



# Generic chaining of LRT webservices using profile matching based on ISOcat data categories

---

Volker Boehlke

University of Leipzig

BoehlkeV@informatik.uni-leipzig.de

NEER109

Helsinki

2009-10-02

# Some facts about me

---



- iba Consulting Gesellschaft (programmer & consultant)
  - reporting software for energy trade
  - real estate asset management
- University of Leipzig
  - member of the scientific staff of the NLP group
  - DSpin/Clarin project (WS, Infrastructure)

UNIVERSITÄT LEIPZIG

Institut für Informatik



Automatische Sprachverarbeitung

# matchmaking I

---



- we are searching „perfect matches“
- mathematics/programming:  $c(b(a()))$
- basically the same problem to be solved like on the type checking level of a compile run: Check if all input needs of a certain function are satisfied: correct number and type of parameters (format is given by programming language).
- What are our “parameters” (NLP-specific):
  - information is encoded using a certain format (TEI, ...)
  - a certain concept/kind of information is present inside of a valid document (tokens, tags, ...) => parameter
  - this concept/information is encoded using a certain datatype (utf8, tagset A, ...)

# services I – binary service



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry**
- manage formats
- chain services
- direct call
- log

Id	Name	Description
47	TEST TCF0.2 converter (deutsch)	
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)	test
49	TEST Tagger-Stuttgart 0.2 (deutsch)	test
50	TEST Tokenizer Leipzig (deutsch)	test
53	TEST Query Wortschatz	test
58	Tokenizer (IMS,TCF0.2,deutsch)	Tokenizer for German text.

Id:  Name:

ShortDescription:

Url:

Description:

Creator:

Contact:

InputWrapper:  OutputWrapper:

Name	Standard

Name	Standard
Text	Utf8
Language	German

Password:

# services II – tokenizer service



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry**
- manage formats
- chain services
- direct call
- log

Id	Name	Description
47	TEST TCF0.2 converter (deutsch)	
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)	test
49	TEST Tagger-Stuttgart 0.2 (deutsch)	test
50	TEST Tokenizer Leipzig (deutsch)	test
53	TEST Query Wortschatz	test
58	Tokenizer (IMS,TCF0.2,deutsch)	Tokenizer for German text.

Id:  Name:

ShortDescription:

Url:

Description:

Creator:

Contact:

InputWrapper:  OutputWrapper:

Name	Standard
Text	Utf8
Language	German

Name	Standard
Sentence	Utf8
Token	Utf8

add modify delete add modify delete

Password:

# services III – tagger service



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry**
- manage formats
- chain services
- direct call
- log

Id	Name	Description
47	TEST TCF0.2 converter (deutsch)	
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)	test
49	TEST Tagger-Stuttgart 0.2 (deutsch)	test
50	TEST Tokenizer Leipzig (deutsch)	test
53	TEST Query Wortschatz	test
58	Tokenizer (IMS,TCF0.2,deutsch)	Tokenizer for German text.

Id:  Name:

ShortDescription:

Url:

Description:

Creator:

Contact:

InputWrapper:  OutputWrapper:

Name	Standard	Name	Standard
Text	Utf8	POS	STTS
Sentence	Utf8	Lemma	Utf8
Language	German		
Token	Utf8		

Password:

# chaining I



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry
- manage formats
- chain services**
- direct call
- log

Id	Name
----	------

Id	Name
47	TEST TCF0.2 converter (deutsch)
53	TEST Query Wortschatz
94	Plaintext Converter (Sfs,TCF0.2,deutsch)
95	Plaintext Converter (BBAW,TCF0.2,deutsch)
96	Plaintext Converter (Sfs,TCF0.3,deutsch)
97	Plaintext Converter (Sfs,TCF0.3,english)
98	RTF Converter (Sfs,TCF0.3,deutsch)
99	RTF Converter (Sfs,TCF0.3,english)
100	PDF Converter (Sfs,TCF0.3,deutsch)
101	PDF Converter (Sfs,TCF0.3,english)
102	Microsoft Word Converter (Sfs,TCF0.3,deutsch)
103	Microsoft Word Converter (Sfs,TCF0.3,english)
104	Negra Converter (Sfs,TCF0.3,deutsch)

clear chain

execute chain

service	result size	time
---------	-------------	------

save

# chaining II



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry
- manage formats
- chain services
- direct call
- log

