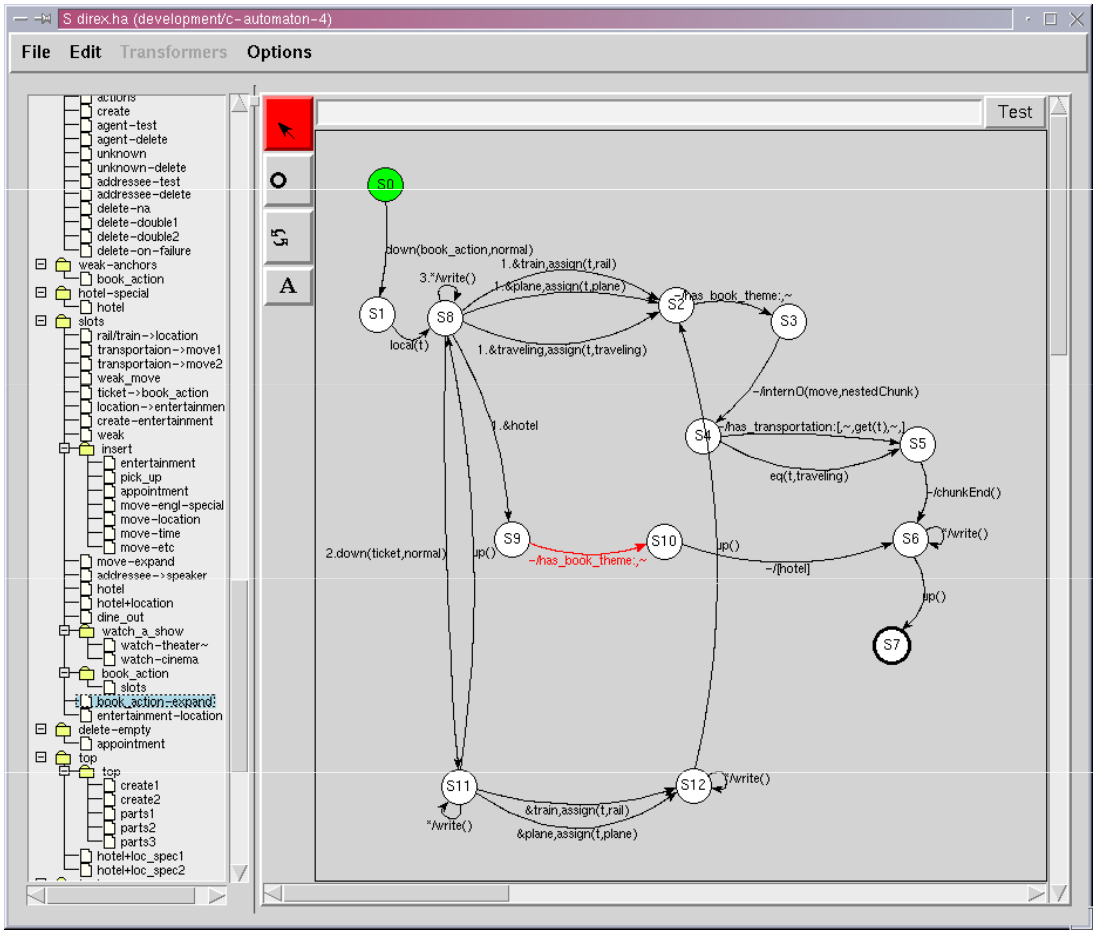


# Dialog Acts - The Hierarchy

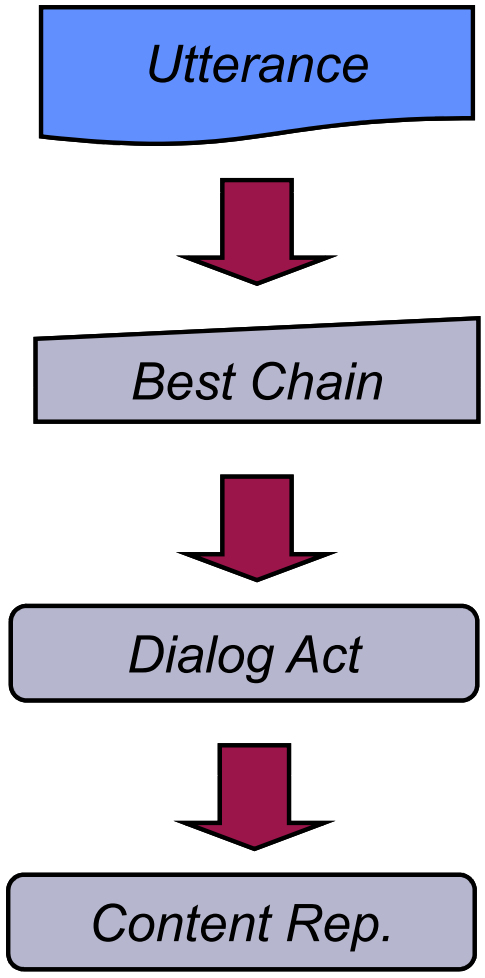


# Representation of Information and Extraction

- **Semantic representation language, used also in the dialog and context modules**
- **Extraction using Finite State Transducers**
- **Semi-automatic creation exploiting semantic databases and lexica**
- **Comfortable development platform**



# Processing Steps




*good so we will leave Hamburg on the first*

I would so we were to leave Hamburg on the first

INFORM

has_move:[move,has_source_location:[city,has_name =‘hamburg‘],
has_departure_time:[date,time=[dom:1]].

# Generation

- **Generation templates (>140), depending on dialog act, topic, content**
- **Translated in Finite State Transducers**
- **Examples:**

```
suggest scheduling $has_date
        g:ich w"urde $* vorschlagen &loc_mode_dat
        e:how about $*

suggest entertainment or($has_location,$has_theme)
        g:wir k"onnten $* gehen &loc_mode_acc
        e:we could go $*

request_suggest
        g:was schlagen Sie vor
        e:what do you suggest
        j:itsu ga yoroshii deshou ka
```

- **Result for our example:** *also wir fahren ab Hamburg am ersten*



# Statistical Translation

- Task:

  Provide approximative correct translations
- Input:

  Prosodically annotated best hypothesis (flat WHG)
- Method:

  Use statistical language and translation models
- Result:

  Translation and a confidence value
- Benefit:

  Approximative correct translation for spontaneous speech
- Responsible:

  RWTH Aachen



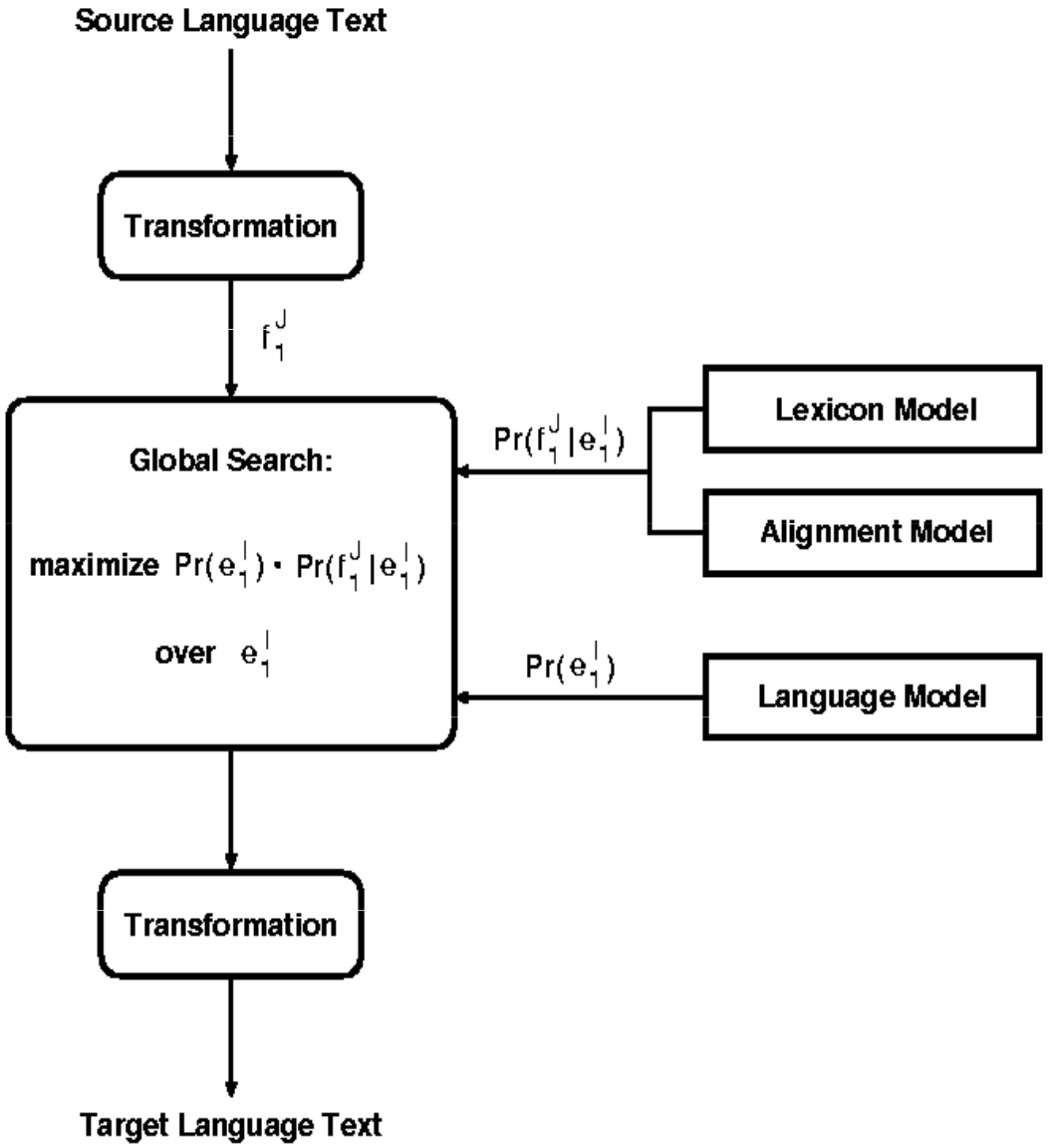
# The Statistical Translation Model

- **Task: translate the source string *f* in the most probable target string *e*:**

$$\hat{e}_1^I = \arg\max_{e_1^I} \{p(e_1^I | f_1^J)\}$$

$$= \arg\max_{e_1^I} \{p(e_1^I) \cdot p(f_1^J | e_1^I)\}$$

- **Bayes' rule needs language model of the target language, and lexicon and alignment models**
- **Learned from aligned corpus**



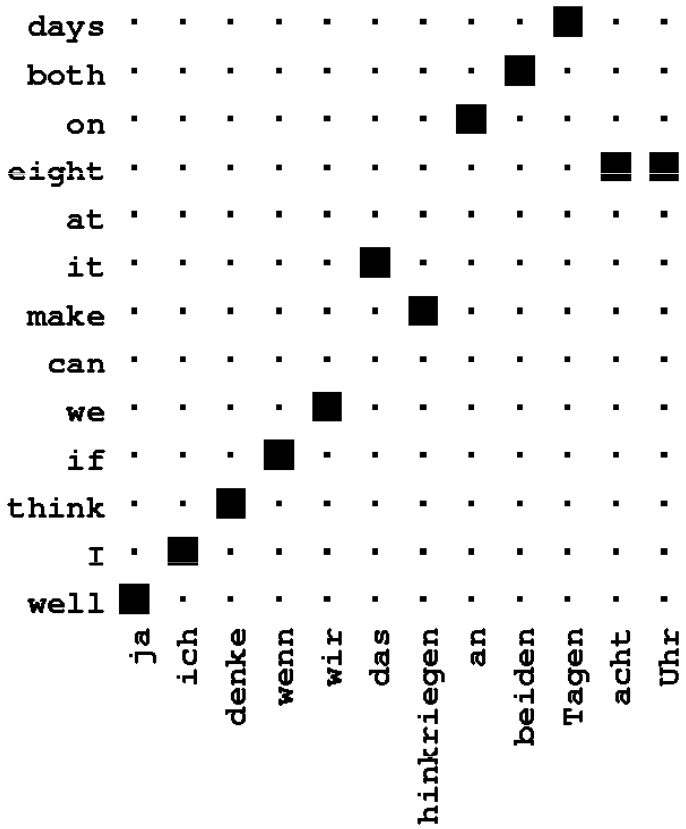
# Alignment Templates

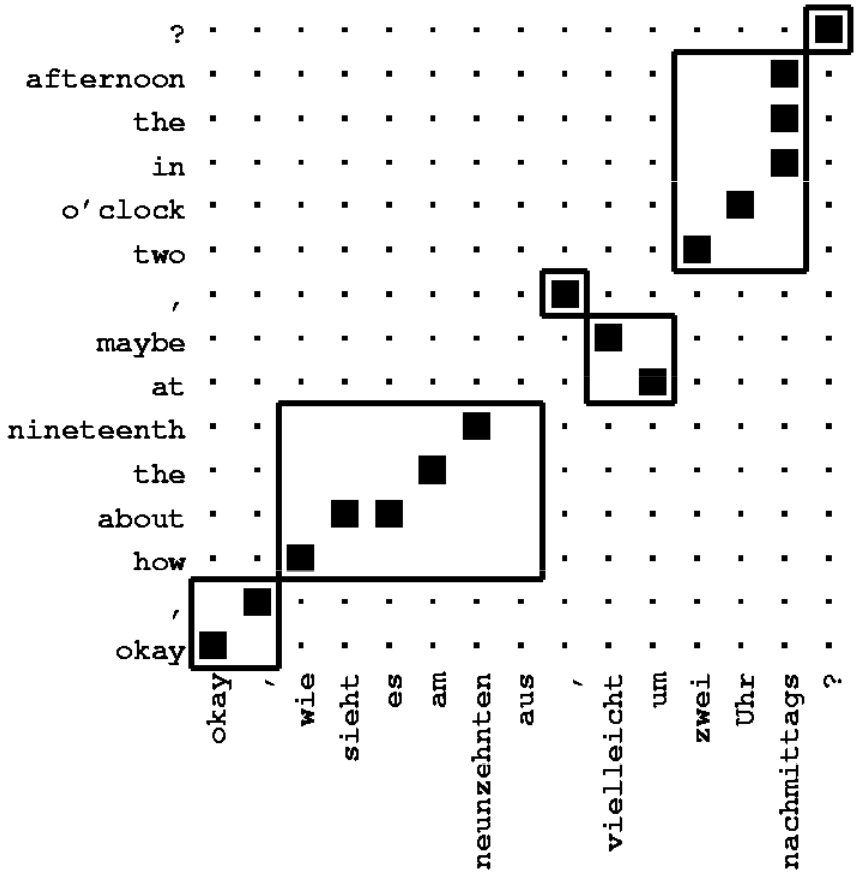
- **Find corresponding words in source and target language sentences**
- **Difficult for language pairs with different word order**
- **Solution: alignment templates**
  - based on word classes (sparse data problem: approx. 40% of the words in the training corpus are singletons)
  - first step: statistically learn alignment of words for each translation direction
  - second step: combine the alignments of both directions
  - third step: statistically learn alignment of “phrases”, i.e. word sequences



# Alignment

Word-to-Word vs. Alignment Templates



# Deep Translation

- Task:

  Provide high quality translations
- Input:

  Prosodically annotated WHG and contextual information
- Method:

  Use syntactic and semantic approaches to analysis, transfer, and generation
- Result:

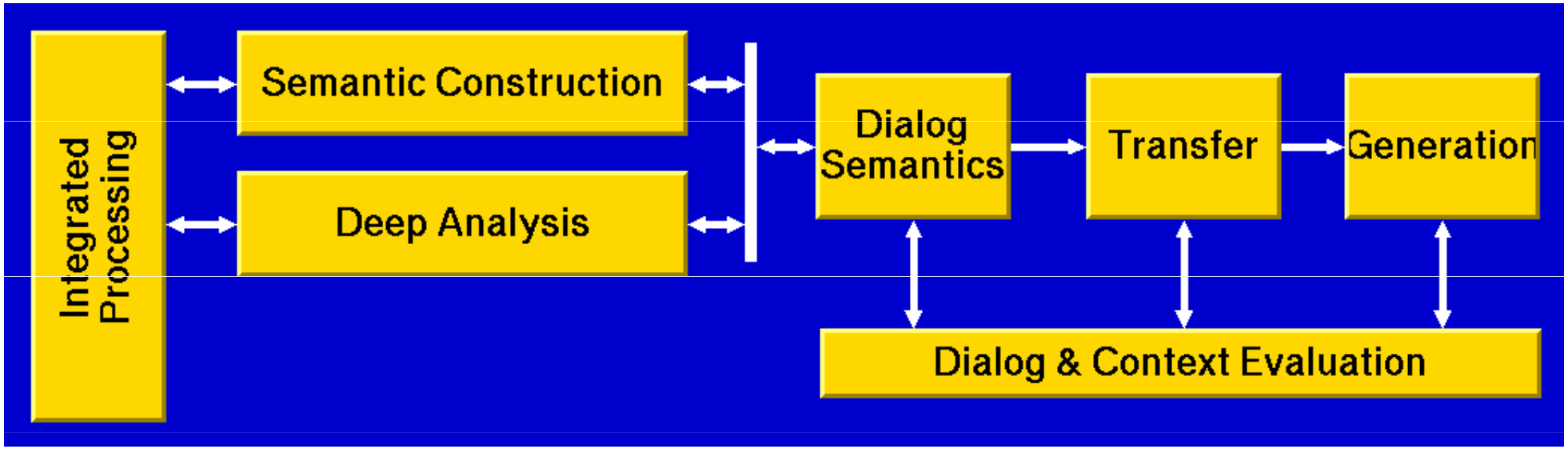
  Translation containing content information, suited for high quality speech synthesis
- Benefit:

  Delivers the highest quality, but is sensitive to recognition errors and spontaneous speech phenomena
- Responsible:

  Siemens AG, DFKI Saarbrücken, Universität Tübingen, Universität des Saarlandes, Universität Stuttgart, TU Berlin, CSLI Stanford



# Modules Involved




- Integrated processing comprises
  - search through the WHG
  - statistic parser
  - chunk parser
- Semantic Construction provides VITs from statistic and chunk parser output
- Deep Analysis: HPSG Parser
- Dialog Semantics:combination of parsing results, and semantic resolution
- Transfer: VIT to VIT transfer
- Generation: TAG generation from VITs
- Dialog+Context: provides contextual information

# The Multi-Parser Approach

- **Verbmobil uses three different syntactic parsers:**
  **an HPSG parser, a chunk parser, and a probabilistic LR parser.**

- **Every parser implements another level of parsing accuracy, depth of syntactic analysis, and robustness of the analyzing process.**

  - Chunk parser: Most robust but least accurate analysis
  - HPSG parser: Most accurate by least robust analysis
  - Probabilistic parser: Level of accuracy and robustness between HPSG and chunk parser



# Integrated Processing

- **Gets WHGs for the English, German, or Japanese speech input and dispatches WHG information to the three parsers**
- **Provides an A* search algorithm that allows any connected parser to find the best scored path using**
  - acoustic score of the speech recognizer
  - Verbmobil trigram language model
- **Parsers analyze the same utterance simultaneously**



# VIT: Verbmobil Interface Term

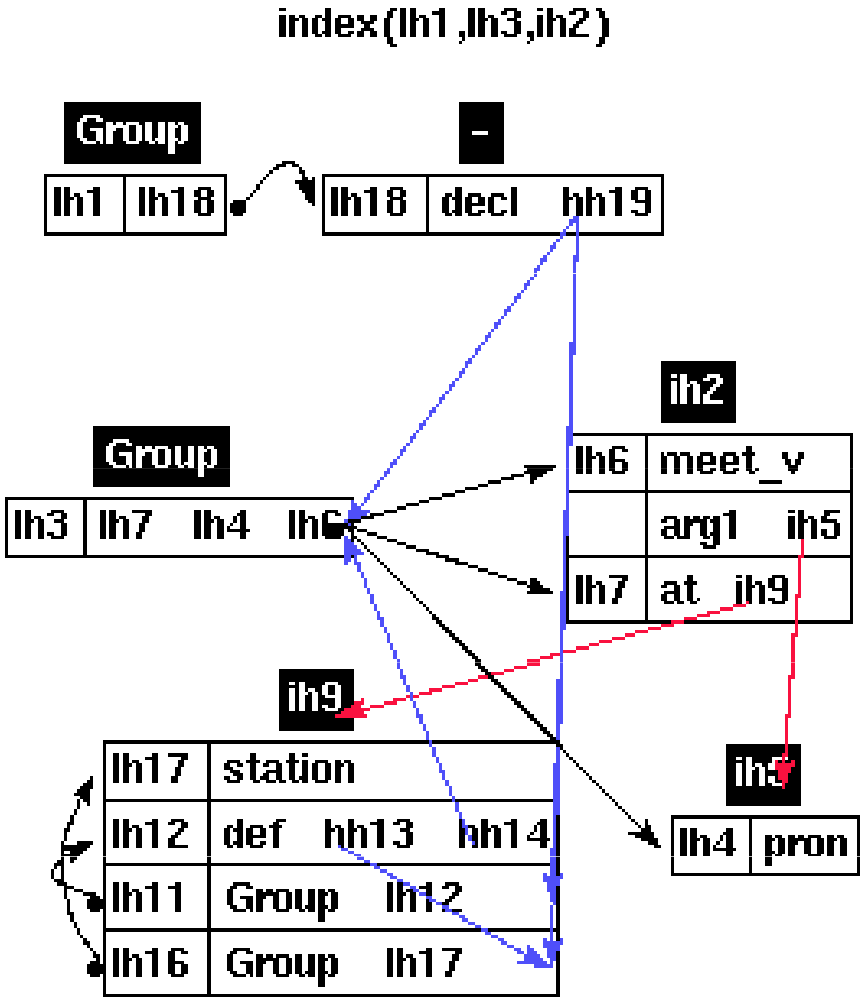
- Common syntactic-semantic interface
- Contains all linguistic information relevant for translation
- Record-like data structure: variable-free lists of non-recursive terms
- ``Flat'' set representations: semantic, scopal, sortal, morpho-syntactic, prosodic, and discourse information
- Labels relate different kinds of information
- Abstract Data Type implements construction, access, update, check, print, etc. facilities

# VIT: Verbmobil Interface Term

```
vit(vitID(sid(...),                                  %Segment ID
    []),                                             %WHG-String
    index(l250,l234,i72),                            %Index
    [start_v(l248,i72),                              %Conditions
     arg1(l248,i72,i75),
     nop(l240,h85),
     quest(l249,h84),
     time(l238,i73),
     abstr_vacation(l247,i75),
     pron(l242,i74),
     poss(l244,i75,i74),
     temp_loc(l239,i72,i73),
     def(l245,i75,h87,h86),
     whq(l235,i73,h83,h82)],
    [in_g(l235,l237), ...                            %Constraints
     leq(l234,h85), ...],
    [s_class(l240,mp), ...],                         %Sorts
    [ana_ante(i74,[i75,i69,i67,i66]),                %Discourse
     prontype(i74,third,std), ...],
    [gend(i75,masc), num(i75,sg)],                   %Syntax
    [ta_mood(i72,ind), ...],                         %Tense and Aspect
    [...])                                           %Prosody
```

*When do your vacations begin?*

# VIT: Verbmobil Interface Term




*We meet at the station.*

# HPSG Processing

- Task:

  Thorough syntactic analysis

- Input:

  Word chains from integrated processing

- Method:

  Apply HPSG analysis

- Result:

  Source language VITs

- Benefit:

  Delivers the highest quality, but is sensitive to recognition errors and spontaneous speech phenomena

- Responsible:

  DFKI Saarbrücken, CSLI Stanford



# Head Driven Phrase Structure Grammar

- Well known  advanced grammar theory in linguistics
- Based on the concept of a *sign* as integrated information structure for all types of linguistic information
- Inherently multilingual by distinguishing universal principles from language specific aspects
- Typed feature structures with inheritance
- Small  number of rules, due to general principles
- Independent of specific processing strategies, usable for analysis and generation



# HPSG Basic Principles

- *Lexicalism*: Words carry all the important information about what they can be combined with, thus allowing to deal with regular and idiosyncratic properties in a uniform way
- *Heads*: Phrases contain a head which determines their combinatory potential, e.g. verbs as heads determine what complements must be present, and what modifiers they can combine with
- *Principles*: Few language independent general projection principles stating, e.g., how to combine a head with complements and modifiers
- *Unification*: Monotonically combines constraints from different sources



# HPSG Parsing in Verbmobil

- **active chart parser allowing bidirectional and island parsing on word hypotheses graphs or strings**
- **fast processing by**
  - eliminating disjunctions, enabling fast conjunctive unification
  - precompiling type unifiability, avoiding runtime computations
  - quick checks on mostly relevant features, avoiding full unification
  - quick checks on possibly discontinuous constituents, e.g. separable verb prefixes in German, reducing the chart size
  - precompiling rule filters on possible rule sequences
  - scoring rule applications
- **anytime behavior**
- **robust: best partial analyses even for ungrammatical input**



# Statistical Parser

- Task:

  Robust probabilistic parsing

- Input:

  n-best hypotheses

- Method:

  LR-Parser trained on Verbmobil´s tree-bank

- Result:

  Syntactic tree representation of the input sentence

- Benefit:

  Increasing robustness in Verbmobil´s multi-engine parser strategy

- Responsible:

  Siemens AG



# Statistical Parser – Approach

- (Non-probabilistic) LR-parsing worked quite well for parsing speech in Verbmobil's first phase.
- LR-parsing is well known to be able to parse huge amounts of input very efficiently.
- Probabilistic chart parsing of spontaneous speech input had some problems i.e. the combinatorical explosion of edges in the chart on a word graph

$\Rightarrow$ try probabilistic LR-Parser

# Statistical Parser – Training and Transformations

- **Training process: derivation of an LR table and the estimation of unknown probabilistic parameters from the Verbmobil tree bank**
  - Find the set of all context free rules (G) contained in the tree bank.
  - Construct an LR table from G using well known standard
  - Problems: sparse data, different annotation styles
    ⇒ **eliminate rules that do occur less than N times**
- **Transformations:**
  - Needed after parsing to correct errors of the probabilistic context free parser
  - Rules are learned automatically from the training corpus



# Chunk Parser

- Task:

  Robust and efficient partial parsing, even on ill-formed input
- Input:

  N-best hypotheses
- Method:

  Cascaded Finite State Transducers
- Result:

  Syntactic tree representation of the input sentence
- Benefit:

  Increasing robustness in Verbmobils multi-engine parser strategy
- Responsible:

  Universität Tübingen

# Parsing Based on *Chunks*

***1st Step: Chunk Parsing using Cascaded Finite State Transducers***

***"Chunks are non-recursive cores of 'major' phrases, i.e. NP, VP, PP, …"***

***2nd Step:***

***Building a syntactic tree out of the parsing results***

**Benefit:** **Robust and efficient parsing**

**But:** **Partial parsing: Often no spanning analysis**



# Example for Chunks

*“Ich habe bei meinem letzten Besuch in Hannover so eine nette Kneipe entdeckt”*

***Chunks:***

- **[NX Ich] [VX habe] [PX bei [ NX meinem letzten Besuch]] in [NX Hannover] [PX so [NX eine nette Kneipe]] [VX entdeckt].**

***where***

- **[NX]: Extends from the beginning to the head of a NP**
- **[VX]: Includes all modals, auxiliary verbs and medial adverbs, but ends at the head verb or predicate adjective**
- **[PX]: Extends to the end of an [NX]**



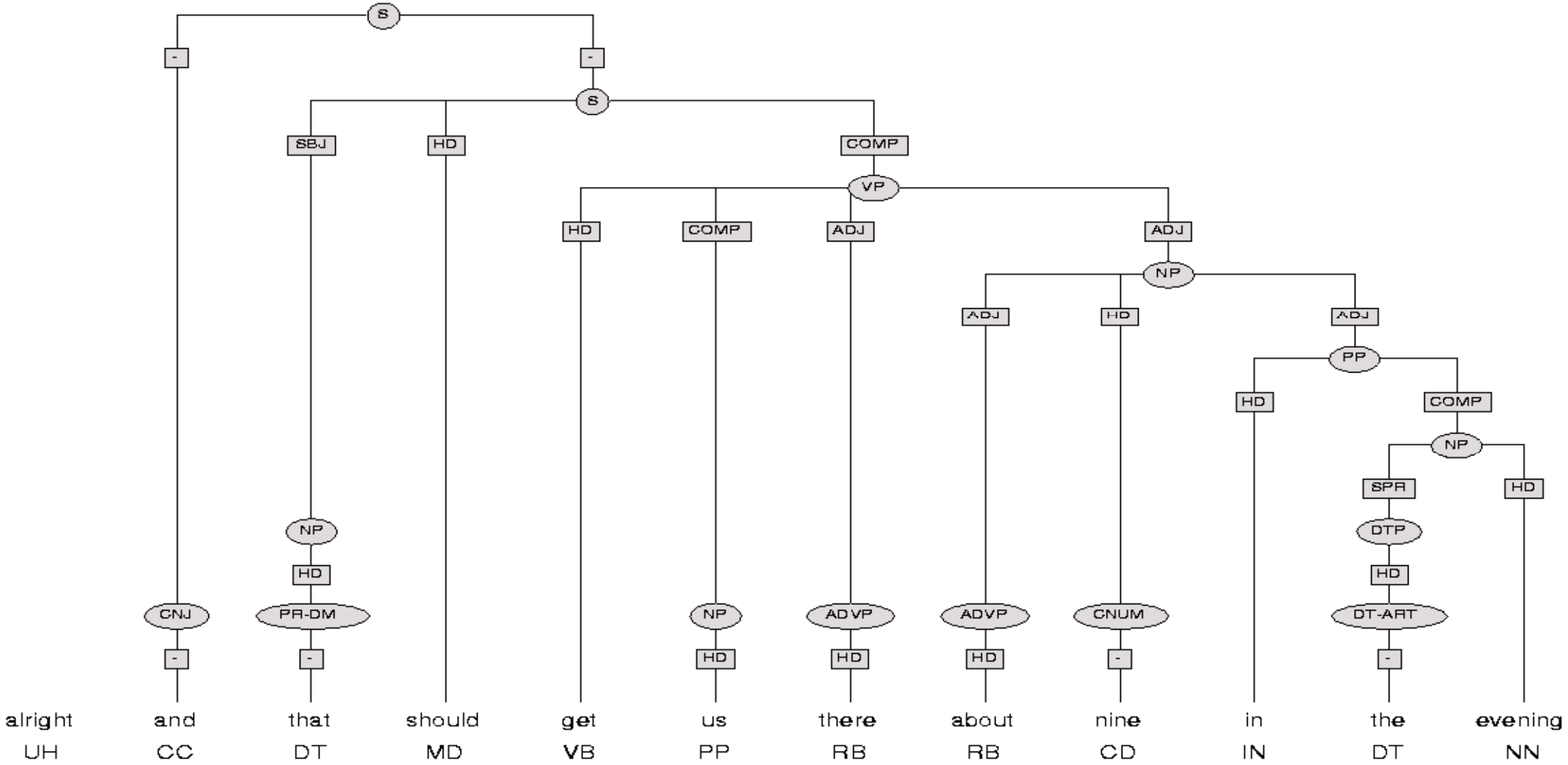
# Tree-Building Tasks

- Determine the chunk position inside the syntactic tree
- Complete the internal chunk structure
- Determine functional categories and topological fields
- Rearrange chunks to obtain a complete syntactic tree



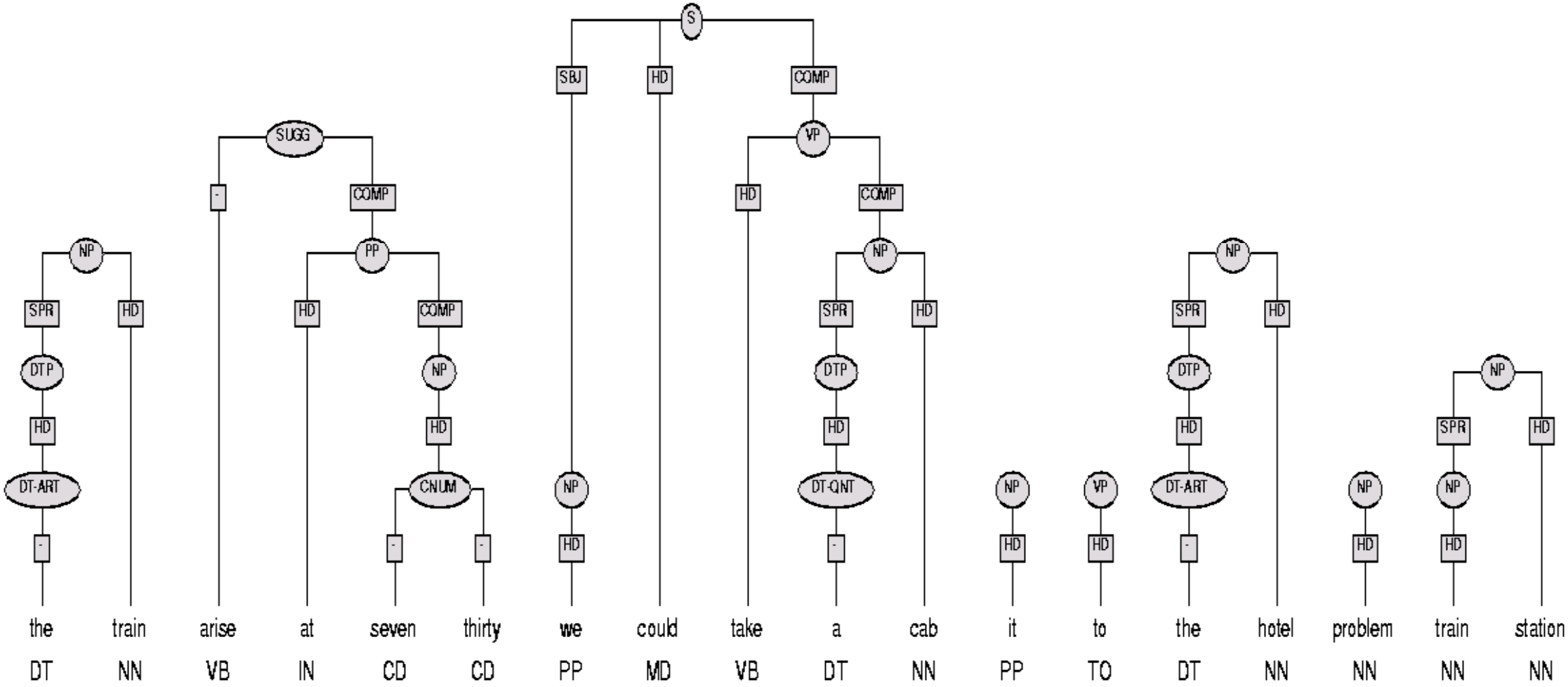
# The Result is a Syntactic Tree

***"Alright, and that should get us there about nine in the evening."***



# ... but analysis is not always spanning

***"The train arise at seven thirty. We could take a cab it to the hotel problem train station."***



# Semantic Construction

- Task:

  Convert and extend syntax trees to VITs
- Input:

  Syntax tree from statistical and chunk parsers
- Method:

