Question Answering & the Semantic Web

Günter Neumann

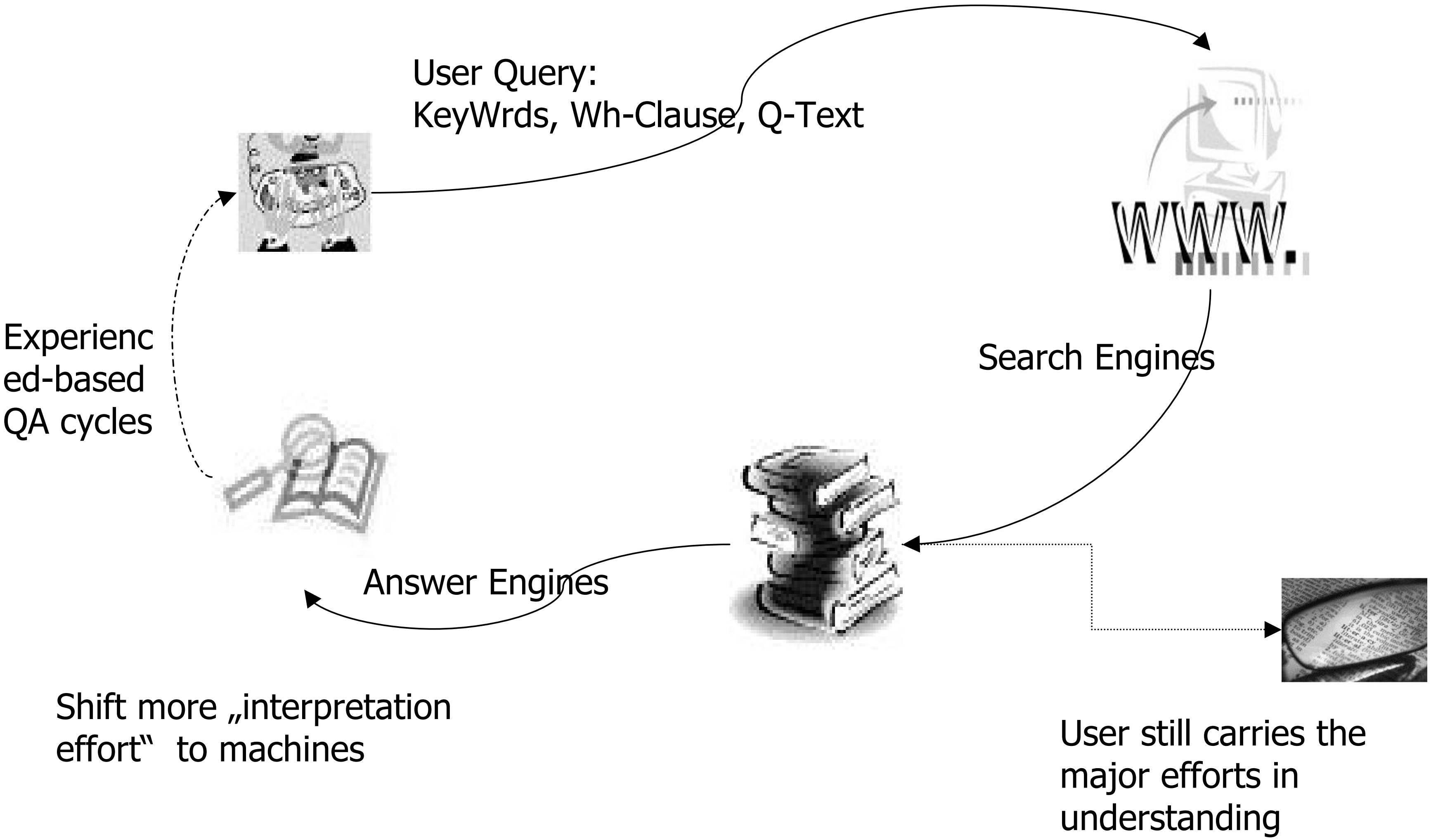
Language Technology-Lab

DFKI, Saarbrücken

Overview

- Hybrid Question Answering
- Language Technology and the Semantic Web

Motivation: From Search Engines to Answer Engines

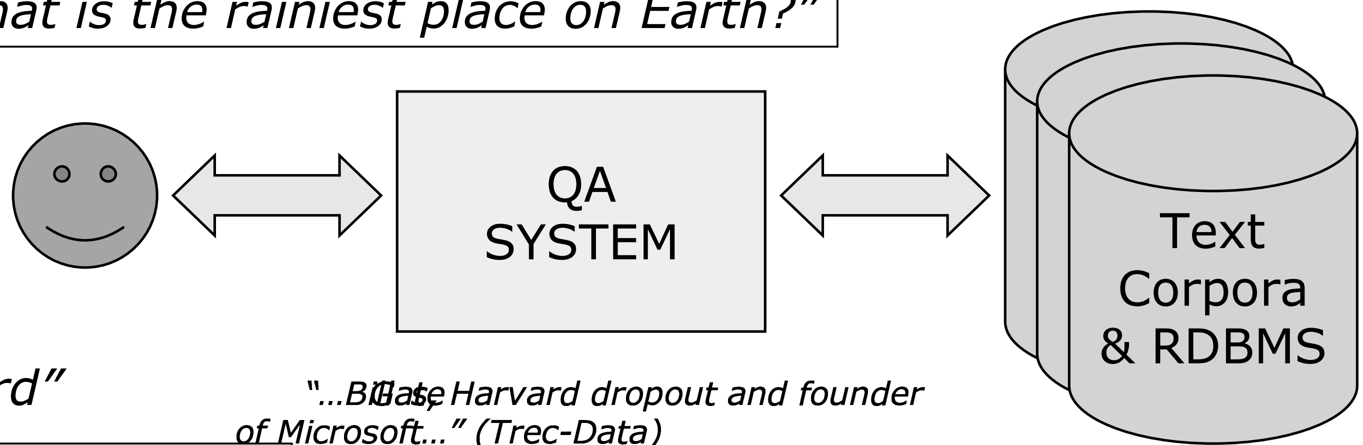


Question Answering

- Input: a question in NL; a set of text and database resources
- Output: a set of possible answers drawn from the resources

"Where did Bill Gates go to college?"

"What is the rainiest place on Earth?"



"Harvard"

"…Bill Gates, Harvard dropout and founder of Microsoft…" (Trec-Data)

"Mount Waialeale"

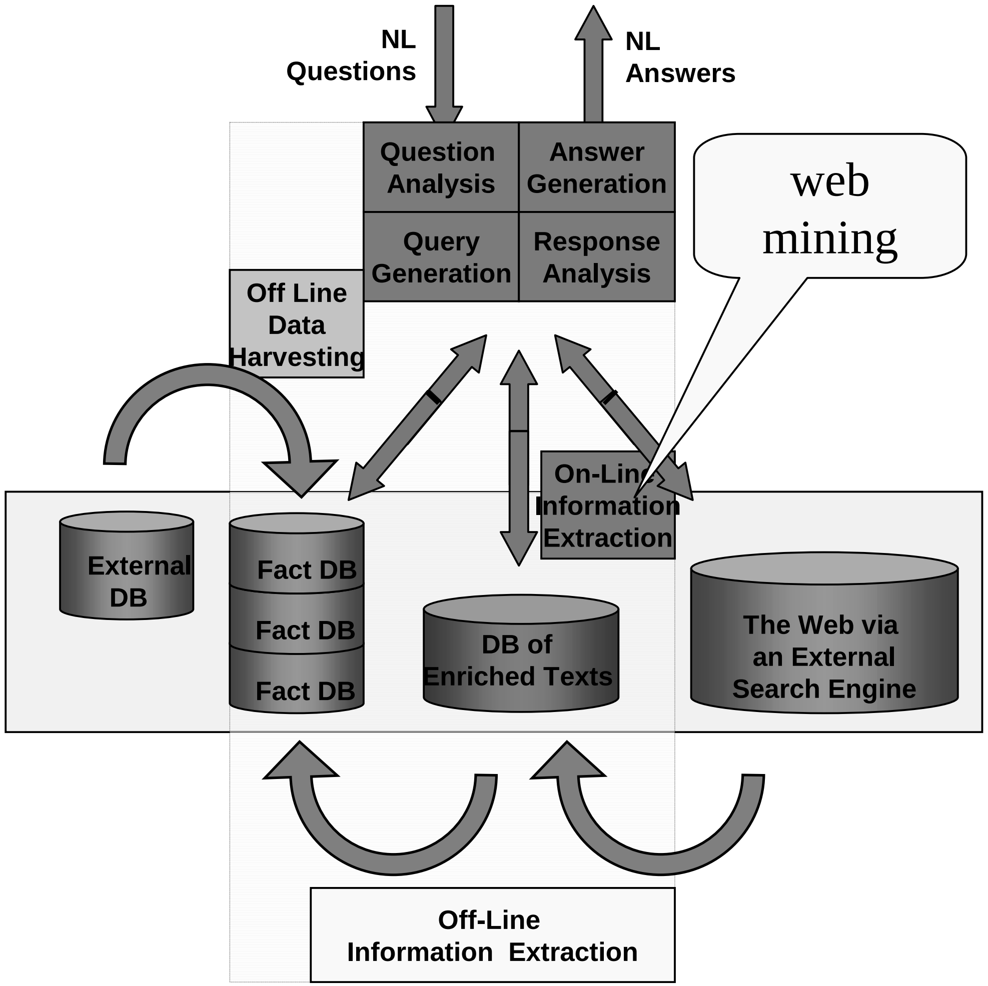
"… In misty Seattle, Wash., last year, 32 inches of rain fell.
Hong Kong gets about 80 inches a year, and even Pago Pago,
noted for its prodigious showers, gets only about 196 inches annually.
(The titleholder, according to the National Geographic Society,
is Mount Waialeale in Hawaii, where about 460 inches of rain falls each year.) …"
(Trec-Data; but see Google-retrieved Web page.)

