Analyse de Concepts Logiques *(LCA)* et Traitement Automatique des Langues *(TAL)*

Annie Foret

Email: foret@irisa.fr

Équipe SemLIS (Logical Information Systems), département Gestion des Données et de la Connaissance

Université Rennes1, Irisa, France

24 mars 2016

TAL & IA

Table of Contents

1. On LIS and LCA
   - Logical Concept Analysis (LCA)
   - An introduction to Logical Information System (LIS)
2. LCA & LIS applied to terminological resources
   - A specialized terminology
   - An under-resourced language
3. LCA & LIS applied to syntactic lexicons
4. LCA & LIS applied to information extraction
5. Discussion and conclusion
6. [Annex]

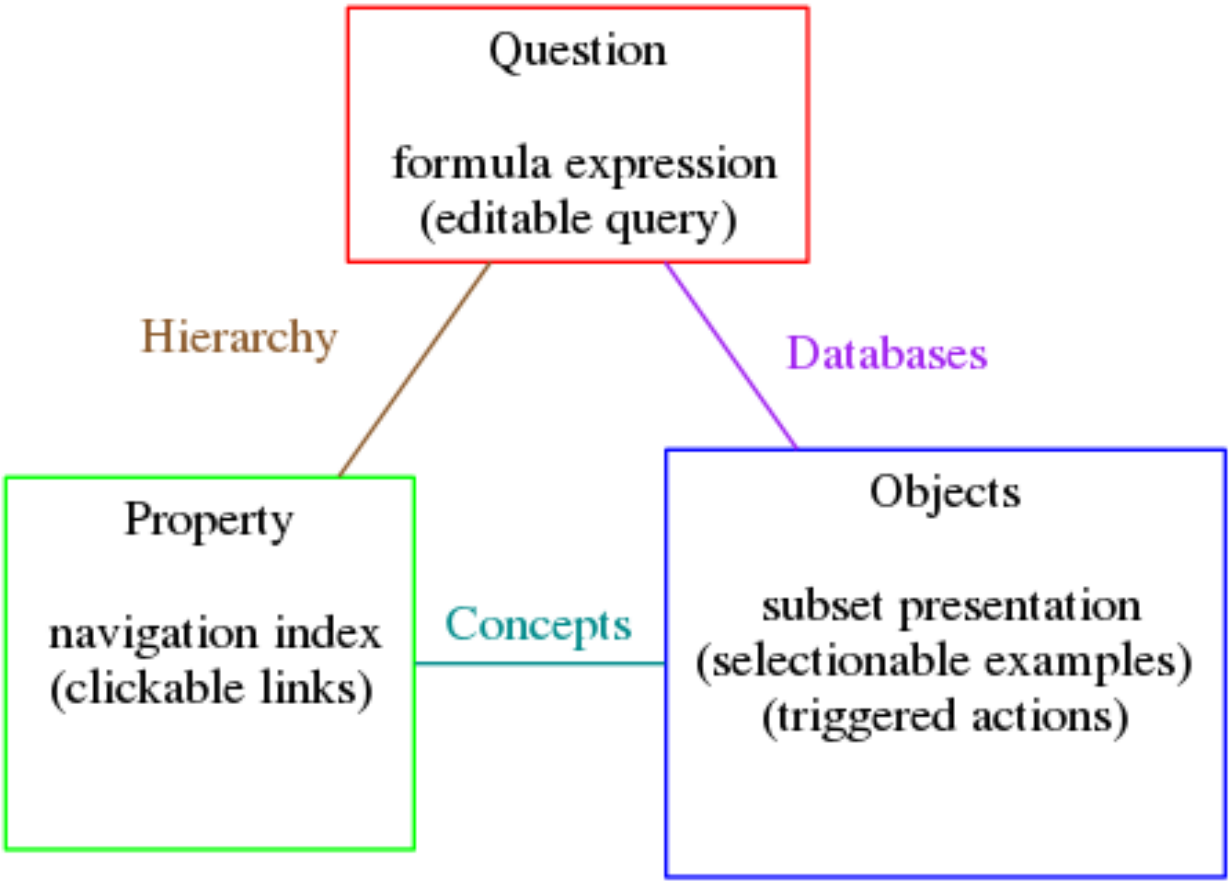
LCA = Logical Concept Analysis [Ferré and Ridoux, 2004]
(an extension of FCA)

LCA methods provide guidance and serendipity (cf experiment)
using a logical information context management system (Camelis)

- allows to explore the information content : linguistic info.,
  in a **flexible** way,
  without a priori knowledge,
  with data quality feedbacks
  can ease other treatments, after appropriate selection
- in an **extensible** way, it can be enriched :
  by adding diverse objects without confusion
  by adding properties to expand the browsing possibilities
  by adding triggered actions on objects, according to contextual arguments

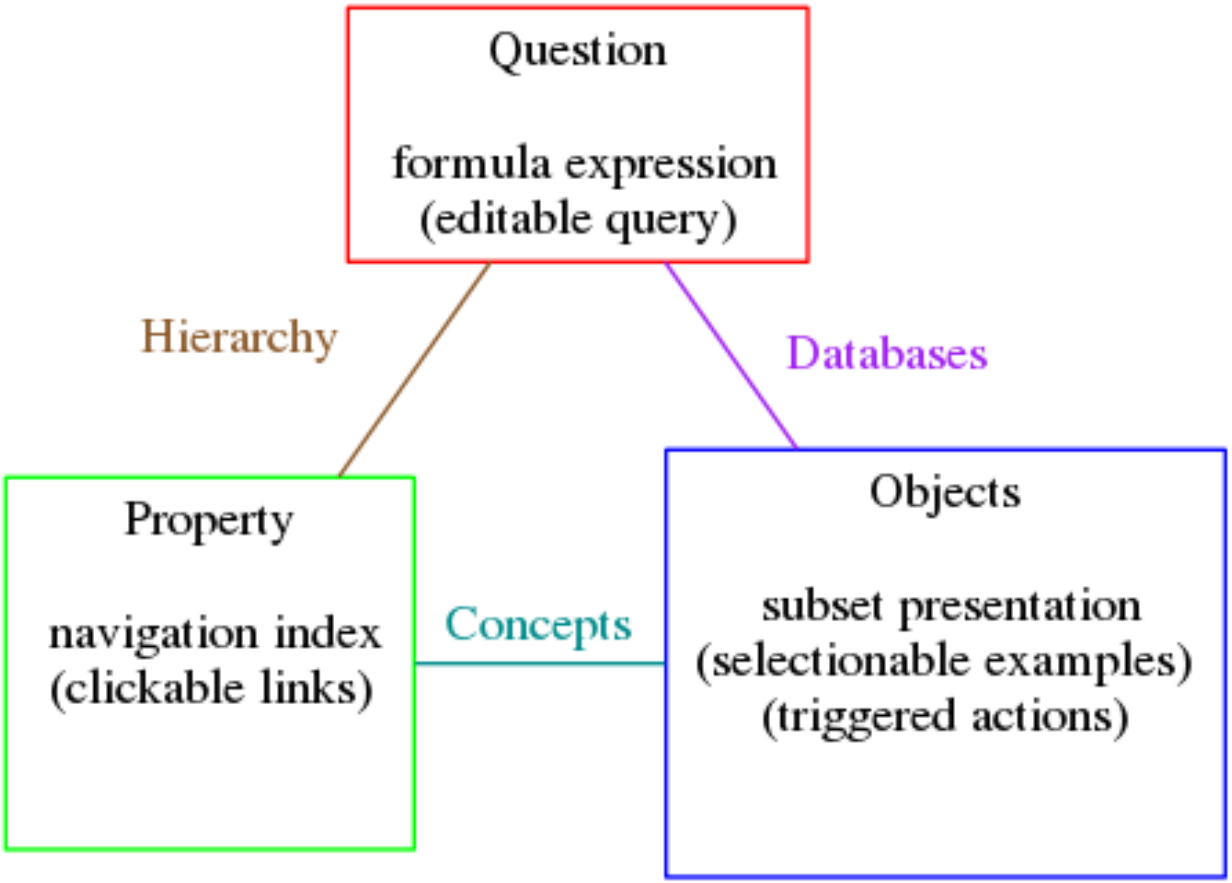
in a safe way.