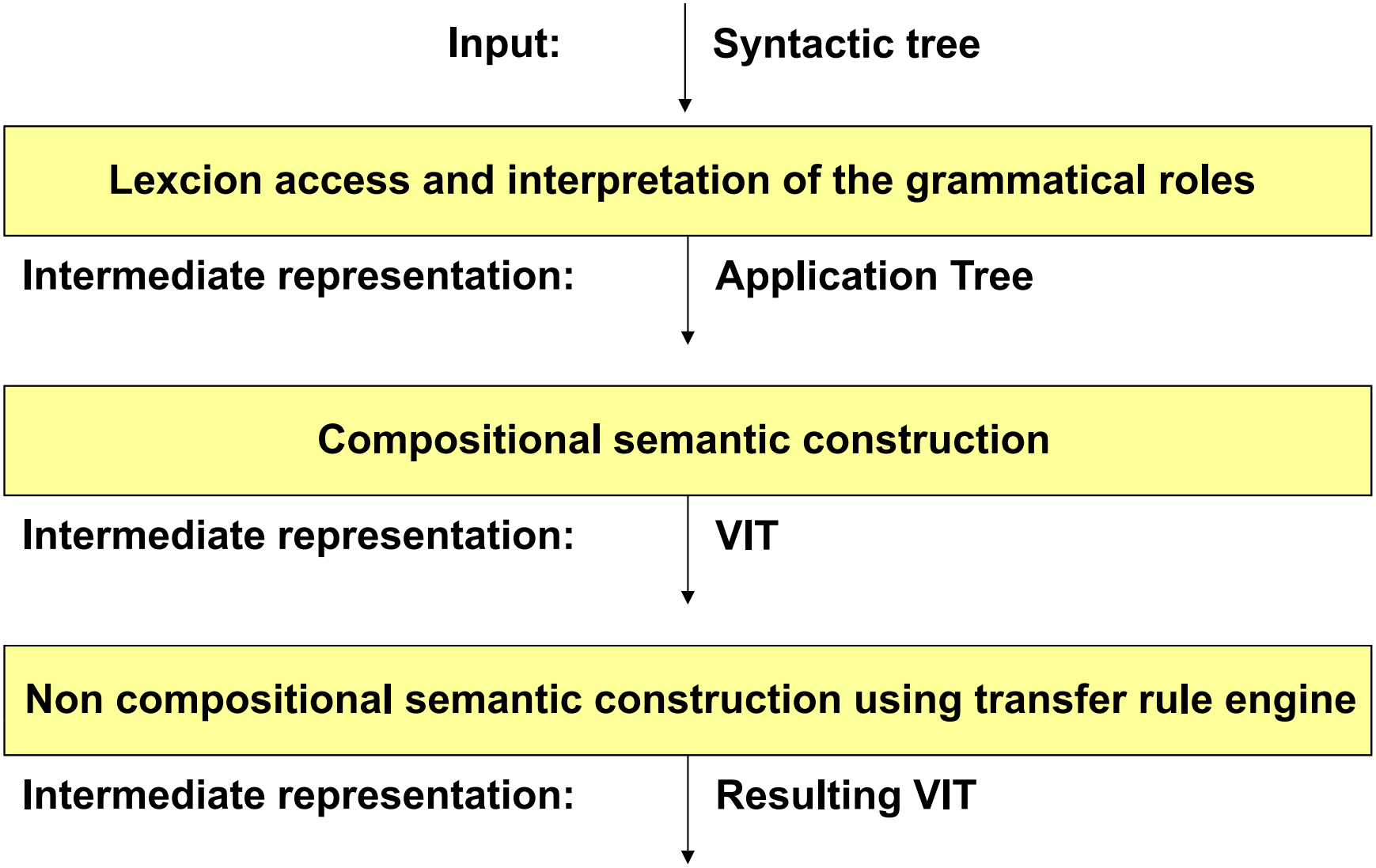
  Compositional construction using semantic lexicon
- Result:

  VITs
- Benefit:

  Providing results of shallow parser to the deep analysis track
- Responsible:

  Universität Stuttgart (IMS)

# Schematic Processing



# Dialog Semantics

- Task:

  Combining results from various parsers, reinterpret and correct VITs, and resolve non-local ambiguities
- Input:

  VITs from different parsers
- Method:

  VIT models and rule based approaches
- Result:

  VIT ready for transfer
- Benefit:

  Enhances robustness of deep analysis and provides vital information for transfer
- Responsible:
  Universität des Saarlandes, Saarbrücken



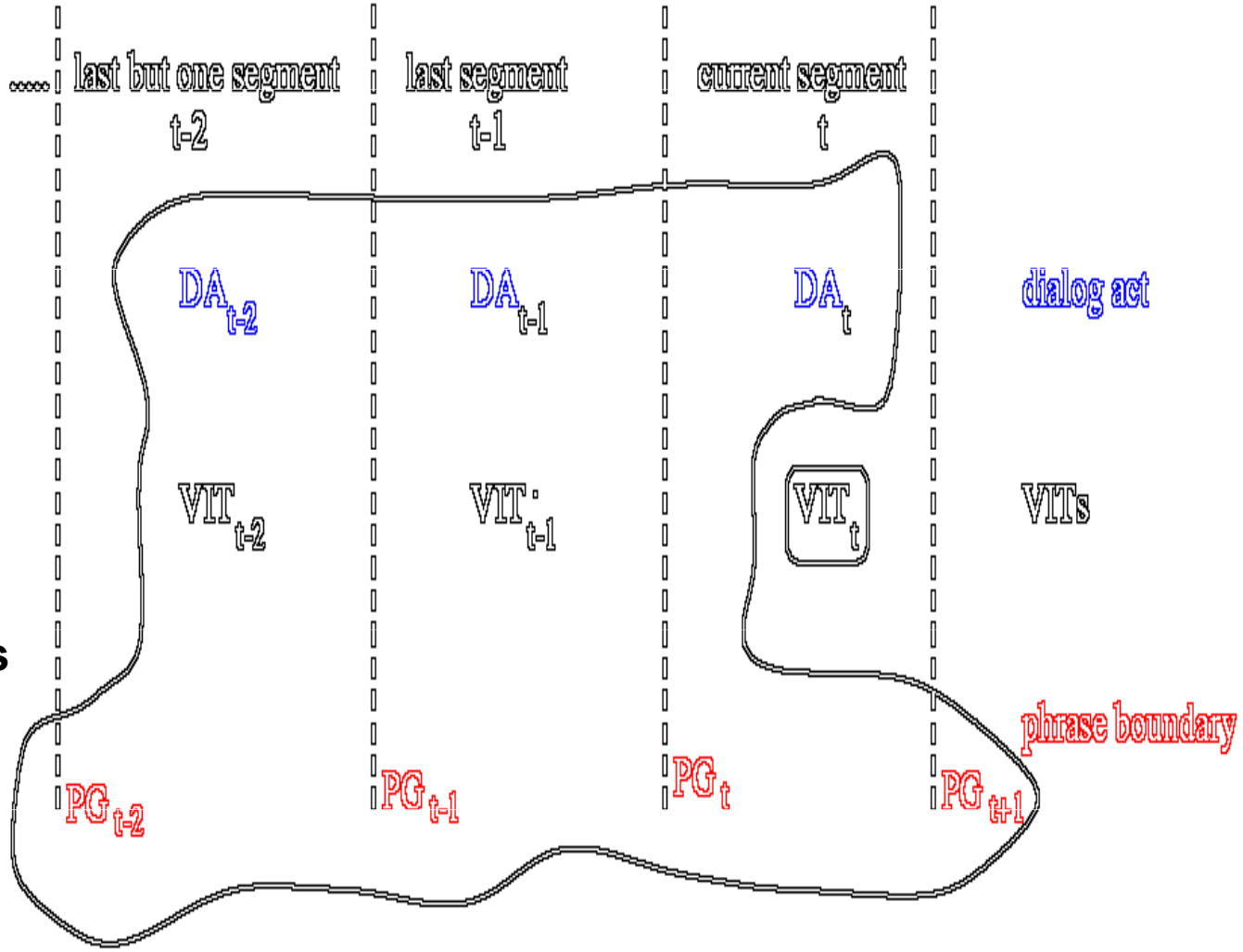
# Combining Analyses from Various Parsers

- **Parsers deliver VITs for segments of a turn**
- **May be spanning analyses or just partial fragments**
- **Combination necessary, both analyses of one parsers, but also analyses from various parsers**
- **Combination criteria**
  - HPSG is better than statistical parsers is better than chunk parser
  - Integrated results are better than fragments
  - Longer results are better than short ones



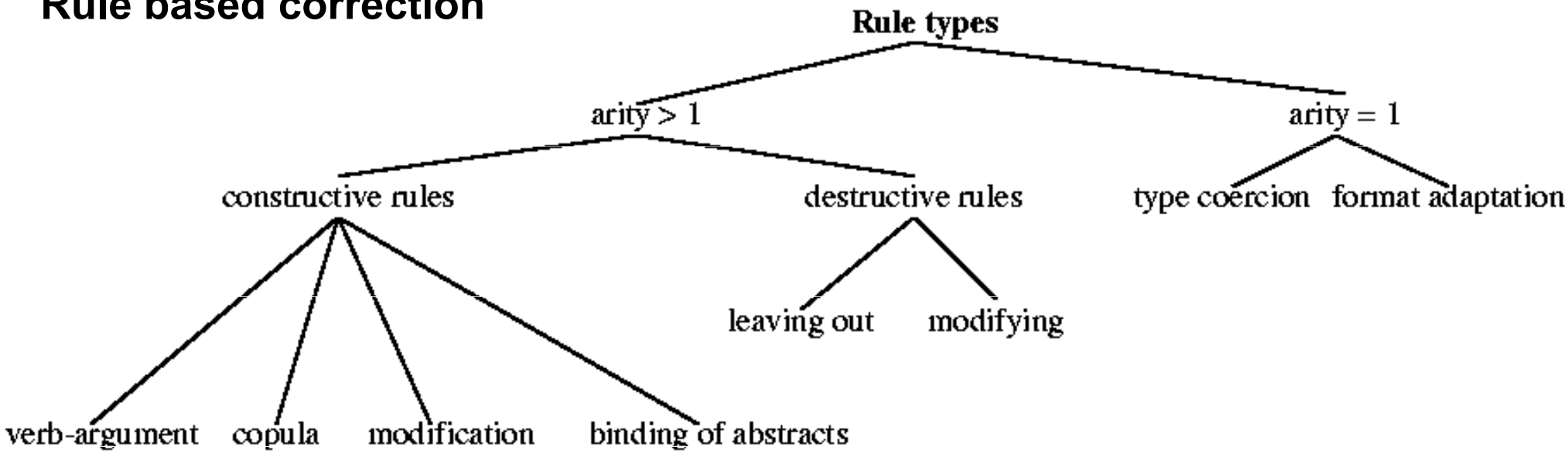
# Stochastic Choice of Spanning Results

- Parser internal scores not normalized ⇒ external scoring necessary
- Statistical model based on VIT content and dialog act (Tetragram language models)
- Search through Vit Hypotheses Graph VHG comparable to search through WHG

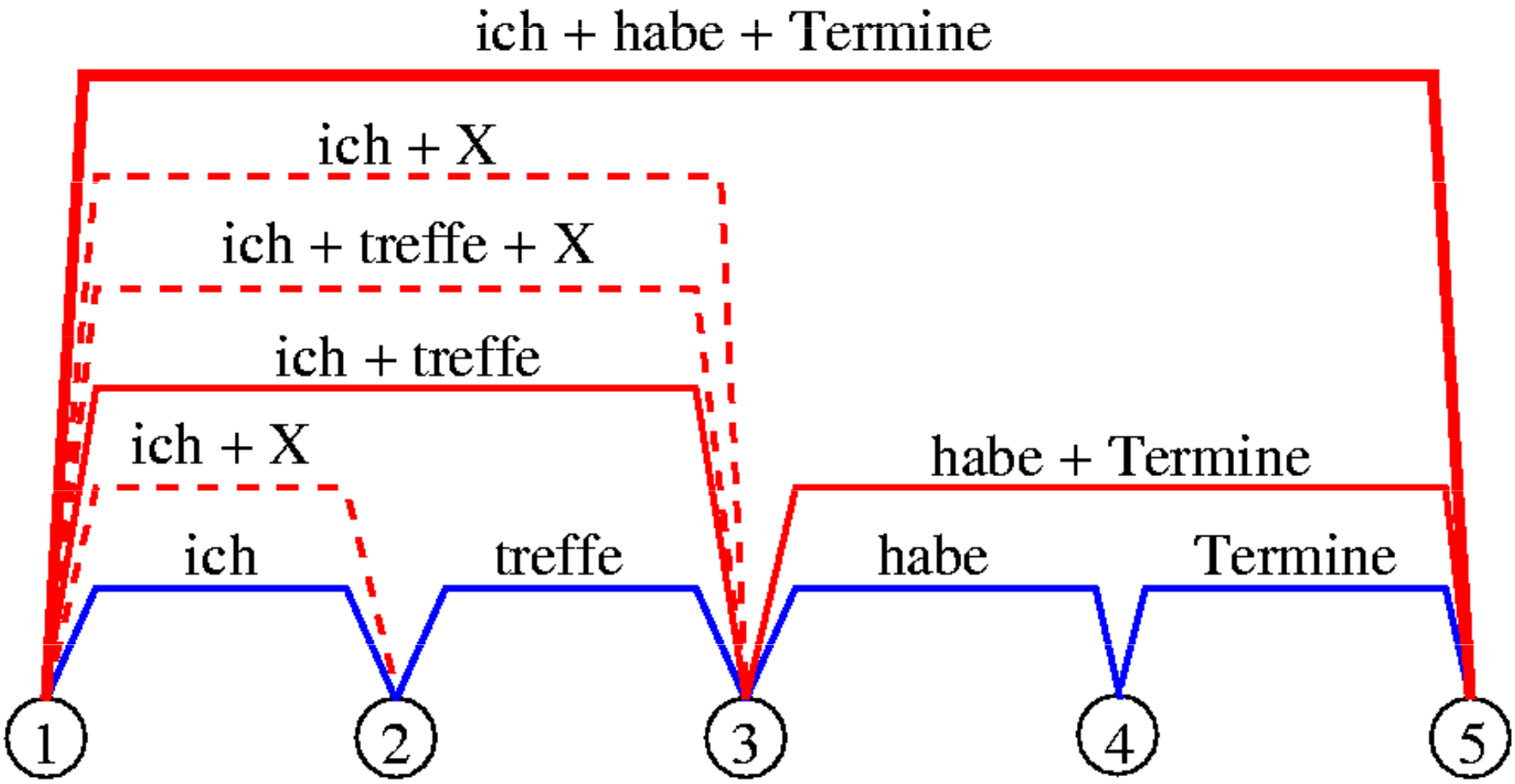


# Robust Semantic Processing

- **Partial results don't necessarily fit together**
  - phenomena of spontaneous speech
  - recognition errors
  - parsing errors
- **Rule based correction**



# Bridging Mechanism for False Starts



# Resolving Non-Local Ambiguities

- **Based on prosody and dialog act information**
- **Ambiguities processed:**
  - Verb disambiguation:
    *Wir gehen in's Theater* (We go to the theater)
    *Montag geht bei mir nicht* (Monday does not suit me)
  - Sentence mood
    *Wir gehen in's Theater* **!** **vs.** *Wir gehen in's Theater***?**
  - **Adverb disambiguation**
    *Wir gehen eher in's Theater* (We go to the theater earlier)
    *Montag geht bei mir eher nicht* (Monday does not really suit me)
  - **Anaphora and ellipsis resolution**
  - **Japanese: Definiteness, topic phrases, zero anaphora**



# Semantic Based Transfer

- Task:

  Transfer VITs from the source to the target language
- Input:

  VITs
- Method:

  Rule based transfer
- Result:

  VITs for generation
- Benefit:

  Translate VITs inside the deep translation path
- Responsible:

  Universität Stuttgart (IMS)



# The Transfer Approach: Rule Based Transfer

- VITs are mapped onto VITs: Transfer is a VIT rewriting system
- Rule based, context conditions restrict application
- Transfer rules remove matching source language expressions from the VIT
- Efficient implementation
- Examples:
  - Simple Rules: `adelig(L,I) -> noble(L,I)`
  - Simple Templates: `@mod(adelig, noble, L, I)`
  - Selectional restrictions:

```
#sort_check(I,human) -> true
@mod(gross,tall,_,I)
#sort_check(I,location) -> true
@mod(gross,large,_,I)
```



# Advanced Features of Transfer

- **Structural changes:**
  - Adjective to PP: tagsüber -> during the day
  - Insertion: übernachte -> spend the night
  - …
- **Disambiguation:**

| type of ambiguity | kinds of knowledge needed for disambiguation | modules that contribute to the resolution |
|---|---|---|
| lexical | syntactic, semantic, contrastive, domain, prosodic | parsers, semantic construction, discourse semantics, transfer, context |
| structural | syntactic, semantic, domain | parsers, semantic construction, transfer |
| anaphora and ellipsis | syntactic, semantic, domain | discourse semantics, context |
| semantic focus and operator scope | prosodic, syntactic, semantic, contrastive, domain | discourse semantics transfer |

# Performance of Transfer

- **Rules are compiled and packed**
- **18088 rules German ⇔ English**
- **4694 rules German ⇔ Japanese**
- **Mean runtime per sentence: 80 msec (Sun Ultra II, 300 MHz)**



# Context Evaluation

- Task:

  Resolving ambiguities in the dialog context during semantic transfer
- Input:

  Requests from transfer
- Method:

  Using world knowledge and rules
- Result:

  disambiguated transfer requests
- Benefit:

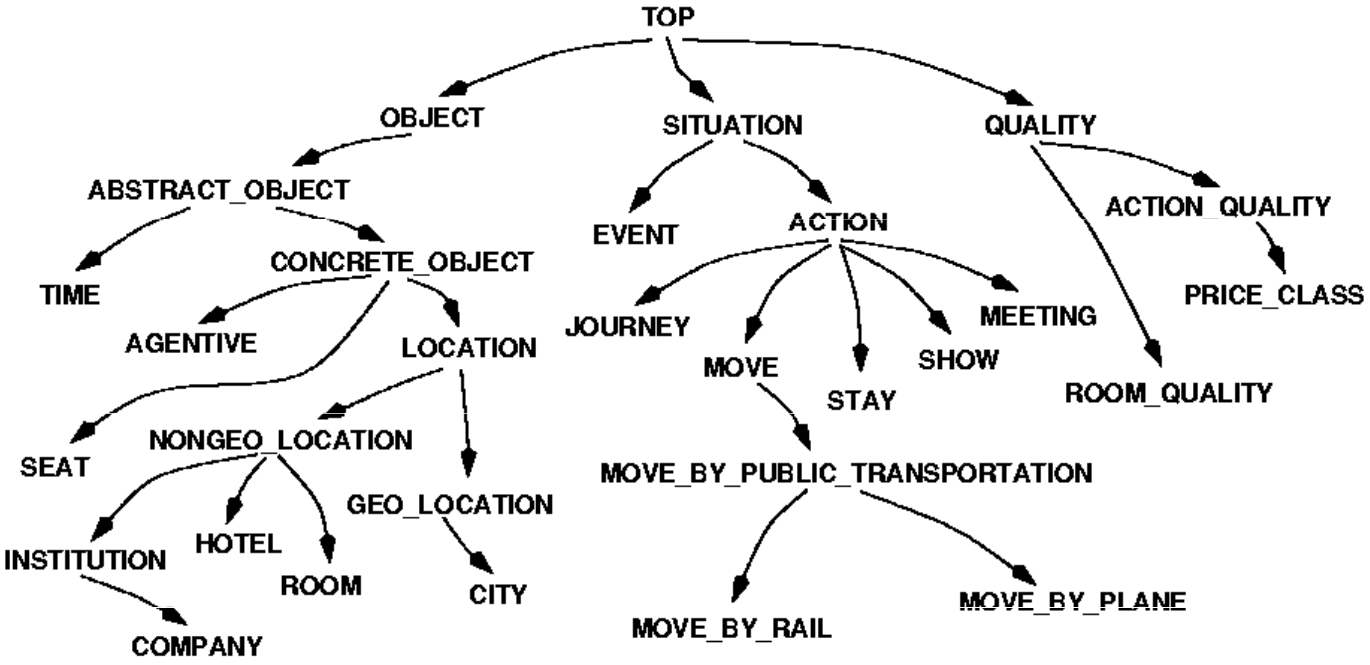
  Higher quality of transfer results
- Responsible:

  Technical University (TU) Berlin



# Context Evaluation - Tasks and Methods

- Supports semantic transfers and processes VITs
- Gets information from dialog module from shallow tracks
- Extends disambiguation of the dialog semantic module and uses ontological information



# Using World Knowledge for Transfer

**Example: Platz ➔ room / table / seat**

1. Nehmen wir dieses Hotel, ja. ➔ Let us take this hotel.
   Ich reserviere einen **Platz**. ➔ I will reserve a **room**.

2. Machen wir das Abendessen dort. ➔ Let us have dinner there.
   Ich reserviere einen **Platz**. ➔ I will reserve a **table**.

3. Gehen wir ins Theater. ➔ Let us go to the theater.
   Ich möchte **Plätze** reservieren. ➔ I would like to reserve **seats**.

# Dialog Processing

- Task:

  Provides dialog context for all tracks and computes main information for dialog summaries
- Input:

  Data from a lot of modules
- Method:

  Frame-like topic structuring and rules
- Result:

  context information and dialog summaries and minutes
- Benefit:

  Verbmobil knows what happens throughout the dialog and can present it
- Responsible:
  DFKI, Saarbrücken

# Dialog Processing

- **Dialog Memory:**
  - Stores information from each track
  - Only dialog act based and semantic transfer provide abstract representations:
    Discourse Representation Language DRL:

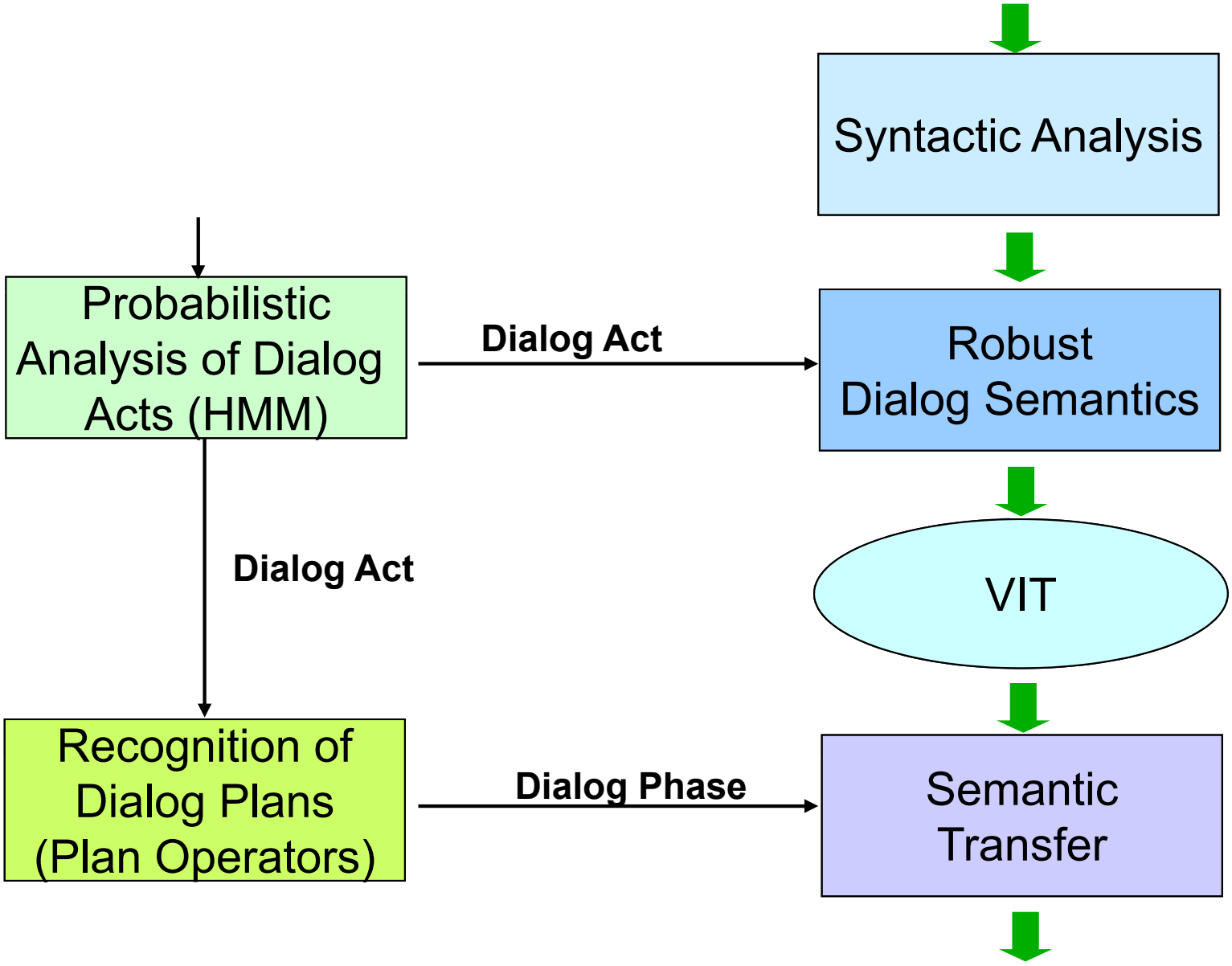
    *I would so we were to leave Hamburg on the first*

    ```
    [INFORM,has_move:[move,has_source_location:[city,has_name='hamburg',
                                    has_departure_time:[date,time='day:1'
    ```

- **Discourse Interpretation:**
  - Groups information into topics
  - Completes information
  - Keeps tracks of negotiation structure



# Dialog Information in Semantic Transfer

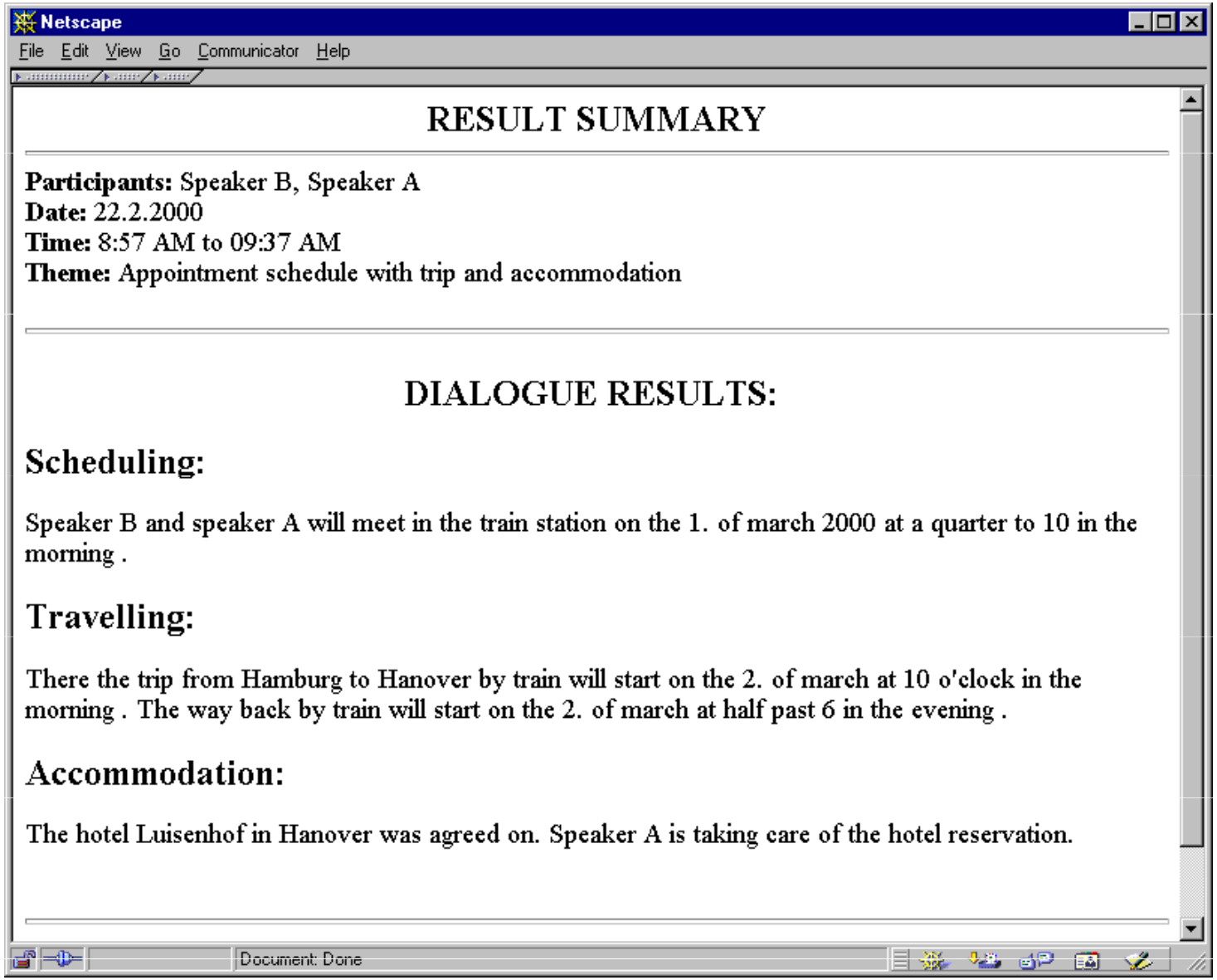


# Collaboration for a New Functionality: Result Summaries

- **Provide the users with a summary of the topics that were agreed**
- **Two benefits**
  - have a piece of information to use in calendars etc.
  - control the translation
- **Approach: exploit already existing modules for**
  - content extraction
  - dialog interpretation
  - planning the summary
  - generation
  - transfer



# Result Summary



# Generation

- Task:

  Robustly generate the output of the semantic transfer in German, English, or Japanese
- Input:

  VITs from transfer
- Method:

  Constraint system for micro-planning, TAG grammar (reusing HPSG grammars) for syntactic realization
- Result:

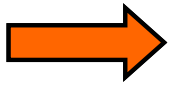
  Strings, enriched with content-to-speech (CTS) information to support synthesis
- Benefit:

  Output from the semantic transfer track
- Responsible:
  DFKI, Saarbrücken



# Architecture

**VIT** (Verbmobil Interface Term)



| | |
|---|---|
| ***Robustness Preprocessing*** | ***•Repairing structural problems<br>•Heuristics for generation gap*** |

| | |
|---|---|
| *Microplanning Module* | *•Selecting planning rules<br>•Lexical choice constraints* |
| *Syntactic Realization Module* | *•Selecting LTAG trees<br>•Tree combination* |
| *Surface Realization Module* | *•Inflection<br>•Synthesis Annotation* |

**Annotated String**

# Preprocessing for Robustness

**Why *pre*-pocessing:**

- **Check and repair inconsistencies as early as possible**
- **Keep robustness and standard modules separate**
- **Alternative: relax constraints**

**Preprocessing for robustness means:**

- **Executing a set of solution submodules in sequence**
- **For each problem found, the preprocessor lowers a *confidence value* for the generation output which measures the reliability of our result**



# How much robustness?

- **PRO:**

  **In a dialog system, a poor translation might still be *better than none* at all,**

- **CON:**

  **one of the shallow modules can be selected when deep processing fails,**

  **so respect the *inherent limitations of robustness*.**

  **⇒ Generation knows its limits and sometimes decides not to produce a string**

- **Selection module: uses training corpus and confidence values to select from the different translation paths**



# Microplanning: Create Syntactic Building Blocks

**Method: Mapping of dependency structures**

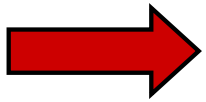
**Example: Time Expressions**

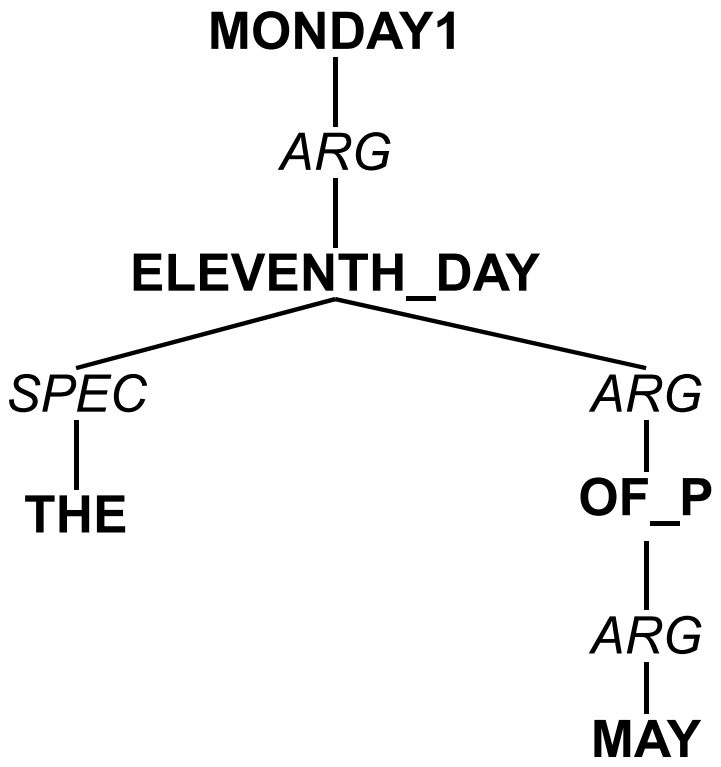
**DEF (L,I,G,H)**

**DOWF (L1,I,mo)**

**ORD (L2,I,11)**

**MOFY (L3,I,may)**






**Semantical dependency: VIT**

**Syntactical dependency: TAG**



# Multilingual Generation for Translation in Speech-to-Speech Dialogues and its Realization in Verbmobil

**Tilman Becker . Anne Kilger . Peter Poller . Patrice Lopez**

DFKI GmbH

Stuhlsatzenhausweg 3

66123 Saarbrücken

Tilman.Becker@dfki.de



# VM-GECO:
# VerbMobil's GEneration COmponents

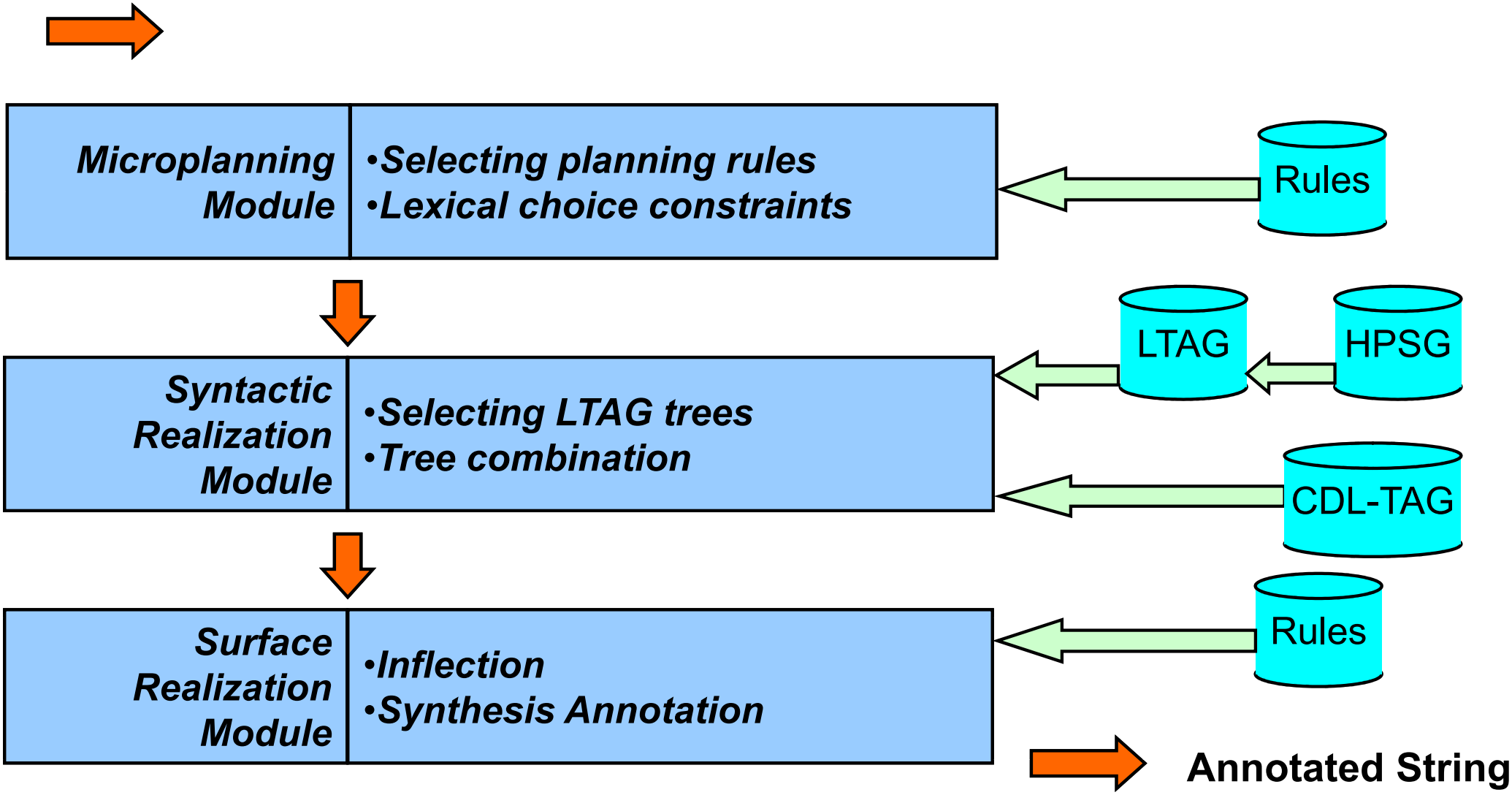
- Multilingual Generation: German, English, Japanese
- Language-independent kernel algorithms
- Language-specific knowlegde sources
- Extended "standard" pipeline architecture:
  - Microplanning
  - Syntactic Realization
  - Surface Realization