Hybrid QA Architecture

Hypothesis

real-life QA systems will perform best if they can

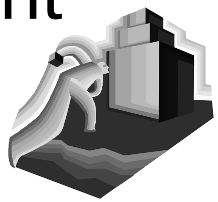
- *combine* the virtues of domain-specialized QA with open-domain QA
- *utilize* general knowledge about frequent types and
- *access* semi-structured know-ledge bases



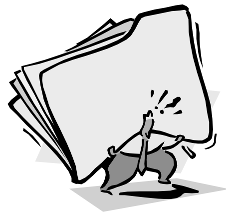
Advertisement:
DFKI project Quetal
2003-2005

Design Issues

- Foster bottom-up system development
  - Data-driven, robustness, scalability
  - From shallow & deep NLP



- Large-scale answer processing
  - Coarse-grained uniform representation of query/documents
  - Text zooming
    - From paragraphs to sentences to phrases
  - Ranking scheme for answer selection

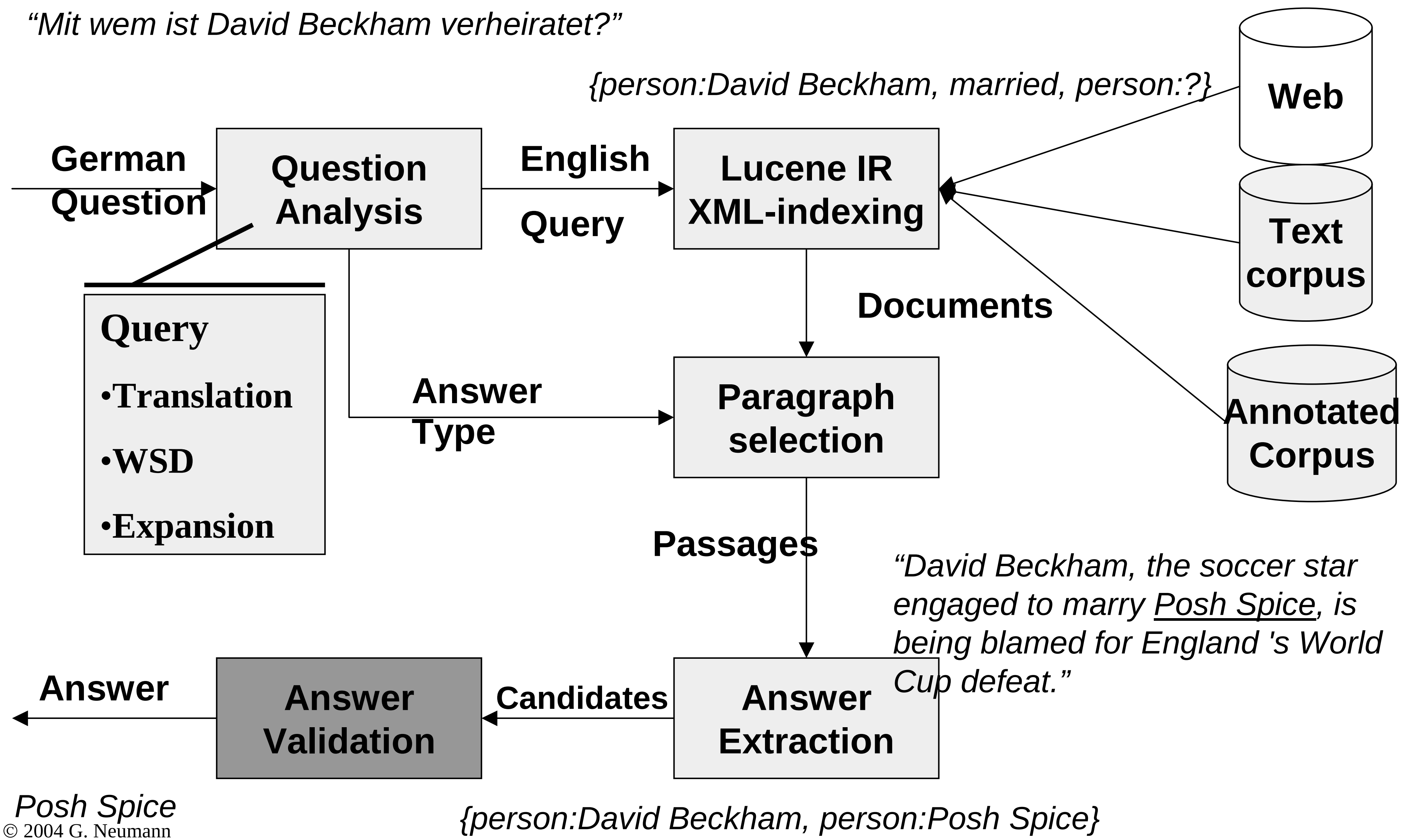


- Common basis for
  - Online Web pages
  - Large textual sources

BiQue: A Cross-Language Question-Answering System
(cf. Neumann&Sacaleanu, 2003)

- Goal:
  - Given a question in German, find answers in English text corpora
- Sub-tasks
  - Integration of existing components
    - IR-engines, our IE-core engine, EuroWordNet
  - Development of methods/components for
    - Question translation & expansion
    - Unsupervised NE recognition
  - Participation at QA-track at Clef –2003/2004

Major control flow of BiQue



Query Translation & Expansion

- First idea:
  - Only use EuroWordNet
  - Defines a word-based translation via synset offsets
- Experience
  - EuroWordNet too sparse on German side
  - Neverless introduced too much ambiguity
  - NE-translation is crucial
- So far, not very much of help

- Second idea:
  - Use EuroWordNet
  - Use external MT-services
  - Overlap-mechanism for query expansion
- Crosslingual because
  - Q-type & A-type from DE-Question Analysis
  - Synsets from EuroWN direct query expansion (online alignment)
- Experience
  - External MT services also used for Word-Sense-Disambiguation WSD
  - Reduced degree of ambiguity

Example (cf. Neumann&Sacaleanu, 2003)

1. Translation services for Word Sense Disambiguation (WSD)

Wo wurde das Militärflugzeug Strike Eagles 1990 *eingesetzt*?

FreeTranslation: *Where did the military airplane become would strike used Eagles 1990?*

Systran: *Where was the military aircraft Strike Eagle used 1990?*

Logos: *Where was the soldier airplane Strike Eagles installed in 1990?*

BoO_{EN} := {soldier, airplane, strike, eagle, install, 1990, military, become, strike, *use*, aircraft}

2. Query expansion using EuroWordNet

$\forall x \in BoO_{EN}$: lookup(EuroWN);
If x is unambiguous: extend BoO_{EN}
Else $\forall$readings(x):
get its aligned German readings &
Look them up in BoO_{GN}
If successfully then add English terms to BoO_{EN}

~~Reading-697925~~
~~EN: {handle, *use*, wield}~~
~~DE: {handhaben, hantieren}~~

~~Reading-1453934:~~
~~EN: {behave toward, use}~~
~~DE: not aligned~~

Reading-658243:
EN: {apply, employ, make use of, put to *use*, use, utilise, utilize}
DE: {anbringen, anwenden, bedienen, benutzen, *einsetzen*, …}

What we learned ...

- Different MT services can help each other
  - Logos suitable for EN-query parsing
    - Necessary to determine A-type, Q-focus on EN side
  - Systran/FreeTranslation better in NE-translation
- Problem: MT-services often compute
  - Ill-formed strings: bad for query parsing
  - “partial” translation (mixed strings): problem for IR/paragraph selection
- Our envisaged approach
  - Use DE-query analysis as control object for determining EN query object
  - Prefer DE-determined EAT, NE, Q-focus
    - Further decrease role of external MT services; only used for WSD

Even more to learn ...

- Off-line Annotation of corpus would help defining more controlled IR
- Query/Answer processing
  - Question analysis as “deep” as possible
  - Question classification as basis for answer strategy selection
  - Answer strategies for definition/list-based questions
- Had led to substantial improvements of our Clef-2003 system for Clef-2004

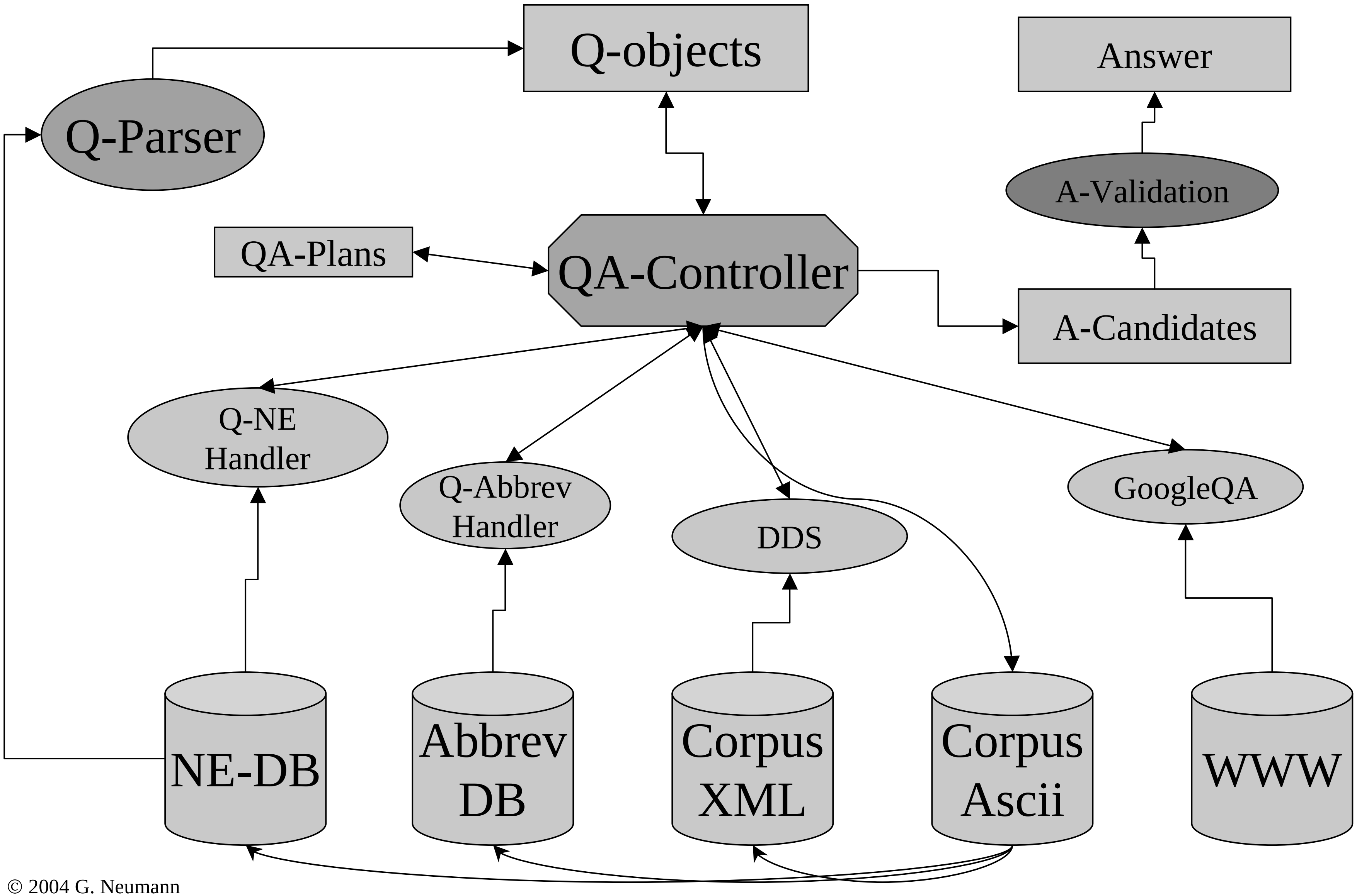
DFKI@CLEF-2004

- We participated in two tasks
  - Cross-lingual German -> English
  - Monolingual German
- Results:
  - DE-EN: 23.5% (23.8%/20%)
    - Best result among 7 groups/13runs/5 languages
  - DE-DE: 25.35 (28.25%/0)
    - Only two participating groups
- Experience from DFKI@CLEF-2003
  - Combination of statistical and symbolic query parser
    - Not of much help
  - Paragraph-based selection of answer sources
    - Too coarse grained
  - Use of MG IR-engine
    - Too inflexible query language