Logical Information Systems Approach



LCA = Logical Concept Analysis [Ferré and Ridoux, 2004]
(an extension of FCA)

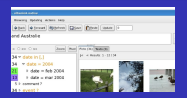
LIS Tools : Camelis



LCA = Logical Concept Analysis
(an extension of FCA)

LIS Tools : Camelis from [http://www.irisa.fr/LIS/softwares]

Introduction LIS



Logic

A *logic* is a pre-order $\mathcal{L}_T = (L, \sqsubseteq_T)$, where:

- L is a set of formulas,
- T is a customizable parameter of the logic,
- and $\sqsubseteq_T$ is a *subsumption* relation that depends on T.

$f \sqsubseteq_T g$ reads "f is more specific than g" or "f is subsumed by g",
also denotes the partial ordering induced from the *pre-order* *(refl., trans.)*

Information Context

For any object o, a formula $d(o)$ denotes the description of o

$\rightarrow$ *A query is a logical formula.*

A *logic functor* is a function $\mathcal{F}$ that:

- takes logics $\mathcal{L}_1, ..., \mathcal{L}_n$ as arguments $(n \geq 0)$,
- returns a composed logic $\mathcal{L}_T = \mathcal{F}(\mathcal{L}_1, ..., \mathcal{L}_n)$.

System Tool

$\rightarrow$ *Customized logics for querying, navigating data.*

Context = collection of objects
+ their logical description (vs set of attributes)

Def. A *logical context* is a tuple $K = (\mathcal{O}, \mathcal{L}_T, X, d)$, where:

- $\mathcal{O}$ is a finite set of objects, ←
- $\mathcal{L}_T = (L, \sqsubseteq_T)$ is a Logic,
- $X \subseteq L$ is finite, called the *navigation vocabulary*, ←
- $d \in (\mathcal{O} \to \mathcal{L}_T)$ is a mapping from objects to formulas. ←

For any object o, the formula $d(o)$ denotes the *description* of o.

Answer (extent) of a *query* formula q
= the set of objects whose description is subsumed by this formula

Def. Extent

Let K be a logical context, and $q \in \mathcal{L}$ be a *query* formula.
The extent of q in K is:

$$K.ext(q) = \{o \in \mathcal{O} \mid d(o) \sqsubseteq_T q\}.$$

LIS – Logic Components

A *logic functor* is a function $\mathcal{F}$ that:

- takes logics $\mathcal{L}_1, ..., \mathcal{L}_n$ as arguments ($n \geq 0$),
- returns a composed logic $\mathcal{L}_T = \mathcal{F}(\mathcal{L}_1, ..., \mathcal{L}_n)$.

As for any logic, one has $\mathcal{L}_T = (L, \sqsubseteq_T)$, but L (resp. $\sqsubseteq_T$) is function of the sets of formulas (resp. subsumption) of arguments $\mathcal{L}_i$

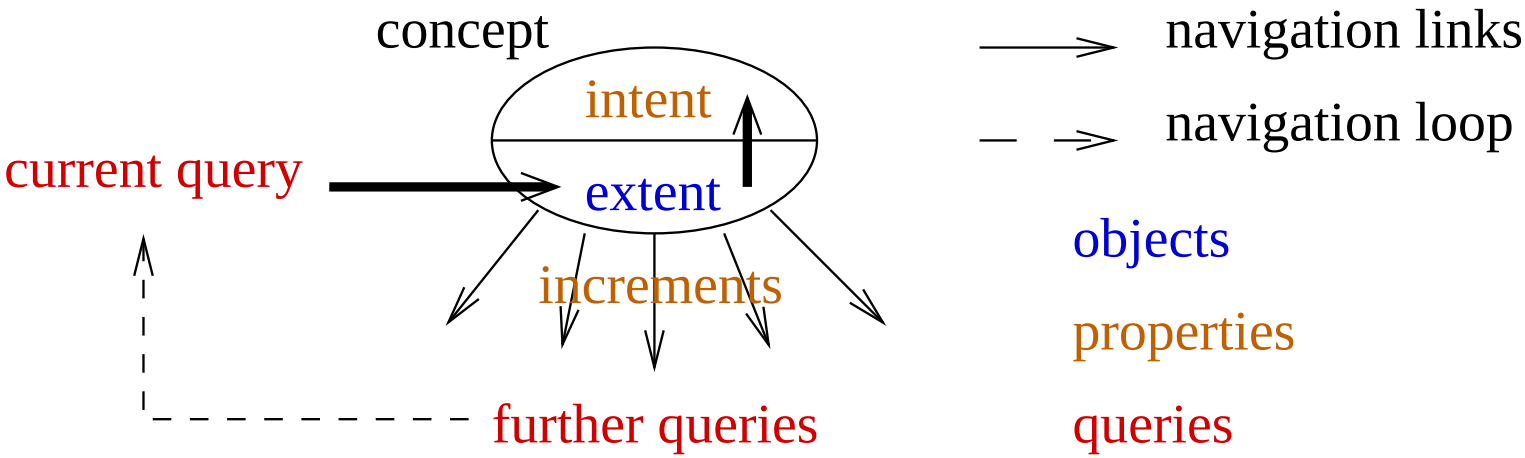
- LogFun ToolBox: in Objective Caml

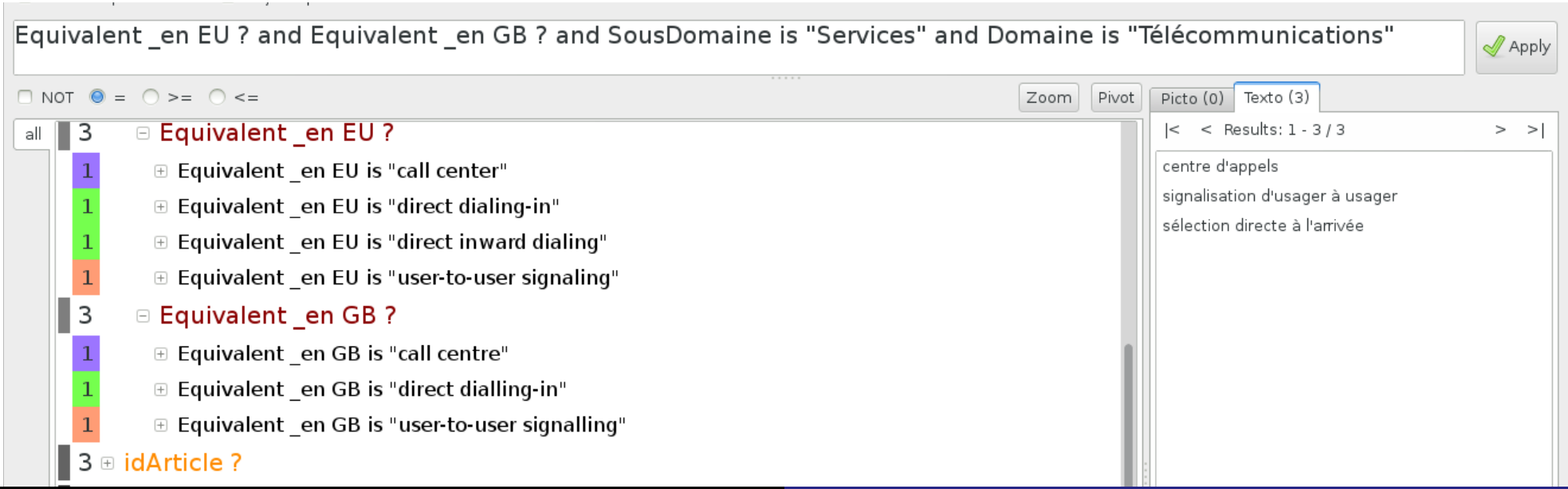
  http://www.irisa.fr/LIS/ferre/logfun/
- Logical Components, and "Glue"
  - Prop, ...
  - Concrete domains (Atom, Int, Interval,...); Structured Data (Product,...)
- Provers (decidable fragments)
- Customized logics : Prop(Atom), Prop(Interval(Int)), ...
- Querying, Navigating in logical contexts

  http://www.irisa.fr/LIS/ferre/camelis/

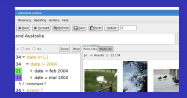
LIS – Navigating in a Logical Context

The concept lattice
→ local view + moves





LIS – Characteristics and Advantages



Properties

- **Navigation Completeness**
- **No dead-end**

Expected qualities:

- extensible descriptions (by users)
- rich datatypes
- entailment relations between values
- expressive querying: patterns, AND, OR, NOT
- flexible navigation (guiding users)

and interactions with other tools

In the scientific and technical fields

XML and open data

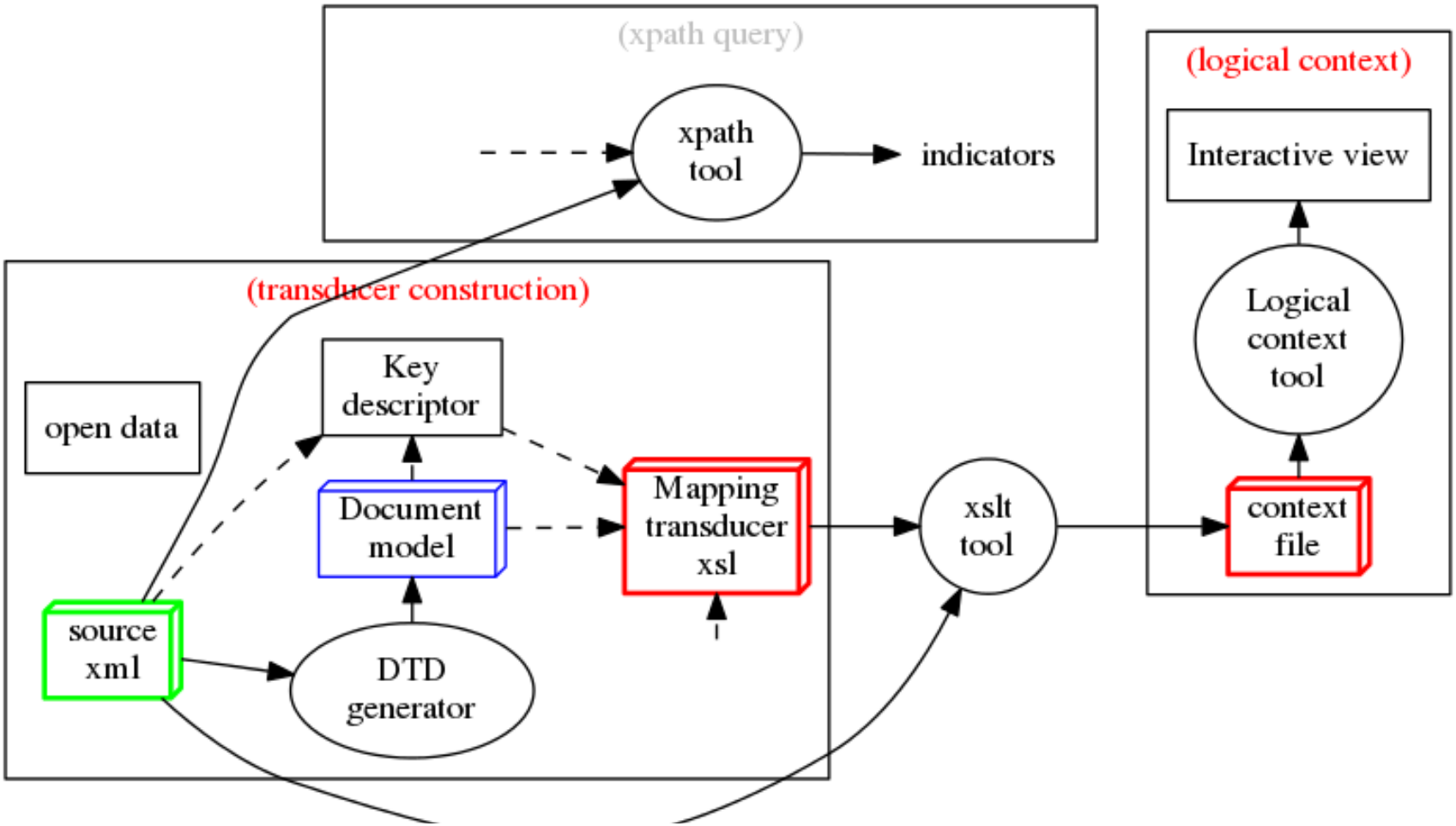
In the scientific and technical fields

XML and open data

```
<Article numero="SPAT333" commission="CSTN Sciences et Techniques spat
publie="1" annule="0" administration="1">
<Domaine>
<Dom numero="1"><![CDATA[Sciences et techniques spatiales]]></Dom>
<S-dom numero="1"><![CDATA[Mécanique spatiale]]></S-dom>
</Domaine>
<DatePub>18/04/2001</DatePub>
<Definition>
Correction d-orbite consistant à réduire l-altitude moyenne d-un satel
artificiel.</Definition>
<Notes>
</Notes>
<Terme statut="privilegie">
abaissement d-orbite</Terme>
<Terme_balise>
abaissement d-orbite</Terme_balise>
<Terme_exergue><strong>
abaissement d-orbite</strong></Terme_exergue>
<Equivalent langue="en" admis="étranger">
<Equi_prop>
orbit lowering</Equi_prop>
</Equivalent>
<Voir>
<A href="visualisation_fiche.html?FIC_ID=815">déclin d-orbite</A>
<A href="visualisation_fiche.html?FIC_ID=1888">relèvement d-orbite</A>
<A href="visualisation_fiche.html?FIC_ID=1892">rétrécissement d-orbite
<A href="visualisation_fiche.html?FIC_ID=1839">surbaissement d-orbite<
</Voir>
</Article>
<Article numero="SPAT1" commission="CSTN Sciences et Techniques spatia
publie="1" annule="0" administration="1">
<Domaine>
<Dom numero="1"><![CDATA[Sciences et techniques spatiales]]></Dom>
<S-dom numero="1"><![CDATA[Matériaux]]></S-dom>
<S-dom numero="2"><![CDATA[Thermique]]></S-dom>
</Domaine>
<DatePub>22/09/2000</DatePub>
<Definition>
Propre à l-ablation.</Definition>
```