**Annotated String**



# Standard Architecture

**VIT** (Verbmobil Interface Term)

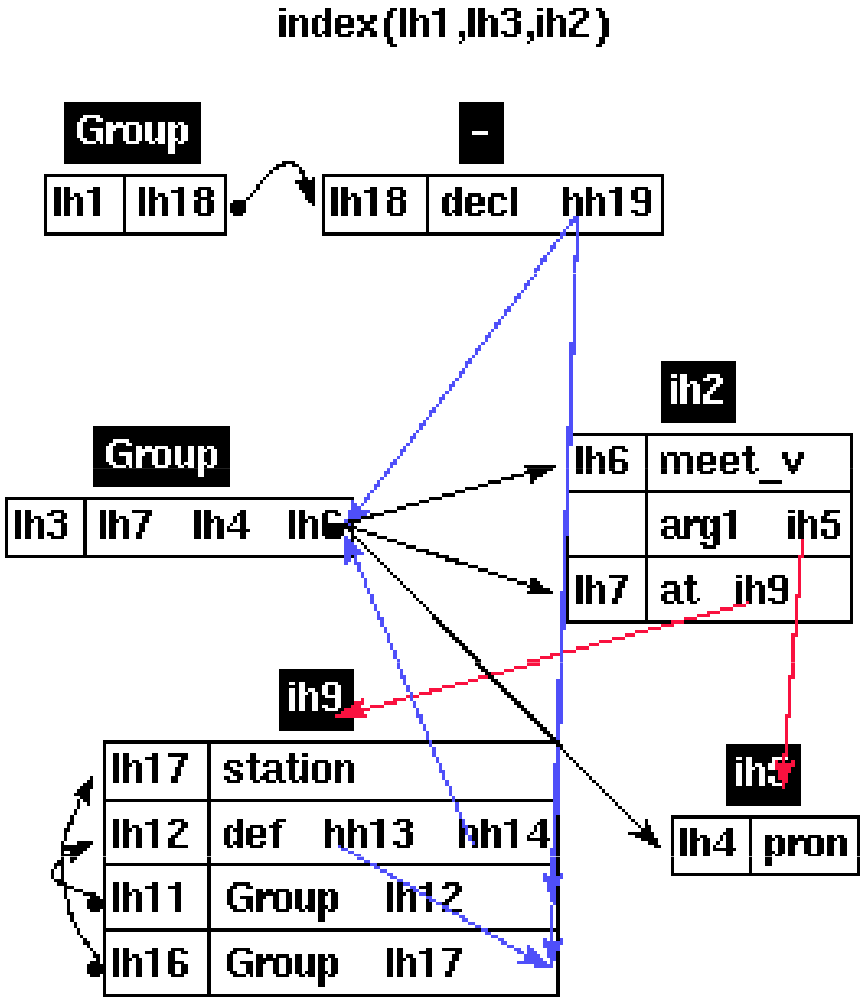


# VIT: Verbmobil Interface Term

```
vit(vitID(sid(...),                                  %Segment ID
    []),                                             %WHG-String
    index(l250,l234,i72),                            %Index
    [start_v(l248,i72),                              %Conditions
     arg1(l248,i72,i75),
     nop(l240,h85),
     quest(l249,h84),
     time(l238,i73),
     abstr_vacation(l247,i75),
     pron(l242,i74),
     poss(l244,i75,i74),
     temp_loc(l239,i72,i73),
     def(l245,i75,h87,h86),
     whq(l235,i73,h83,h82)],
    [in_g(l235,l237), ...                            %Constraints
     leq(l234,h85), ...],
    [s_class(l240,mp), ...],                         %Sorts
    [ana_ante(i74,[i75,i69,i67,i66]),                %Discourse
     prontype(i74,third,std), ...],
    [gend(i75,masc), num(i75,sg)],                   %Syntax
    [ta_mood(i72,ind), ...],                         %Tense and Aspect
    [...])                                           %Prosody
```

*When do your vacations begin?*

# VIT: Verbmobil Interface Term




*We meet at the station.*

# Microplanning: deriving a sentence plan

- Microplanning tasks:
- determine type of utterance
- determine syntactic structure
- execute word choice

- Microplanning rules map parts of VIT input to partial dependency structures

- Implemented as constraint solving problem
- Approx. 7,200 microplanning rules (German)



# Microplanning: deriving a sentence plan

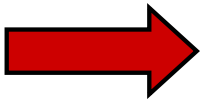
- **An example:** *"the eleventh of May"*

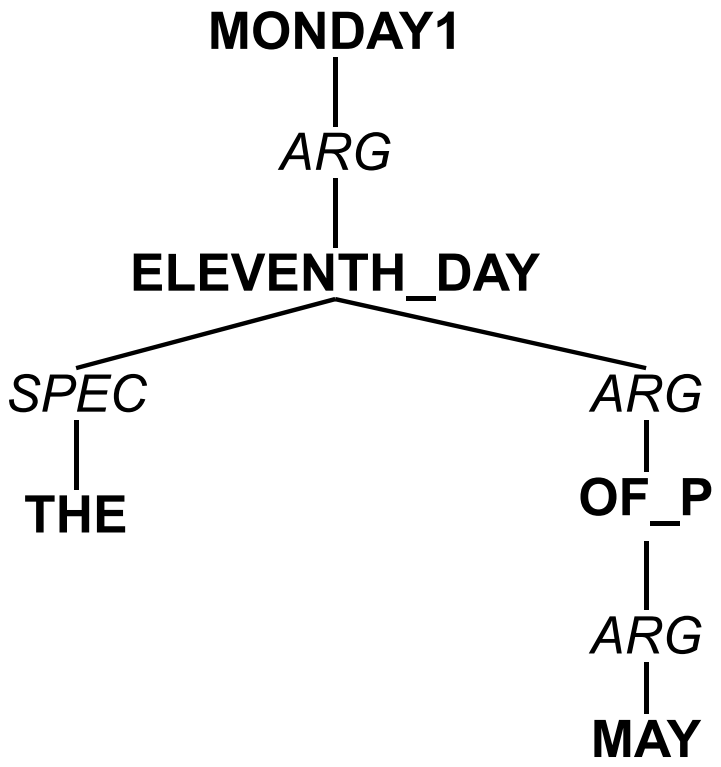
**DEF (L,I,G,H)**

**DOWF (L1,I,mo)**

**ORD (L2,I,11)**

**MOFY (L3,I,may)**






**Semantic dependency: VIT**

**Syntactic dependency: TAG**

# Syntactic Realization

- **Tasks of syntactic realization:**
- **selecting lexicalized (TAG) trees**
- **constructing a phrase structure tree**
- **provide all information for surface realization:**
  - inflection and annotation for CTS (content to speech) synthesis

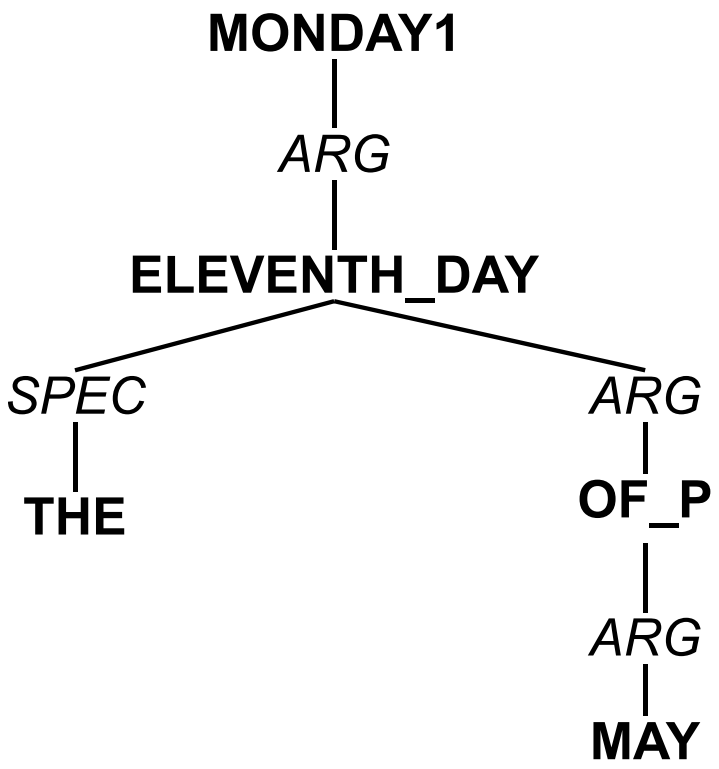
- **Based on FB-LTAG:**

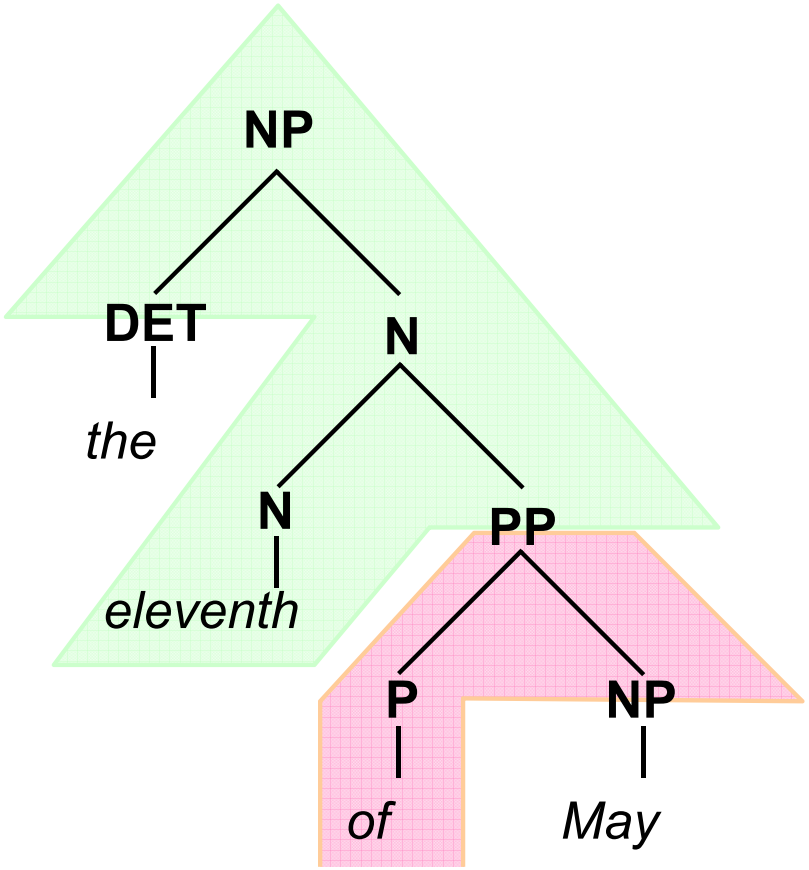
  **Feature-Based Lexicalized Tree Adjoining Grammars**
- **Compiled from HPSG grammars**



# Syntactic Realization:

- **An example:** *"the eleventh of May"*




**Syntactic dependency:**

**TAG derivation tree**

**Syntactic phrase structure:**

**TAG derived tree**

# HPSG to TAG Compilation

- **HPSG: context-free rules (schemas)**
- **TAG: extended local lexical structures (trees)**

- **Off-line compilation computes all projections from lexical types**
- **Generates approx. 2,300 TAG trees from 250 lexical types**
  - Reuse existing Resources:
    - Spontaneous speech, syntactic/lexical coverage of Verbmobil domain
  - Speed vs. space
  - TAG captures dependencies
  - HPSG include syntax-semantics interface, vast body of linguistic work



# Problems for generation

- **Technical problems**
  - should be eliminated
  - hard to eliminate in a large-scale system
  - better to be robust
- **Task-inherent problems**
  - Spontaneous speech input
  - Insufficiencies in the analysis and translation
  - Generation gap:
    mismatch between semantic input and coverage of the grammar

→ **Robust generator necessary**



# Problems for generation (2)

**(Task-inherent) problems manifest themselves as fault**

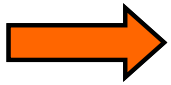
**wrt. the interface language definition**

- **Problems with the *structure* of the semantic representation:**
  - unconnected subgraphs
  - multiple predicates referring to the same object
  - omission of obligatory arguments
- **Problems with the *content* of the semantic representation:**
  - contradicting information
  - missing information (e.g. agreement information)



# Extended Architecture

**VIT** (Verbmobil Interface Term)



| | |
|---|---|
| ***Robustness Preprocessing*** | ***•Repairing structural problems<br>•Heuristics for generation gap*** |

| | |
|---|---|
| *Microplanning Module* | *•Selecting planning rules<br>•Lexical choice constraints* |
| *Syntactic Realization Module* | *•Selecting LTAG trees<br>•Tree combination* |
| *Surface Realization Module* | *•Inflection<br>•Synthesis Annotation* |

**Annotated String**

# Extended Architecture (2)

**Why *pre*-pocessing:**

- **Check and repair inconsistencies as early as possible**
- **Keep robustness and standard modules separate**
- **Alternative: relax constraints**

**Preprocessing for robustness means:**

- **Executing a set of solution submodules in sequence**
- **For each problem found, the preprocessor lowers a *confidence value* for the generation output which measures the reliability of our result**



# How much robustness?

- **PRO:**

  **In a dialogue system,**

  **a poor translation might still be *better than none* at all,**

- **CON:**

  **one of the shallow modules can be selected when deep processing fails,**

  **so respect the *inherent limitations of robustness*.**

- **Selection module: uses training corpus and confidence values to select from the different translation paths**



# Content-to-Speech (CTS) Output

- **Output annotated with information like speech act, syntactic grouping, word classes, prominence, ...**
- **Enhances synthesis quality**
- **Example:**

{SpeechAct:begin}{SpeechActType: Inform}{Language:English}{Utterance:begin} {SentenceType:Aussagesatz}{WordClass:N}Verbmobil{WordClass:AUX}is {WordClass: DET-ART} a{Prominence:2} {WordClass:ADJ}speaker_independent{WordClass:N} system{BorderProminence:5} {WordClass:CONJ-SYN}that {Prominence:15}{WordClass:V}offers {Prominence:4}{WordClass:N}translation_assistance{BorderProminence:2} {WordClass:PREP-SYN}in {Prominence:4}{WordClass:N}dialog {WordClass:N}situations {Utterance:end}



# Minutes and Summaries

- Dialog module keeps track of the dialog:

  dialog model, context extraction, translations: dialog history

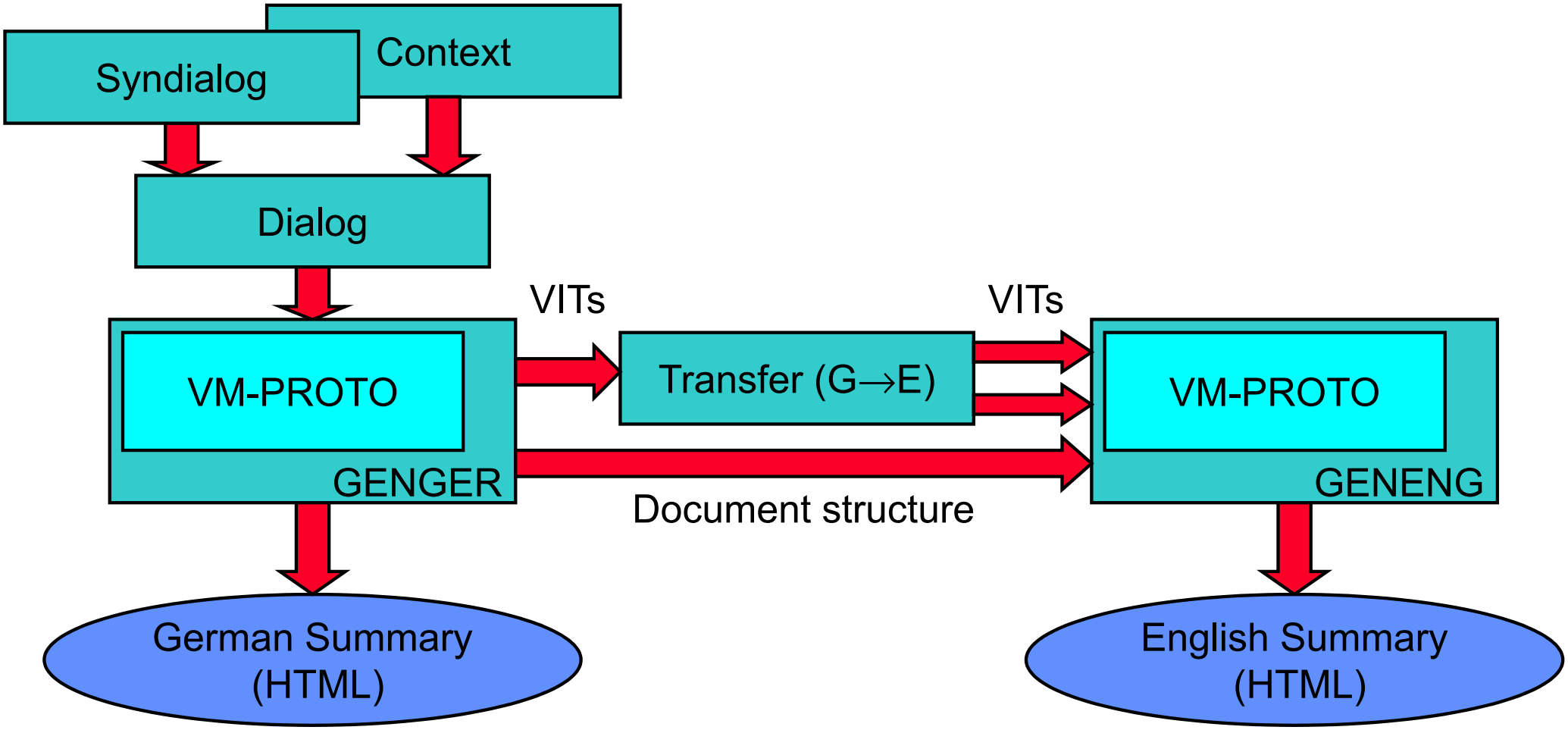
- Three types of “protocols”:

- Minutes: relevant exchanges
- Summary: dialog results
- Scripts: complete dialog script



# Multilingual Minutes and Summaries

- **Multilinguality: Integration of transfer module:**



# Conclusion

- **Multilingual generation:**
  - kernel algorithms
  - multilingual knowledge sources
- **Robustness is necessary and useful**
  - within limits
- **Output of classified, graded quality**
- **Generation of minutes and summaries**

- **The Verbmobil book: 2 articles on Generation**



# Selection and Speech Synthesis



# Selection of Translations

- Task:

  Select the “best” translation out of all deep and shallow translation paths
- Input:

  Translations (text or content)
- Method:

  Learning inequalities
- Result:

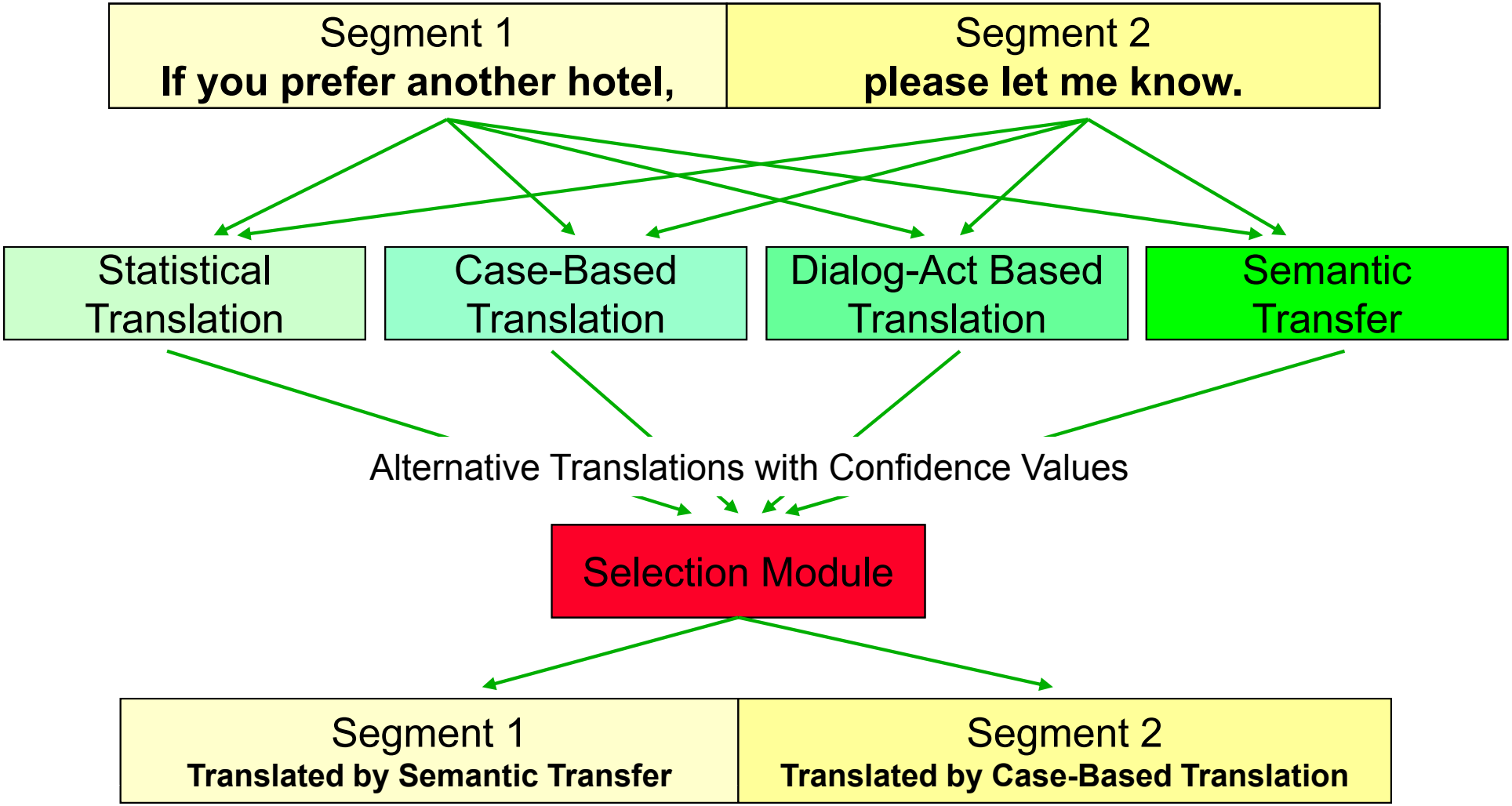
  Selected Translation (text or content)
- Benefit:

  Use the expertise of all translation paths for a particular utterance
- Responsible:

  TU Berlin



# Integrating Deep and Shallow Processing



# The Selection Problem

**Selection is a difficult business:**

- **confidence values are difficult to compare**
  - probabilistic vs. knowledge based approaches
  - no bird's eyes view possible
- **re-training necessary after changes in the engines**
- **training data must be produced**



# Speech Synthesis

- Task:

  Synthesize the translation

- Input:

  text or content

- Method:

  Multilevel selection and concatenation of speech units from large speech corpora

- Result:

  Audio signal

- Benefit:

  “End of the chain” of the speech-to-speech system

- Responsible:

  Universität Bonn
  TU Dresden
  Universität Bochum
  Daimler Chrysler



# Different Types of Synthesis

- ***Text-to-Speech (TTS):*** **reading machine from arbitrary text in orthographic form. Unlimited domain. The machine does not know what it is saying.**
- ***Concept-to-Speech*** **[or *content-to-speech*] *(CTS):* spoken out-put from a database inquiry or from a dialog system. The input of the synthesizer comes from a semantic representation via a generation module. The machine should have full knowledge of what it is saying.**
- ***Reproductive Speech Synthesis:*** **spoken output from pre-recorded samples. For strictly limited domains.**



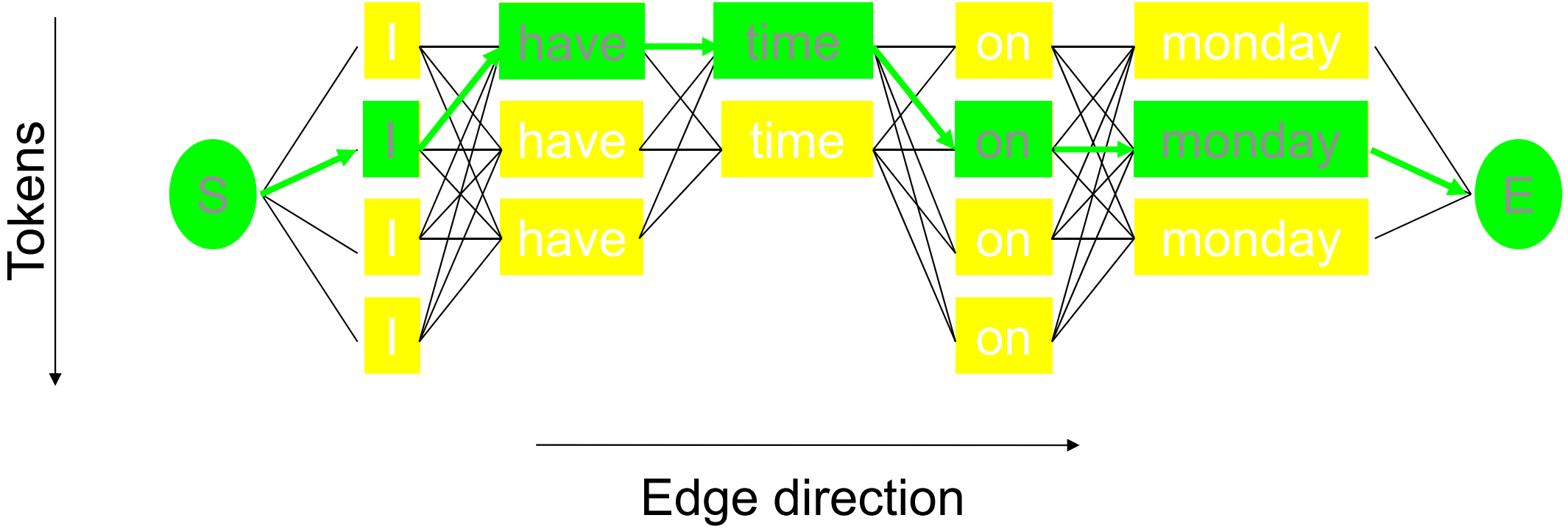
# Corpus-Based Synthesis

- Target utterances are synthesized from a corpus of utterances from within the domain.
- All units – whatever they are – have multiple instances in the corpus.
- No predefined units: the unit selection algorithm selects contiguous chunks of speech from the data base – the longer, the better.
- When units of word size and above are applied, much of the natural prosody is preserved.
- Problem: coverage. Words not in the database cannot be synthesized in this way.



# Unit Selection Algorithm

Sentence to synthesize: I have time on monday.

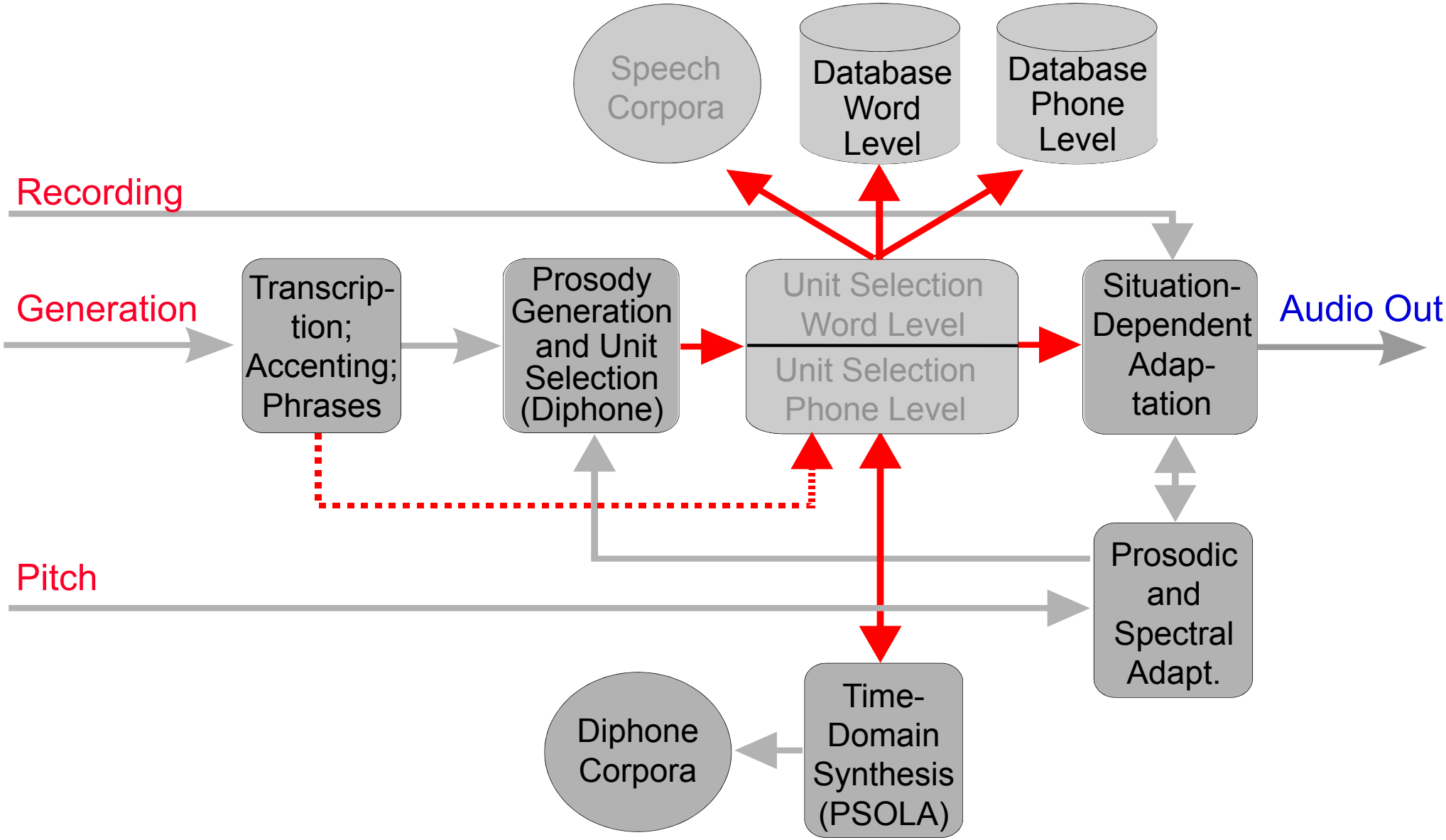


# Implementation

- *Word* is the central unit and the starting point for all processing.
- Only if no suitable instance of a word is available in the database, an algorithm is invoked that composes a word from subword units which are currently phones.
- The principal strategy on both the word and the sub-word levels is to concatenate chunks that are as long as possible (up to a whole sentence).
- Like in CHATR, no prosodic manipulation is performed in this synthesis.
- In principle each word is needed in up to three positions (initial/medial, final declarative, final interrogative) and in both accented and unaccented mode.
- For Verbmobil this would mean that we need about 80000 word tokens to be recorded (which is prohibitive).
- Good coverage is reached by a selection of typical phrases from within the domain (dialogs from the Verbmobil dialog database).
- Additional utterances realize frequent words in relevant contexts (e.g., opening phrase, names of big cities).