Our Clef-2004 QA system

- Same system in both tasks
- Robust semantic analysis of german queries
  - High coverage for different question types
  - Underspecifed dependency analysis
  - Soft retrieval to ontological information
- Hybrid answer selection strategies
  - Preprocessing of corpus with NE, sentence analysis, tenary relations
- Flexible IR-query term construction

Hybrid Architecture



QA Track Setup – Task Definition

Given **200 questions** in a source language, find **one exact answer** per question in a collection of documents written in a target language, and provide a justification for each retrieved answer (i.e. the **`docid`** of the unique document that supports the answer).

S \ T	DE	EN	ES	FR	IT	NL	PT
BG							
DE							
EN							
ES							
FI							
FR							
IT							
NL							
PT							

6 monolingual and 50 bilingual tasks.

18 Teams participated in 19 tasks, submitting 48 runs.

Next slides from Alessandro Vallin ITC-irst, Trento - Italy

Evaluation Exercise – Results (EN)

Results of the runs with English as target language.

Run Name	R	W	X	U	Overall Accuracy (%)	Accuracy over F (%)	Accuracy over D (%)	NIL Accuracy		CWS
								Precision	Recall	
bgas041bgen	26	168	5	1	13.00	11.67	25.00	0.13	0.40	0.056
dfki041deen	47	151	0	2	**23.50**	23.89	20.00	0.10	0.75	0.177
dltg041fren	38	155	7	0	19.00	17.78	30.00	0.17	0.55	-
dltg042fren	29	164	7	0	14.50	12.78	30.00	0.14	0.45	-
edin041deen	28	166	5	1	14.00	13.33	20.00	0.14	0.35	0.049
edin041fren	33	161	6	0	16.50	17.78	5.00	0.15	0.55	0.056
edin042deen	34	159	7	0	17.00	16.11	25.00	0.14	0.35	0.052
edin042fren	40	153	7	0	**20.00**	20.56	15.00	0.15	0.55	0.058
hels041fien	21	171	1	0	10.88	11.56	5.00	0.10	0.85	0.046
irst041iten	45	146	6	3	**22.50**	22.22	25.00	0.24	0.30	0.121
irst042iten	35	158	5	2	17.50	16.67	25.00	0.24	0.30	0.075
lire041fren	22	172	6	0	11.00	10.00	20.00	0.05	0.05	0.032
lire042fren	39	155	6	0	19.50	20.00	15.00	0.00	0.00	0.075

Evaluation Exercise – Results (Monolingual)

Results of the runs with Italian as target language.

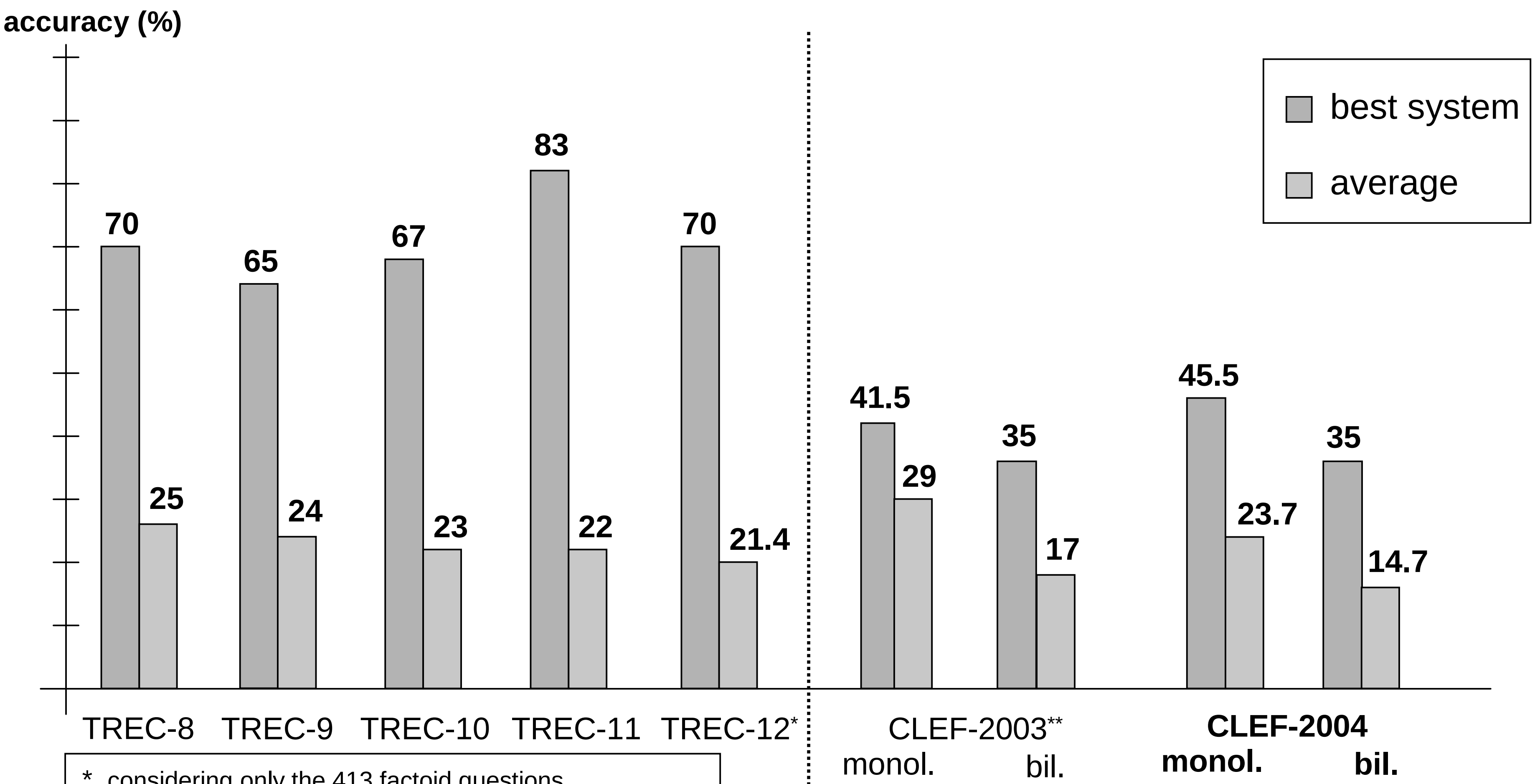
Run Name	R	W	X	U	Overall Accuracy (%)	Accuracy over F (%)	Accuracy over D (%)	NIL Accuracy		CWS
								Precision	Recall	
ILCP-QA-ITIT	51	117	29	3	25.50	22.78	50.00	0.62	0.50	-
irst041itit	56	131	11	2	**28.00**	26.67	40.00	0.27	0.30	0.155
irst042itit	44	147	9	0	22.00	20.00	40.00	0.66	0.20	0.107

Results of the runs with German as target language.

Run Name	R	W	X	U	Overall Accuracy (%)	Accuracy over F (%)	Accuracy over D (%)	NIL Accuracy		CWS
								Precision	Recall	
dfki041dede	50	143	1	3	25.38	28.25	0.00	0.14	0.85	-
FUHA041-dede	67	128	2	0	**34.01**	31.64	55.00	0.14	1.00	0.333