Id	Name
47	TEST TCF0.2 converter (deutsch)

Id	Name
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)
50	TEST Tokenizer Leipzig (deutsch)

clear chain

execute chain

service	result size	time
---------	-------------	------

save

# chaining III



ASV DSpin Registry Management Tool v0.01a

add service  
browse registry  
manage formats  
**chain services**  
direct call  
log

Id	Name
47	TEST TCF0.2 converter (deutsch)
49	TEST Tagger-Stuttgart 0.2 (deutsch)
50	TEST Tokenizer Leipzig (deutsch)

Id	Name
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)

clear chain

execute chain

service	result size	time
TEST TCF0.2 converter (deutsch)	6 KB	1293
TEST Tokenizer Leipzig (deutsch)	58 KB	331
TEST Tagger-Stuttgart 0.2 (deutsch)	122 KB	1202

save

# chaining IV



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry
- manage formats
- chain services
- direct call
- log

Date	Url	Direction	Size
Thu Oct 01 08:15:19 CEST 2009	http://gelbaugenpinguin.ims.uni-stuttgart.de/cgi-bin/dspin/newtagger	DOWNLOAD	122 KB
Thu Oct 01 08:15:18 CEST 2009	http://gelbaugenpinguin.ims.uni-stuttgart.de/cgi-bin/dspin/newtagger	UPLOAD	58 KB
Thu Oct 01 08:15:18 CEST 2009	http://aspra18.informatik.uni-leipzig.de:5000/webservices/Tokenizer	DOWNLOAD	58 KB
Thu Oct 01 08:15:18 CEST 2009	http://aspra18.informatik.uni-leipzig.de:5000/webservices/Tokenizer	UPLOAD	6 KB
Thu Oct 01 08:15:18 CEST 2009	http://clarin.sfs.uni-tuebingen.de:8080/convert/dspindoc/from/plain/text/to/textcorpus/de	DOWNLOAD	6 KB
Thu Oct 01 08:15:17 CEST 2009	http://clarin.sfs.uni-tuebingen.de:8080/convert/dspindoc/from/plain/text/to/textcorpus/de	UPLOAD	6 KB
Thu Oct 01 08:15:07 CEST 2009	http://aspra18.informatik.uni-leipzig.de:5000/webservices/RepositoryChain	DOWNLOAD	758 Bytes
Thu Oct 01 08:15:07 CEST 2009	http://aspra18.informatik.uni-leipzig.de:5000/webservices/RepositoryChain	UPLOAD	123 Bytes