# Architecture



# Verbmobil From a Software Engineering Point of View

**System Design and Software Integration**



# Software Technology Challenges

## The goal

- Build an integrated system

## The situation

- Researchers do research
- Using different programming languages
- Researchers don’t want to be bothered with technical details

## The solution

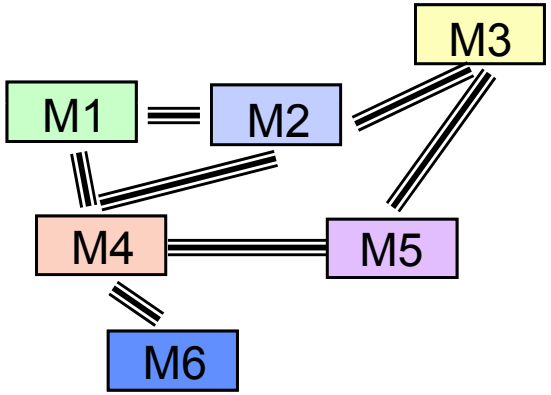
- Introducing: the System Group
- Maximal technical support for the researchers/developers



# The System Architecture

## Verbmobil I

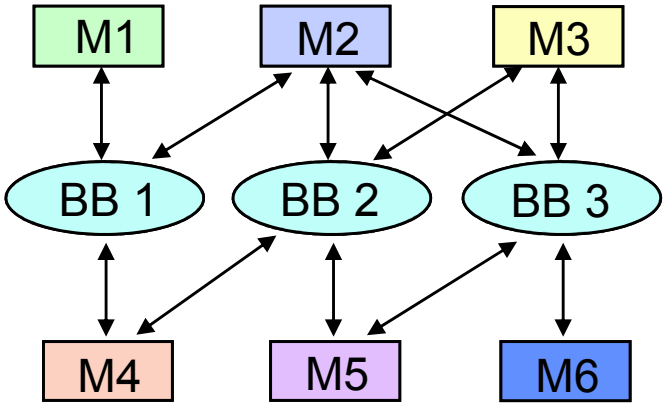
**Multi-Agent Architecture**




- Modules know all communication partners
- Direct communication between modules
- Reconfiguration difficult
- Software: ICE and ICE Master
- Basic Platform: PVM

## Verbmobil II

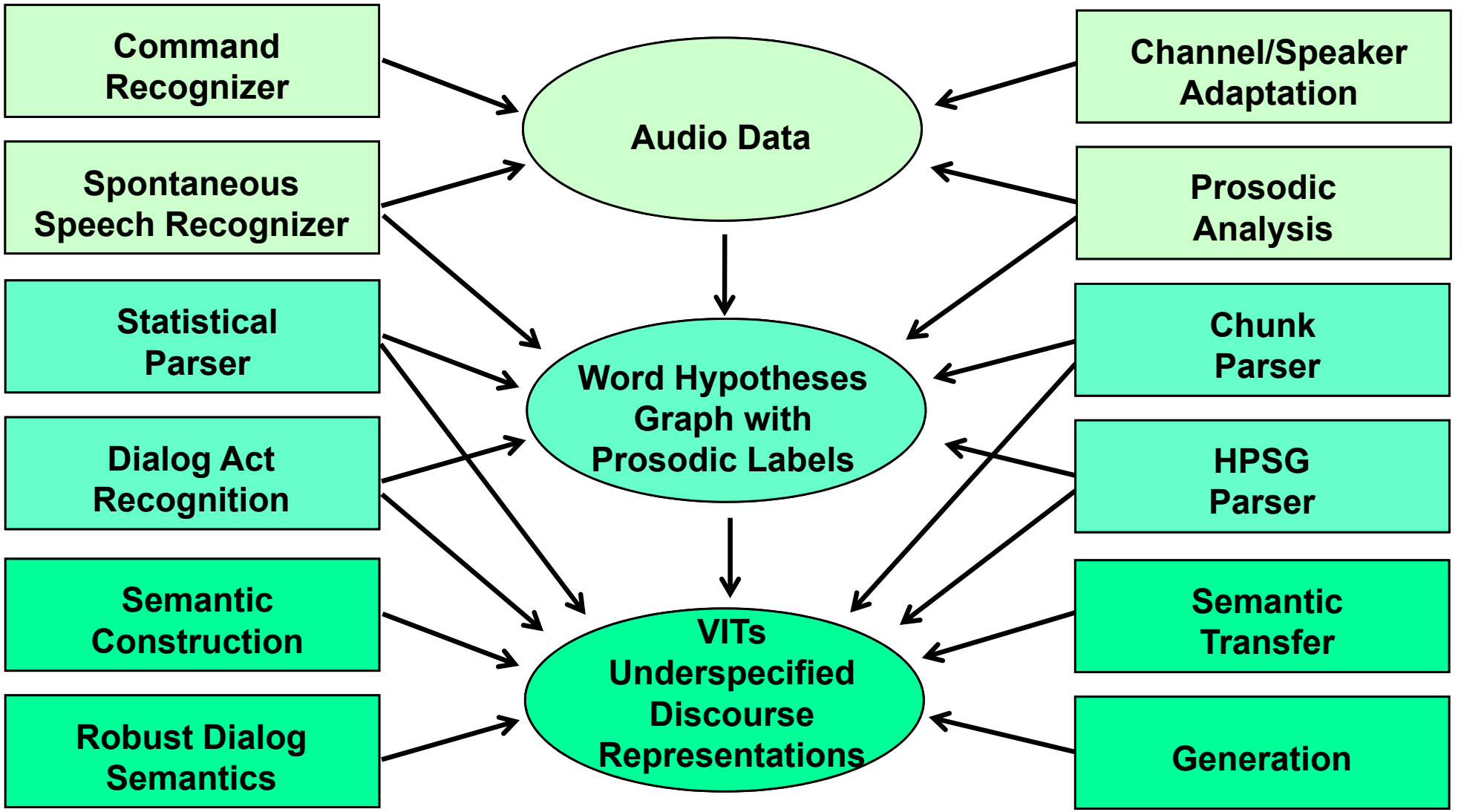
**Multi-Blackboard Architecture**



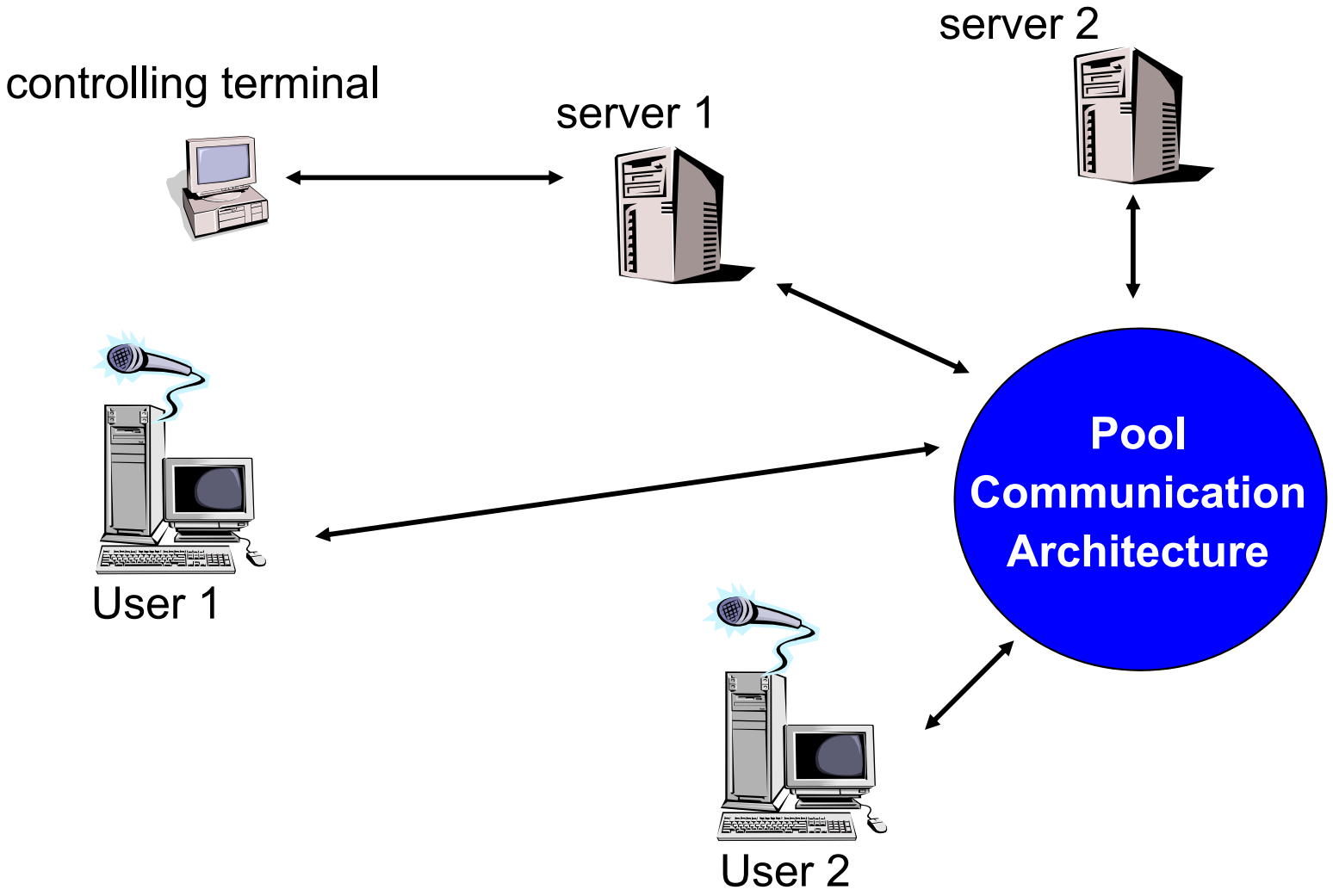

Blackboards

- Modules know their I/O data pools
- No direct communication between modules
  **198 blackboards vs. 2380 direct comm. paths**
- Reconfiguration easy
- Several instances of one module/functionality
- Software: PCA and Module Manager
- Basic Platform: PVM

# Sample Pool Structure



# Distributed Execution Supports Distributed Development



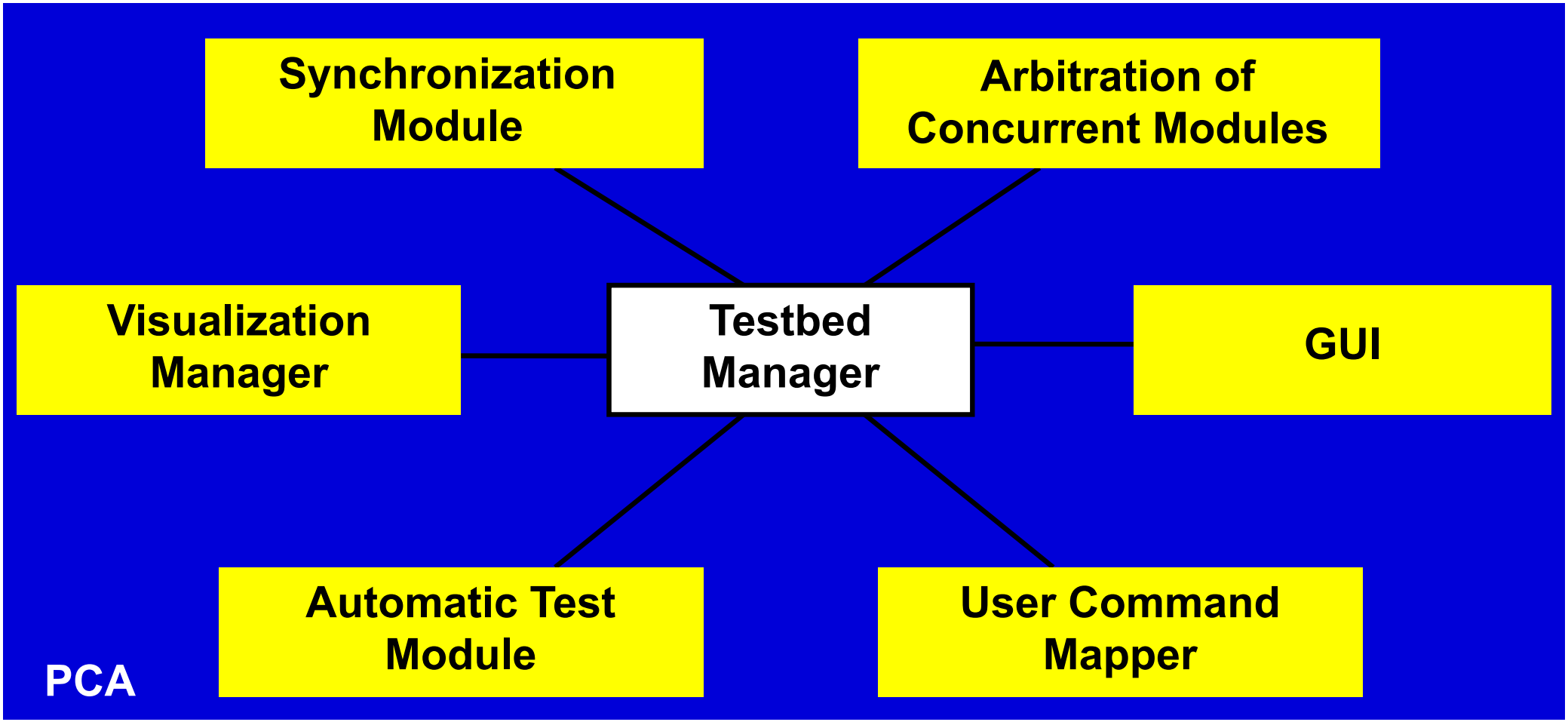
# Support from the System Group (1)

**Integration framework (Testbed) with**

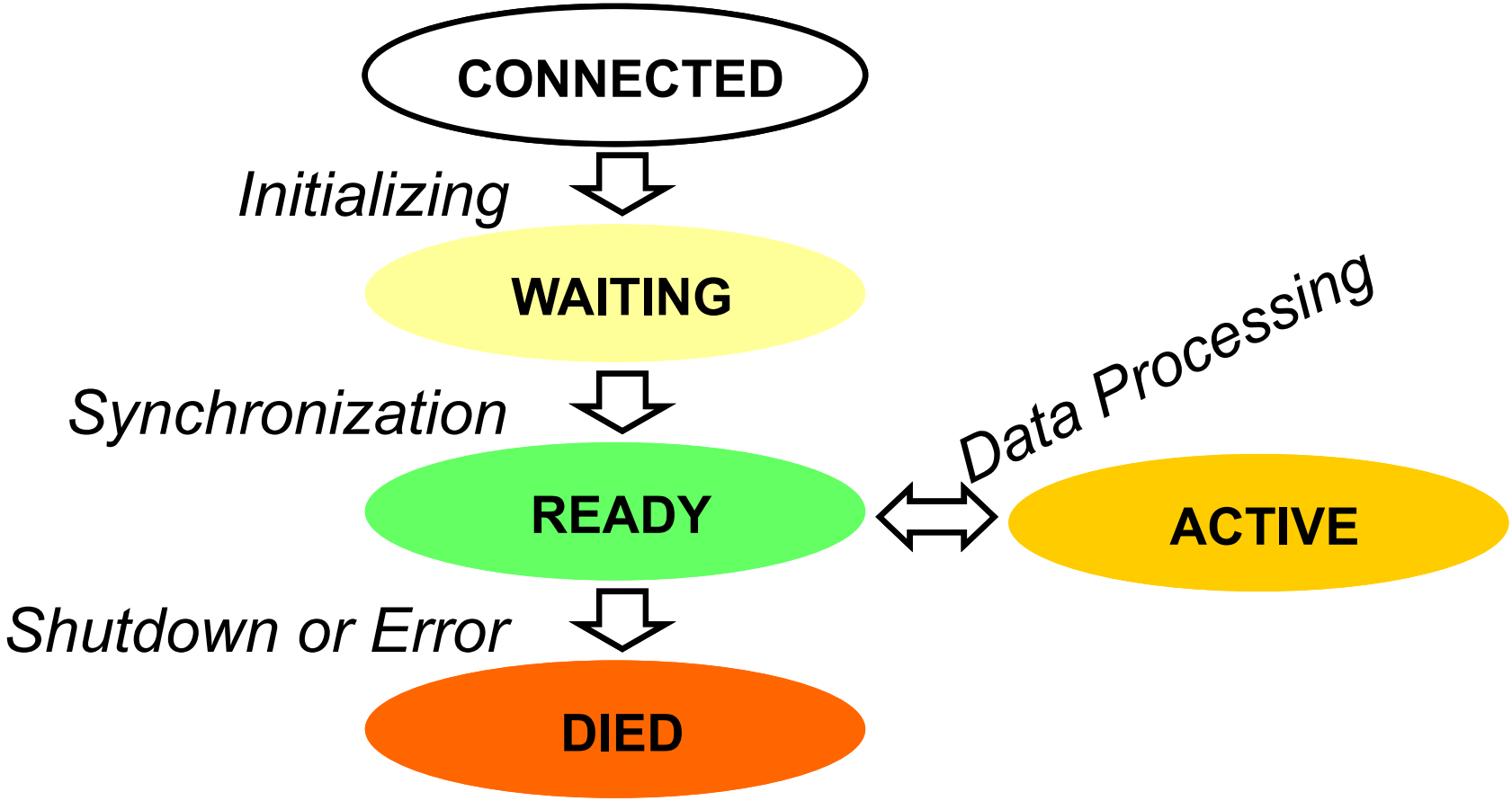
- **common communication mechanism for all used programming languages (C, C++, Lisp, Prolog, Java, Fortran, Tcl/Tk)**
- **Narrow interface for all used programming languages**
- **Overall system control infrastructure**
- **Standards on various levels**
  - Installation
  - Compilation
  - Communication formats between modules
  - ...
- **Toolbox for recording, replaying, testing, inspecting data exchanged between modules, ...**



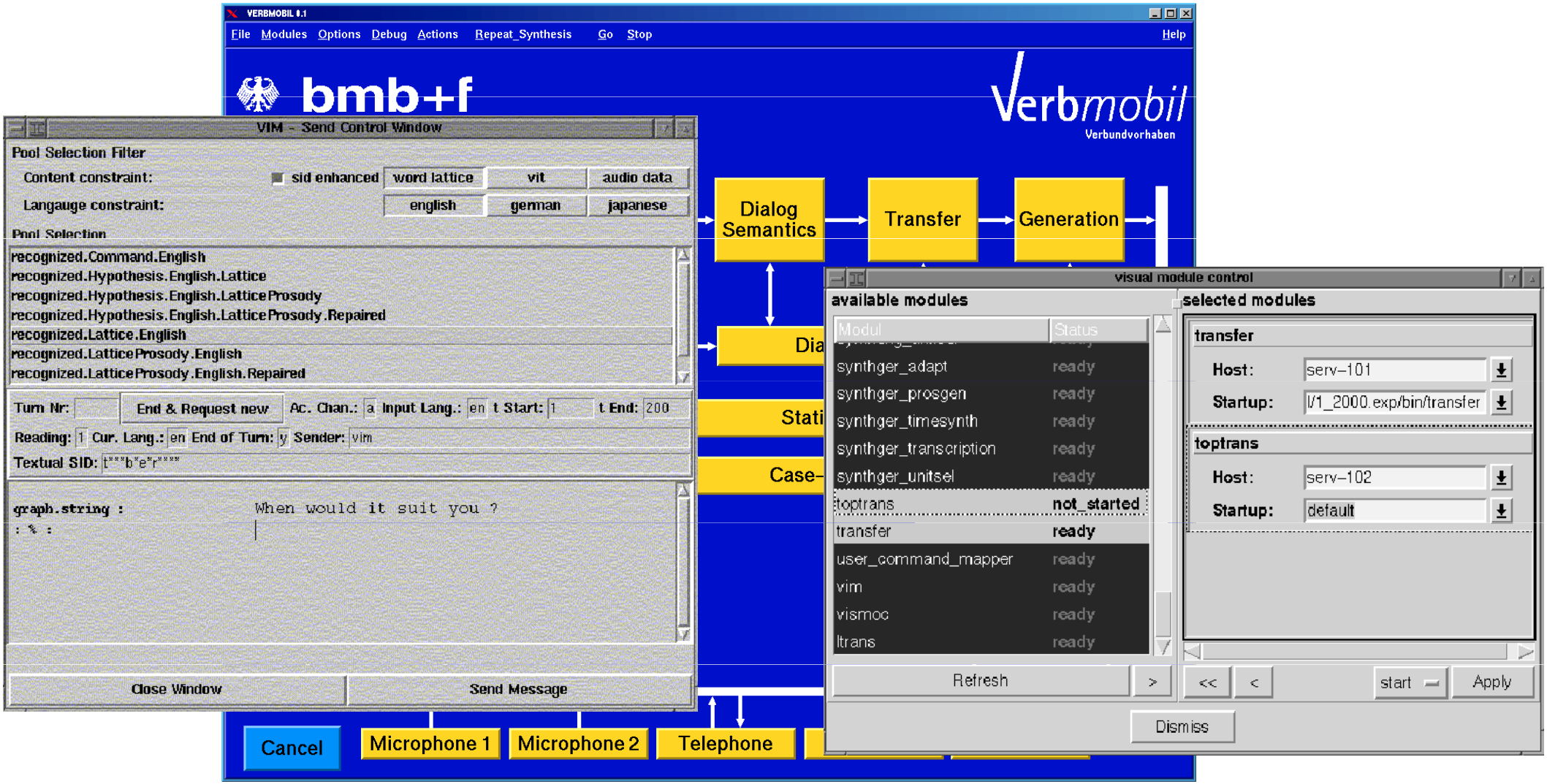
# The Testbed is the Integration Framework for the Verbmobil System



# The Testbed controls the System: Module States



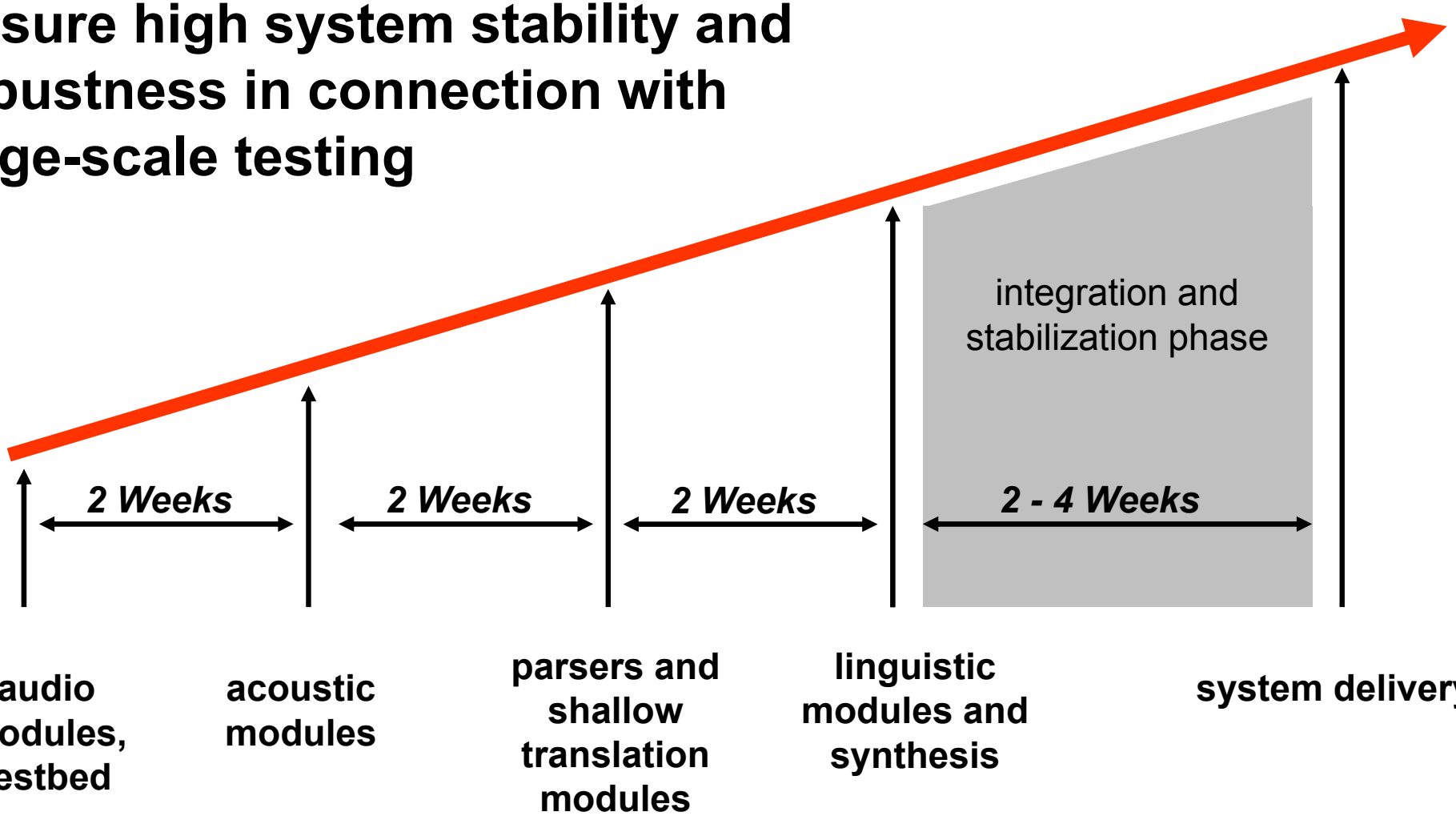
# The GUI- Visualization and Debug Tool




**.... and much more**

# Support from the System Group (2): Regular Integration Cycles

**Assure high system stability and robustness in connection with large-scale testing**



# Human Factors



# A Remark about Project Duration

**8 years is a long time, especially since the invention of Internet time**

**1993**

- "You will need special hardware!"
- "1500 words speaker independent is impossible!"
- "Aren't your goals unrealistic?"

**2000**

- "Does it run on my notebook?"
- "Only 10 000 words?"
- "Why can't it also translate in the domains X, Y, and Z?"

**but**

**it is a unique chance for**

- **large scale, continuous research and development**
- **training people, collaborating, gaining experience**
- **collecting and annotating data**



# Management Challenges

## The goal

- Build an integrated system

## The situation

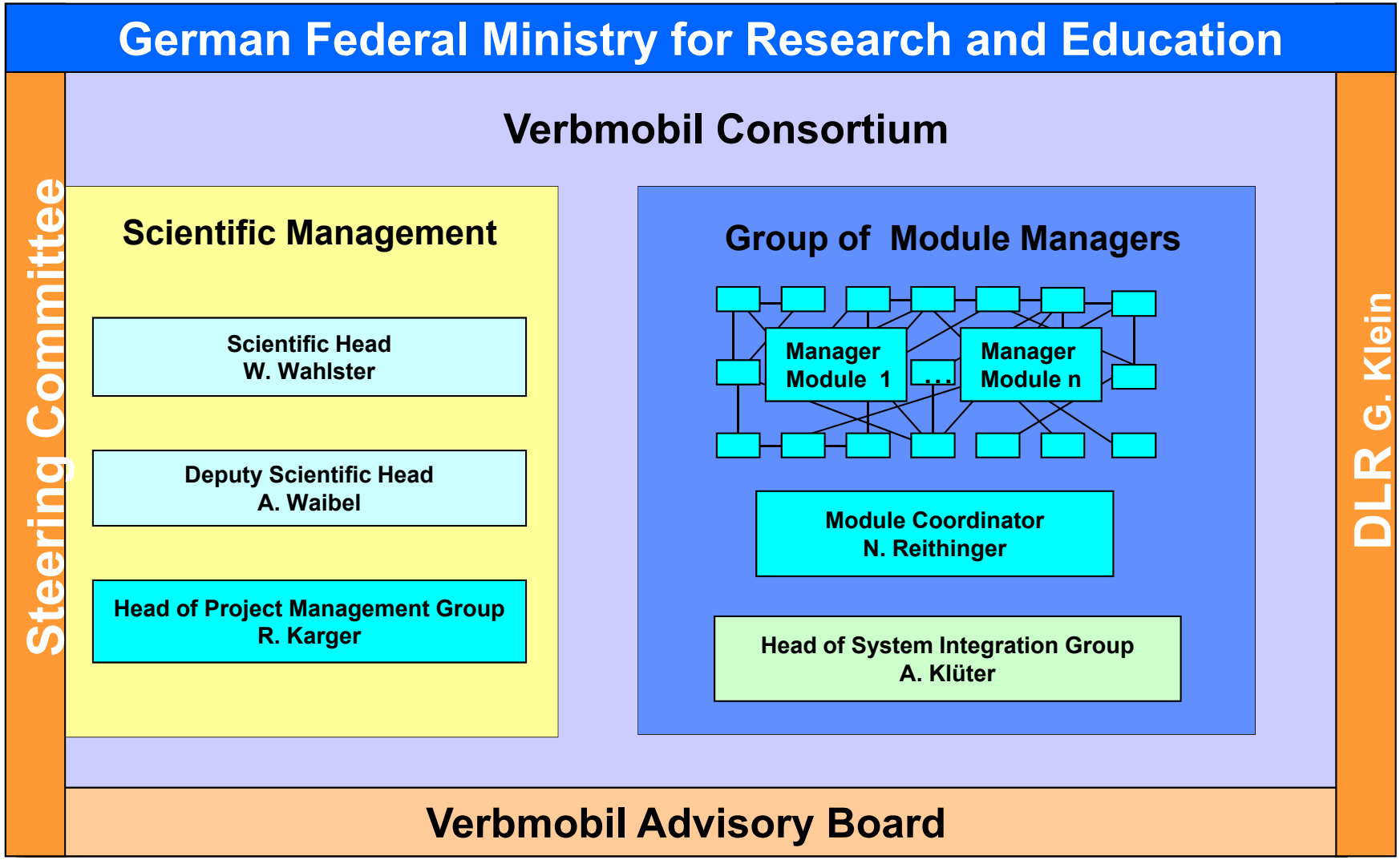
- Partners distributed and pretty independent
- Great variation in project and background experience
- Adjustment of project plan and goals over time needed

## The solution

- Define a flat management structure
- Create a group spirit



# Project Organization



# Module Managers

- Have technical hands on experience
- Responsible for one module, even if it is developed at different sites
- Volunteers (sort of ...)
- Meet regularly, despite e-mail, phone and other devices
- Define next milestones
- Define data and software integration plans

Module coordinator coordinates the efforts and is the link to the scientific management

# Example: Optimization Schedule 2000

- 21.02.  Delivery of CeBit system
- 21.02. - 30.04. Optimization phase
  - 15.03. - 28.04. End-To-End evaluation with feedback to developers
  - 27.03. - 07.04. Workshop Deep Processing
- 09.05. Delivery Verbmobil System 1.0
- starting 09.05
  - speech recognizer evaluation
  - turn evaluation



# Experience

- The group of module managers is a Good Thing™
- Common goals motivate
- Friendly peer pressure works most of the time
- Early problem detection and resolution in most cases
- Regular integration cycles focus and motivate

❑ Proactive consensus management (PCM)



# Experience

- The System Group is a Good Thing™
- The multi blackboard architecture is a Good Thing™
- Crucial for the success of Verbmobil
- Software foundation for (almost) hassle free module development

❑ Controlled distributed development possible

# Verbmobil-Symposium

## 30.7.2000, 10:30-18:00
## Saarbrücken, Kongresshalle

**Programm**

(Keine Teilnahmegebühr)

Zeitraster für das Verbmobil-Abschlussymposium

Datum: 30.07.2000

Ort: Neue Congresshalle Saarbrücken

10:30 - 10:35 Eröffnung
10:35 - 10:45 Grußworte des BMBF (B. Reuse, BMBF)
10:45 - 11:30 Verbmobil (W. Wahlster)
11:30 - 12:00 Präsentation des Verbmobil-Systems (R. Karger)
12:00 - 12:45 Spracherkennung und Prosodieanalyse (A. Waibel, E. Nöth)

12:45 - 13:30 Imbiss

13:30 - 14:15 Multilinguale Analyse (U. Block, H. Uszkoreit)
14:15 - 15:00 Symbolische und Statistische Übersetzung (C. Rohrer, H.Ney)

15:00 - 15:30 Kaffee

15:30 - 16:15 Generierung und Synthese (T. Becker, W. Hess)
16:15 - 16:45 Evaluierung der End-to-End-Übersetzungsleistung des Systems (W. v.Hahn)

16:45 - 17:00 Verlesen des schriftlichen Abschlussgutachtens

17:00 - 18:00 Podiumsdiskussion: Sprachtechnologie und New Economy

# SmartKom


- Overview
- Architecture
- Core Areas: Analysis, Fusion, Generation, ...
- Dialogue Processing

# Overview

- **Introduction**
  - Why Multimodal Interaction Systems?
  - Reference Architecture for Multimodal Systems
- **SmartKom: A Multimodal Interaction System**
  - SmartKom: A Transportable Interface Agent
  - Situated Delegation-oriented Dialog Paradigm: Collaborative Problem Solving
  - Modes in SmartKom
  - More About the System
  - M3L: XML based Multimodal Markup Language
  - Multimodal Coordination



# Why Multimodal Interaction Systems? (Oviatt&Cohen, CACM March 2000)

- **Accessibility for diverse users and usage contexts**
  - Selection of modes by the user and by the system
    e.g. lean- forward/lean-backward mode in a home environment, car
- **Performance stability and robustness**
  - Users can select robust mode
  - Mutual disambiguation and presentation
- **Expressive power and efficiency**
  - Interface more powerful
  - Faster
  - Increased task completion



# Reference Architecture for Multimodal Systems

**2 Nov. 2001**
Dagstuhl Seminar
Fusion and Coordination
in Multimodal Interaction
edited by: M. Maybury

**Media Input Processing**

Biometrics

G
T
A
G
V

User(s)

A
G
A
V
G

**Media Output Rendering**

**Mode Analysis**
- Language
- Graphics
- Gesture
- Sound

**Mode Design**
- Language
- Graphics
- Gesture
- Sound
- Animated Presentation Agent

**Mode Coordination**

**Multimodal Fusion**
- Multimodal Reference Resolution

**Presentation Design**
- Select Content
- Design
- Allocate
- Coordinate
- Layout

**Interaction Management**

**Discourse Management**
- Reference Resolution

**Context Management**

**Expectation Management**

**Intention Recognition**

**Action Planning**

**User Modeling**
- User ID

**Application Interface**
- Initiate
- Terminate
- Request
- Respond
- Integrate

Information, Applications, People

User Model | Discourse Model | Context Model | Domain Model | Task Model | Media Models | Application Models

**Representation and Inference, States and Histories**

# Overview

- **Introduction**
- **SmartKom: A Multimodal Interaction System**
  - SmartKom: A Transportable Interface Agent
  - Situated Delegation-oriented Dialog Paradigm: Collaborative Problem Solving
  - Modes in SmartKom
  - More About the System
  - M3L: XML based Multimodal Markup Language
  - Multimodal Coordination
- **MIAMM**
  - Main Objectives
  - Interaction using Haptics
- **Research Roadmap of Multimodality**
- **Conclusion**



# Human-Technology Interaction Lead Projects

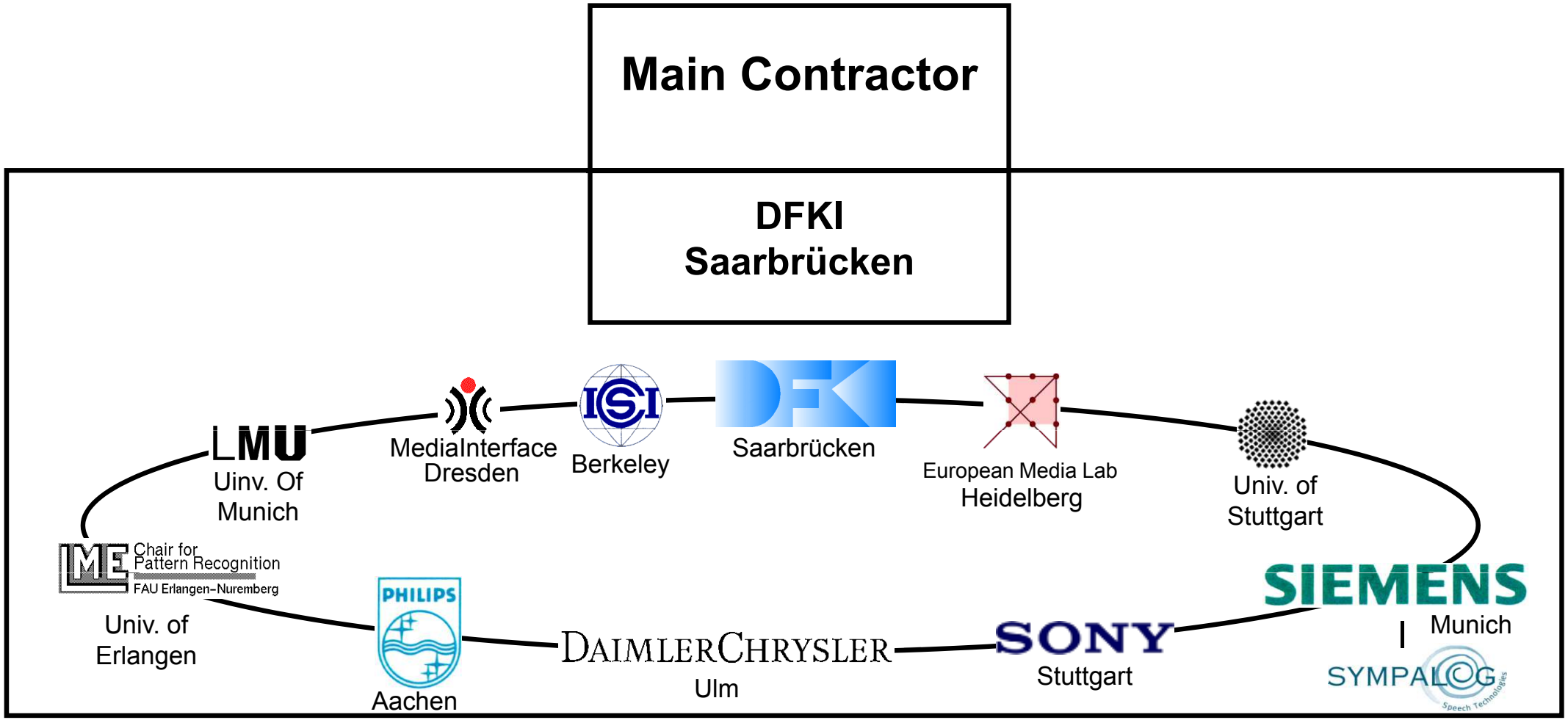
| Project | Title | Coordinator | Funding Period |
|---|---|---|---|
| INVITE | Intuitive Mensch-Technik-Interakt. für die vernetzte Informationswelt der Zukunft | ISA GmbH, Stuttgart | 07/99 - 06/03 |
| MORPHA | Intelligente anthropomorphe Assistenzsysteme | Delmia GmbH, Fellbach | 07/99 - 06/02 |
| EMBASSI | Elektronische Multimediale Bedien- und Service-Assistenz | Grundig GmbH, Fürth | 07/99 - 06/03 |
| ARVIKA | Augmented Reality für Entwicklung, Produktion und Service | Siemens AG, Nürnberg | 07/99 - 06/03 |
| SMARTKOM | Dialogische Mensch-Technik-Interaktion durch koordinierte Analyse und Gener. multipler Modalitäten | DFKI GmbH, Saarbrücken | 09/99 - 09/03 |
| MAP | Multimedia Arbeitsplatz der Zukunft | AlcatelSel AG, Stuttgart | 04/00 - 03/03 |