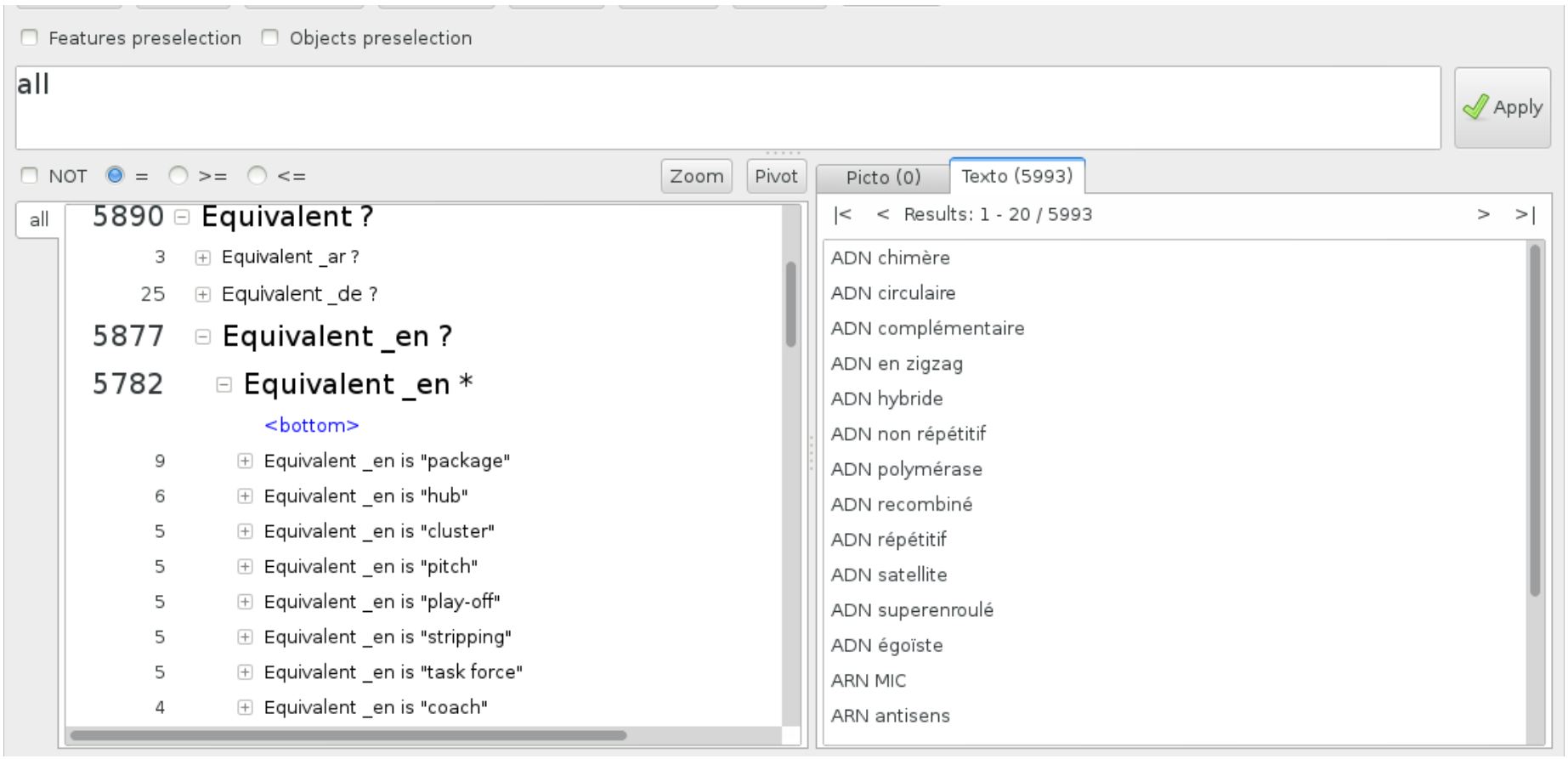
The transducer methodology

designed to present data in a logical information system
without losing information, but gaining in ease of exploration.



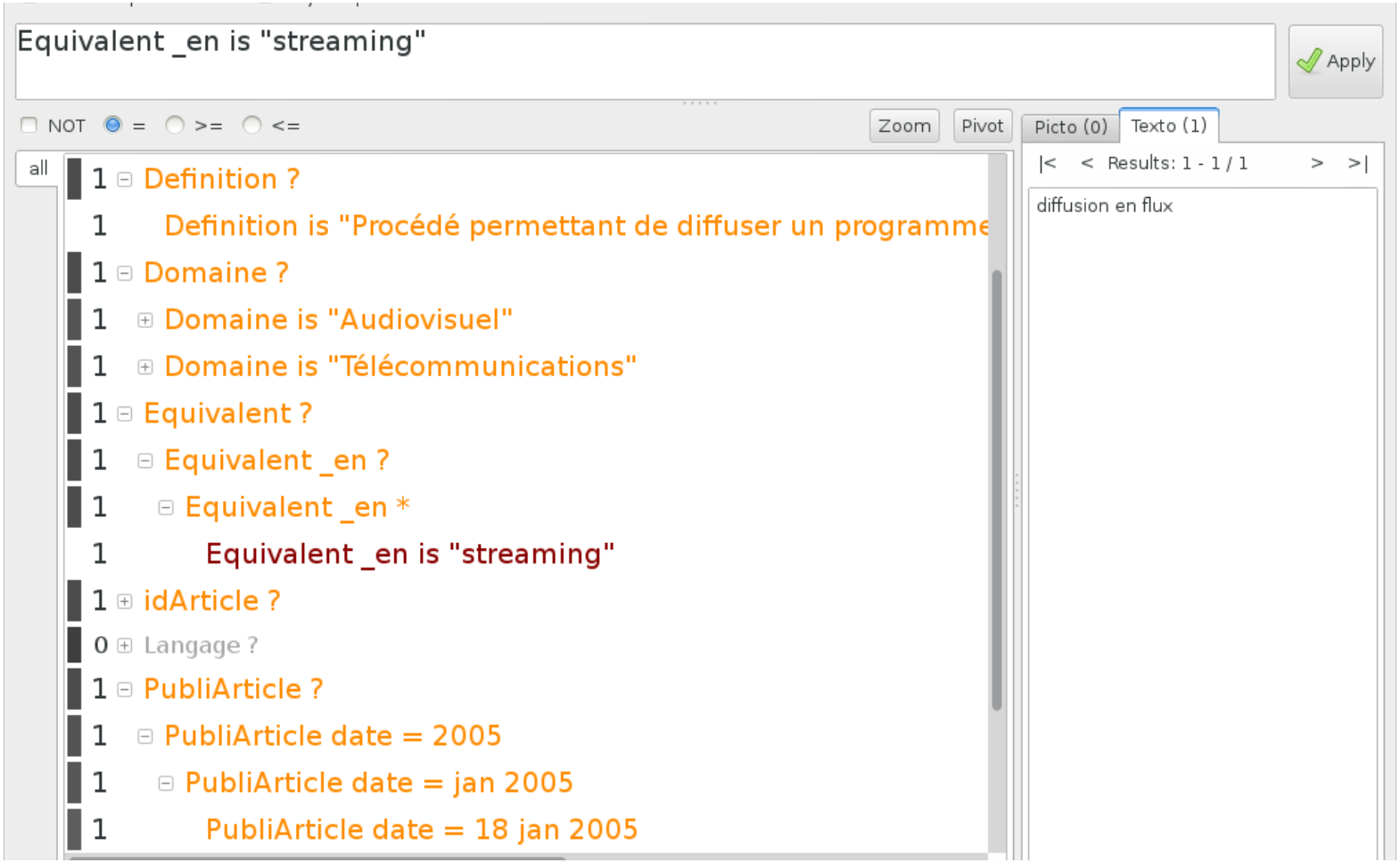
Sub-context Cardinalities.

In the property index-tree, we may choose an order for displaying a given facet. This is useful for example to read directly which `Equivalent _en` correspond to the greatest number of French terms



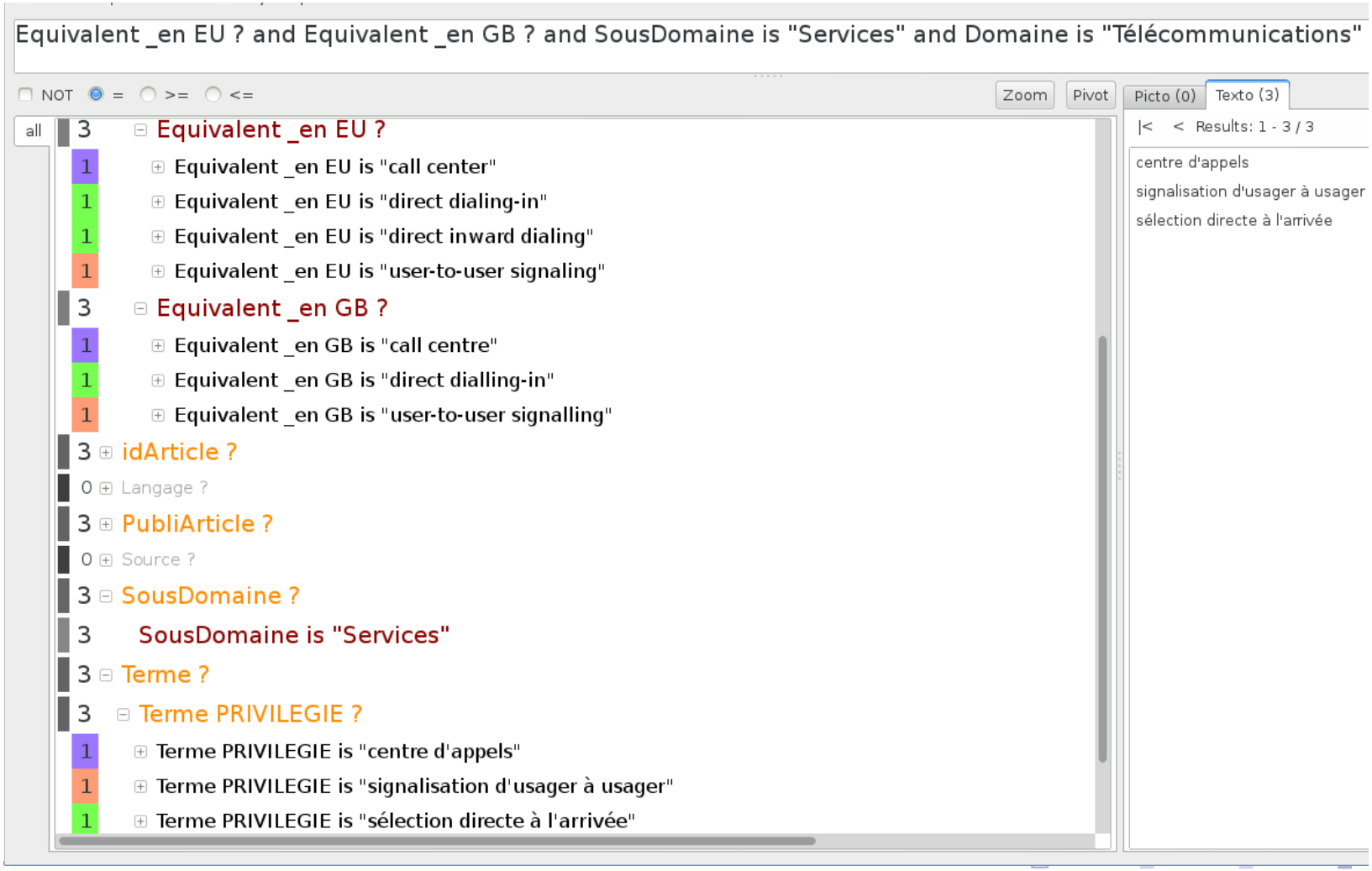
Data types.