tokID="t64">PPER</tag> <tag tokID="t65">APPR</tag> <tag tokID="t66">ART</tag> <tag tokID="t67">NN</tag> <tag tokID="t68">VVINF</tag> <tag tokID="t69">VMFIN</tag> <tag tokID="t70">\$</tag> <tag tokID="t71">ADJD</tag> <tag tokID="t72">VAFIN</tag> <tag tokID="t73">NN</tag> <tag tokID="t74">VFIN</tag> <tag tokID="t75">\$</tag> <tag tokID="t76">VFIN</tag> <tag tokID="t77">PPER</tag> <tag tokID="t78">ADJD</tag> <tag tokID="t79">PPOSAT</tag> <tag tokID="t80">NN</tag> <tag tokID="t81">NN</tag> <tag tokID="t82">NE</tag> <tag tokID="t83">\$</tag> <tag tokID="t84">ART</tag> <tag tokID="t85">NN</tag> <tag tokID="t86">VAFIN</tag> <tag tokID="t87">ADJD</tag> <tag tokID="t88">KON</tag> <tag tokID="t89">VFIN</tag> <tag tokID="t90">PPOSAT</tag> <tag tokID="t91">NN</tag> <tag tokID="t92">ART</tag> <tag tokID="t93">ADJA</tag> <tag tokID="t94">NN</tag> <tag tokID="t95">VVIZU</tag> <tag tokID="t96">\$</tag> <tag tokID="t97">NN</tag> <tag tokID="t98">VAFIN</tag> <tag tokID="t99">ART</tag> <tag tokID="t100">NN</tag> <tag tokID="t101">NE</tag> <tag tokID="t102">APPR</tag> <tag tokID="t103">ART</tag> <tag tokID="t104">NN</tag> <tag tokID="t105">\$</tag> <tag tokID="t106">NN</tag> <tag tokID="t107">NN</tag> <tag tokID="t108">\$</tag> <tag tokID="t109">PRELS</tag> <tag tokID="t110">NE</tag> <tag tokID="t111">VVINF</tag> <tag tokID="t112">VFIN</tag> <tag tokID="t113">\$</tag> <tag tokID="t114">VFIN</tag> <tag tokID="t115">PPER</tag> <tag tokID="t116">\$</tag> <tag tokID="t117">NE</tag> <tag tokID="t118">VFIN</tag> <tag tokID="t119">ADV</tag> <tag tokID="t120">ART</tag> <tag tokID="t121">NN</tag> <tag tokID="t122">\$</tag> <tag tokID="t123">KOKOM</tag> <tag tokID="t124">ADV</tag> <tag tokID="t125">VFIN</tag> <tag tokID="t126">PRF</tag> <tag tokID="t127">ART</tag> <tag tokID="t128">ADJA</tag> <tag tokID="t129">NN</tag> <tag tokID="t130">NE</tag> <tag tokID="t131">APPRART</tag> <tag tokID="t132">NN</tag> <tag tokID="t133">APPR</tag> <tag tokID="t134">ART</tag> <tag tokID="t135">NN</tag> <tag tokID="t136">APPR</tag> <tag tokID="t137">\$</tag> <tag tokID="t138">NN</tag> <tag tokID="t139">VFIN</tag> <tag tokID="t140">ADJA</tag> <tag tokID="t141">NN</tag> <tag tokID="t142">VFIN</tag> <tag tokID="t143">APPR</tag> <tag tokID="t144">ART</tag> <tag tokID="t145">ADJA</tag> <tag tokID="t146">NN</tag> <tag tokID="t147">ADJD</tag> <tag tokID="t148">\$</tag> <tag tokID="t149">FM</tag> <tag tokID="t150">NN</tag> <tag tokID="t151">\$</tag> <tag tokID="t152">ADV</tag> <tag tokID="t153">CARD</tag> <tag tokID="t154">\$</tag> <tag tokID="t155">APPR</tag> <tag tokID="t156">CARD</tag> <tag tokID="t157">NN</tag> <tag tokID="t158">ADJD</tag> <tag tokID="t159">VAFIN</tag> <tag tokID="t160">APPR</tag> <tag tokID="t161">ART</tag> <tag tokID="t162">NN</tag> <tag tokID="t163">ADV</tag> <tag tokID="t164">ADJD</tag> <tag tokID="t165">APPR</tag> <tag tokID="t166">ART</tag> <tag tokID="t167">NN</tag> <tag tokID="t168">APPR</tag> <tag tokID="t169">NN</tag> <tag tokID="t170">VVPP</tag> <tag tokID="t171">\$</tag> <tag tokID="t172">ADJD</tag> <tag tokID="t173">APPR</tag> <tag tokID="t174">ART</tag> <tag tokID="t175">NN</tag> <tag tokID="t176">NN</tag> <tag tokID="t177">NN</tag> <tag tokID="t178">VAFIN</tag> <tag tokID="t179">ART</tag> <tag tokID="t180">NN</tag> <tag tokID="t181">ADV</tag> <tag tokID="t182">ADJD</tag> <tag tokID="t183">\$</tag> <tag tokID="t184">APPR</tag> <tag tokID="t185">ART</tag> <tag tokID="t186">ADJA</tag> <tag tokID="t187">NN</tag> <tag tokID="t188">ART</tag> <tag tokID="t189">ADJA</tag> <tag tokID="t190">NN</tag> <tag tokID="t191">PTKZU</tag> <tag tokID="t192">VVINF</tag> <tag tokID="t193">\$</tag> <tag tokID="t194">ART</tag> <tag tokID="t195">NN</tag> <tag tokID="t196">VMFIN</tag> <tag tokID="t197">ADV</tag> <tag tokID="t198">ADJD</tag> <tag tokID="t199">VVPP</tag> <tag tokID="t200">VAINF</tag> <tag