# The SmartKom Consortium

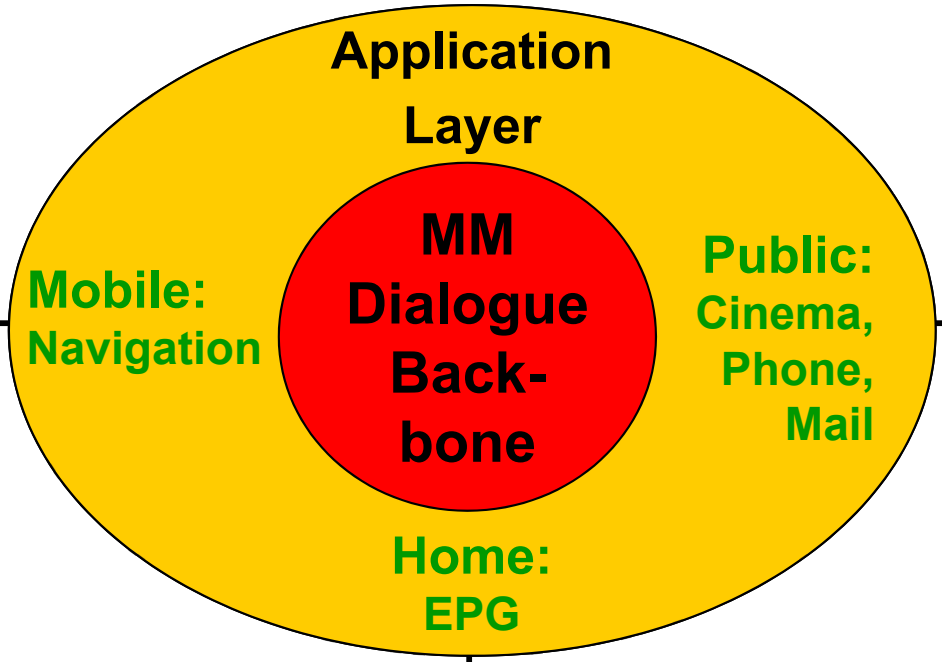
**Project Budget:** € 25.5 million

**Project Duration:** 4 years (September 1999 – September 2003)



# SmartKom: A Transportable Interface Agent


**SmartKom-Mobile**:
A Handheld
Communication
Assistant



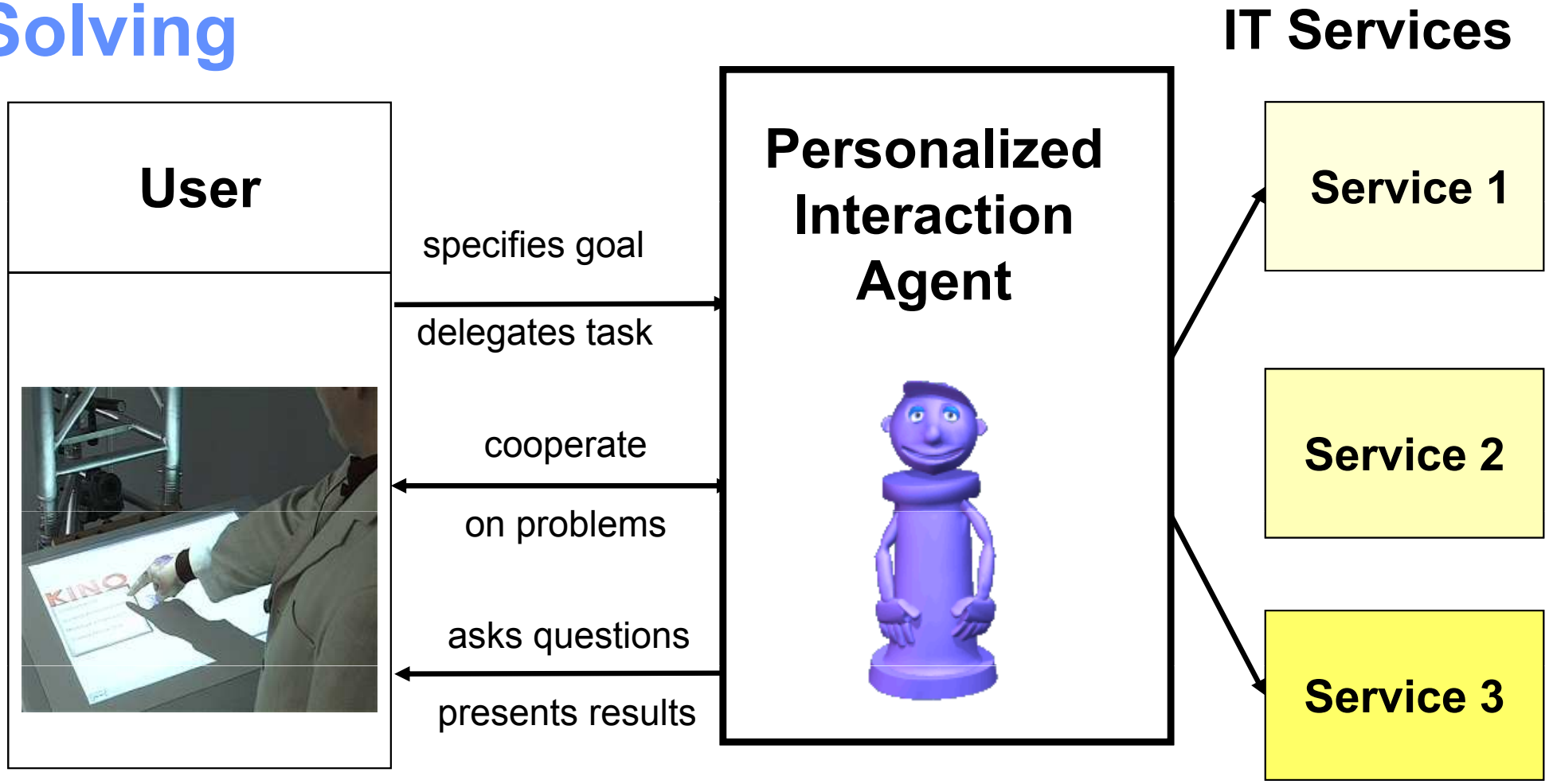


**SmartKom-Home/Office**:
Multimodal Portal
to Information Services

**SmartKom-Public**:
A Multimodal
Communication
Kiosk

# An Example Interaction with SmartKom Mobile

# Situated Delegation-oriented Dialog Paradigm: Collaborative Problem Solving

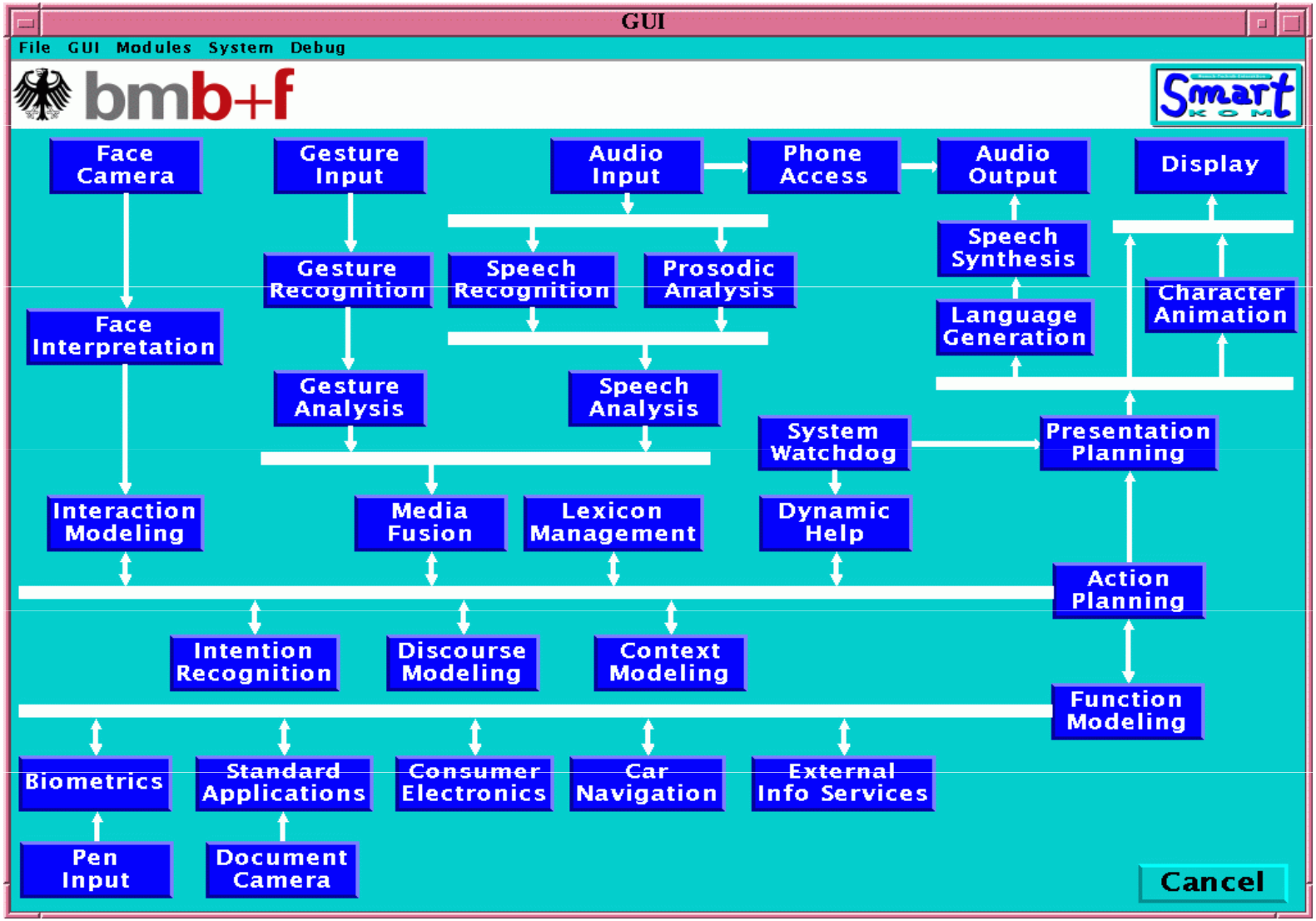


# Modes in SmartKom

- **Speech**
  - Speaker independent speech recognition
  - Prosodic input processing
  - Synthesis
- **Gesture**
  - Input
    - Natural gestures (SIVIT)
    - Pen-based
  - Presentation agent
- **Facial/body expression**
  - User state recognition
  - System state presentation



# The Main Modules on the Control GUI

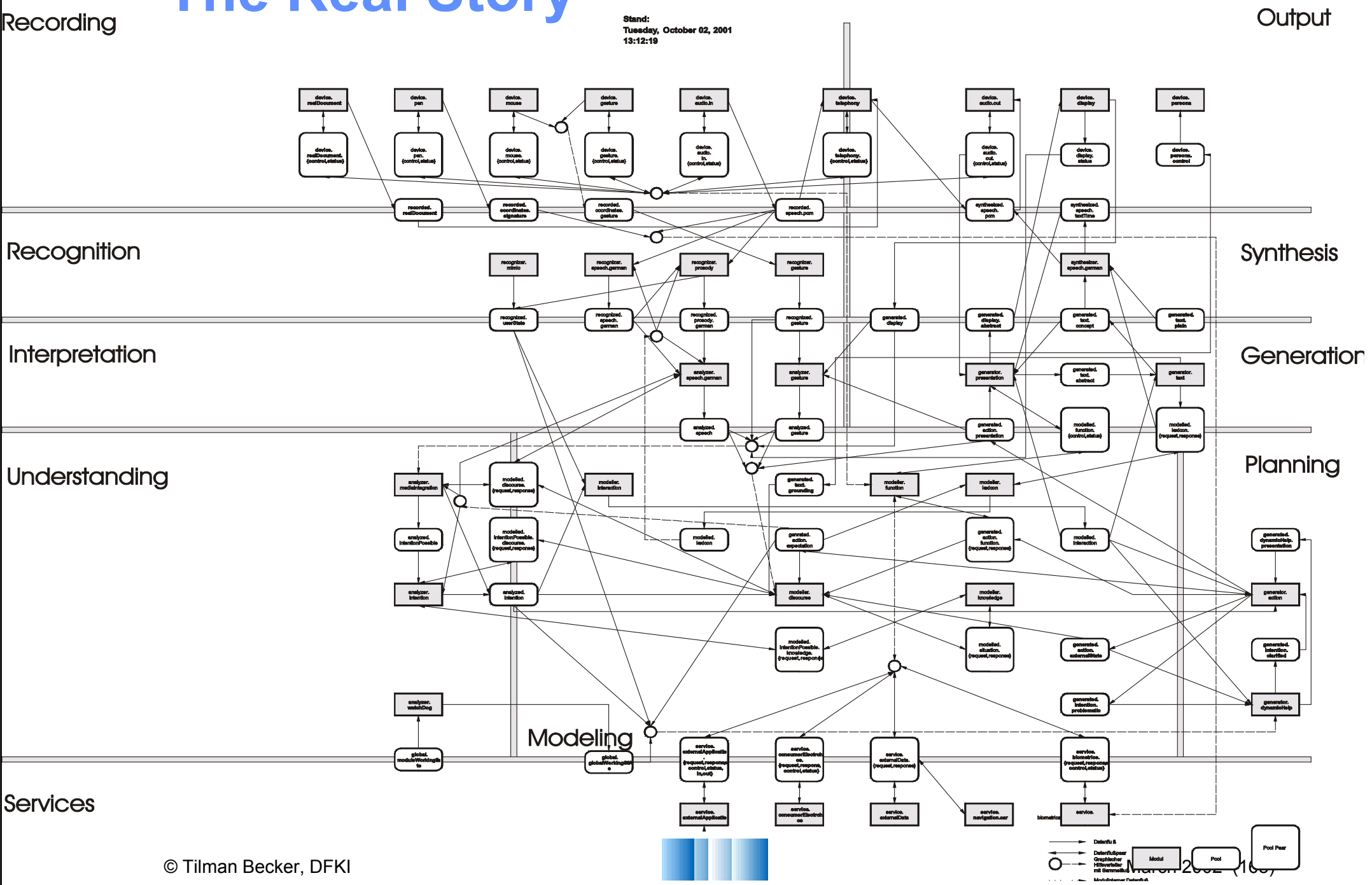


# More About the System

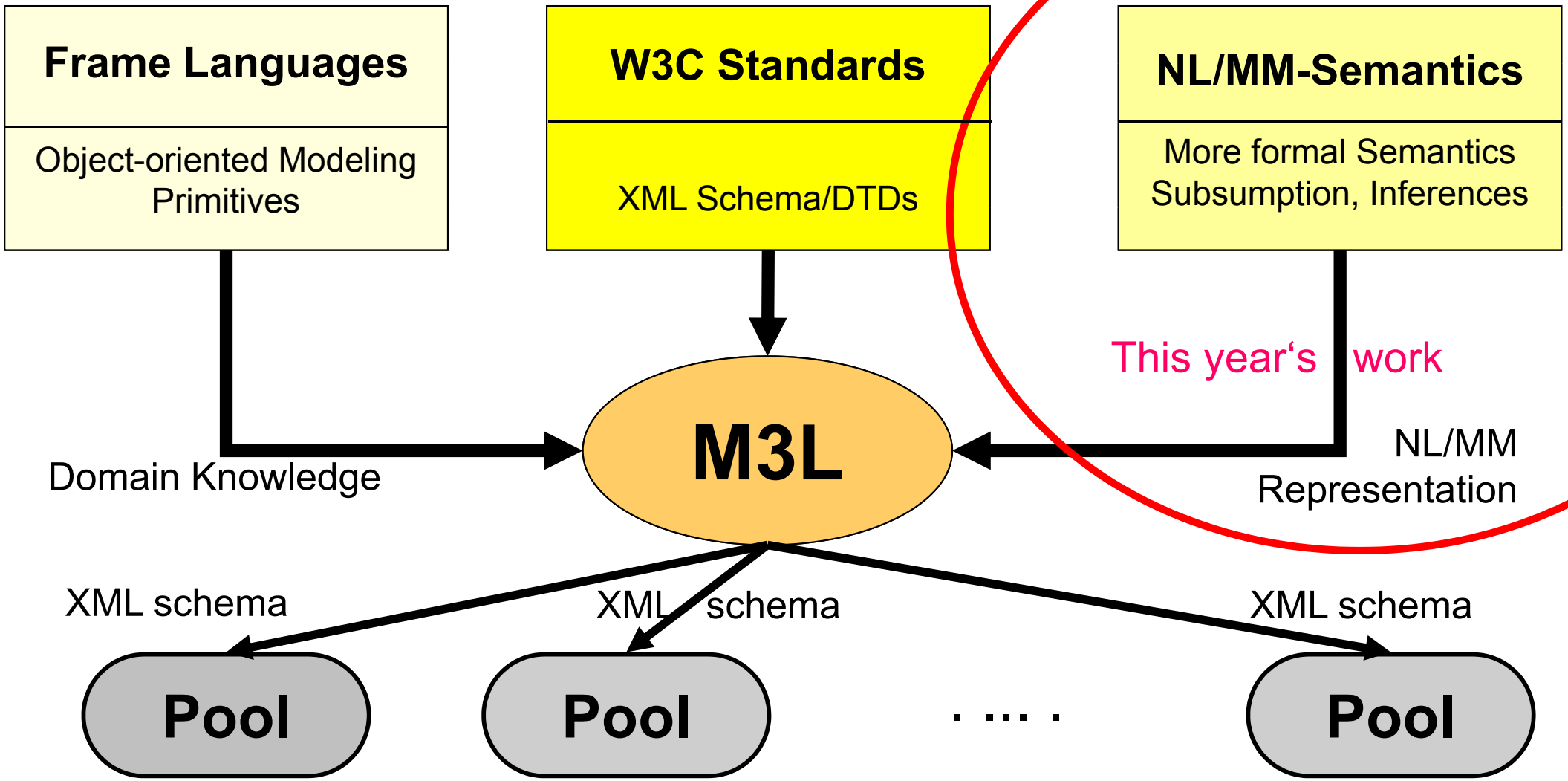
- **Modules realized as independent processes**
- **Not all must be there (critical path: speech or graphic input to speech or graphic output)**
- **(Mostly) independent from display size**
- **Pool Communication Architecture (PCA) based on PVM for Linux and NT**
  - Modules know only about their I/O pools
  - Literature:
    - Andreas Klüter, Alassane Ndiaye, Heinz Kirchmann: *Verbmobil From a Software Engineering Point of View: System Design and Software Integration.* In Wolfgang Wahlster: *Verbmobil - Foundation of Speech-To-Speech Translation.* Springer, 2000.
- **Data exchanged using M3L documents**
- **All modules and pools are visualized here ...**



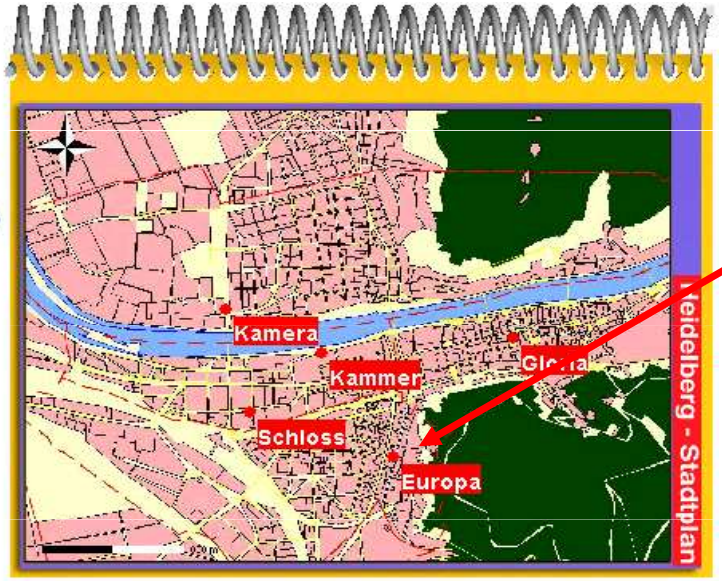
# The Real Story



# The “Glue“ - M3L: XML based Multimodal Markup Language



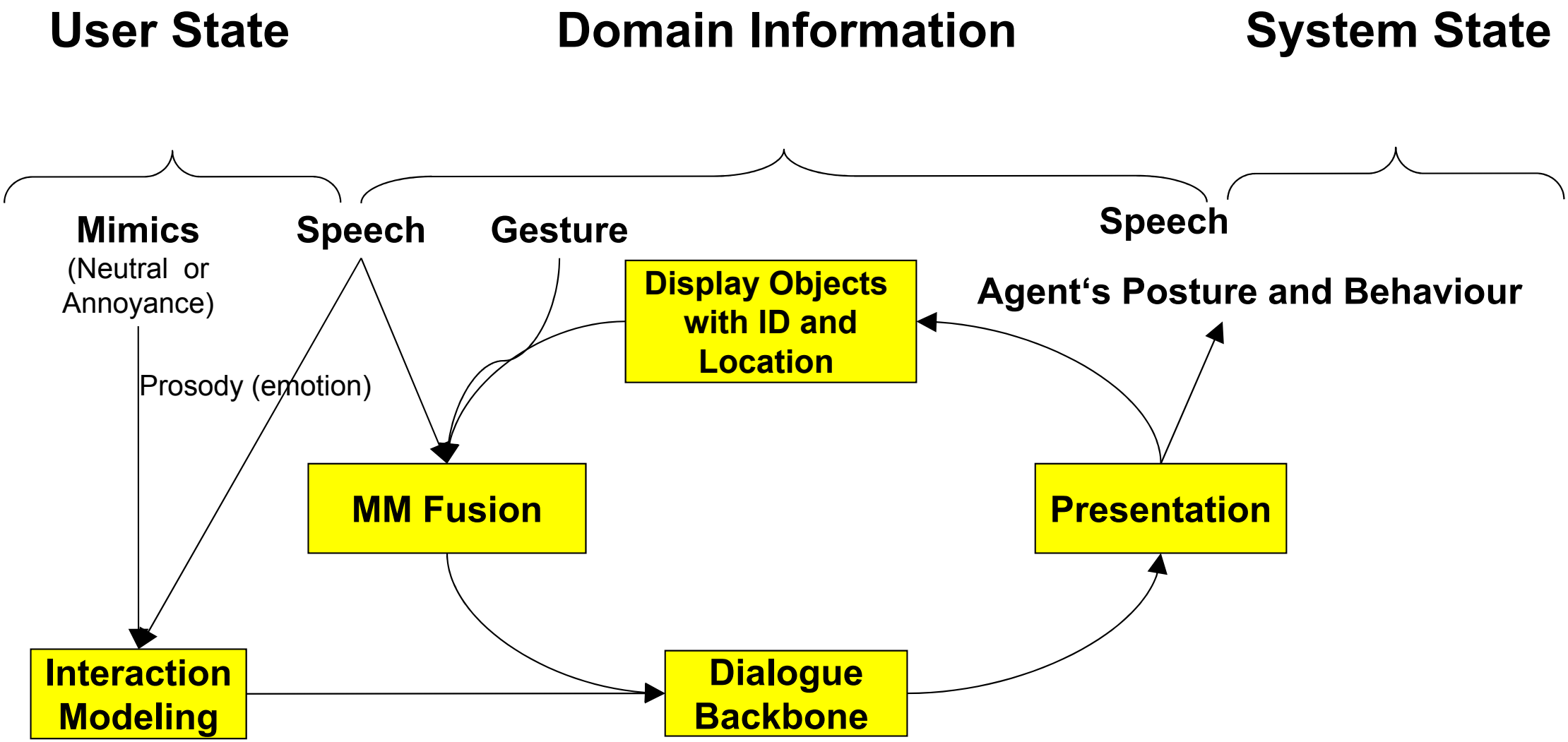
# An Example of the M3L Representation of the Multimodal Discourse Context



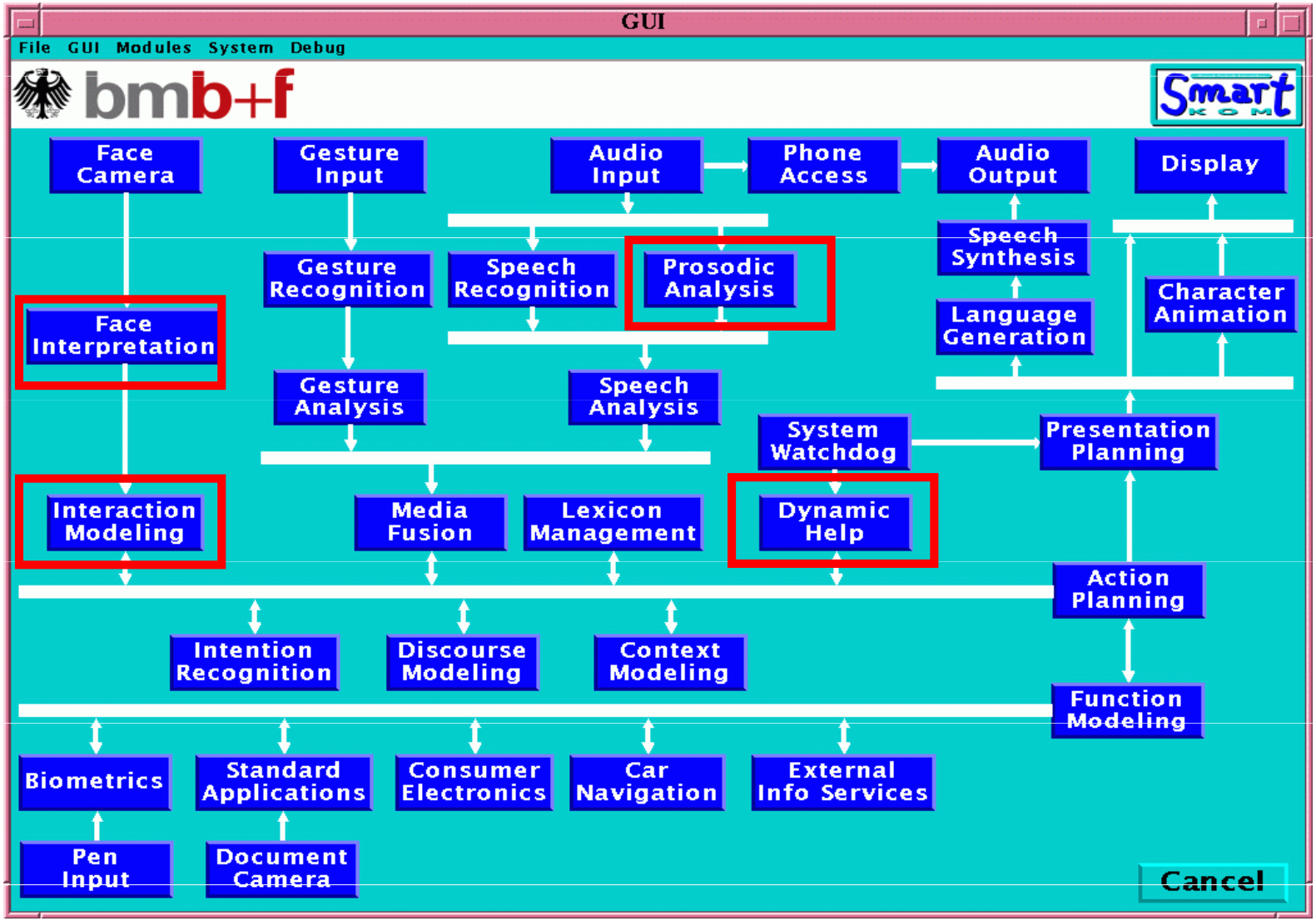

```
<?xml version="1.0"?>
<presentationContent>
[...]
   <abstractPresentationContent>
      <movieTheater structId=„pid3072”>
         <entityKey> cinema_17a </entityKey>
         <name> Europa  </name>
            <geoCoordinate>
             <x> 225 </x> <y> 230 </y>
            </geoCoordinate>
      </movieTheater>
   </abstractPresentationContent>
[...]
    <panelElement>
      <map  structId="PM23">
       <boundingShape>
        <leftTop>
         <x>  0.5542 </x>  <y>  0.1950 </y>
        </leftTop>
        <rightBottom>
         <x>  0.9892 </x>  <y>  0.7068 </y>
        </rightBottom>
       </boundingShape>
       <contentReference> pid3072 </contentReference>
      </map>
    </panelElement>
[...]
</presentationContent>
```

***„No presentation without representation!“***

# Mode Processing: The Data Flow



# Processing the User‘s State



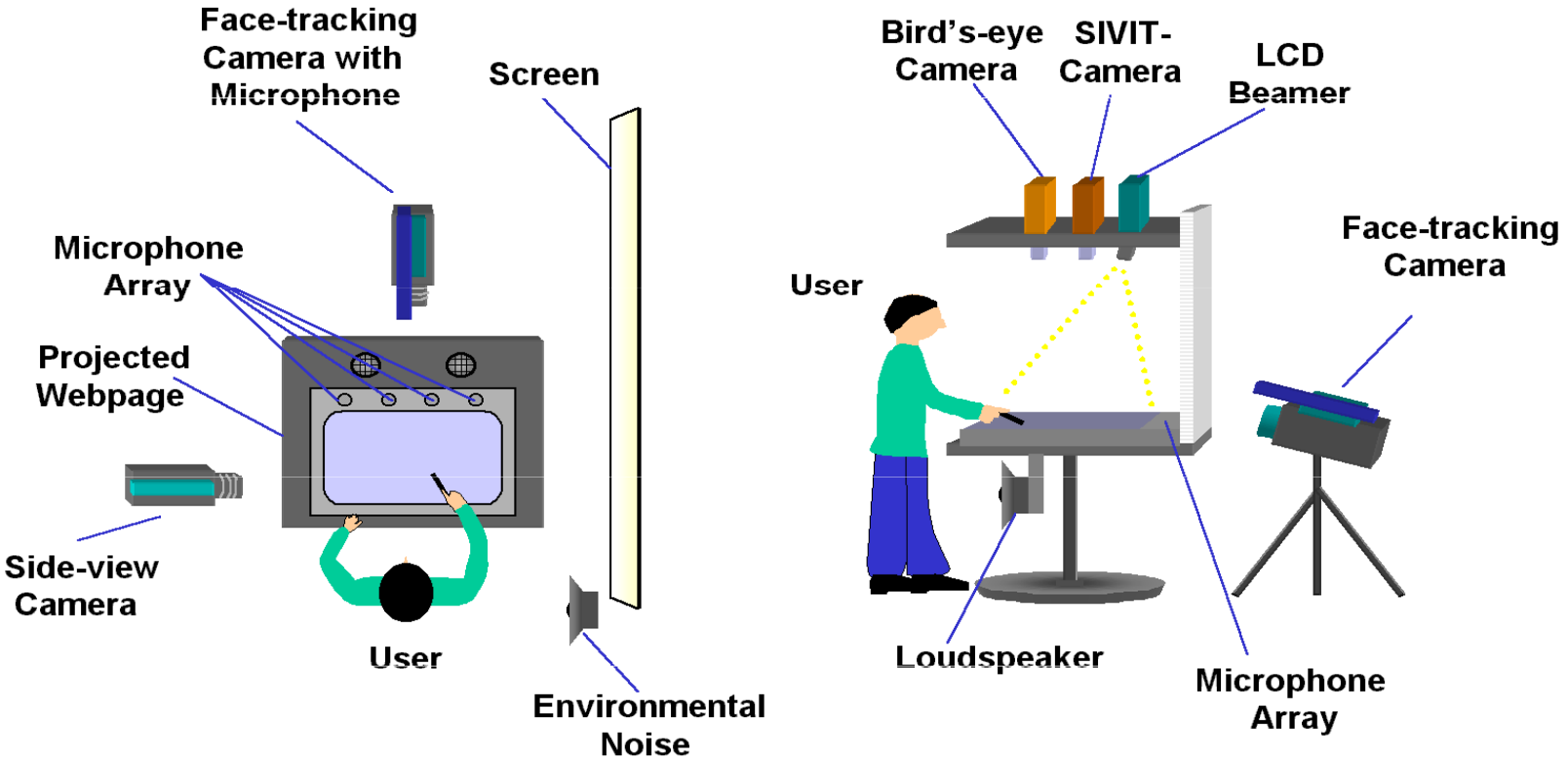
# Processing the User‘s State

- **Different reference levels:**

| Object level | Meta level |
|---|---|
| *This is great! Show me more!* | *That was quick!* |
| *One moment, let me think.* | *OK now, what are you doing?* |
| *Oh no, that’s ugly! A new one!* | *What the .... is going on?* |

- **Annotated in the data from the data collection**
- **Recognized using mimics and prosody**
- **In case of anger activate the dynamic help**

# Wizard of Oz Data Collection
## (LMU Munich)




**Data distributed on DVD (1 DVD per 5 minute dialogue)**

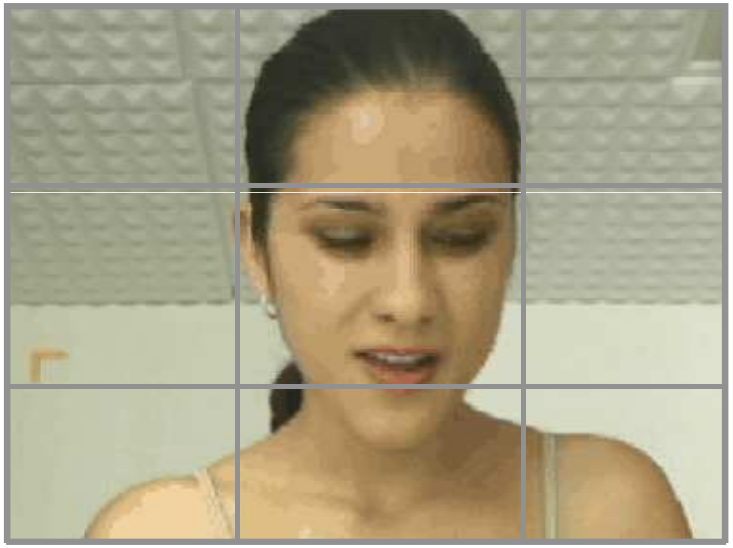
# User States Annotated in 45 dialogues

| | |
|---|---|
| Neutral | 681 |
| Joy/success | 31 |
| Reflection | 59 |
| Perplexity | 31 |
| Surprise/Astonishment | 11 |
| Annoyance/Failure | 16 |