Data types other than attributes and strings can be handled, such as dates, allowing for more or less fine-grained selections



Exploring variants and false-friends.

A state from selections in the index tree ;
links of a same color characterize the same set of objects (concept)



Exploring variants and false-friends.

A state from selections in the index tree ;
links of a same color characterize the same set of objects (concept)

Figure: Variants

User help :

- potential linguistic errors (in a domain / subdomain) identified
- other examples can be highlighted as ambiguous, such as
  "package" (`Equivalent _en`)
  or some abreviations ("ABS")
  the dynamic navigation links (domains, etc.) then provide hints to disambiguate.

Summaries and exceptions

Using the `Not` button, we can arrive at the subcontext

$$A_2 = not(A_1)\ and\ A_0$$

property A_2 expresses a search for exceptions to A_1,
we get a kind of subcontext summary and extra informations

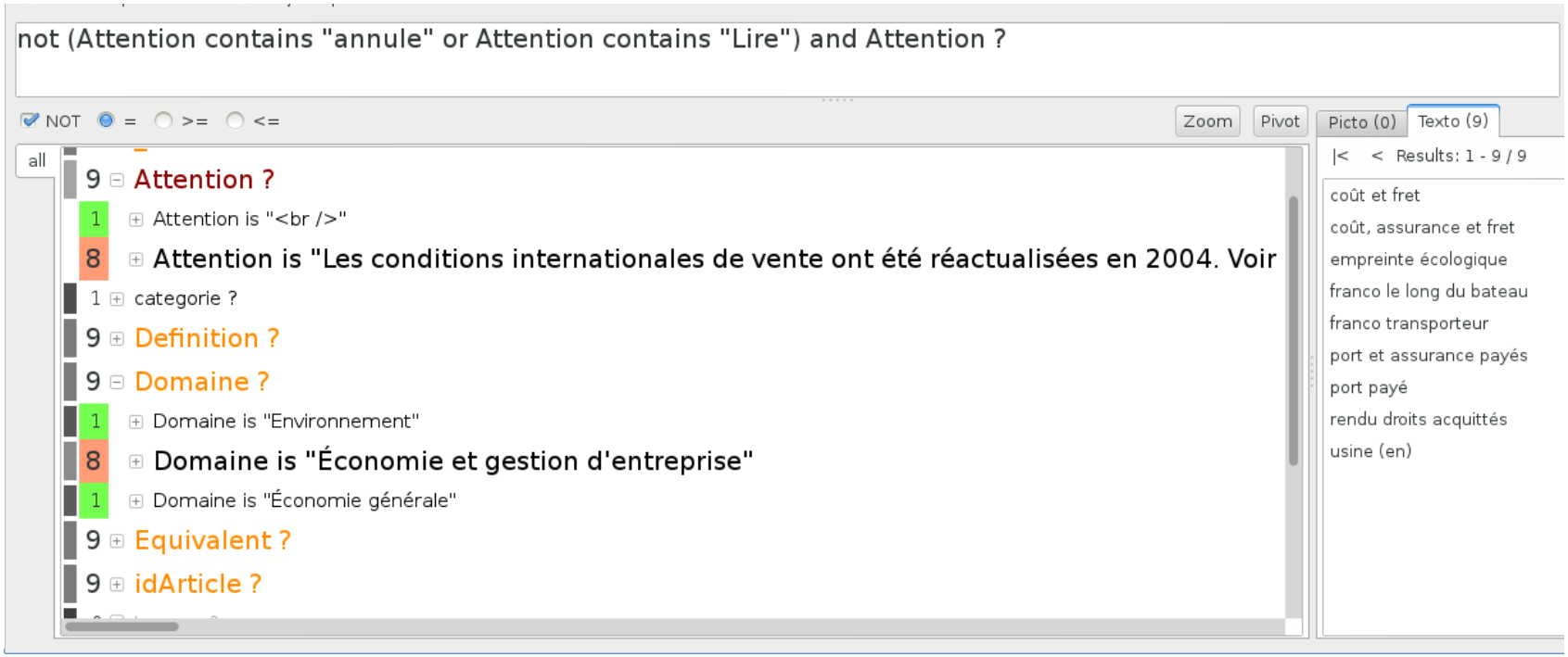


Figure: Focus on `Attention`

Data quality - feedback

This navigation mode allows to detect abnormalities, in particular

- pseudo-empty properties appear on other facets
  (through a link like Definition is "")
  these cases often correspond to existing but empty XML elements (but are not XML errors).
- low cardinalities in the navigation tree may suggest to explore the link, by selecting it and opening other facets simultaneously ;
  we can analyse this way "the words without translation, following the not Equivalent? link.
- in case of redundancies, noticeable through browsing:
  exploring the Antonymes facet,
  we can see XML structuring redundancies (this information being carried by two source elements).

Refinements 1 – translating index

Adapting facets using rules

Domains and SubDomains data have been translated.
The resulting two context files contain update rules of the form:

```
rule_extr Domaine is "Acoustique" -> Domain is "Acoustic"
```

when loaded in the context, properties on the right hand side are added to all objects verifying the left hand side.

Refinements 2 – harmonization

Improving grammatical categories using rules and axioms

In the original context, grammatical categories are listed with various values

- a disjunction, as in:

```
categorie is "adj.  ou n.m." and Equivalent _en is "crossmedia (n.  ou adj.)"
```

which selects the term `transmédia`;

but we also observe its permutation `categorie is "n.m.  ou adj."`;