# matchmaking II

---



- A (simple) service interface may look like this:
- Input:
  - Format: TEI
  - parameters and datatypes: sentences/utf8, tokens/utf8
- Output:
  - Format: TEI
  - parameters and datatypes: sentences/utf8, tokens/utf8, pos-tags/STTS
- All information - format, each parameter and each datatype - needs to be expressed using unique id's => ISOcat datacategories
- There needs to be a specification which parameters are valid in which format and which datatypes may be used for a parameter.

# matchmaking III



ASV DSpin Registry Management Tool v0.01a

add service  
browse registry  
manage formats  
chain services  
direct call  
log

Id	Name	Description
47	TEST Tokenizer-Converter (deutsch)	
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)	test
49	TEST Tagger-Stuttgart 0.2 (deutsch)	test
50	TEST Tokenizer Leipzig (deutsch)	test
53	TEST Query Wortschatz	test
58	Tokenizer (IMS,TCF0.2,deutsch)	Tokenizer for German text.
61	POS Tagger (IMS,TCF0.2,deutsch)	Generate POS tags for tokenized German text.

Id: 50 Name: TEST Tokenizer Leipzig (deutsch)

ShortDescription:

Url: http://aspra18.informatik.uni-leipzig.de:5000/webservices/Tokenizer

Description: test

Creator: test

Contact: test

InputWrapper: DSpinTextCorpus OutputWrapper: DSpinTextCorpus

Name	Standard
Text	Utf8
Language	German

Name	Standard
Sentence	Utf8
Token	Utf8

add modify delete

Frequency Integer

TokenBoundaries modify delete

Text delete service

Token

Language

Password: mod

# matchmaking IV



ASV DSpin Registry Management Tool v0.01a

- add service
- browse registry**
- manage formats
- chain services
- direct call
- log

Id	Name	Description
48	TEST Tokenizer-Stuttgart 0.2 (deutsch)	test
49	TEST Tagger-Stuttgart 0.2 (deutsch)	test
50	TEST Tokenizer Leipzig (deutsch)	test
53	TEST Query Wortschatz	test
58	Tokenizer (IMS,TCF0.2,deutsch)	Tokenizer for German text.
61	POS Tagger (IMS,TCF0.2,deutsch)	Generate POS tags for tokenized German text.

Id:  Name:

ShortDescription:

Url:

Description:

Creator:

Contact:

InputWrapper:  OutputWrapper:

Name	Standard
Text	Utf8
Language	German

Name	Standard
Sentence	Utf8
Token	Utf8

Language:

Password:

- A format is described by a number of concepts that may be present in a valid „document“.
- A concept may be identified by an ISOcat datacategory
- Each concept in the „document“ is encoded following a certain standard. These standards may also be identified by ISOcat datacategories.
- Matchmaking of webservices is done (easy case) by combining the output specifications of previously executed services and matching them (subset) against the input specification of all other known services working on the same format.

- already done:
  - implemented a basic registry for webservices (url, descriptions etc + metadata on format & input/output)
  - added several services of the DSpin prototype to it
  - implemented a first version of the matchmaking algorithm
  - first successful test in two different workflow/chaining tools (Tübingen, Leipzig)
- future developments:
  - switch from a hard wired formats/concepts/standards-hierarchy to dynamic ones stored in a registry
  - access to ISOcat DR through WS interface and integrate it into a dynamic „format specification process“
  - open up the registry for harvesting
  - implement a generic chain builder: automatically suggest a chain from startpoint (resource) to endpoint (tool, certain information, ...)

# Questions

---



- Is a format itself also an ISOcat datacategory?
- Do we need optional parameters, especially for converter services?
- Do we need „parameter versions“ in order to reflect multiple flavours of information representation in one format?