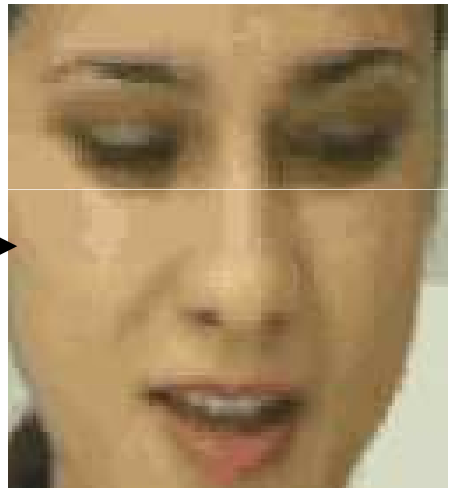
**Only about 18% emotional user state events**



# User Independent Classification of Facial Expressions
## (Univ. Erlangen)



Localization

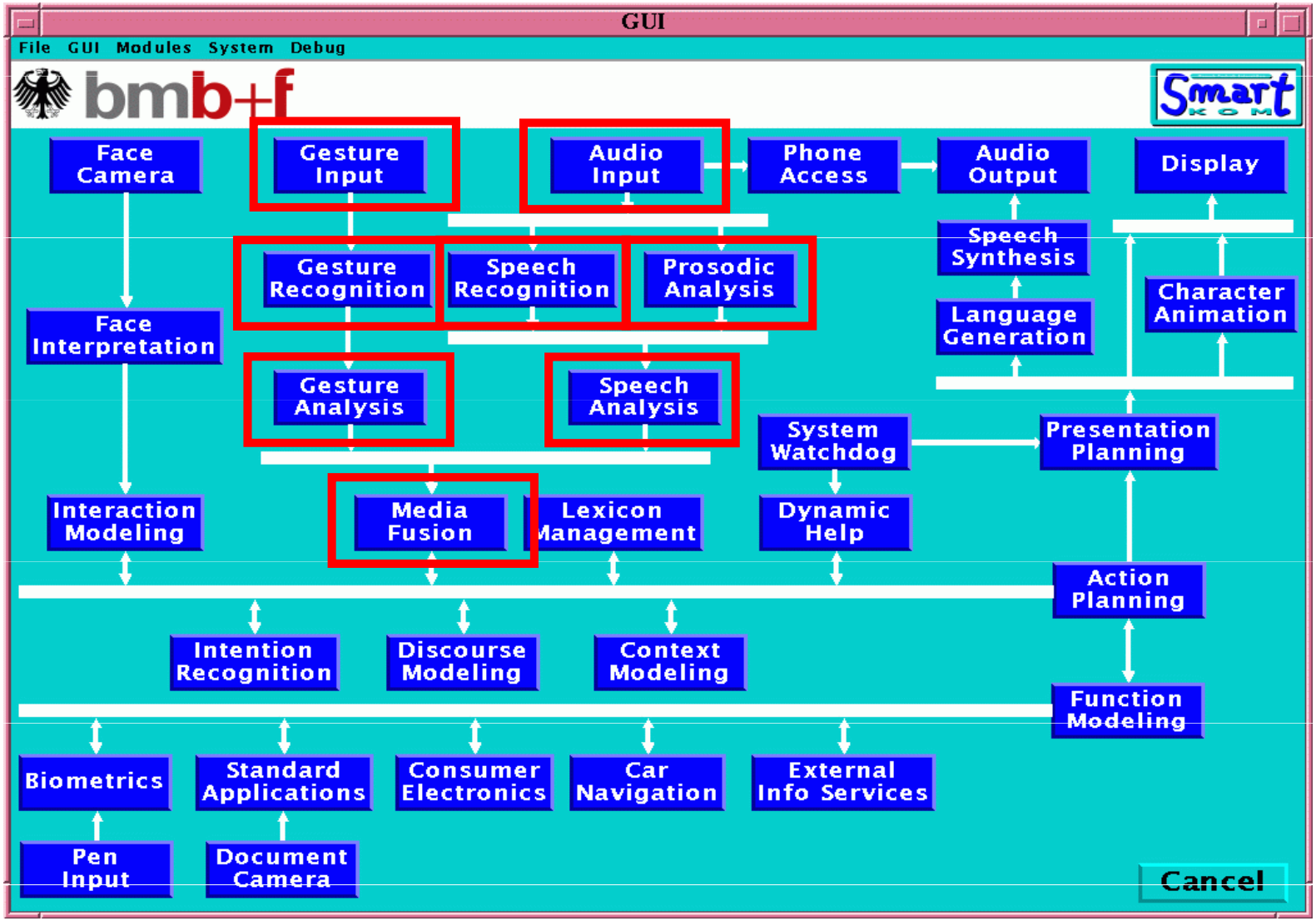


*Annoyance*

*Rest (neutral)*

Classification
(SVM, Eigenfaces)

# Media Fusion



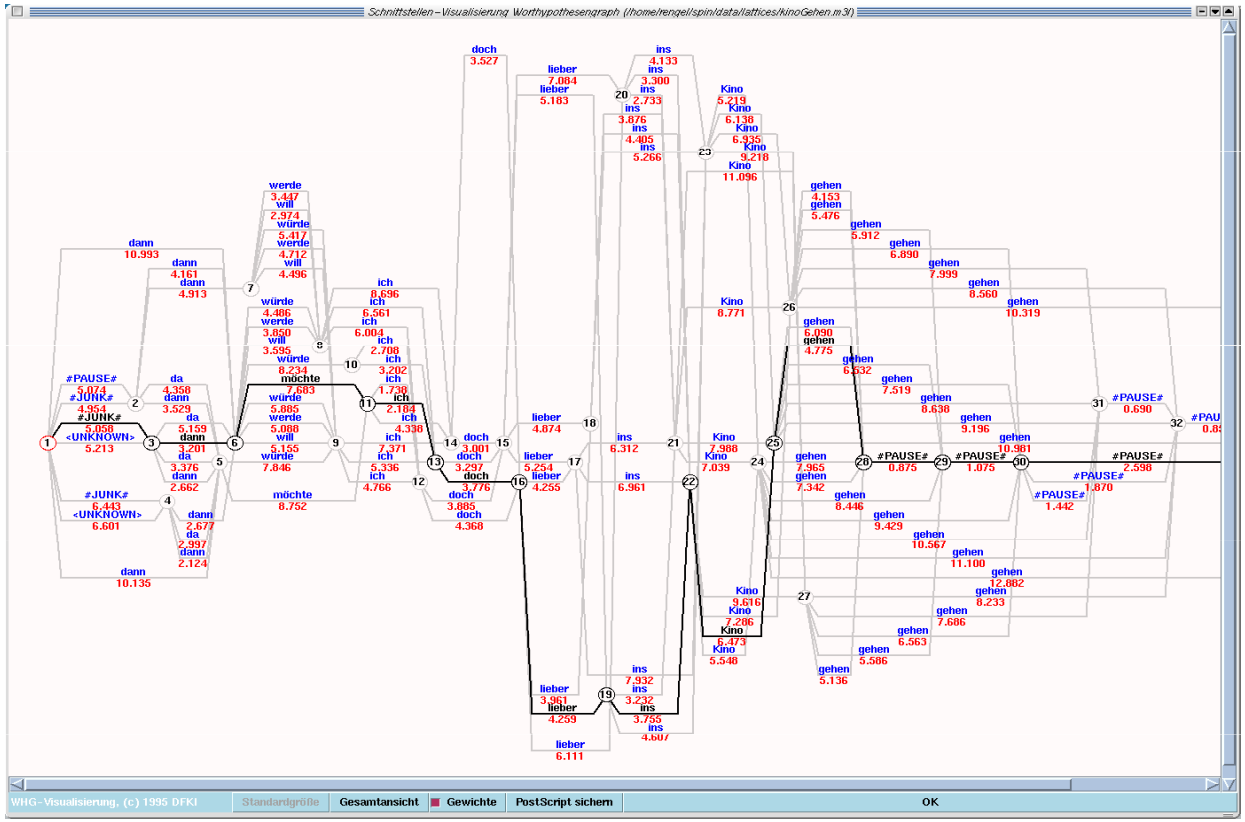
# Gesture Processing

- **Objects on the screen are tagged with IDs and bounding boxes**
- **Gesture input**
  - Natural gestures recognized by SIVIT
  - Touch sensitive screen
- **Gesture recognition**
  - Location
  - Type of gesture: pointing, tarrying, encircling
- **Gesture Analysis**
  - Reference object in the display described as domain model (sub-)objects (M3L schemata)
  - Compute distance to bounding boxes
  - Output: gesture lattice with hypotheses



# Speech Processing

- **Word lattice**




- **Prosody inserts boundary and stress information**
- **Speech analysis creates intention hypotheses**

  *which movies are playing at the Metropol*

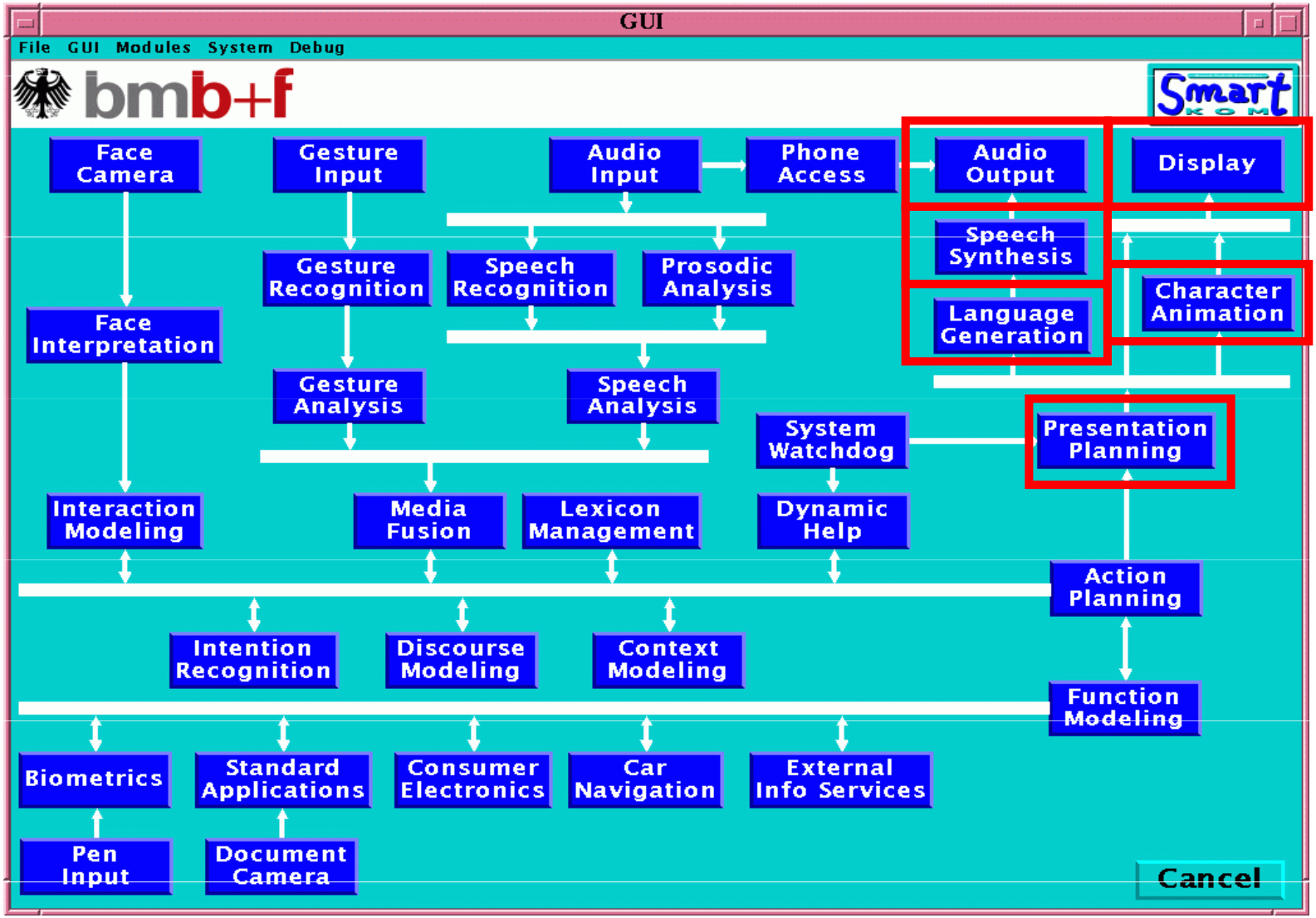
  ***hypothesis(action:info,performance(cinema(name:Metropol)) ..)***

# Media Fusion

- Integrates gesture hypotheses in the intention hypotheses of speech analysis
- Information restriction possible from both media
- Possible but not necessary correspondence of gestures and placeholders (deictic expressions/ anaphora) in the intention hypothesis
- Necessary: Time coordination of gesture and speech information
- Time stamps in *ALL* M3L documents!!
- Output: sequence of intention hypothesis



# Presentation

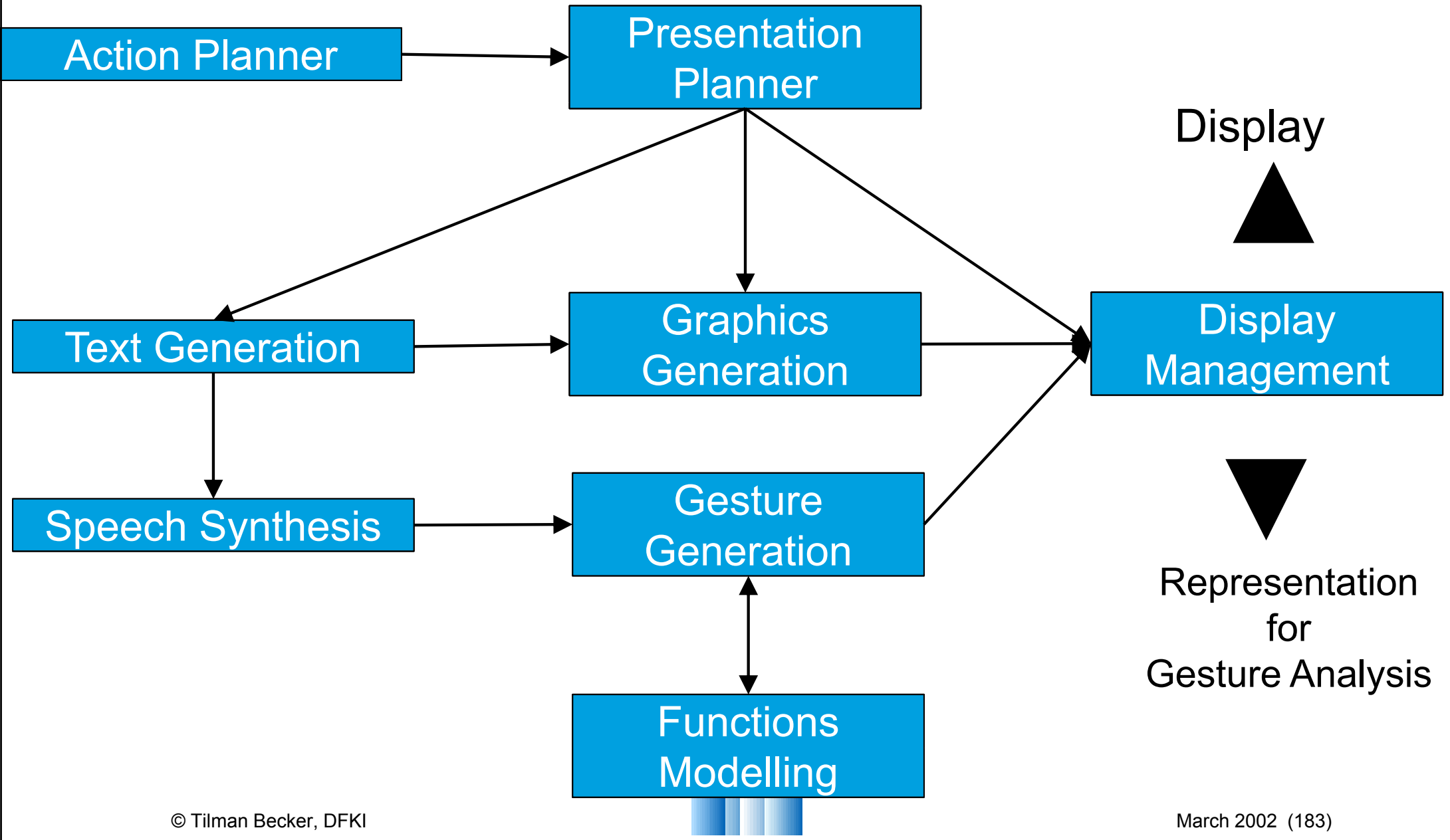


# Presentation

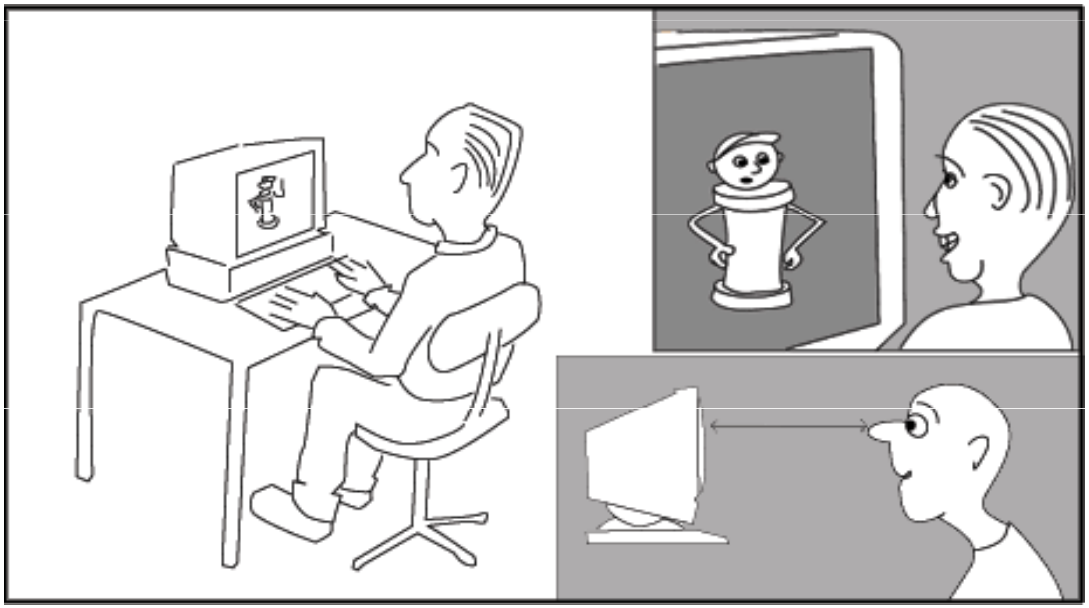
- **Starts with action planning**
- **Definition of an abstract presentation goal**
- **Presentation planner:**
  - Selects presentation, style, mode, and agent‘s general behaviour
  - Activates natural language generator which activates the speech synthesis which returns audio data and time-stamped phoneme/viseme sequence
- **Character animation realizes the agent‘s behaviour**
- **Synchronized presentation of audio and visual information**



# Partial view of SK architecture: Multimodal Presentation



# User Perspective



**Monitor: frontal view**

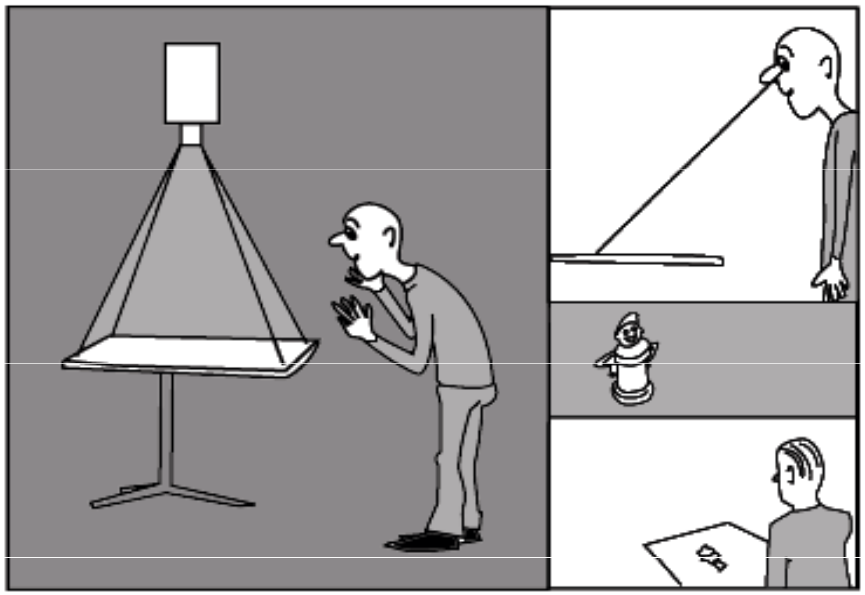


**Table: angled view**

# Lip Synchronization with Visemes

- **Goal: present a speech prompt as natural as possible**
- **Viseme: elementary lip positions**
- **Correspondence of visemes and phonemes**
- **Examples:**

# Behavioural Schemata

- **Goal: the agent (Smartakus) is always active to signal the state of the system**
- **Four main states**
  - Wait for user‘s input
  - User‘s input
  - Processing
  - System presentation

- **Current body movements**
  - 9 vital, 2 processing, 9 presentation (5 pointing, 2 movements, 2 face/mouth)
  - About 60 basic movements

# New animations

Examples for complex movements and speech-synchronized gestures

Pointing to the right

Enumeration of items

Moving in a circle

# Example: Pointing Gestures

base position

preparation

stroke

retraction

composed gesture:

# Details:

**Natural Language Generation in SmartKom**

**Discourse Updates in Interactive Dialogues**

AT&T Research
2 Aug 2001


# Natural Language Generation in SmartKom

Tilman Becker

**Deutsches Forschungszentrum für Künstliche Intelligenz GmbH**
**Stuhlsatzenhausweg 3, Geb. 43. 1 - 66123 Saarbrücken**
**Tel.: (0681) 302-5271**
**Fax.: (0681) 302-5020**
**Email: becker@dfki.de**
**www.smartkom.org**

# Overview

- **Architecture**
- **Presentation Goals**
- **Natural Language Generation**

  **for Speech Synthesis**
  - Architecture
  - Selection of data, sentence templates
  - „fully specified templates“
  - Concept-To-Speech information
- A short look aside: graphics and gestures
- **Outlook**



# Presentation Begins in Action Planning

- **Presentation as planning of a multi-modal dialog act**

- **Abstract presentation goals**

  **(defined in an XML Schema**

  `presentation.xsd`**)**



# Natural Language Generation: Overview

- **Input, Output**
- **Architecture**
- **Knowledge Bases**
- **The steps of generation**
- **Templates**
  - Tree Adjoining Grammars
  - “fully specified templates”
- **Concept-To-Speech information**



# Typical Abstract Presentation Goals

- **Presentation of information (usu. With an implicit request): “Here you can see...” : `<inform>`**
- **Explicit Request to fill a slot: “Please show me where you want to sit” : `<request>`**
- **Feedback: “Your reservation is secured...” `<feedback>`**
- **Canned presentations:**

  **`<goodbye>`**

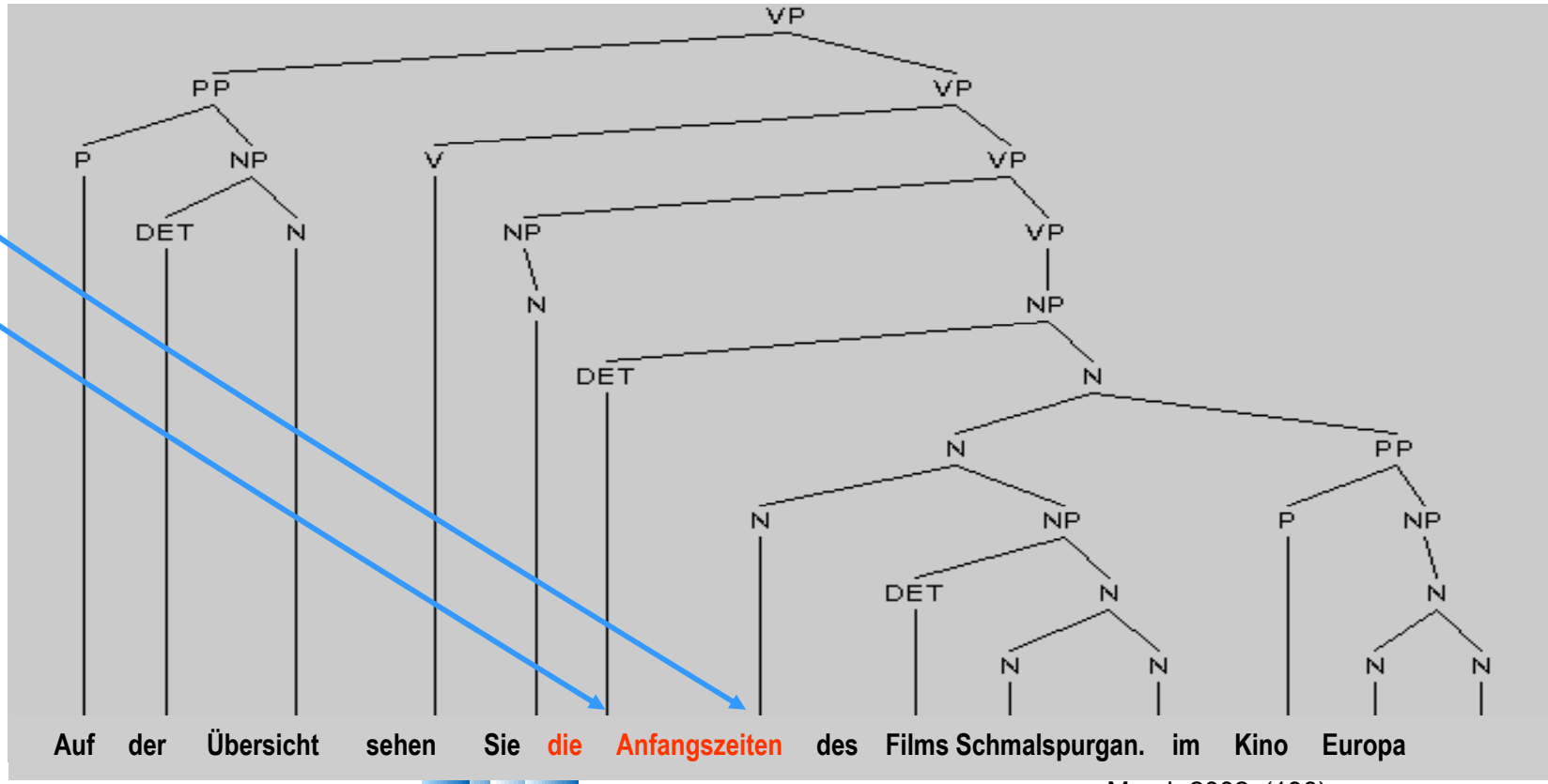


# Input for Natural Language Generation

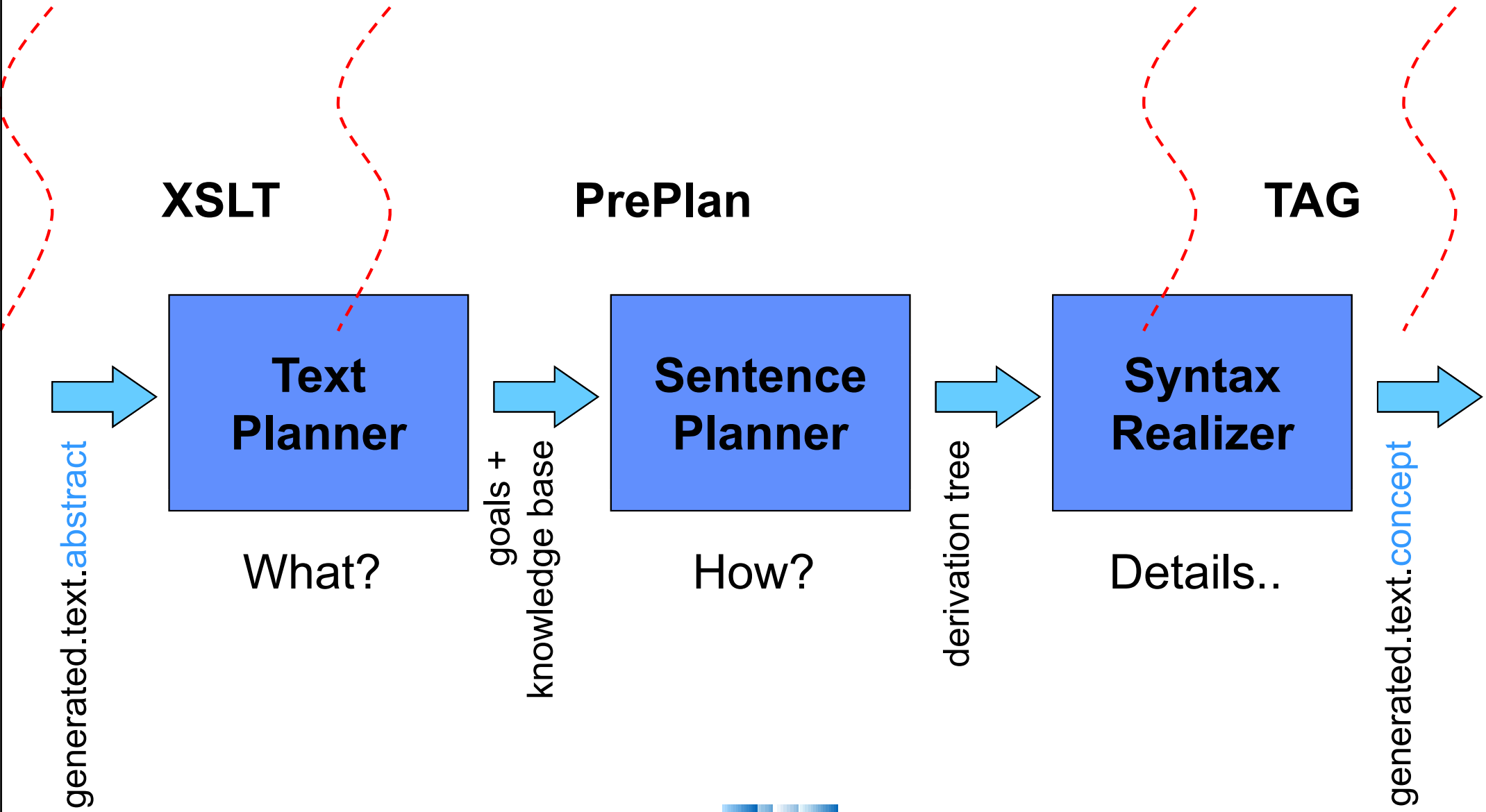
```
<speechGenerationTask  goalKey="11">
 <speechPresentationGoal>
  <inform>
   <comment       commentTyp="onGraphicalPresentation">
     <graphicalRealisationType>        list        </graphicalRealisationType>
     <deepFocus   structReference="struct201"/>
     <content       structReference="struct17"/>
     <content       structReference="struct18"/>
   </comment>
  </inform>
 </speechPresentationGoal>
 <abstractPresentationContent>
      <performance>
       <avMedium>
        <title structId="struct18"> Schmalspurganoven </title>
       </avMedium>
       <cinema>
        <movieTheatre structId="struct17">
         <name>   Europa    </name>
        </movieTheatre>
       </cinema>
      <beginTime structId="struct201"/>
     </performance>
[...]
 </abstractPresentationContent>
</speechGenerationTask>
```

```
<concept   sequence="11">
 <discourseElement     id="9011"  discourseRelation="simple">
  <sentence       id="tsen-9011"   sentenceMode="declarative">
[...]
        <syntaxElement  case="acc" argumentStatus="Object"  syntaxCategory="NP">
         <syntaxElement  syntaxCategory="Det">
          <lexicalElement   partOfSpeechTag="ART">
           <text>                die                </text>
          </lexicalElement>
         </syntaxElement>
         <syntaxElement   syntaxCategory="N">
            <lexicalElement    partOfSpeechTag="NN">
             <text>          Anfangszeiten     </text>
            </lexicalElement>
           </syntaxElement>
[...]
    </syntaxElement>
  </sentence>
 </discourseElement>
</concept>
```

# Output of Natural Language Generation



# Sketch of the Architecture



# Knowledge Bases in NLG

- Defining the goal (XSLT Stylesheet, What?)
- Planning rules (PrePlan, How?)
- (Template-)grammar (TAG, Realizer How?)
- (Morphology)
- Lexicon (TAG, Realizer)

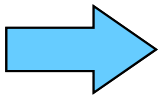
- Discourse memory (anaphora etc.)
- User model (“Interaktionsmodellierung”) (register etc.)