Evaluation Exercise - Results

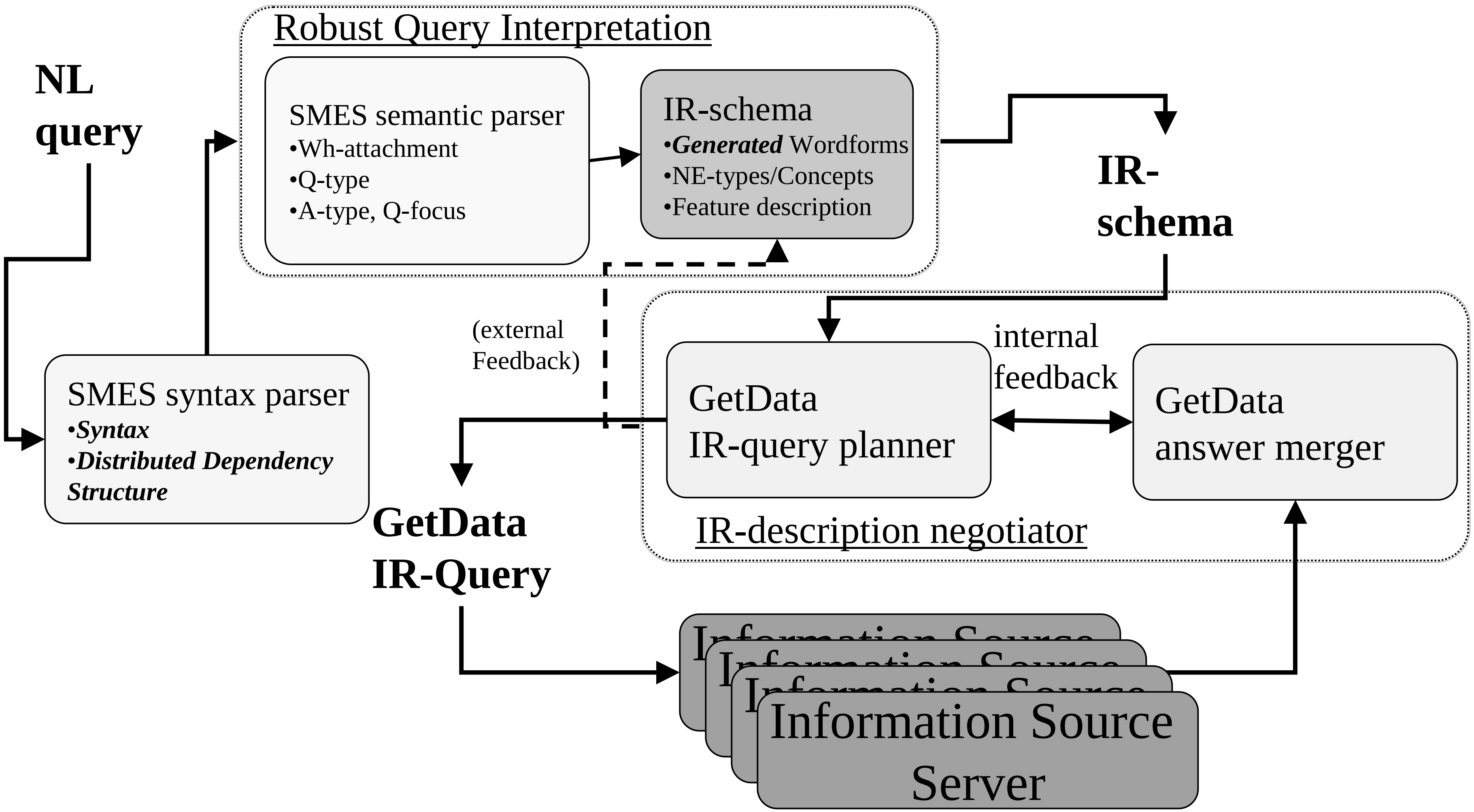
Systems' performance at the TREC and CLEF QA tracks.



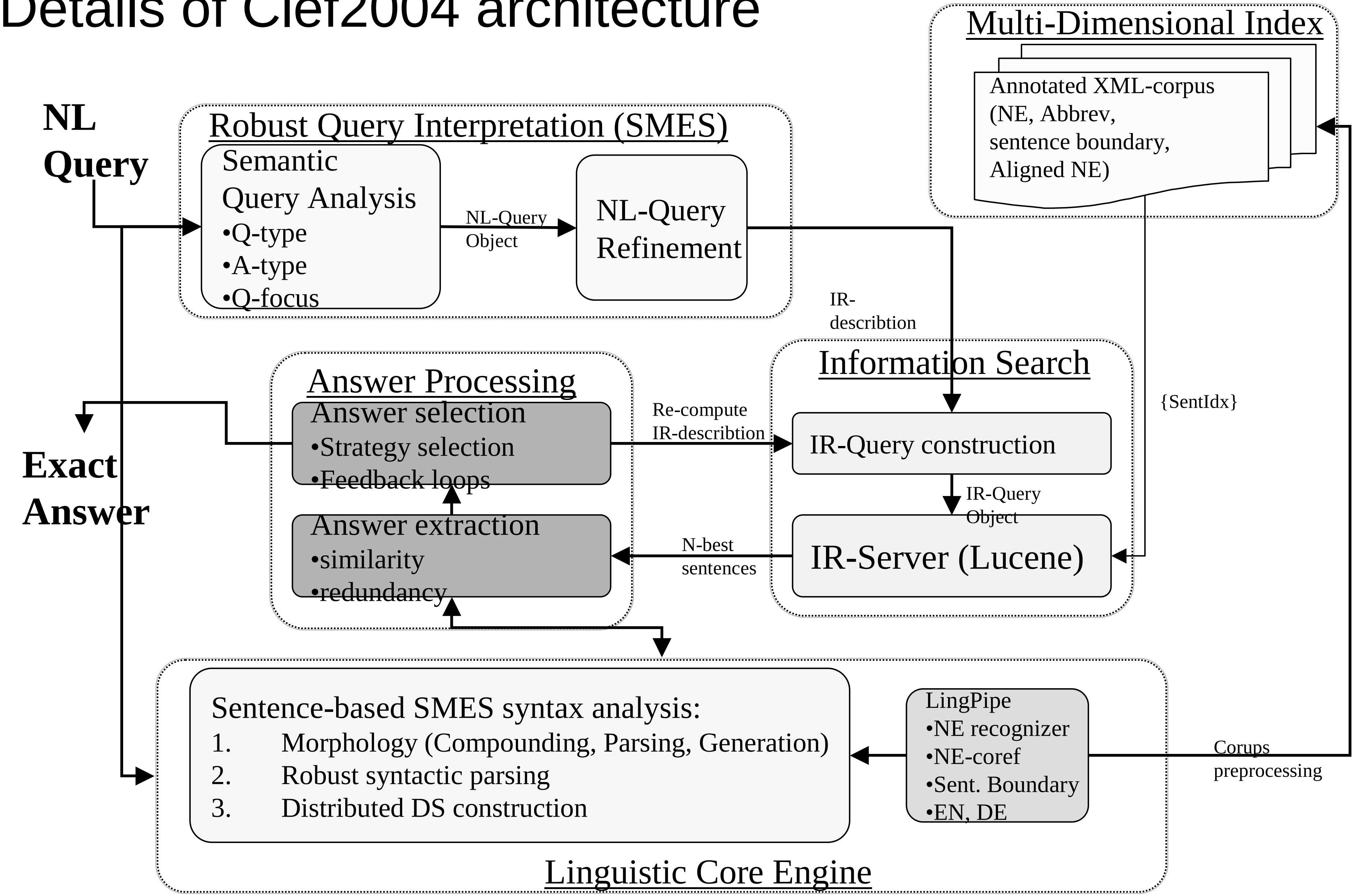
* considering only the 413 factoid questions

** considering only the answers returned at the first rank

SMES QA Interface



Details of Clef2004 architecture



Robust Interpretation of NL Queries

- German syntax (SMES):
  - Topological parsing
  - Local Subgrammars for Wh-phrases
- Re-representation
  - Distributed Representation for Dependency Structure
- Query analysis
  - Major information
    - Q-type (description/definition/…)
    - A-type (Person/Location/…)
    - Scope (further constraints for A-type)
  - Q-type determination using Wh-meta terms
    - “What type of bridge is the Golden Gate Bridge?”
  - Corpus-driven approach for Wh-domain terms
    - “What is the capital of Somalia?”
- Determines control-information for QA-strategy selection

Robust Interpretation of NL Queries in BiQue-2004

- German syntax (SMES):
  - Topological parsing
  - Local Subgrammars for Wh-phrases
- Re-representation
  - Distributed Representation for Dependency Structure
- Query analysis
  - Major information
    - Q-type (description/definition/…)
    - A-type (Person/Location/…)
    - Scope (further constraints for A-type)
  - Q-type determination using Wh-meta terms
    - “What type of bridge is the Golden Gate Bridge?”
  - Corpus-driven approach for Wh-domain terms
    - “What is the capital of Somalia?”
- Determines control-information for QA-controller

Distributed Representation of Dependency Structures

Flat dependency-based structure, only upper bounds for attachment and scoping:

[$_{PN}$Die Siemens GmbH] [$_{V}$hat] [$_{year}$1988][$_{NP}$einen Gewinn] [$_{PP}$von 150 Millionen DM], [$_{Comp}$weil] [$_{NP}$die Aufträge] [$_{PP}$im Vergleich] [$_{PP}$zum Vorjahr] [$_{Card}$um 13%] [$_{V}$gestiegen sind].

“The siemens company has made a revenue of 150 million marks in 1988, since the orders increased by 13% compared to last year.”

<u>BaseObjects</u>

1:[$_{PN}$Die Siemens GmbH]

2:[$_{V}$hat]

3:[$_{year}$1988]

4:[$_{NP}$einen Gewinn]

5:[$_{PP}$von 150 Millionen DM]

6:[$_{Comp}$weil]

7:[$_{NP}$die Auftraege]

8:[$_{PP}$im Vergleich]

9:[$_{PP}$zum Vorjahr]

10:[$_{Card}$um 13%]

11:[$_{V}$gestiegen sind].

<u>LinkObjects</u>

L-1: O:2(O:1,O:3,L-2,L-3)

L-2: O:4(O:5)

L-3: O:6(L-4)

L-4: O:11(O:7,O:8,O:9,O:10)

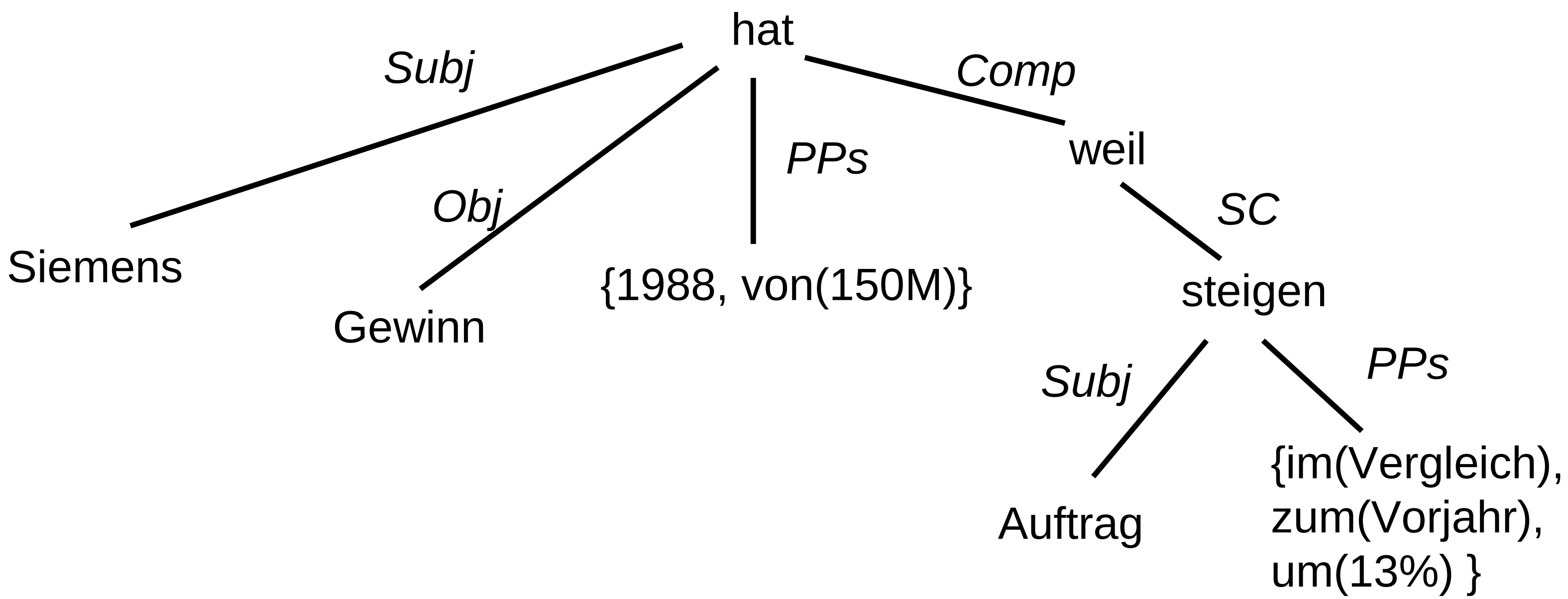
Linguistic and application specific extension are described as operations (typing, re-organisation of attachment) applied on LinkObjects.