# Join us at the WP2 Clarin Workshop 2009 in Leipzig

19.11.2009-20.11.2009



# CLARIN

Common Language Resources and Technology Infrastructure

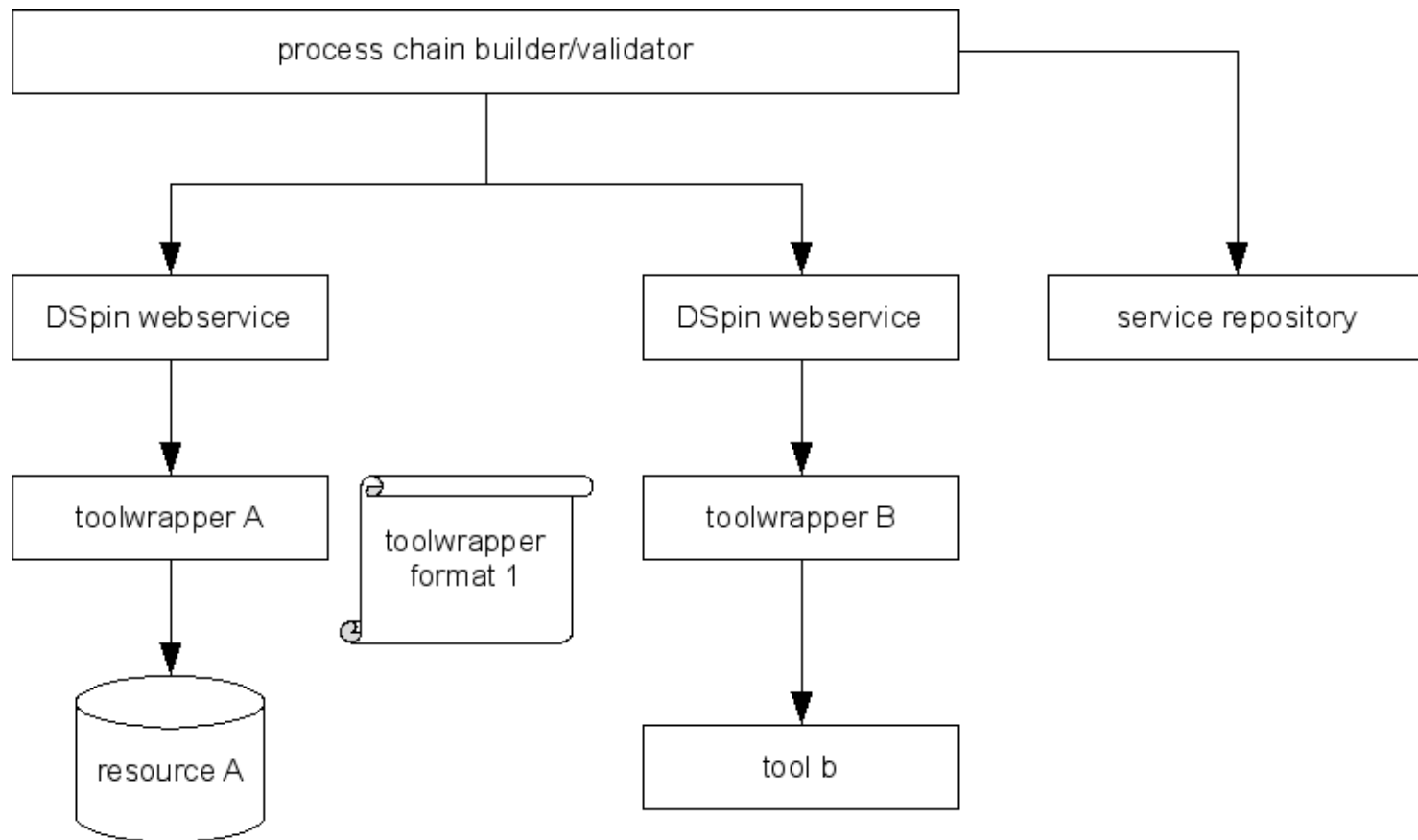


## Thank you for your attention

---

CLARIN has received funding from  
the European Community's Seventh Framework Programme  
under grant agreement n° 212230

# Backup/Questions I



# Backup/Questions II

converter/transformer repository (web 2.0; automatic inference, ...)