# First Step: Defining the Goal

- **XSLT: Mapping abstract goals to realization goals, e.g.:**

```
<request>
  <slotFill><select>
     <modality>gesture</modality>
   </select></slotFill>
  <requestFocus>
    <deepFocus idReference="mf42"/>
  </requestFocus></request>
```



```
(showme mf42)
```

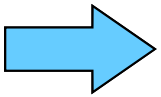
```
<xsl:template match="request/slotFill/select[normalize-space(modality/text())='gesture']">
  (showme
 <xsl:apply-templates select="requestFocus/deepFocus"/>
   )
</xsl:template>
```



# First Step (2): Using Context Information

- **XSLT: Creation of a generation knowledge base from the input, e.g.:**

```
<performance id="mf745">
  <entityKey id="mf746">
    performance_1000030
  </entityKey>
  <avMedium id="mf747">
    <entityKey id="mf748">
      avMedium_1002535
    </entityKey>
    <title>
      O Brother, Where Art Thou?
    </title>
  </avMedium>
</performance>
```



```
(GKB (
 (performance mf745)
 (entitykey
   mf746
   performance_1000030)
 ...
 (title
   mf747
   "O Brother..")
 ...
)
```



# Second Step: Sentence Planning with Templates

- **Result is a derivation tree**
- **PrePlan (a simple planning tool in Java):**
  - (Text and) sentence planning
  - Selection of templates and filling of slots, e.g.:

```
(overview mf42)
  ->
(select “You can see an overview”)
(adjoin “Node Overview-4711”)
(np-realize mf42)
```

  - `Select` and `adjoin` refer to trees and nodes of the (TAG) Grammar



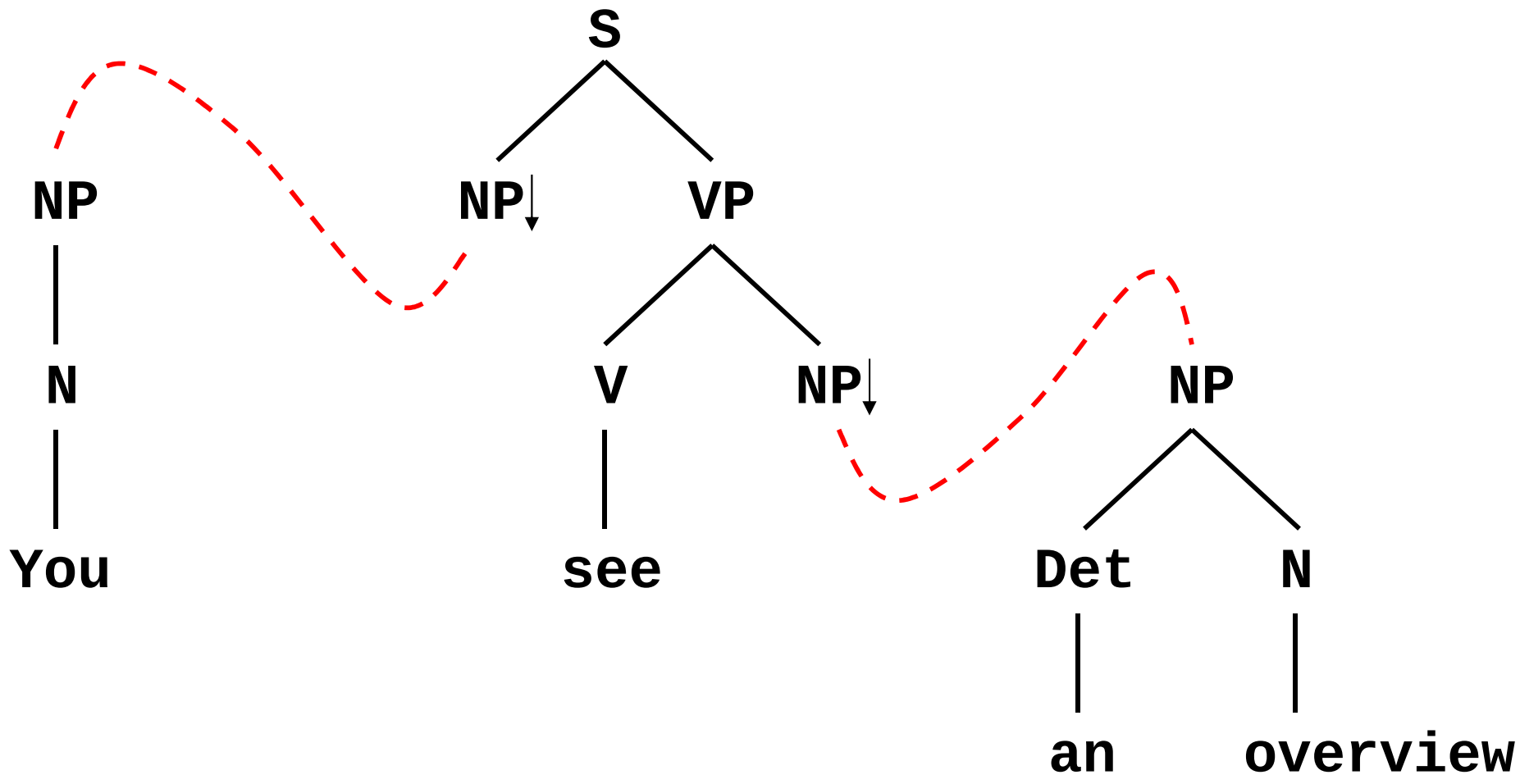
# TAG Grammars

- **Tree Adjoining Grammars (Joshi et al 1975)**
- **A grammar**
  - consists of partial trees,
  - that are combined by two operations:
    - Adjunction
    - Substitution
  - Lexicalized grammars:
    - A set of possible partial trees for every word
    - Every partial tree is a “maximal projection” of the word



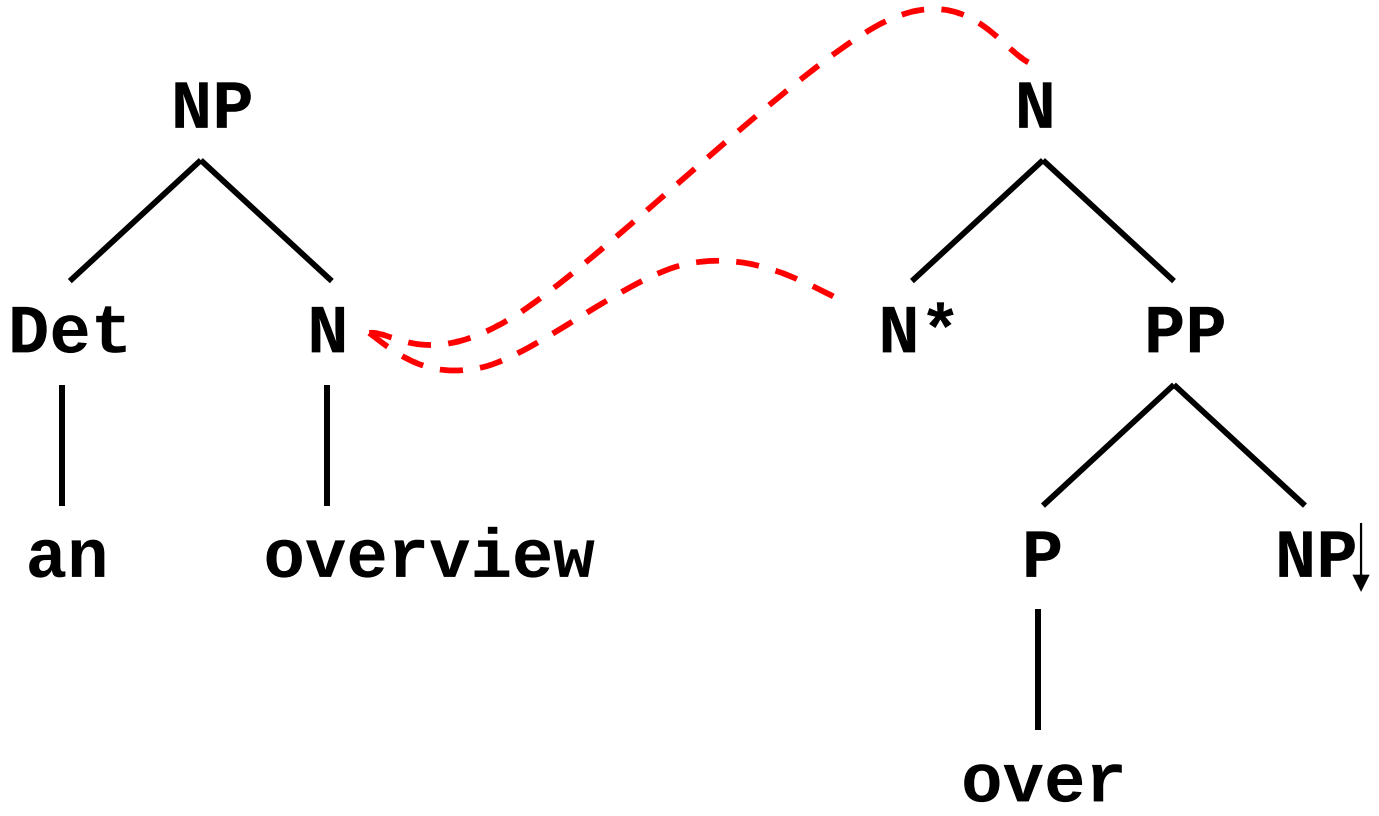
# TAG: Initial Trees

**Substitution** as in context-free grammars:



# TAG: Auxiliary Trees

**Adjunction** is more powerful than context-free grammars:



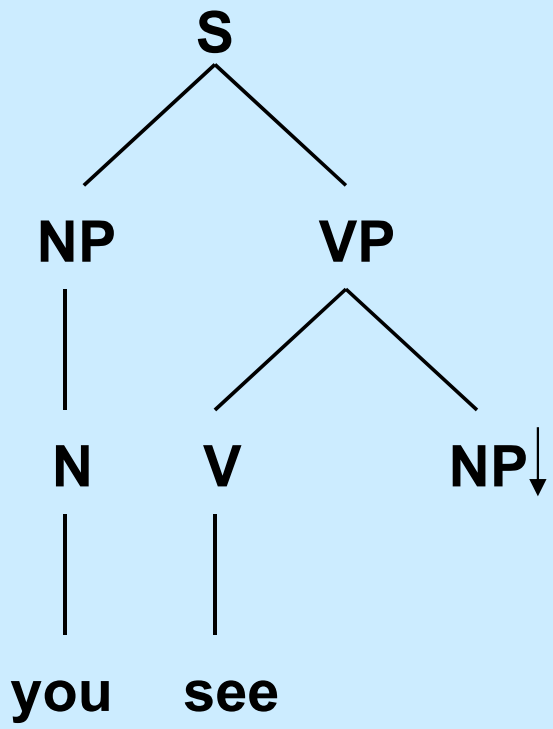
# TAG with Templates

- **Instead of lexicalized trees:**
  - A template tree contains the entire structure of a template
  - …including all words
  - A simplistic „template Grammar“ consists of complete sentences
  - Can smoothly be developed into a complete grammar
- **Problem:**
  - What are the right syntactic(?) structures?
  - General problem with CTS

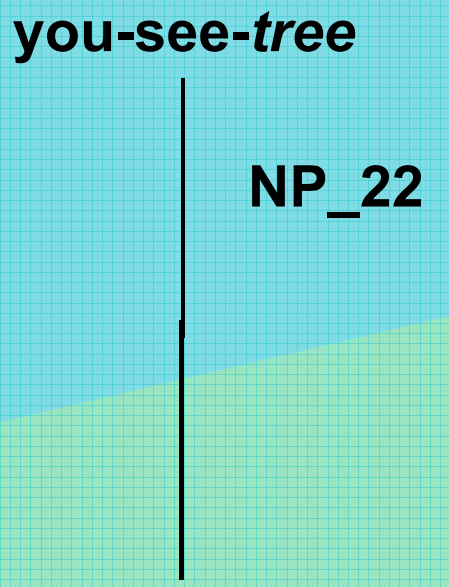


# Planning a Derivation Tree

Commenting on a graphical presentation

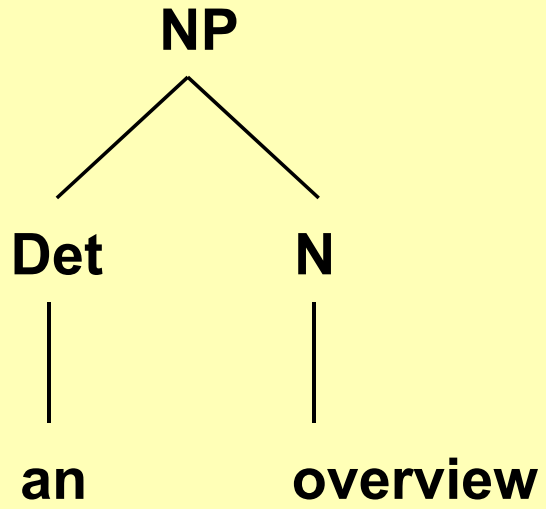



derived tree




derivation trees

Referring to a list




derived tree



# Concept-To-Speech

- Syntactic Information is used to compute Prosodic Information
- Sentences are combined to discourse tree
- Filtering of irrelevant syntactic features

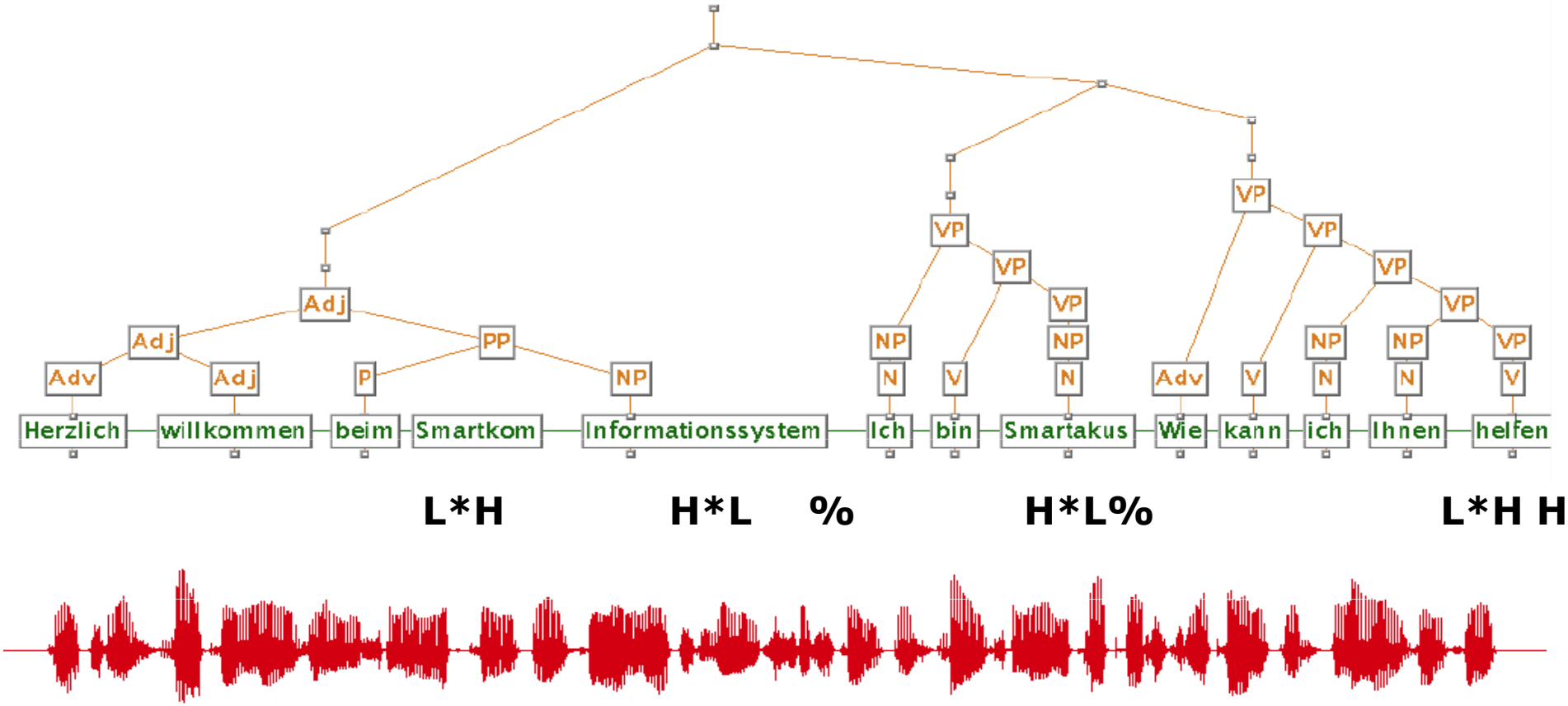
- Synthesis is based on Festival
- Preprocessing traverses syntactic structure (Scheme)

- Work carried out at IMS, Stuttgart, Germany
  Gregor Möhler, Antje Schweitzer
  (Prof. Dogil)



# CTS versus TTS

Institut für Maschinelle Sprachverarbeitung
Universität Stuttgart



# Templates

- **Where do we get the templates from?**
  - Ideally from existing grammars:
    - consistent
    - short development time
    - no/less expertise required
  - Data collection for a new application:
    - example dialogues
    - Wizard of Oz experiments
    - dialogue models
  - Growing collection of “standard templates” (will lead to a real grammar)



# Current work

- **Complete TAG implementation with unification:**
  - Porting an existing Unifier (LISP)
  - XML-Representation of the grammar:
    - Graphical tools
    - XSLT mapping to/from other formats (LISP)
- **Structure of planning rules:**
  - Separate text and sentence planning
- **Extending the set of templates**



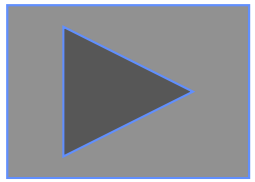
# Future Work

- **Generating referring expressions**
- **Generating text for graphics, esp. for mobile scenario “no audio”**
- **Text planning**
- **Abstract “sentence plans”:**
  - Module within syntactic realization
- **Various tools (next slide)**
- **Language independent steps of NLG**

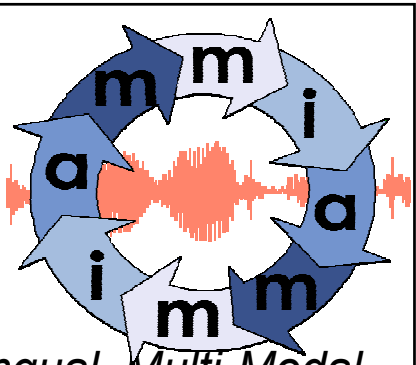


# Future Work

- **Tools for:**
  - PrePlan planning rules
  - Lexicon (morphology)
- **Template tree development scenario:**
  - Parser (with a German grammar -- Kim Gerdes) produces derivation trees
  - (Graphical) tool to
    - select correct analysis
    - relate to existing templates
    - mark fixed/variable parts



# MIAMM

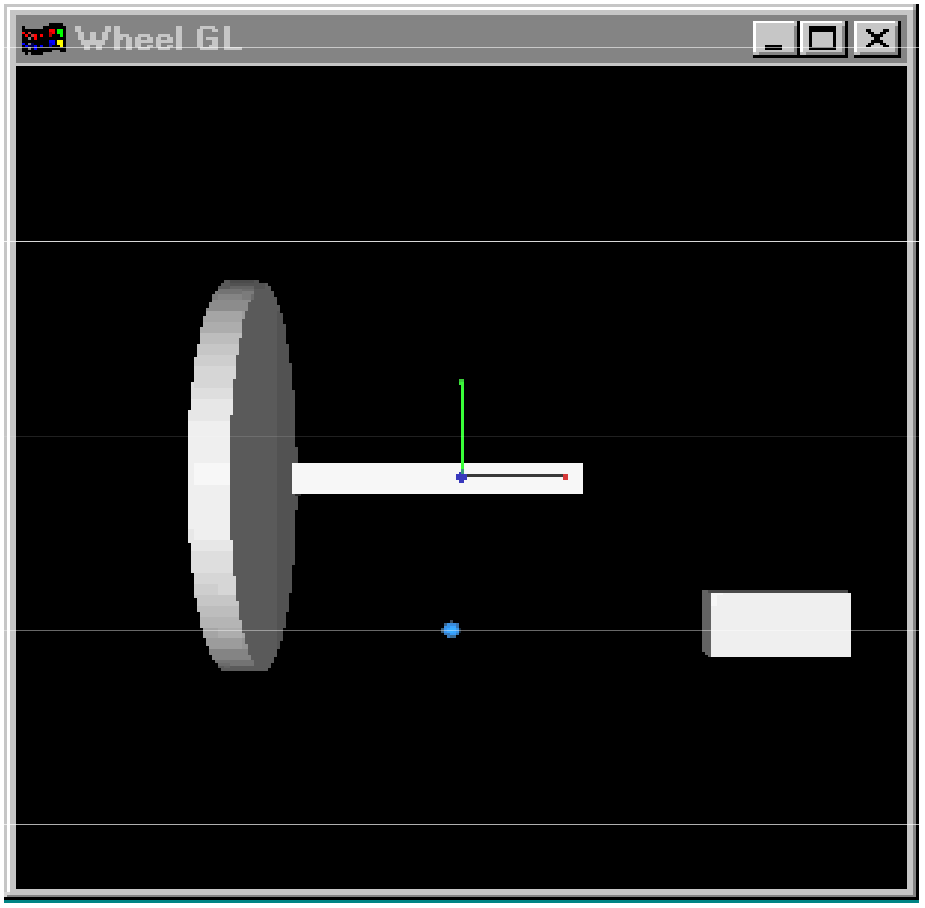



- **Multidimensional Information Access using Multiple Modalities (IST-2000-29487)**
  - **Cross Programme Action 2** *User Friendliness, Human Factors, Multi-Lingual, Multi-Modal dialog modes*
- **Duration: September 2001 - February 2004**
- **Participants**
  - **INRIA (Laboratoire Loria), FR [Coord.]**
    - Speech recognition, language analysis, contextual interpretation
  - **Deutsches Forschungszentrum für Künstliche Intelligenz, DE**
    - Graphical interface, language analysis, dialogue management
  - **Netherlands Organization for Applied Scientific Research (TNO), NL**
    - Task analysis, interaction scenarios, evaluation
  - **Sony International Europe GmbH, DE**
    - Multilingual speech recognition (en, de), software for haptic interaction, domain modeling, hardware interaction
  - **CANON Research Centre Europe (CRE), UK**
    - Multimedia database and search application



# The Haptic Device

**Phantom (www.sensable.com)**

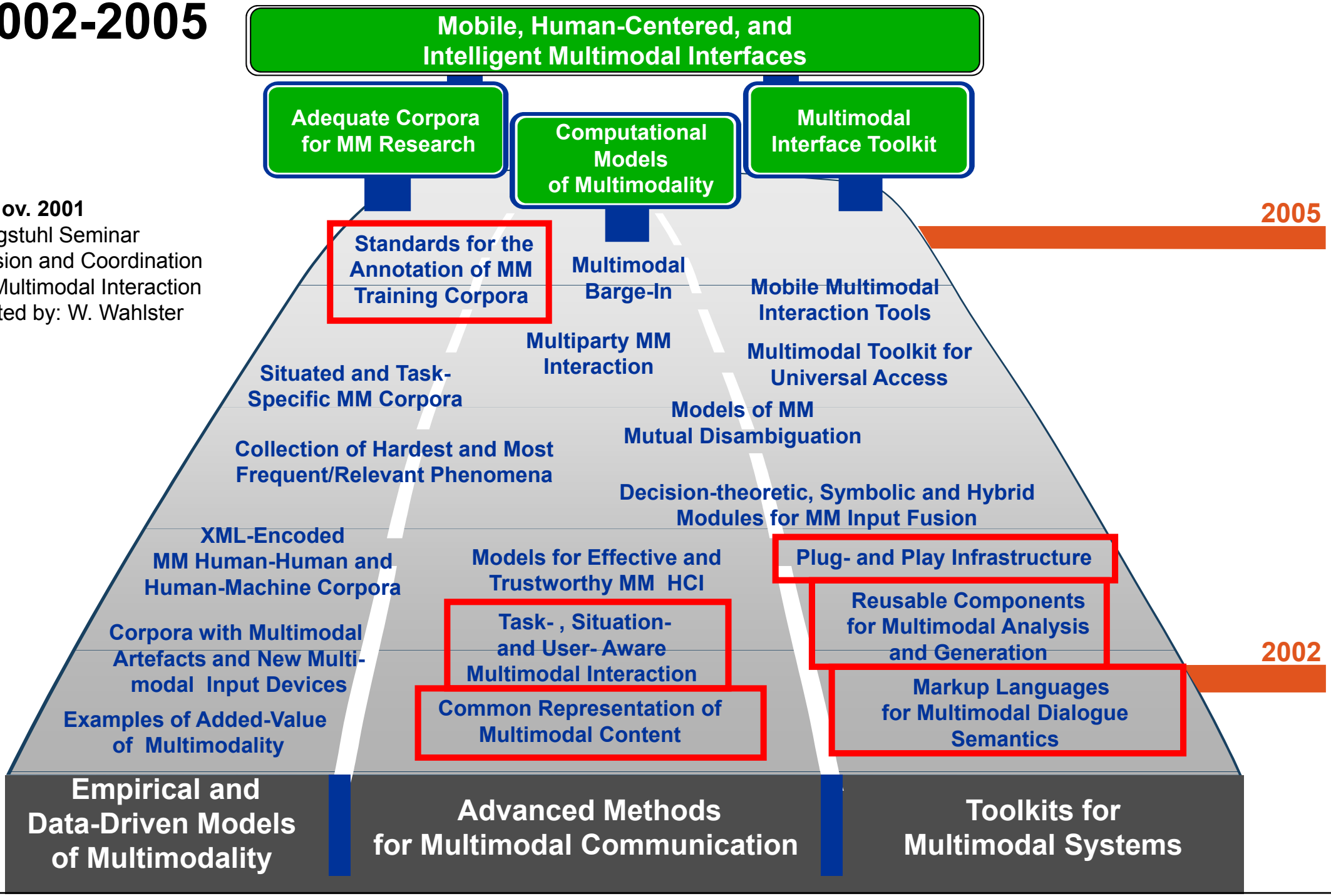



**3 degrees of freedom force feedback unit**

# Research Roadmap of Multimodality

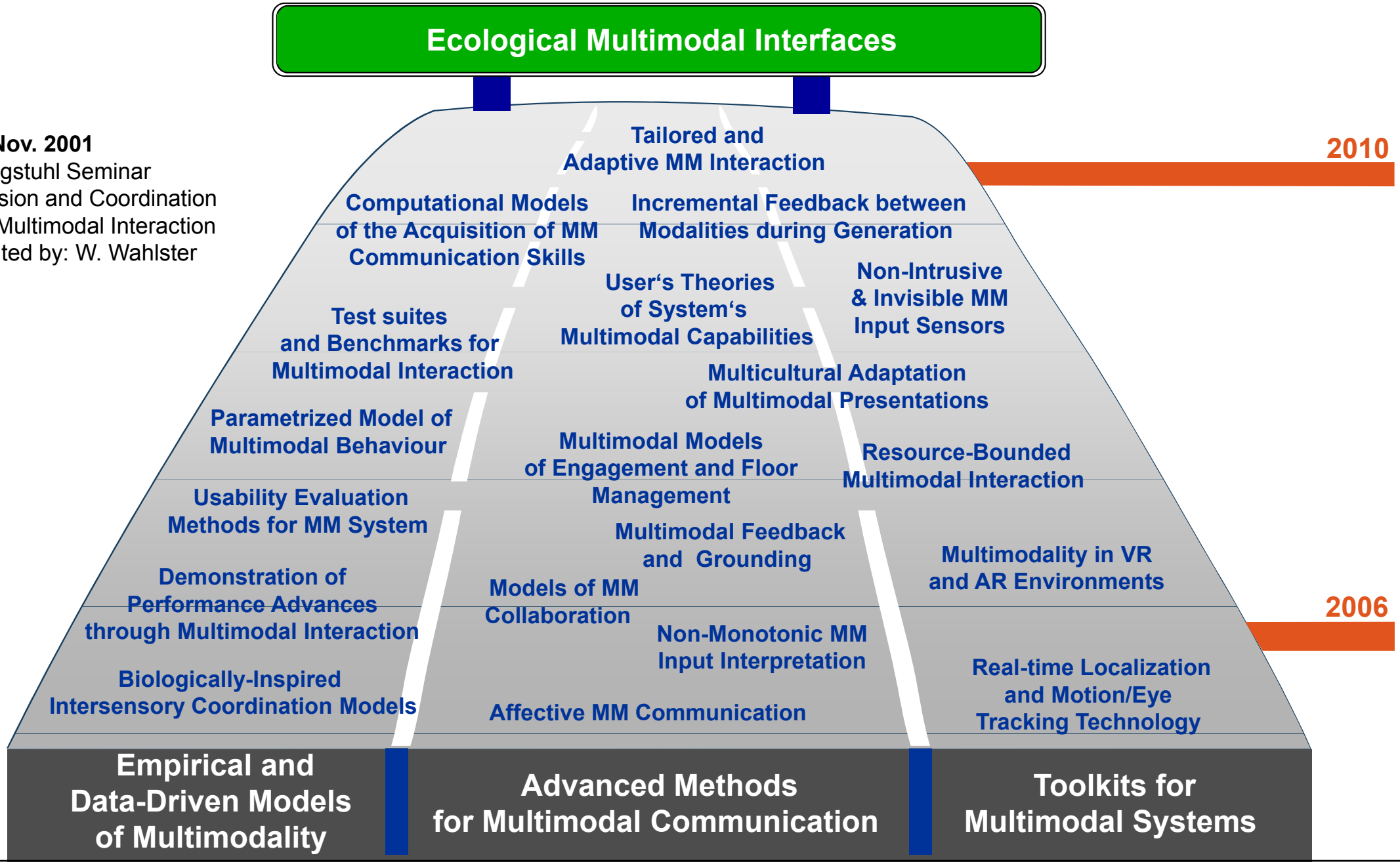
## 2002-2005

**2 Nov. 2001**
Dagstuhl Seminar
Fusion and Coordination
in Multimodal Interaction
edited by: W. Wahlster



# Research Roadmap of Multimodality

## 2006-2010



# Research Roadmap of Multimodality 2001-2010

## Enabling Technologies and Important Contributing Research Areas

**2 Nov. 2001**
Dagstuhl Seminar
Fusion and Coordination
in Multimodal Interaction
edited by: W. Wahlster

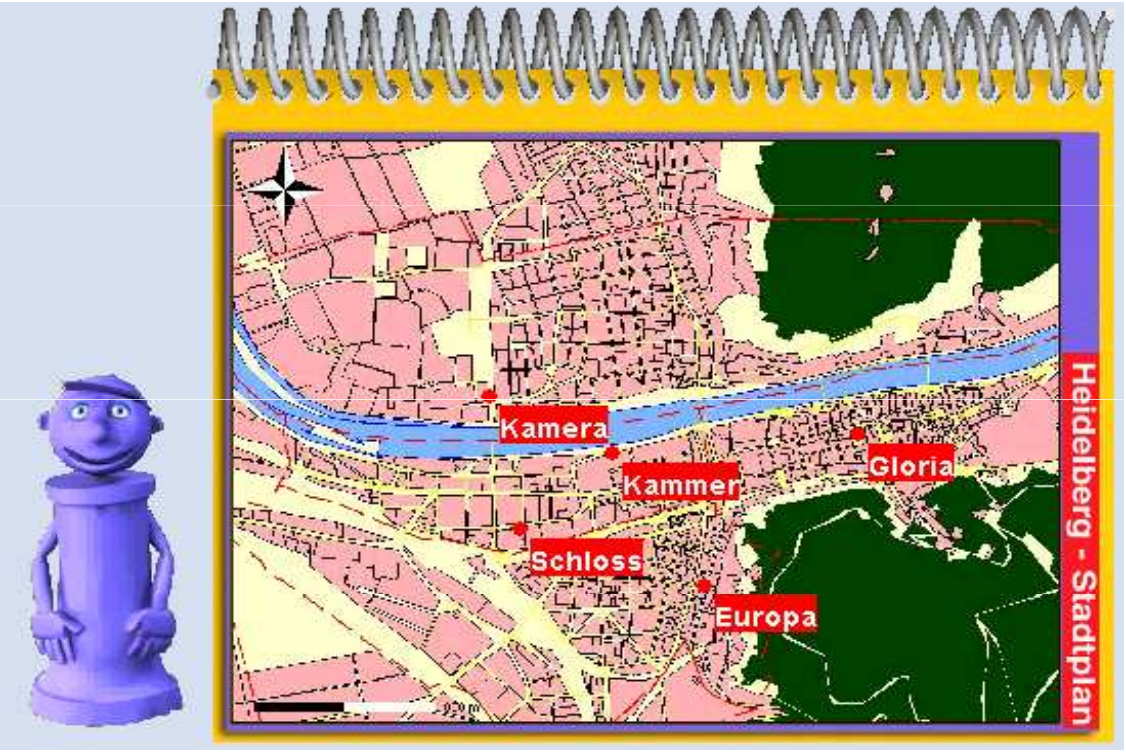
| Multimodal Input | Multimodal Interaction | Multimodal Output |
|---|---|---|
| ● Sensor Technologies | ● User Modelling | ● Smart Graphics |
| ● Vision | ● Cognitive Science | ● Design Theory |
| ● Speech & Audio Technology | ● Discourse Theory | ● Embodied Conversational Agents |
| ● Biometrics | ● Ergonomics | ● Speech Synthesis |

● Machine Learning ● Formal Ontologies ● Pattern Recognition ● Planning



# Multimodal Interaction in SmartKom

**Scenario:**
public (mobile, home)
**Application:**
movie information
(EPG, email, phone, fax,
address book,
tv and vcr control,
routing/tourist info)




**U:** ***I want to make a reservation in (☼) this movie theater***

**S:** This theater does not take reservations

**U:** ***Then a different one, (☼) this one perhaps***

IJCAI 2001
Workshop TASK-4
Seattle, WA, USA

# Overlay as the basic operation in discourse processing

**Jan Alexandersson**

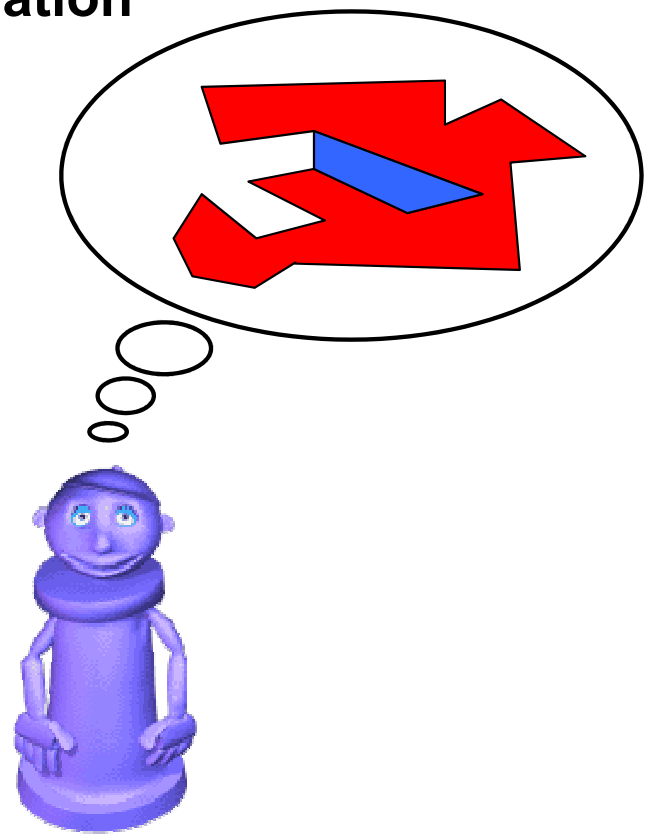
**Tilman Becker**

**Deutsches Forschungszentrum für Künstliche Intelligenz GmbH**
**Stuhlsatzenhausweg 3, Geb. 43.8 - 66123 Saarbrücken**
**Tel.: (0681) 302-5271**
**Email: {janal,becker}@dfki.de**
**www.dfki.de**

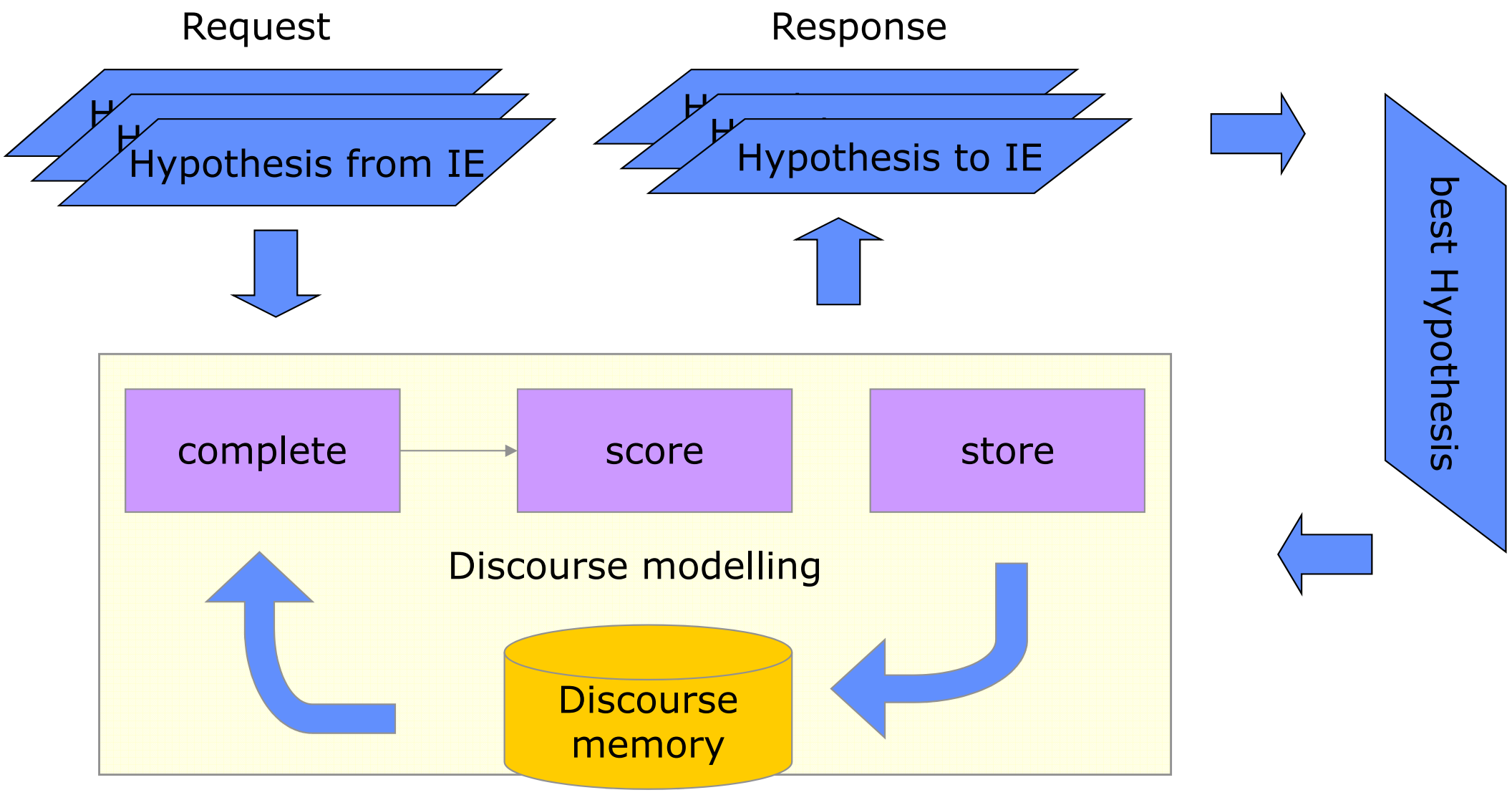


# Discourse modelling tasks

- **Construct a discourse memory of contextual information**
- **Hypotheses:**
  - enrich w/ context information
  - compute scores
- **discourse memory:**
  - enrich
  - retract
  - (partially) overwrite