Underspecified functional description for sentences

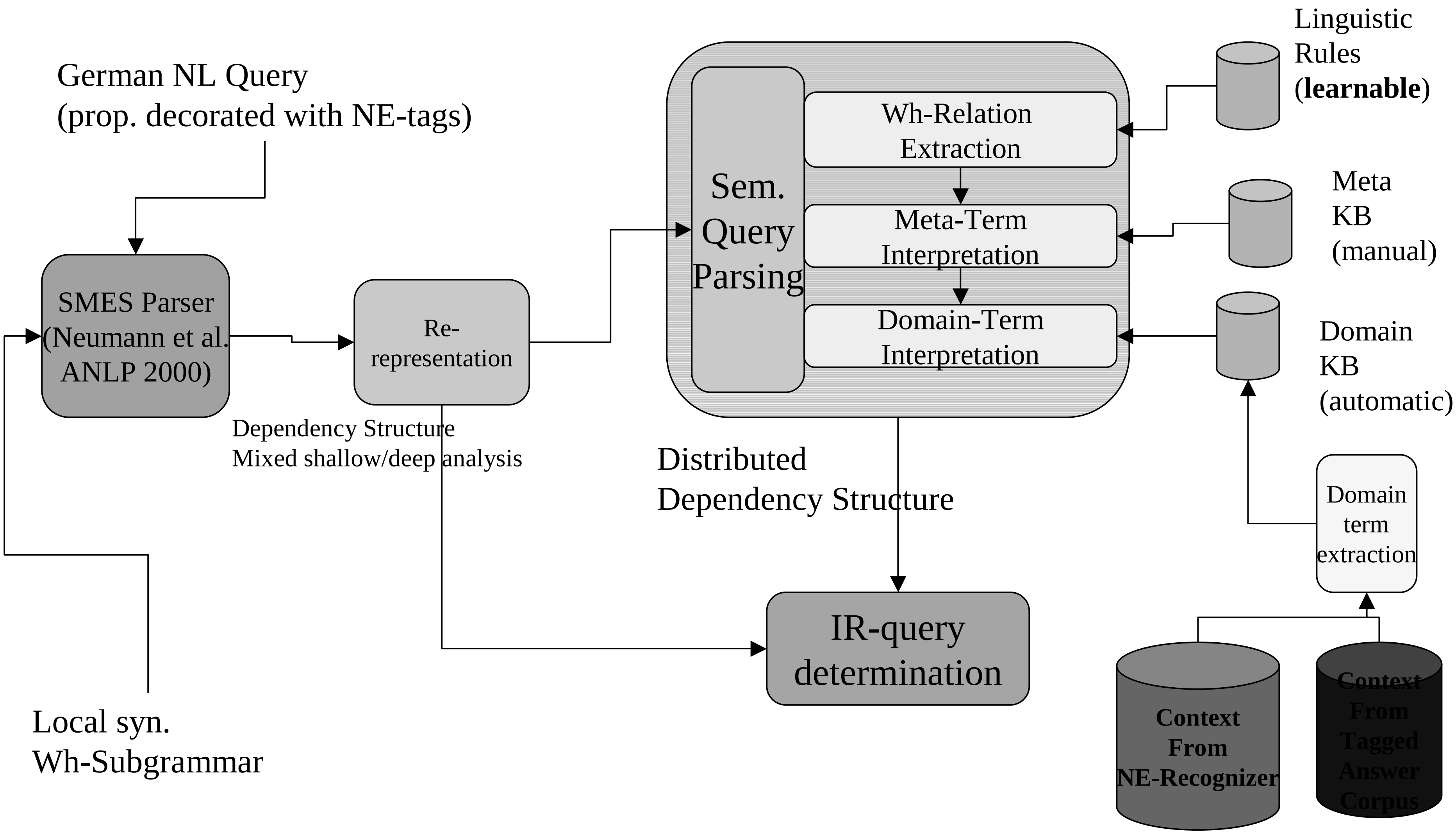
Flat dependency-based structure, only upper bounds for attachment and scoping:

[$_{PN}$Die Siemens GmbH] [$_{V}$hat] [$_{year}$1988][$_{NP}$einen Gewinn] [$_{PP}$von 150 Millionen DM],
[$_{Comp}$weil] [$_{NP}$die Aufträge] [$_{PP}$im Vergleich] [$_{PP}$zum Vorjahr] [$_{Card}$um 13%] [$_{V}$gestiegen sind].

"The siemens company has made a revenue of 150 million marks in 1988, since the orders increased by 13% compared to last year."



Robust Semantic Query Interpretation



Examples (and more)

„Was für eine Art Tier ist der Hund? "

<IOOBJ msg='quest' s-ctr='C-HYPONYM' q-weight='1.0'>

<A-TYPE>ANIMAL</A-TYPE>

<SCOPE>hund</SCOPE>

…

“In welcher Stadt lebte Picasso?"

<IOOBJ msg='quest' s-ctr='C-DESCRIPTION' q weight='1.0'>

<A-TYPE>LOCATION</A-TYPE>

<SCOPE>stadt</SCOPE>

…

Query Translation

- Same approach as in our Clef-2003 system
- Core idea:
  - For German string X, call N-MT system to produce N english strings Y_i
  - Compute internal query objects for Y and compute union
  - Call linguistic ontologies to perform query expansion
- Extensions for Clef 2004 system
  - 6 MT systems instead of 3
  - No use of query expansion
  - Word-level alignment for mapping scopus information
  - Heuristics for coping with NE-translation problems (PERSON names)

Determination of Lucene query

- Task: Compute IR-query from NL-query
- Goal:
  - Use different style of query expression for different analysis of dependency structure
  - Use analysis-controlled NL-generation of query terms
  - Perform feedback loop from most specific to most relaxed syntactic expression
- Our approach:
  - From NL-query compute internal IR-independent representation which also covers control information
  - Map this to specific search engine (Lucene, Google, MSN, etc.)

Example

NL-Query

"Wie viele Analphabeten gibt es auf der Welt?"

Robust Query Interpretation

NL-generated
Word forms

Query-DS

Computation of IR-description

```
(((:OR . :V) "gaben" "gab" "gegeben" "geben" "gibt")
  ((:WORD :N :NEC 4) . "analphabet")
  ((:WORD . :N) . "es")
  ((:WORD . :N) . "welt"))
```

Strict Lucene-string

"(gaben OR gab OR gegeben OR geben OR gibt)
AND analphabet^4 AND es^1 AND welt^1 "

Relaxed Lucene-string

"(gaben OR gab OR gegeben OR geben OR gibt) OR +analphabet^4 OR es^1 OR welt^1 "

Important Issues

- High coverage
  - Factoid, definition, list questions
- Soft retrieval for
  - Meta terms & Domain terms
  - Distinguishes:
    - full-match, compound-match, suffix-match
- Explicitly taking into account compounding

" Zu welcher Tierart gehört der Hund? "

<IOOBJ msg='quest' s-ctr='C-HYPONYM' q-weight='1.0'>

<A-TYPE>ANIMAL</A-TYPE>

<SCOPE>hund</SCOPE>

…

Processing of Definition Questions (IE-perspective of QA)

- Query analysis yields:
  - Definition + focus + type of focus
- Core idea:
  - Assume focus-type specific definition of templates
    - Person: born-where, born-when, business-what
  - Compute a set of slot-oriented IR-descriptions
    - These serve as answer patterns
  - Slots are
    - possible known NE (person, location, date, …) which function as a-types
    - NL-phrases “describing” slot, if no TYPE can be deduced
- Compute for each slot one (multiple) Lucene-query term of kind:
  - NE-type:person & text:<query term>

Example

„Wer ist Thomas Mann?“

Q-type=c-definiton, focus=<Person, „Thomas Mann“>

IR-meta term/pattern: <FOCUS> geboren in <LOCATION>

"(neTypes:LOCATION AND +geboren
(text:\"Thomas Mann\" OR text:Mann))"

Problem

<sent>Der <ENAMEX id="3" type="DATE">1908</ENAMEX> in <ENAMEX id="0" type="LOCATION">München</ENAMEX> geborene Schriftsteller und Journalist war ein Vertrauter des Literarturnobelpreisträgers <ENAMEX id="4" type="PERSON">Thomas</ENAMEX> Mann und ein enger Freund von dessen Familie.</sent>