- a more or less fine granularity, as in: `categorie is "n.m.inv."`

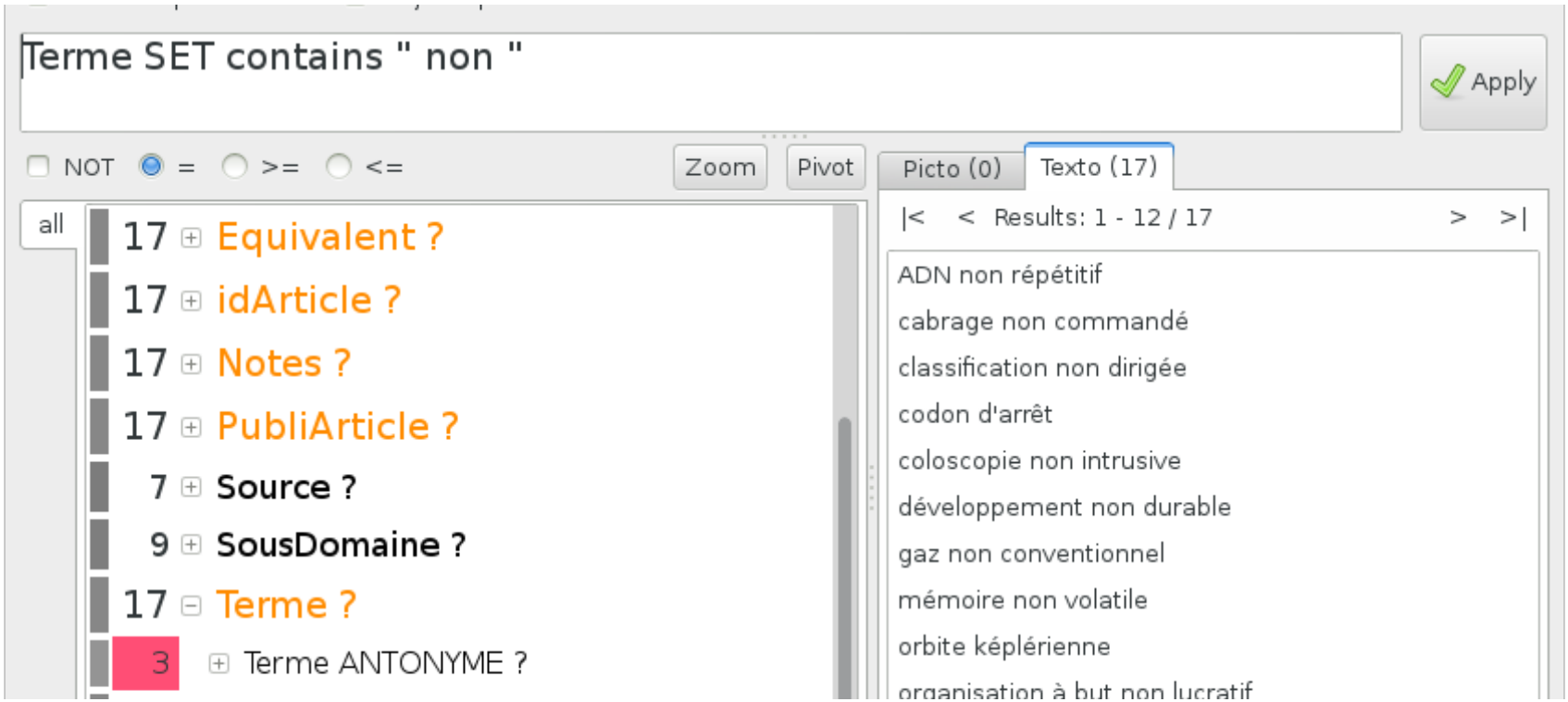
We add rules and axioms in the logical context such as:

```
rule_extr categorie is "n.m.inv." -> Categorie_n_m_inv  ...
Categorie_n_m_inv axiom, Categorie_n_m
Categorie_n_m axiom, Categorie_n
...
```

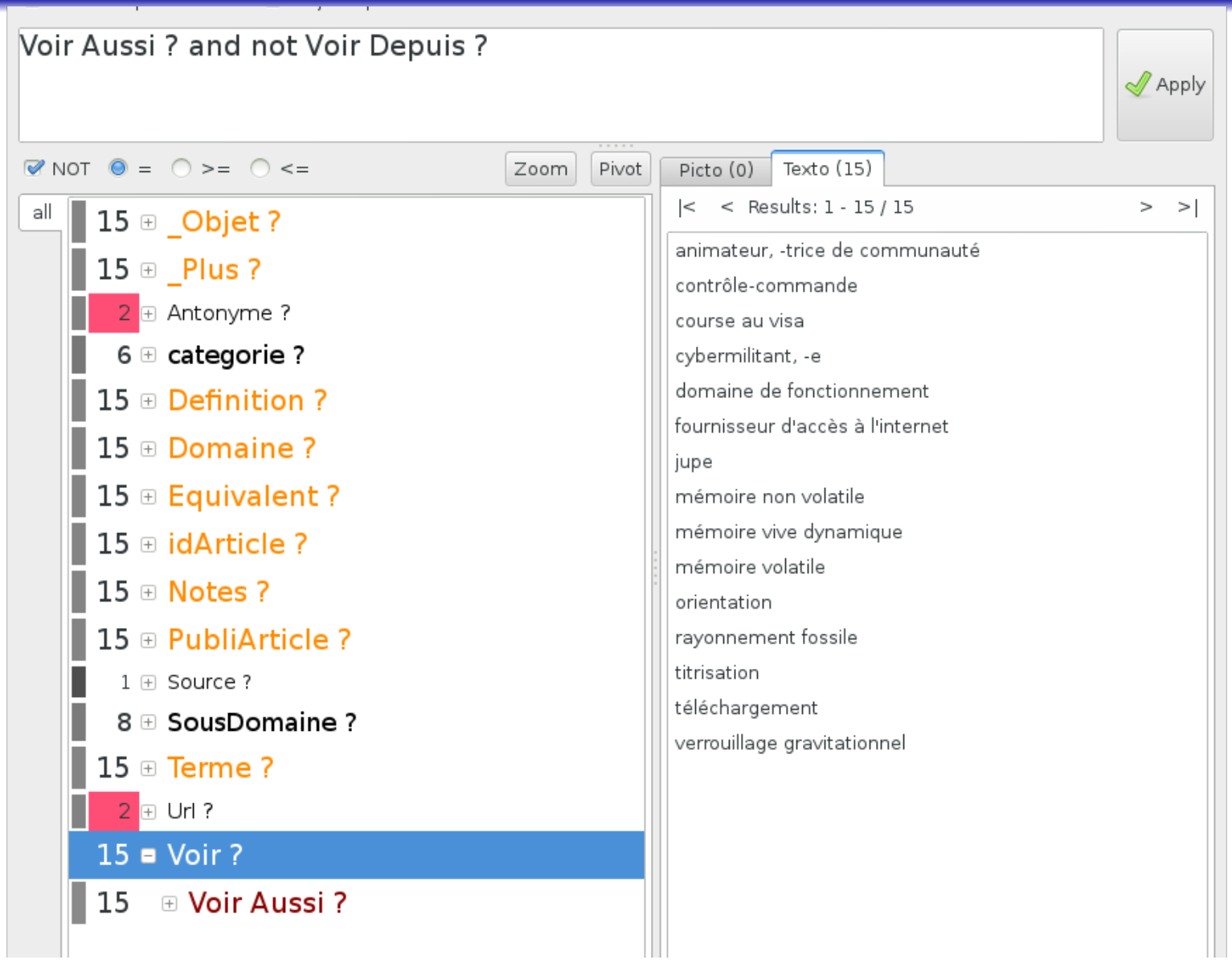
3 – Axioms for property variants – browsing improvements

```
axiom PRIVILEGIE, SET
axiom SYNONYME, SET
axiom SET, ANY
axiom ANTONYME, ANY
```

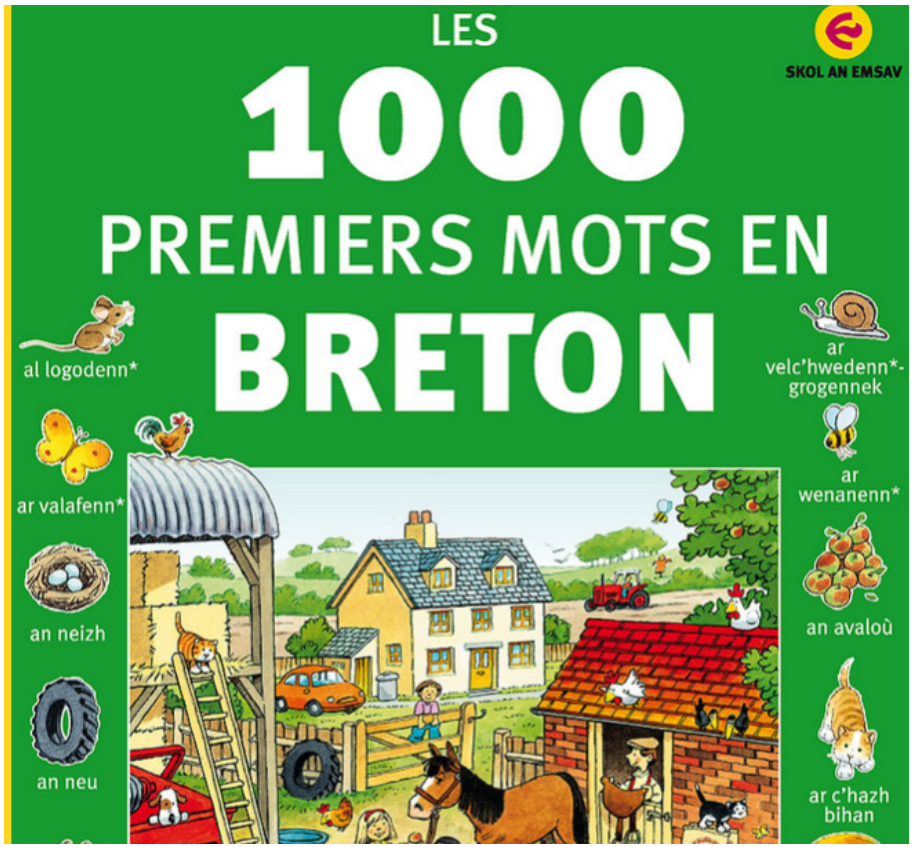
A query may then group several properties (having a status below another expression like SET) as examplified in figure 8.