# Architecture



# Dialog memory

- **A typical dialog situation:**
  - **User**: *I want to see Matrix*
  - **Sytem**: Ok, it runs at 8 and at 10
  - **User**: *At 8*
- **Dialog memory:**
  - structured storage for utterances (and their meaning)
- **“current context:”**
  - data structure representing the currently active context
  - e.g.: Matrix at 8



# Putting the user in context

- New information is added to current context,
- Result:

  updated current context
- used, e.g. for a database query



# Unification-based Integration of Speech and Gesture

**MVPQ, AT&T**
**Johnston 2000**

**Speech recognition**

**Gesture recognition**

**Speech analysis**

**Gesture analysis**

Multimodal Chart Parser

**Unification-based multimodal Grammar**

**Application interface**

# Updating current context with Unification

- **Representing complex discourse objects as typed feature structures (TFS), e.g. Johnston 1998**
- **Used, e.g. in media fusion:**
  - **User**: *I want to see this one* [pointing to movie “Matrix”]
  - Speech: “I want to see X”
  - Gesture: “When is Matrix showing?” “I want to see Matrix.” ....
  - Media Fusion: “I want to see Matrix.”
- **Problem: enumeration of all structures (in deixis)**



# Typed feature structures and XML

- **In the SmartKom project, discourse objects are represented in XML**
- **Mapping from XML to TFS assumed**
- **Example:**

```
<performance>
    <time>...</time>
    <film>
        <title>Matrix</title>
    </film>
...
</performance>
```

$\Leftrightarrow$

$$\begin{bmatrix} type{:}cinemaEntertainment \\ time{:}... \\ film{:}\left[title{:}\text{Matrix}\right] \\ ... \end{bmatrix}$$

# The limits of unification

- **Not all new information is consistent with current context**
- **Even for Mediafusion:**
  - User: This one, (but) in green
- **Some parts must be kept, some be overwritten**
  - “keep and overwrite”, M. Streit

- **Provide a principled method,**

  **based on unification**



# Overlay to the rescue

- Unification is monotonic, reflexive operation
- old information from the current context can be changed, new information is more important

❑ we need a non-monotonic, non-reflexive operation: overlay



# Overlay to the rescue

- **Task: compare new (intention) hypothesis against discourse history**
- **new information consistent with focus:**

  ✞ **Unifikation**
- **new in formation (partially) inconsistent with focus:**

  ✞ **Overlay**



# Example for Unification

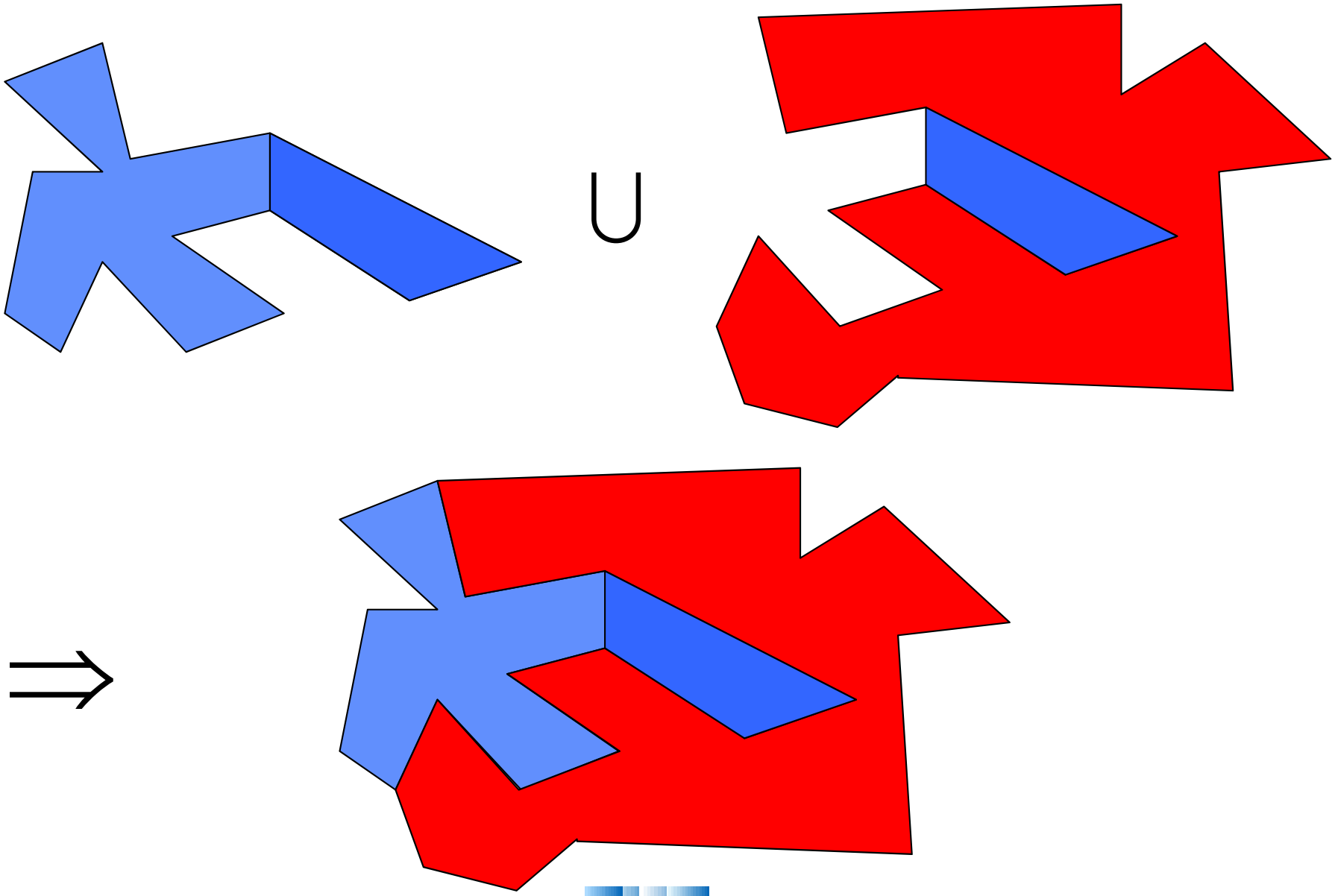
**U:** ***I want to go to the movies tonight***

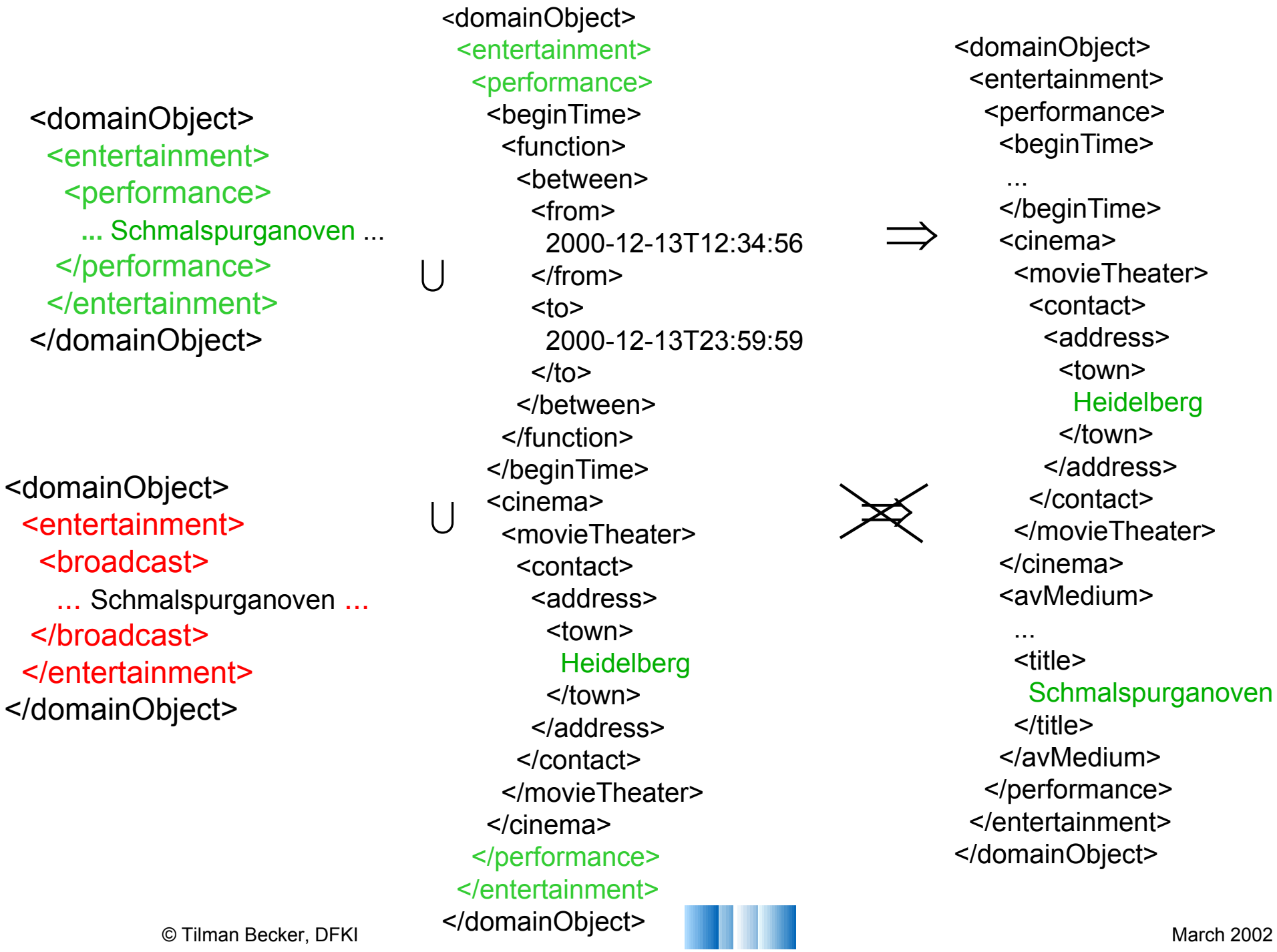
**S:** Here is a list of the films that are shown in Heidelberg tonight**: (SmartKom shows a list)**

**U:** ***I want to see (☼) this one, where is it playing?***

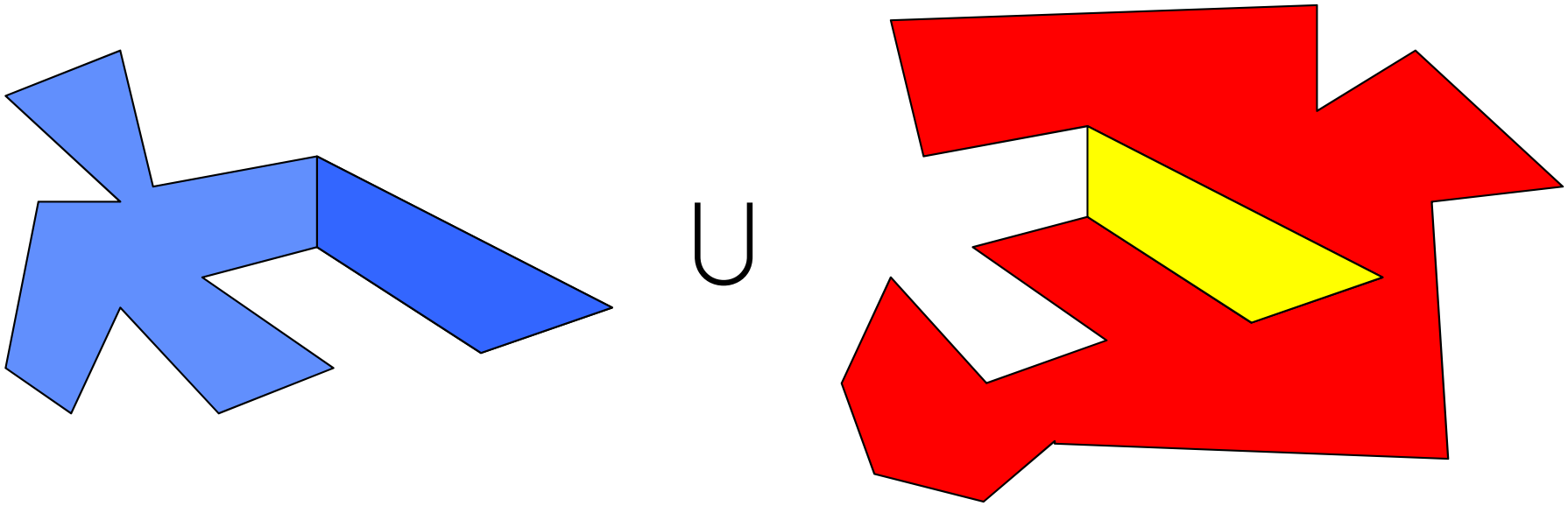


# Unification: monotonic operation



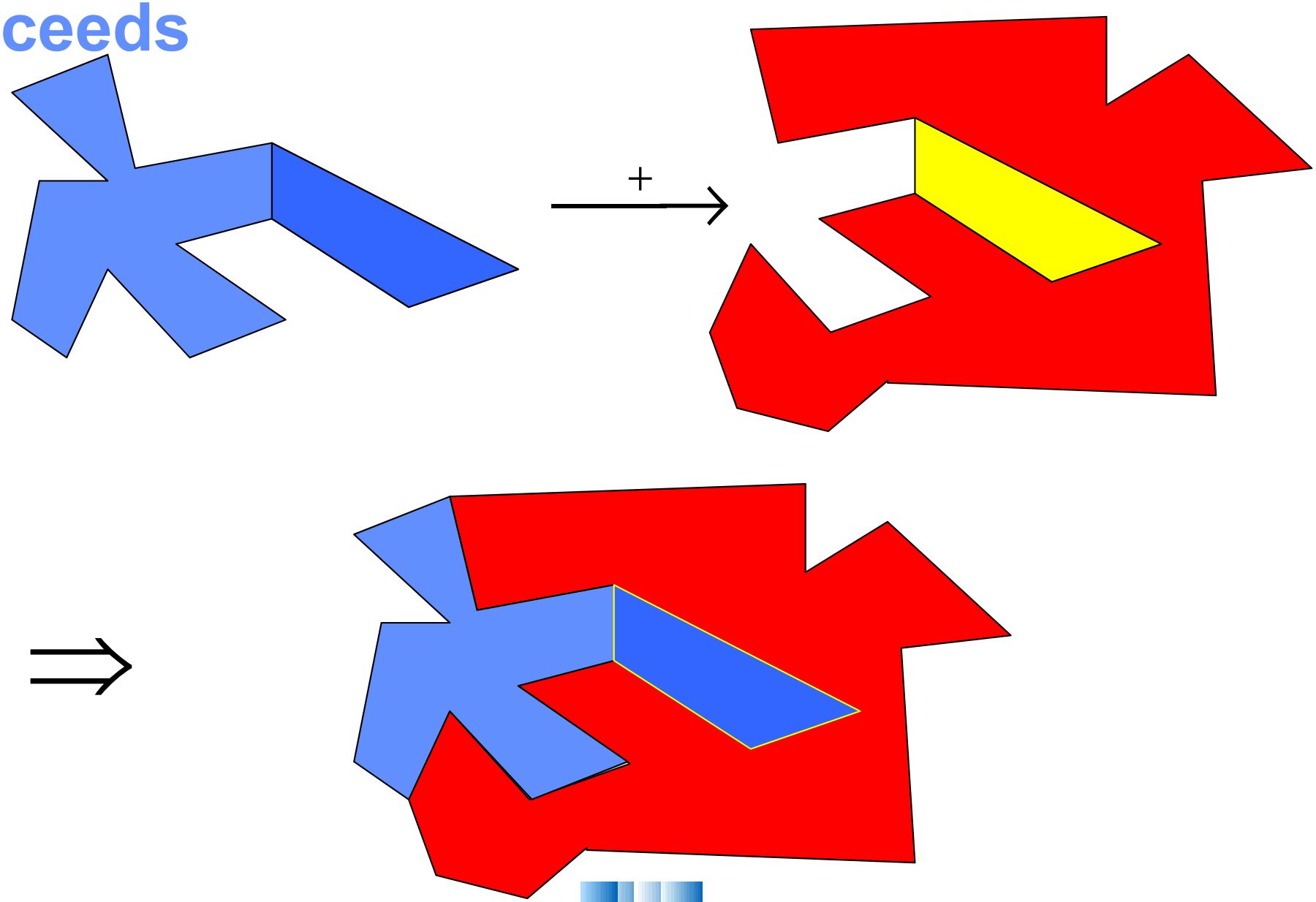


# Unification: compatibility condition



$\Rightarrow$ **fail**

# Overlay: nonmonotonic operation, that always succeeds



# Example for Overlay

**U:** ***I want to make a reservation in (☼) this movie theater***

**S:** This theater does not take reservations

**U:** ***Then a different one, (☼) this one perhaps***



<domainObject>
<entertainment>
<performance>
<cinema>
<movieTheater>
...
<name>Studio Europa</name>
<contact>
...
</contact>
</movieTheater>
</cinema>
</performance>
</entertainment>
<movieTheater/>
</domainObject>

+

<domainObject>
<entertainment>
<performance>
<beginTime>
<function>
...
</function>
</beginTime>
<cinema>
<movieTheater>
...
<name> Kamera </name>
<contact>
...
</contact>
</movieTheater>
</cinema>
<avMedium>
...
<title>
Schmalspurganoven
</title>
</avMedium>
</performance>
</entertainment>
</domainObject>

⇒

<domainObject>
<entertainment>
<performance>
<beginTime>
...
</beginTime>
<cinema>
<movieTheater>
...
<name> **Studio Europa**
</name>
<contact>
...
</contact>
</movieTheater>
</cinema>
<avMedium>
...
<title> Schmalspurganoven
</title>
</avMedium>
</performance>
</entertainment>
</domainObject>

P=0.7

<domainObject>
<movieTheater>
...
<name> Studio Europa </name>
<contact>
...
</contact>
</movieTheater>
<movieTheater/>
</domainObject>

+

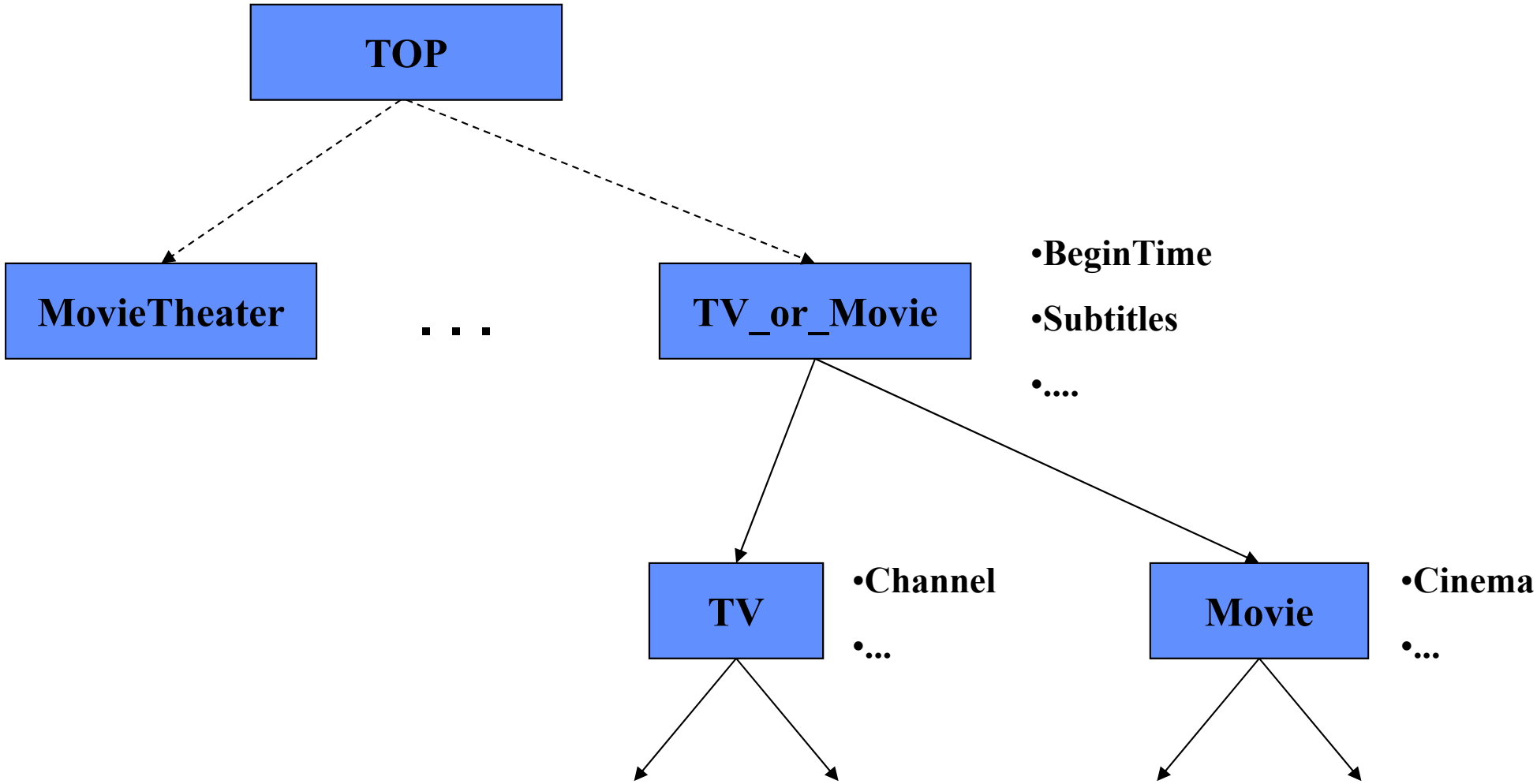
<domainObject>
<movieTheater>
...
<name> Studio Europa </name>
<contact>
...
</contact>
</movieTheater>
<movieTheater/>
</domainObject>

⇒

P=0.3



# Type Hierarchy



# Overlay and Typed Feature Structures (TFS)

- **Two non-unifiable structures (type clash):**
  - Cover is more important than background
  - Keep information from background:
    - Find lub (most specific common supertype)
    - “reduce” background to this type
    - recursively apply overlay on features
    - for atomic values: ignore background



# An Example

**U:** ***What films are showing on TV tonight?***

**S:** [shows list of films]

**U:** ***That‘s a boring program, I‘ll rather go to the movies.***

**Q: How do we save *“tonight”* ?**

# An Example

**U:** ***What films are showing on TV tonight?***

**⇒ Context of type *TV***

**S:** [shows list of films]

**U:** ***That's a boring program, I'll rather go to the movies.***

**⇒ Analysis finds data of type *Movie***

- **incompatible with context**
- **abstract context to lub *TV_or_Movie***

  **(keeps *"tonight"*)**
- **unifiable with analysis**

# Does TFS solve all your problems?

- **An adequate type hierarchy must exist**
  - "most specific common supertype"
  - Carpenter and others on **default unification**
- **Overlay (and unification) of lists and sequences is not well defined -- and content dependent**
- **What about "semantics", e.g. DRS, Verbmobil VIT/MRS?**



# Implementation

- **Mapping of XML Schema to Java classes**

  **see data binding:**

  - Castor Project
  - Java 1.4: JAXB
- **XML documents are represented internally as instances of these classes**
- **Unification and overlay are realized using the Java meta protocol**



# Next steps

- **Treatment of subobjects**
  - find relation to context
- **Grounding**
  - model the presentation-acceptance cycle of discourse objects
- **Inclusion of dialog management plans**
  - expected vs. Possible next states
  - better interpretation in context
- **Fully formalize XML schema to tfs mapping**



# Summary of the Talk

- **Two large-scale spoken dialogue projects: Verbmobil, SmartKom**
- **Spotlight on Aspects of NLG, Discourse Processing**

- **Conclusion:**
  - Large Scale projects offer new insights‘
    See also upcoming 6th framework of EU
  - Modular Architecture (data pool driven middleware)
  - combine shallow and deep approaches
    - multi-engine approach
    - fully specified template approach
  - emerging multi-modal markup language



# Finally

Thank you very much
for your kind attention.



# Verbmobil -The Project

**Some information for those who haven´t heard of Verbmobil recently**

- **speaker independent speech-to-speech translation system for appointment scheduling and travel planning:**
  **German ↔ English (10 175 words German, 6871 words English)**
  **German ↔ Japanese (2566 words Japanese)**
- **69 modules, full configuration 3.5 GB**
- **23 participating institutions (in Verbmobil II)**
- **over 900 full workers and students involved**
- **project duration: 1993 - 2000**

❑ **scientific, software technology, and management challenges**



# Scientific Results

**There are over 600 refereed papers on the various aspects of and achievements in Verbmobil.**
**See also** *W. Wahlster (ed.): Verbmobil: Foundations of Speech-to-Speech Translation, Springer Verlag,* **to appear July 2000** ... at any shop near your office :-)

**Some highlights**

- **Speaker independent speech recognition over various channels**
- **Language ID**
- **Unknown words**
- **Prosodic information (segmentation, stress etc.) used in various modules**
- **Repair of hesitations, repetitions**
- **Combination of parser analysis fragments**
- **Semantic representation: VIT**
- **Context and dialog knowledge supports translation**
- **Efficient semantic transfer**
- **Content to speech generation**
- **Word concatenative speech synthesis**
- **Dialog minutes and summaries**
- **Large data collection with annotation on various levels (e.g. tree-banks, dialog acts)**
- **....**

# Multi-Engine for Translation (DIE)

- Large-Scale Web-based Evaluation: 25 345 Translations, 65 Evaluators
- Sentence Length 1 - 60 Words

| *Translation Thread* | *Word Accuracy ≥ 50% 5069 Turns* | *Word Accuracy ≥ 75% 3267 Turns* | *Word Accuracy ≥ 80% 2723 Turns* |
|---|---|---|---|
| Case-based Translation | 37% | 44% | 46% |
| Statistical Translation | 69% | 79% | 81% |
| Dialog-Act based Translation | 40% | 45% | 46% |
| Semantic Transfer | 40% | 47% | 49% |
| Substring-based Translation | 65% | 75% | 79% |
| **Automatic Selection** | **57% / 78% *** | **66% / 83% *** | **68% / 85% *** |
| **Manual Selection** | **88%** | **95%** | **97%** |

*** After Training with Instance-based Learning Algorithm**



# Agreement between Different Labels

| | B3 | -B3 | D3 | -D3 |
|---|---|---|---|---|
| M3 | 79 | **21** | 52 | **48** |
| -M3 | **3** | 97 | **0** | 100 |

| | B3 | -B3 | M3 | -M3 |
|---|---|---|---|---|
| D3 | 91 | **9** | 97 | **48** |
| -D3 | **8** | 92 | **7** | 93 |

**B3** **prosodic boundary**

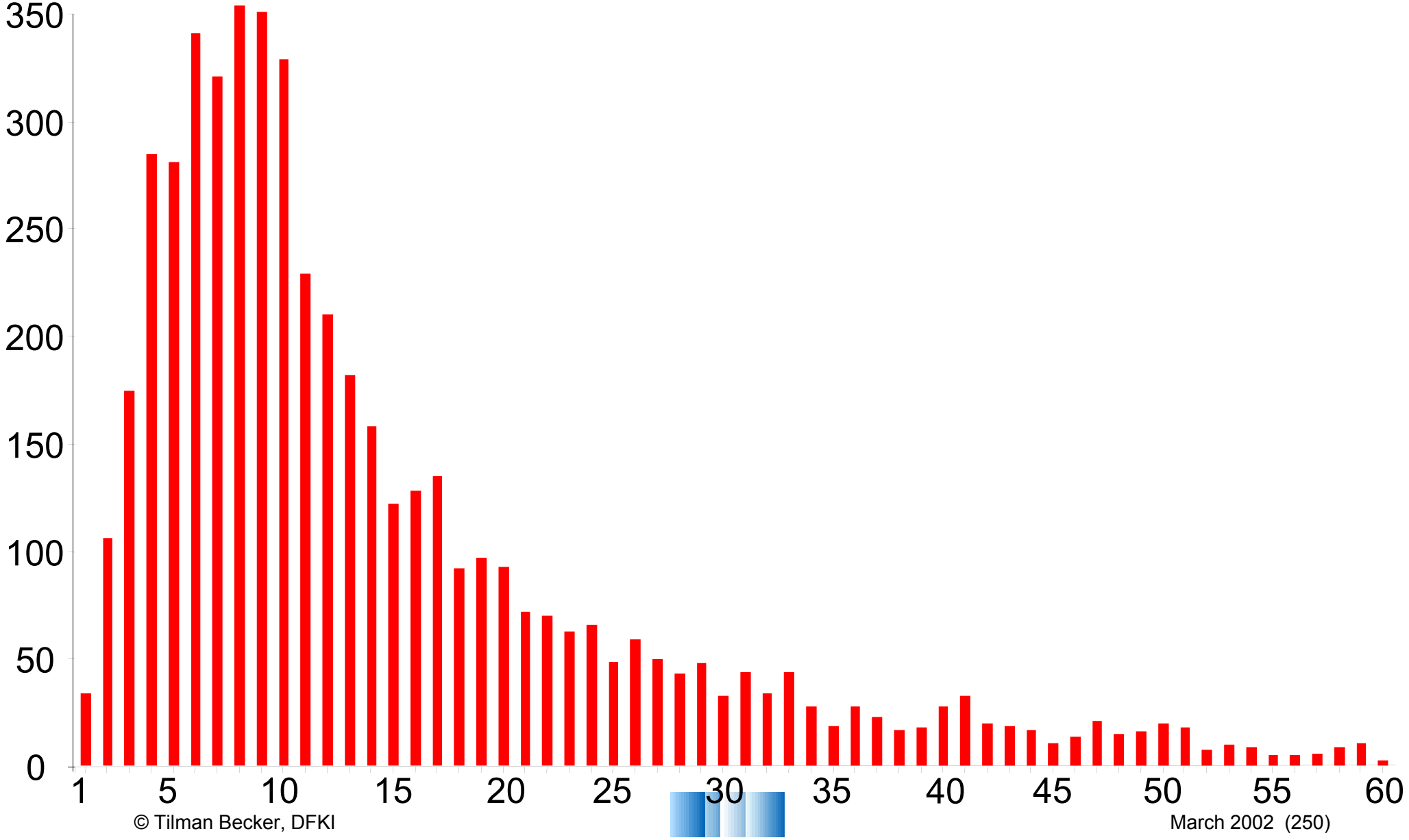
**M3** **syntactic boundary**

**D3** **dialog act boundary**

- **Most M- (79%) and D-bound. (91%) are prosodically marked**
- **About half of the M-boundaries (52%) are D-boundaries**
- **Practically all D-boundaries (97%) are M-boundaries**
- **High agreement between the non-boundaries (92-100%)**
- **Even a prosody with a recognition rate of 100% will not find 21% of the M-boundaries and 9% of the D-boundaries!**



# Distribution of Sentence Length in Large-Scale Evaluation



# Results of End-to-End Evaluation Based on Dialog Task Completion for 31 Trials

| Topic | Successful Completions | Attempts | Percentage of Successful Task Completions | Frequency-Based Weighting Factor |
|---|---|---|---|---|
| Meeting time | 25 | 28 | 89,3 | 0,90 |
| Meeting place | 21 | 27 | 77,8 | 0,87 |
| Means of transport | 30 | 30 | 100 | 0,97 |
| Departure place | 22 | 25 | 88 | 0,81 |
| Arrival time | 22 | 26 | 84,6 | 0,84 |
| Place of arrival | 17 | 19 | 89,5 | 0,61 |
| Who reserves the hotel | 28 | 31 | 90,3 | 1 |
| How to get to departure place | 7 | 9 | 77,8 | 0,29 |
| Means of return transportation | 23 | 24 | 95,8 | 0,77 |
| Departure place for return trip | 16 | 17 | 94,1 | 0,55 |
| Meeting time for return trip | 3 | 4 | 75 | 0,13 |
| Meeting place for return trip | 3 | 4 | 75 | 0,13 |
| Arriving place for return trip | 10 | 11 | 90,9 | 0,35 |
| **Total Number of Dialog Tasks** | 227 | 255 | | |
| **Average Percentage of Successful Task Completions** | | | 86,8 | |
| **Weighted Average Percentage of Successful Task Completions** | | | 89,6 | |

# Test Results for the current Repair Module

| | Detection | | Correct scope | | gen. correct scope | |
|---|---|---|---|---|---|---|
| | Recall | Precision | Recall | Precision | Recall | Precision |
| Test 1 | 49% | 70% | 47 % | 70% | – | – |
| Test 2 | 71% | 85% | 62% | 83% | 64% | 84% |

**Remember:**

***The output of the Repair module are additional hypotheses for the linguistic analysis. The original hypotheses remain in the WHG***



# Examples

| | |
|---|---|
| Text | Wie wäre es denn mit dem achtzehnten, weil ich am siebzehnten noch verhindert bin. |
| Transl. | How about the eighteenth, because I am still booked on the seventeenth. |
| Speech | Wie wäre es denn mit dem achtzehnten, weil ich am siebzehnten noch verhindert, dann |
| Transl. | How about the eighteenth, because I still booked on the seventeenth then. |
| Text | Sehr gut, ja. dann fahren wir da los. alles klar. danke schön. |
| Transl. | Very good, yes. then we will go then leave. all right. thank you. |
| Speech | Sehr gut , ja ich dann fahren wir da uns , alles klar dann schon |
| Transl. | Very good, well then we will go then I us, all right then already. |
| Text | Mittwoch, den sechsten, geht nicht. Montag, der elfte. |
| Transl. | Wednesday, the sixth, isn't possible. Monday, the eleventh. |
| Speech | Wie Mittwoch den sechsten geht, nicht, Montag , der elfte? |
| Transl. | How is, not Wednesday the sixth, Monday, the eleventh? |
| Text | Ah, ja, ja, die haben einen guten Service. |
| Transl. | Oh, well, well, they have a good service. |
| Speech | Ah, ja, die ja guten Service. |
| Transl. | Oh, yes, good yes the service. |
| Text | Genau, das wäre dann eine Übernachtung. |
| Transl. | Exactly, then , that would be an overnight stay. |
| Speech | Genau, das wäre dann eine Übernachtung. |
| Transl. | Exactly, then, that would be an overnight stay. |