Therefore:

need for deeper NL analysis on document side as well as knowledge reasoning

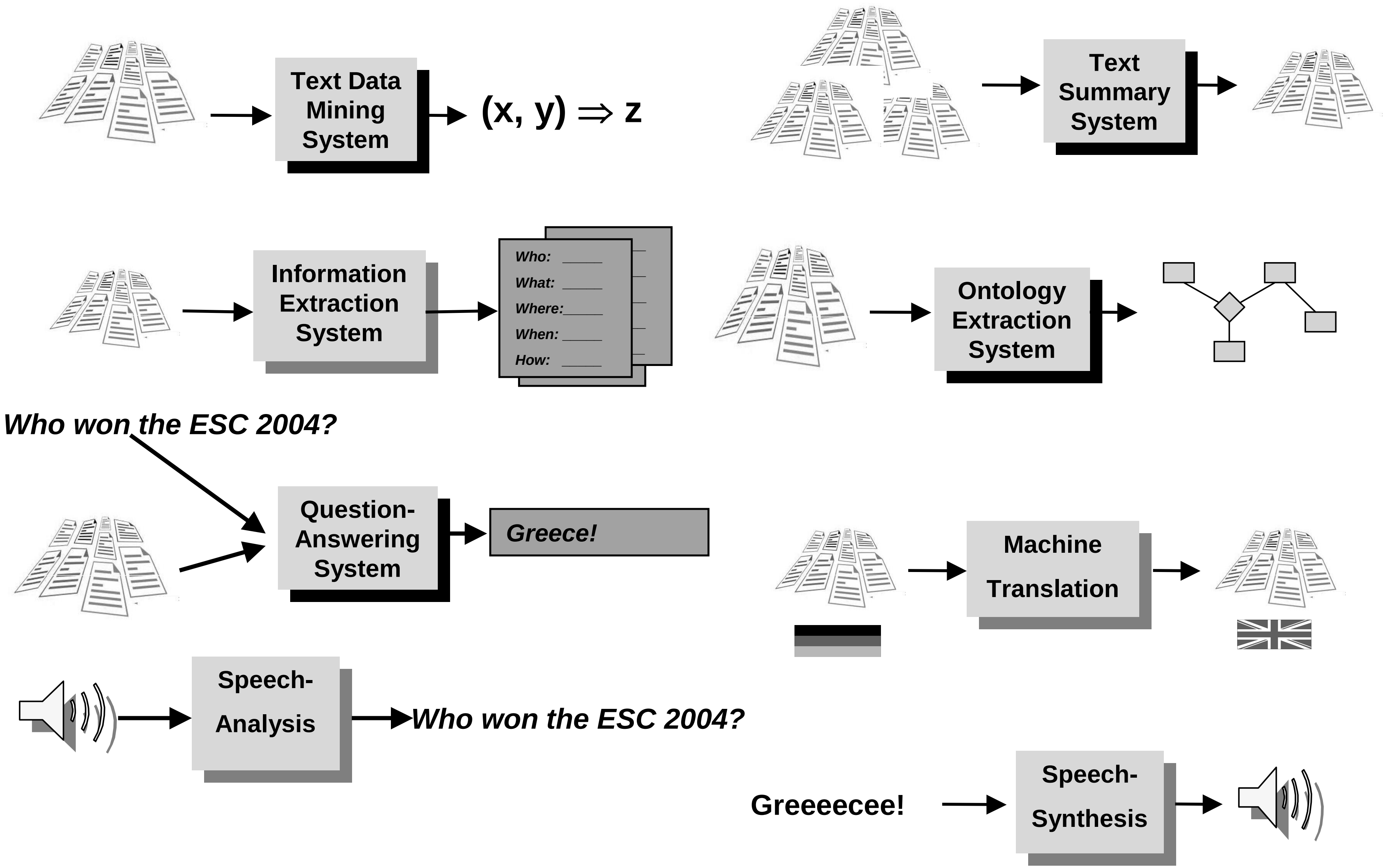
Language Technology and Semantic Web

A kind of course summary and
“future work”

Human Language Technology

- *Human Language Technology LT* – covers
  - The design and implementation of algorithms, data and electronic devices for processing of natural language (text and speech), and
  - Their integration into real-world applications and products
- Language Technology defines the engineering part of computational linguistic

LT-methods cover many areas



Multi/cross-linguality is of great importance in all these areas!

LT as embedded part of applications

- Human-Machine Communication
- Data-oriented Knowledge Acquisition

Integration

- Modularity
- Multi-media
- Software-Engineering standards

High Performance

- Real-time
- Robustness
- Scalability
- Adaptation
- Evaluation

Language Technology

- **Core technology**
  - Efficient data structures
  - Weighted finite state automata
  - Machine learning
  - Statistical inference

- **LT-Methods**
  - Named Entity-Recognition
  - PoS/Sem-Tagging
  - Controlled Languages
  - Integration of shallow & deep NLP („text zooming“)
  - Reference-resolution
  - NL-oriented ontologies

- Already a successful technology transfer
  - Industry (Microsoft, IBM, Siemens, Telekom, ...) & Spin-offs, competence centers, ...
  - Speech-systems, MT, Editors, Text-Mining, Knowledge-Mining Content-Management, ...

- Newest Technology Hype: the Semantic Web
  - What role does it play for LT?

The Semantic Web (SW)

- Tim Berners-Lee, 1998:
  - “This document is a plan for achieving a set of connected applications for data on the Web in such a way as to form a consistent logical web of data (semantic web).”

- Tim Berners-Lee et al., 2001
  - “… an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.”

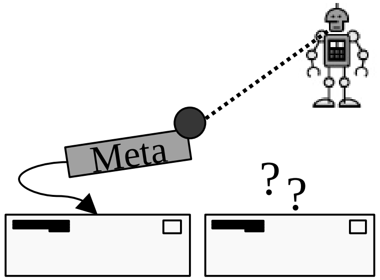
SW – illustrated

1 Extension of the Current Web

The existing web will further emerge, so that computers can understand content on-line, to better help humans to organize, search, and exchange information.

2 Add meta-data

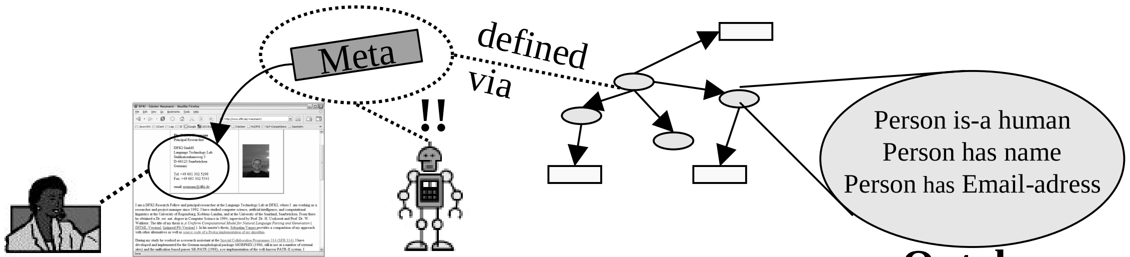
Data over data; Structural linkage of heterogeneous data sources



3 Ontologies associate meaning to meta-data

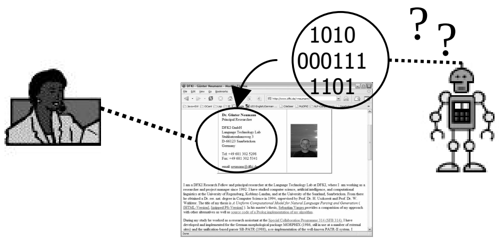
SW exists of meta-data and links to global ontologies, which define the meaning of terms.

An ontology serves as a structural vocabulary for the interpretation of domain-specific terms.

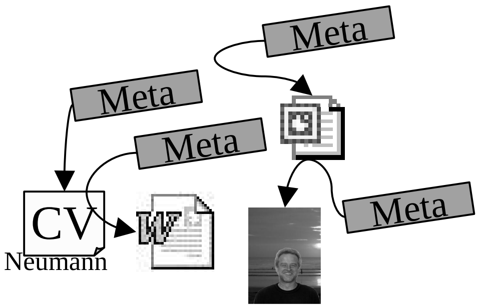


Ontology

4 Structured Web of data



5 The SW does not only consider Web-pages



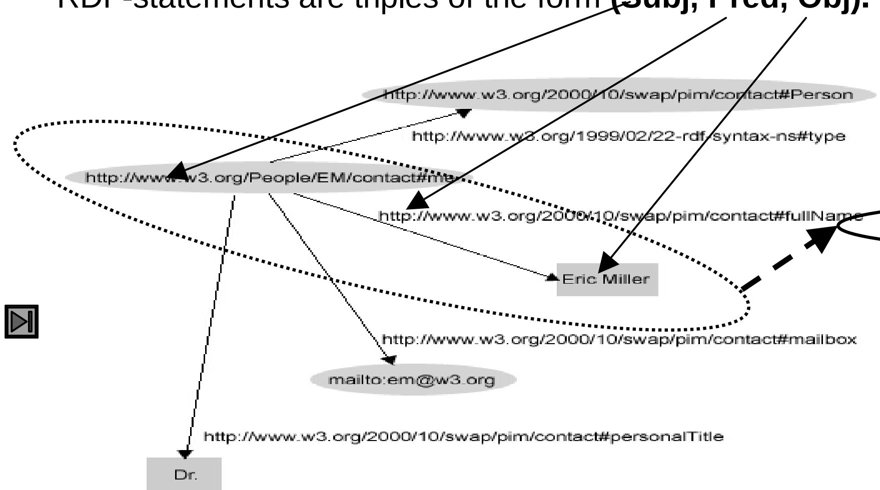
6 How will I use the SW?

- Intelligent information search;
- Automatic support for the management of my personal information on the SW

RDF and OWL: Modeling data on the SW

1 RDF: Resource Description Framework

RDF is language for the representation of meta-data over web resources.
RDF-statements are triples of the form **(Subj, Pred, Obj).**



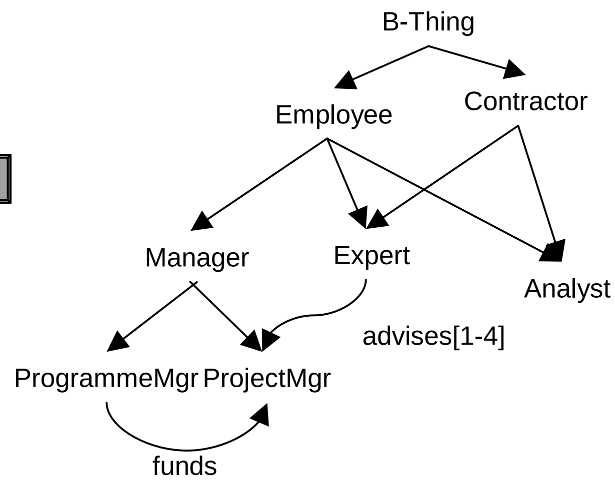
2 XML & N3 sind alternative RDF-Syntaxen

XML schematically: <Subj> <Pred> Obj </Pred> </Subj>

N3:
@prefix rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns# .
@prefix contact: http://www.w3.org/2000/10/swap/pim/contact# .
@prefix EM: http://www.w3.org/People/EM/contact# .

EM:me rdf:type contact:person .
EM:me contact:full-name "Eric Miller" .
EM:me contact:personalTitle "Dr." .
EM:me contact:mailbox rdf:resoure "mailto:em@w3.org .