4 – Linking data and resources



From a small breton lexicon (and Apertium data)



by theme : "at home", "at school"
name with det, gender : al logodenn*

→

Lexicon (simple xml)

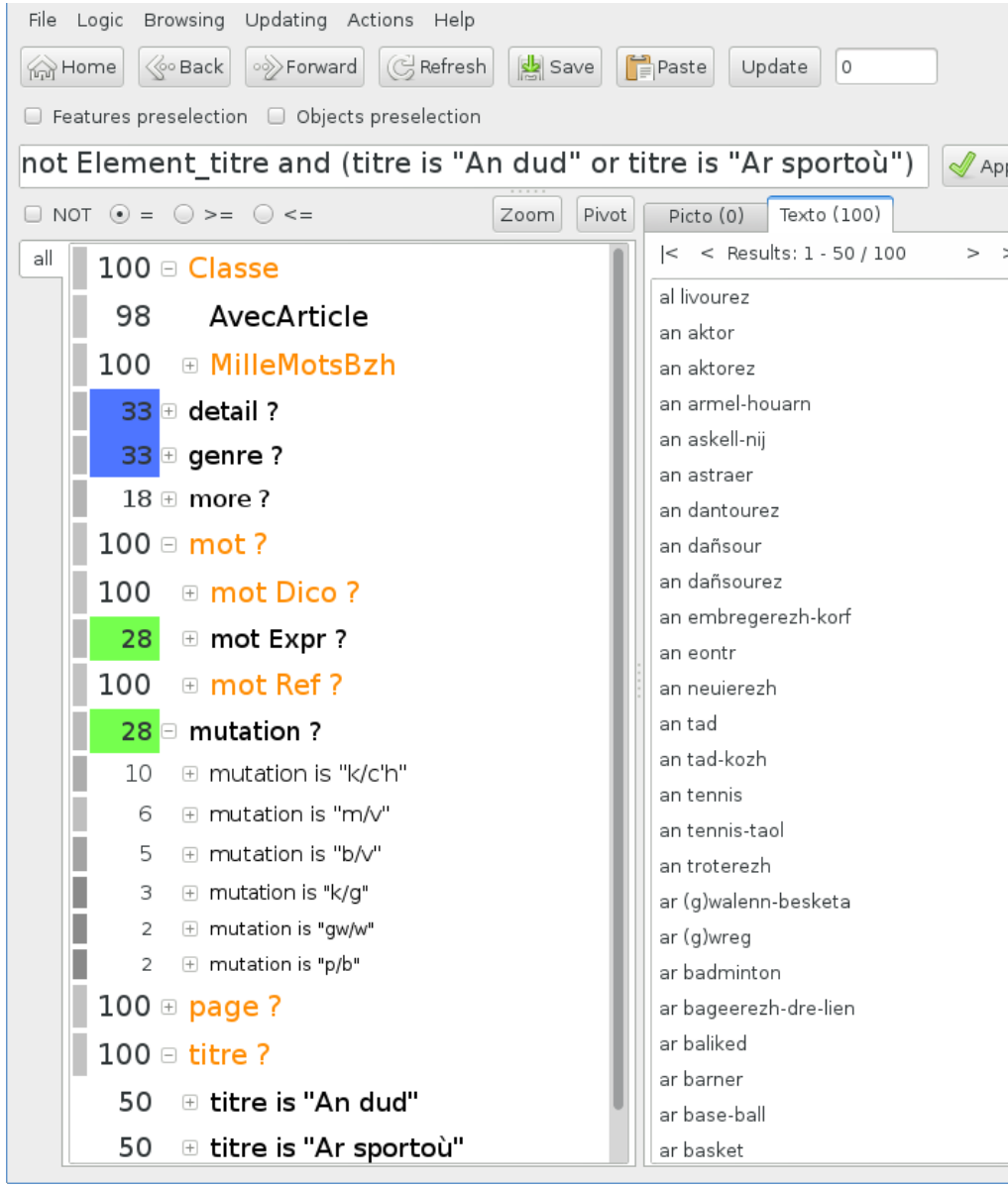
Expressions (has image)
written by theme :
- with gender
- no image
- no translation

enriched with :
- mutation marks
- comments

↓

LIS inform. context

Monolingual context
→ after a new selection in the index on the left (dynamic, reorded)
→ coherence between windows
→ indicators :
- size
- same color when same subset
- new links for exploring
(local view of the
LCA lattice concept)



Resource integration : Apertium.

The "home" context is now enriched, to integrate multilingual information coming from Apertium.
For every added language, the words that are unknown in Apertium are marked by * (at the beginning).

It can be a starting point to diverse evaluations, such as
(1) a cover analysis of the translator
suggestions for additions
and
(2) characterization of the cases without translation.
Certain incoherence can be also highlighted.

Multilingual context

Context br + fr

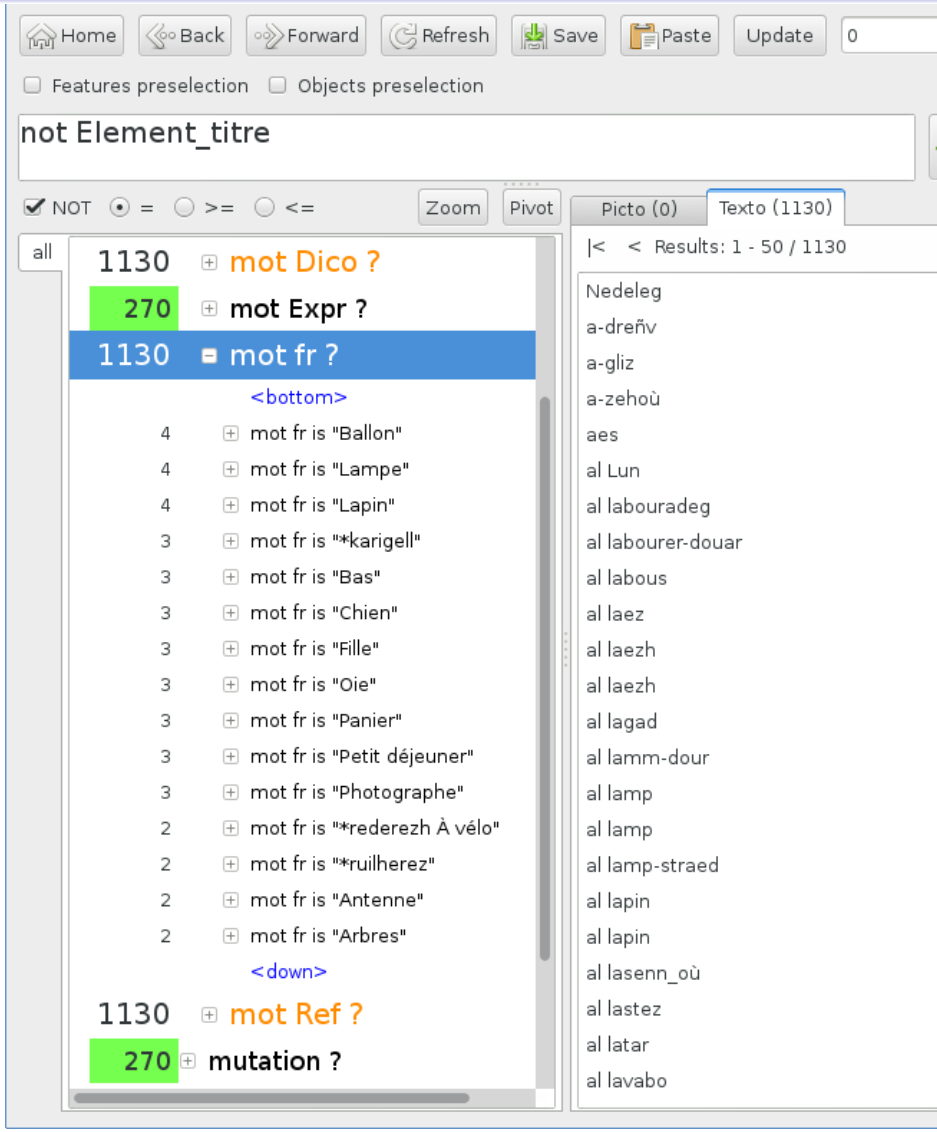
For french :