3 OWL: Web Ontology Language



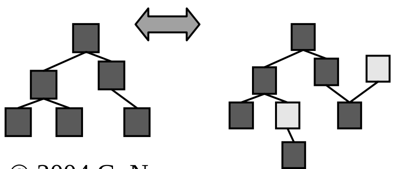
•some RDF-statements have a fix interpretation (is-a, =, inverseOf, card, ...)

•**Sharing** of information between individuals from multiple documents ⇒ Web of data from heterogeneous sources
•Semantic of OWL as basis for inference mechanism over these data structures.

4 Relevant aspects for SW

standardization, Web-globalization, distribution of resources

5 Ontology Mapping



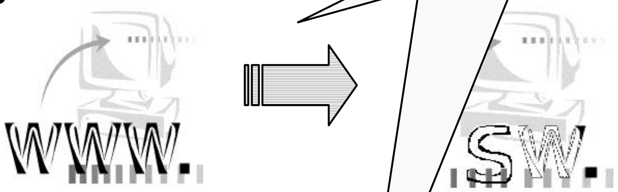
Mapping between distributed, local ontologies

Relevance of LT for SW

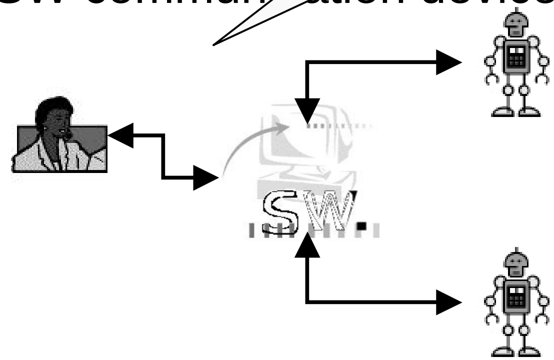
Intelligent Information Extraction

Intelligent Information Access

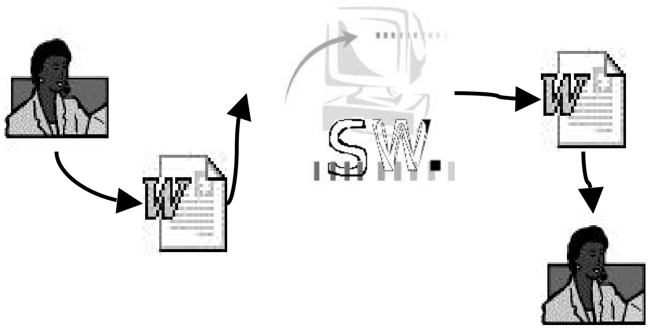
1 During the transi
WWW to the SW, L
technology



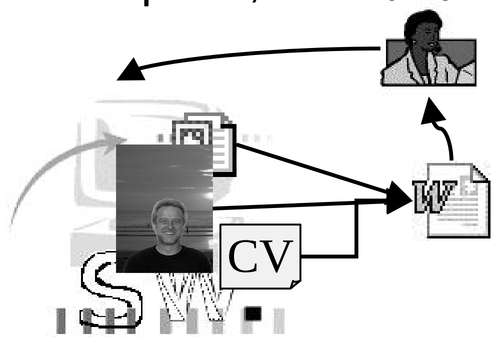
2 As long as the hum … et
Loop", NL will remain to … core
Human-SW communication device.



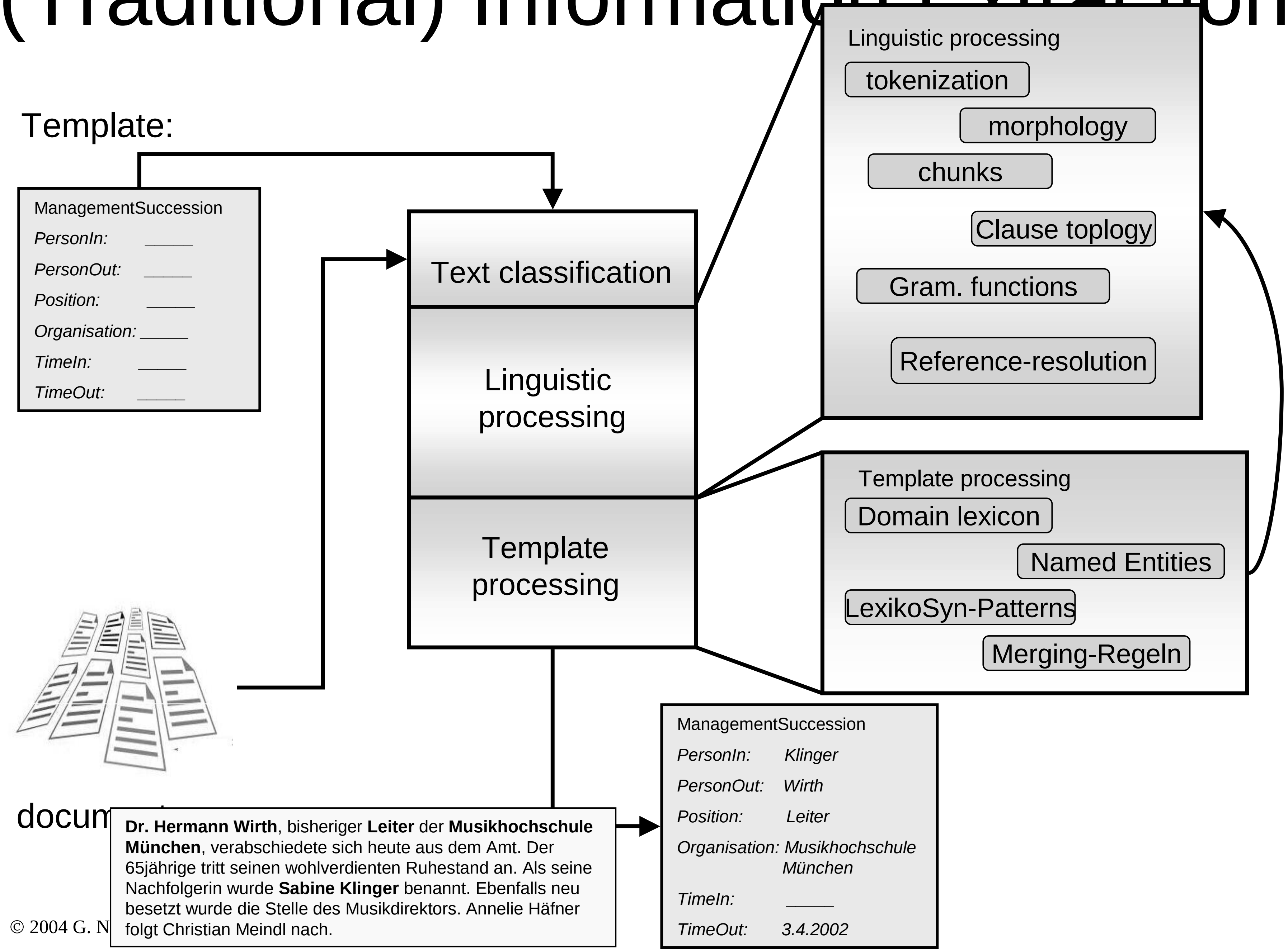
3 Humans will also in the future exchange knowledge via NL documents: Semantically annotated documents as Human-SW interface



4 NL-generation of information in form of NL-Text, e.g., heterogeneous resources, dynamically created reports, newspapers, …



(Traditional) Information Extraction



IE for semantic annotation

Identification of IE-sub-tasks:
• basic entities (e.g., proper names)
• binary relations between entities
• n-ary relations/events

→ **Machine learning!**

Automatic Content Extraction (ACE)

• Spezification of an IE-core-ontology
• Annotation-specification & -tools
• Templates as specializations of the IE-core-ontology (also multi-templates)

IE as core for semantic annotation
• identification
• discovery
• validation
• evaluation
of semantic relationships & as basis for the automatic creation of meta data

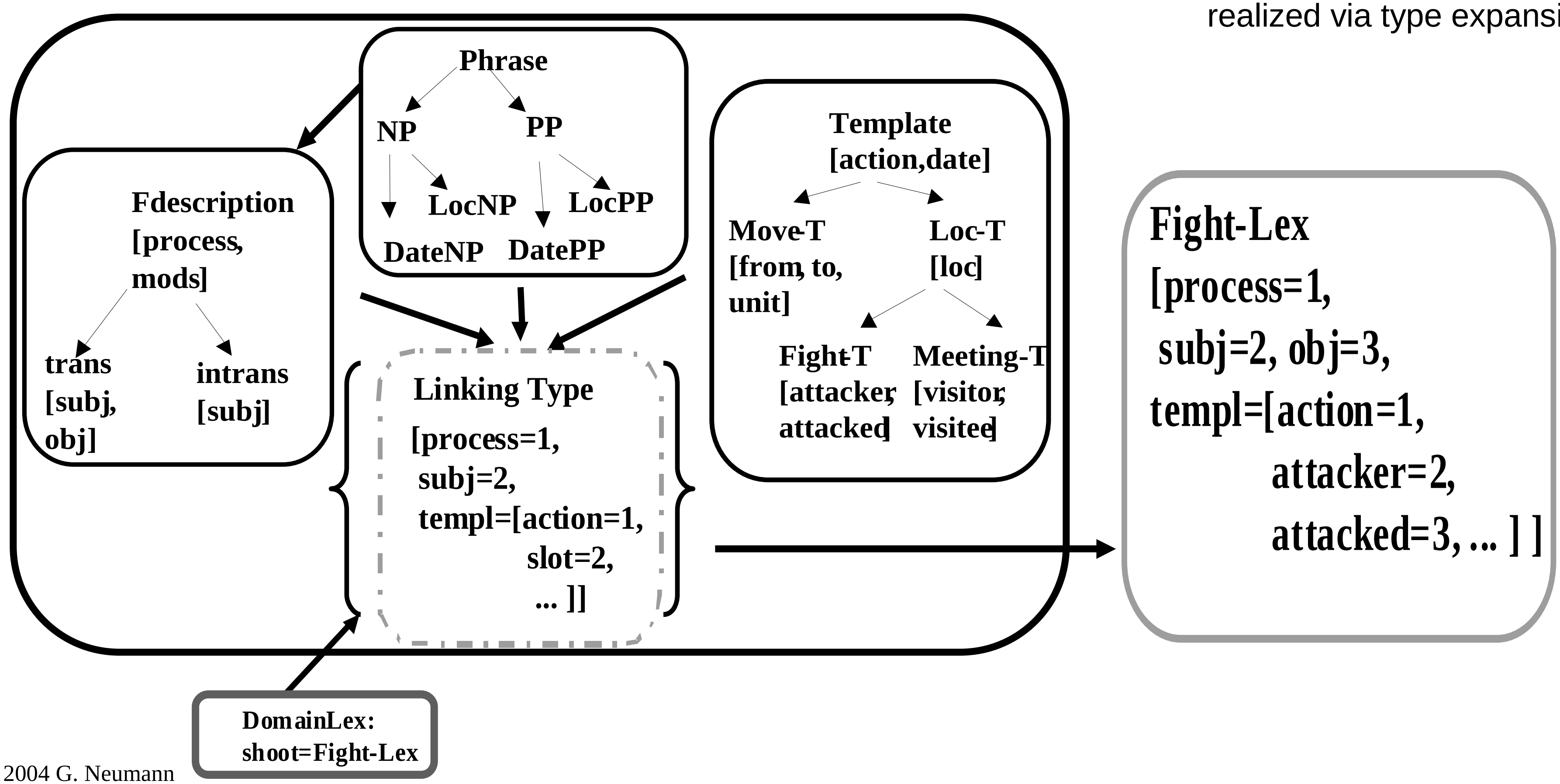
LT-challenges

Identification of verbalizations/mentioning of concepts/instances

- Linking of domain ontology and NL-oriented ontology (e.g., WordNet)
- Paraphrasing
- Metonymy   (“Peking organizes the Olympic Games 2008.”)
- Reference identification (“Chancellor Schröder, Schröder, the German chancellor, he, …”)
- Analysis of sublanguages as basis for adaptive IE (cf. Grishman, 2001)

Domain modeling in DFKI system SMES is realised using typed feature structures

- Domain modeling via hierarchy of templates (black box), using the formalism TDL, which is also used to model hierarchies of linguistic objects (yellow boxes).
- The interface between domain knowledge and linguistic entities is specified via *linking types* (green box), which represent a close connection between concepts of the different layers, and which are accessible via the domain lexicon (brown & green box). Template-filling is then realized via type expansion.



NL-annotations for the SW

Starting point: START multi-media QA system, by Boris Katz et al

T-expression
<subject relation object>

Central issues

1. Sentence-based NL-Analysis
2. NL-annotations for multi-media information segments

Bill surprised Hillary with his answer

<<Bill surprise Hillary> with answer>
<answer related-to Bill>

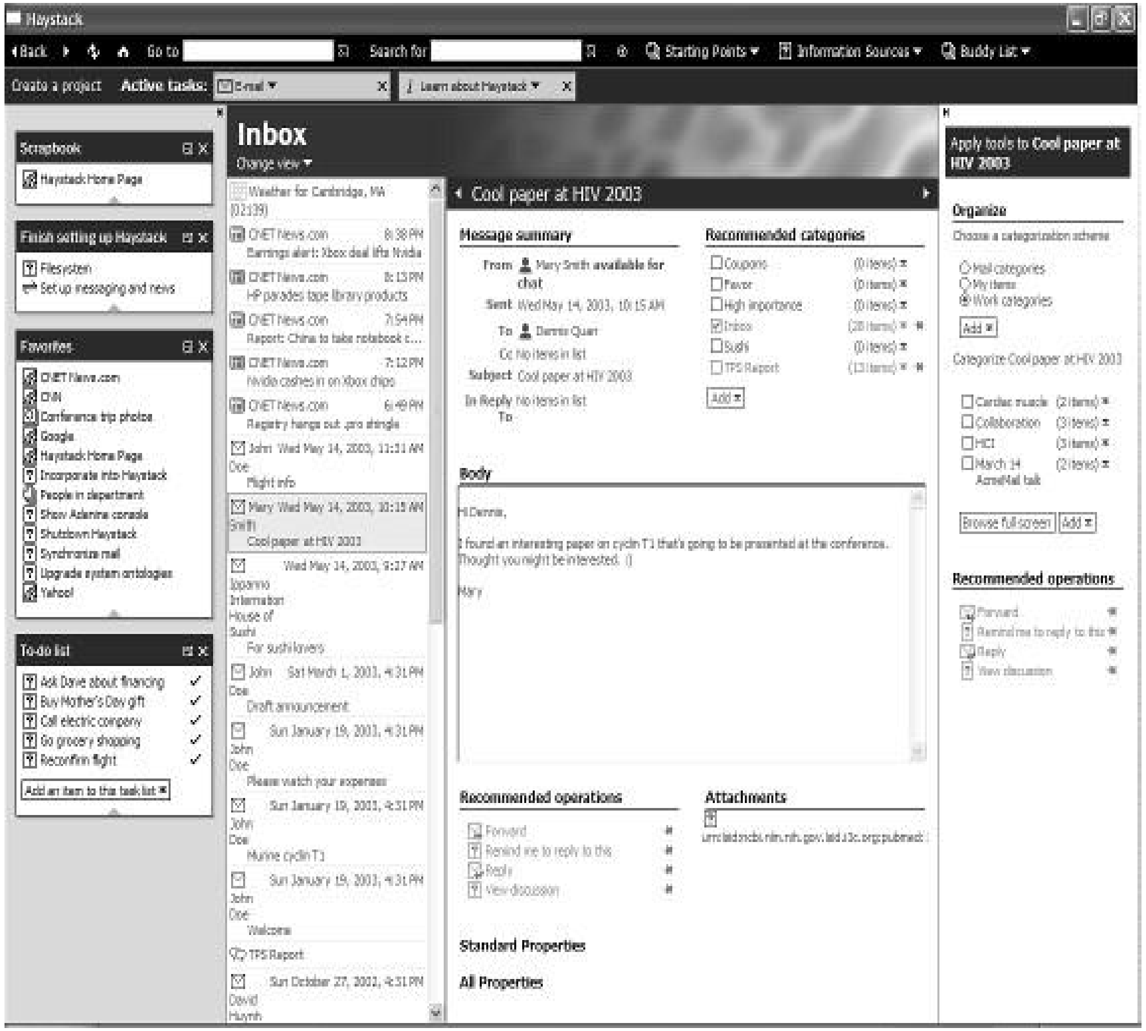
Processing of huge text collections:

1. Extraction of relevant sentences from texts.
2. Syntax analysis
3. Annotation of the texts with syntax

NL-Question
Whose answer surprised Hillary?
<answer surprise Hillary>
<answer related-to ***whom***>

Haystack: the universal information client

http://haystack.lcs.mit.edu/



Motivation:
semantic annotation should be a side-effect of daily use of computer.

Idea:
Personalized information portal for all relevant services, like email, documents, calender, Web-pages, ...

Collection of all data uniformly via RDF-database

Programming language Adenine for the manipulation of frequent (i.e., as support for the implementation of specific service programs).

Haystack RDF-database:

```
@prefix dc: http:77purl.org/dc/elements/1.1/
@prefix :  http://www.50states.com/data#

{ :State
        rdf:type     rdfs:Class ;
        rdfs:label  „State“
}
{ :bird
        rdf:type       rdf:Property ;
        rdfs:label     „State bird“  ;
        rdfs:domain :State
}
{ :alabama
        rdf:type :State ;
        dc:title „Alabama“ ;
        :bird „Yellowhammer“ ;
        :flower „Camellia“ ;
        :population „4447100“ ;
        ...
}
```

Natural language schema:

```
@prefix nl: http://www.ai.mit.edu/projects/infolab/start#

Add{ :stateAttribute
          rdf:type         nl:NaturalLanguageSchema ;
          nl:annotation @( :attribute „of“ :state) ;
          nl:code          :stateAttributeCode
}
Add{ :attribute
          rdf:type          nl:Parameter ;
          nl:domain        rdf:Property ;
          nl:descrProp    rdf:label ;
}
Add{ :state
          rdf:type          :Parameter ;
          nl:domain       :State ;
          nl:descrProp   dc:title;
}

Method
:stateAttributeCode : state=state :attribute=attribute
          return (ask {state  attribute ?x })
```

Ask{state=:alabama, attribute=:bird, ?x }

:bird ⇐ :attribute=„state bird“
:alabama ⇐ :state=„Alabama“

⇒ ?x= „Yellowhammer“

Antwort: ***Yellowhammer***

Frage: ***What is the state bird of Alabama?***

Example: Linking of t-expressions & RDF

```
@prefix nl: http://www.ai.mit.edu/projects/infolab/start#

Add{ :Person
        rdf:type        rdfs:Class ;
}

Add{ :homeAddress
        rdf:type        rdf:Property ;
        rdfs:domain     :Person ;

        nl:annotation  @(nl:subj „lives at“ nl:obj) ;
        nl:annotation  @(nl:subj „‘s home adress is“ nl:obj) ;
        nl:annotation  @(nl:subj „‘s apartment“ nl:obj) ;

        nl:generation  @(nl:subj „‘s home address is“ nl:obj) ;

}
```

Remarks:

- NL-annotations as a means for controlling the paraphrasing potential of NL expressions
- Richer linguistic annotations are possible (e.g., fine-grained grammatical functions, agreement)
- Also relevant for user-oriented adaptation of service programs

Natural language annotations for the SW

- NL used as meta-data
  - Readability of RDF
  - Supports transition from WWW to SW
  - NL-annotation specifies which kind of (NL)-question a meta-data is able to answer
    $\Rightarrow$ controlled question-answering systems

- Information access (IA) within SW
  - Development of programs, which help a user to locate, to collect, to compare and to link information

- NL is the most natural way for user to perform IA
  - SW should support in the same way IA using specialized languages/exchange formats & NL

Relevance

- Approach is open for future extensions:
  - statistical-based models (add weight to the NL-annotations)
  - Machine Learning of NL-annotations on basis of ontology-oriented IE (cf. Hovy et al. 2002)
- The current mechanism of NL-annotations is idiosyncratic, however at DFKI we plan the following:
  - Exploration of a linking mechanism between dependency structure and RDF/OWL
  - Foundation for novel template-based QA-strategies

Concluding remarks

- LT is a key technology for the construction of the Semantic Web
- Very high requirements on
  - Performance
  - Modularity & integration
  - scalability & on-demand availability
  - Domain & user adaptation
- Systematic evaluation of LT-methods
  - Driving power & revisions of future developments
- In the future, cognitive-based methods will be considered
  - as inspiration for more effective LT-methods, e.g., deterministic parsing/generation, intelligent memory management