→ `nb = ...` (dico br-fr)

→ `mot fr is ...`

as *translation* Apertium *for*

`mot Dico is ...`



Multilingual context

Context br + fr

For french :

→ `nb = ...` (dico br-fr)

→ `mot fr is ...`

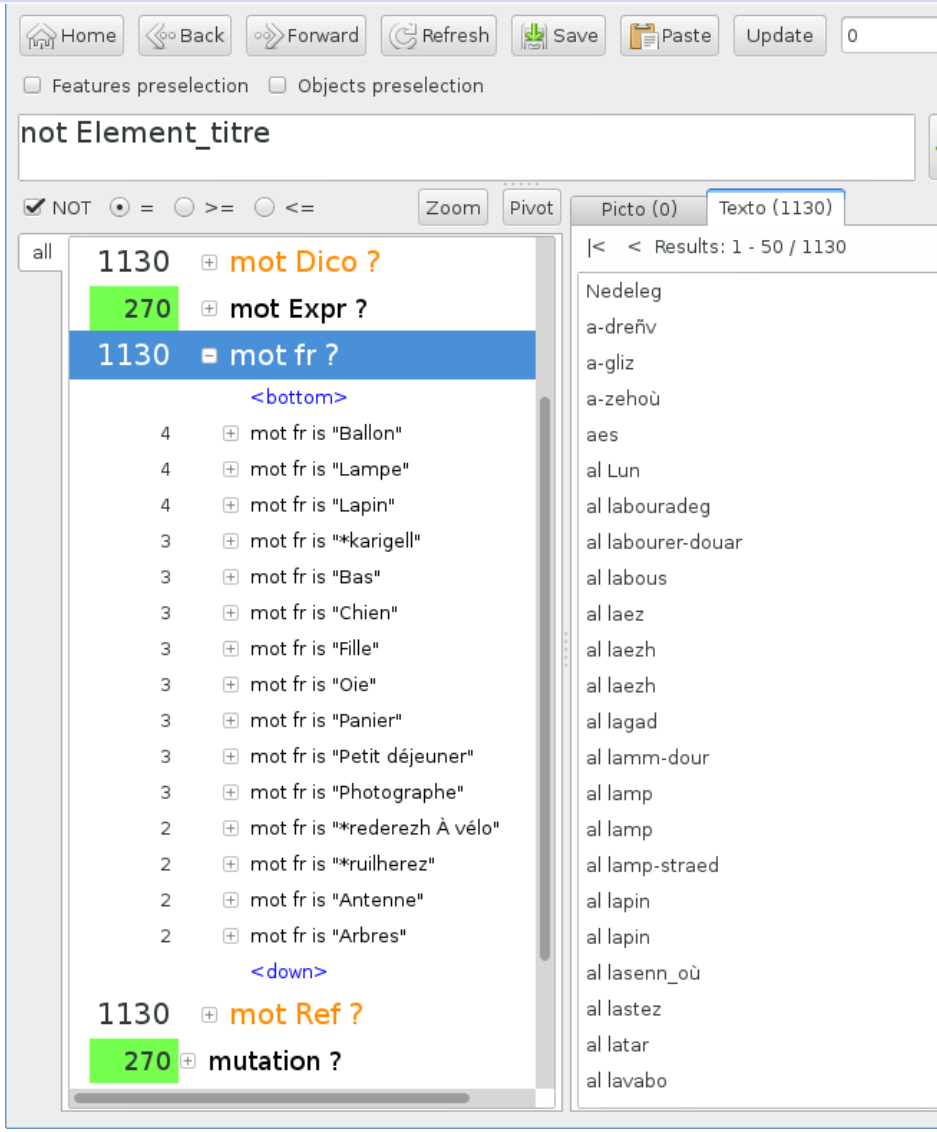
as *translation* Apertium *for*

`mot Dico is ...`

Then :

→ `mot eo is ...`

→ `mot gb is ...`



Multilingual context

Context br + fr

For french :

→ `nb = ...` (dico br-fr)

→ `mot fr is ...`

as *translation* Apertium *for*

`mot Dico is ...`

Then :

→ `mot eo is ...`

→ `mot gb is ...`

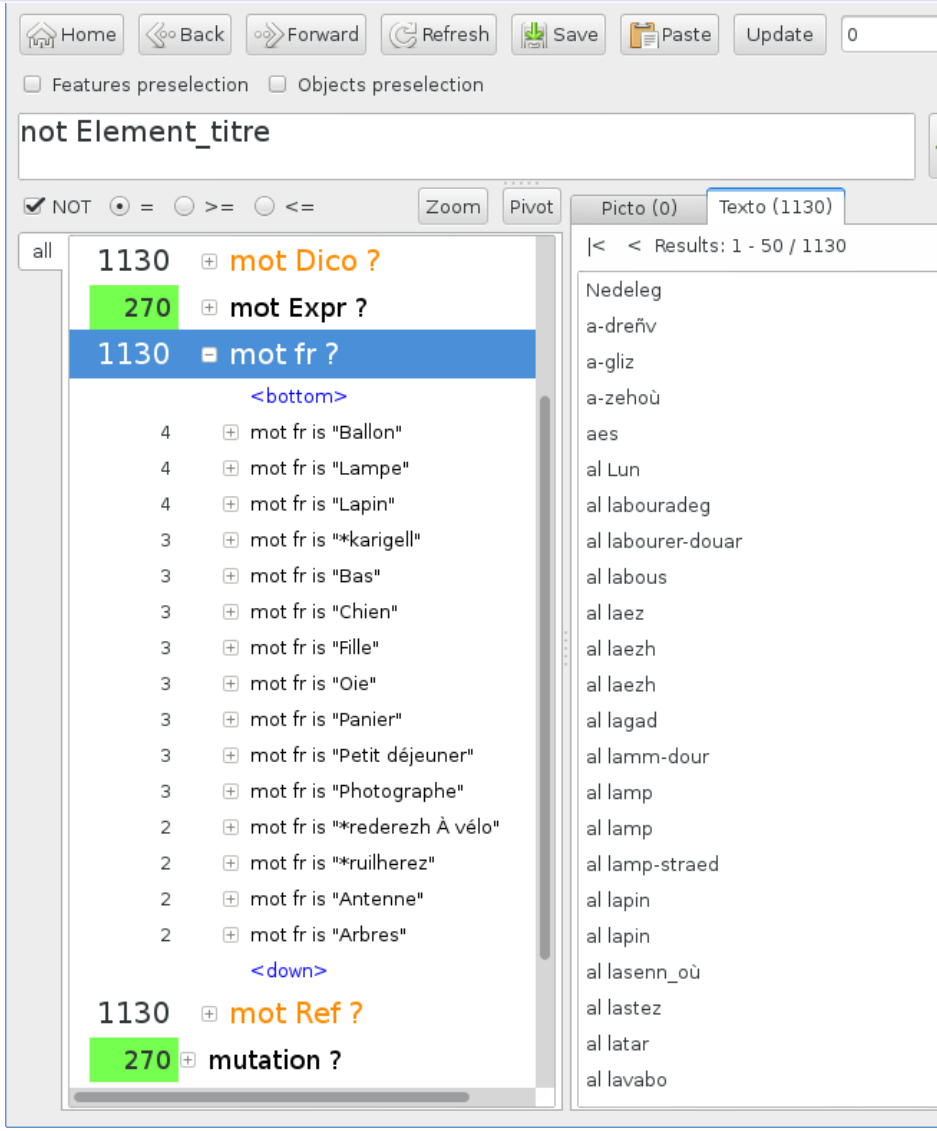
Transformation2 using

apertium br-fr, apertium fr -eo,

apertium eo-en

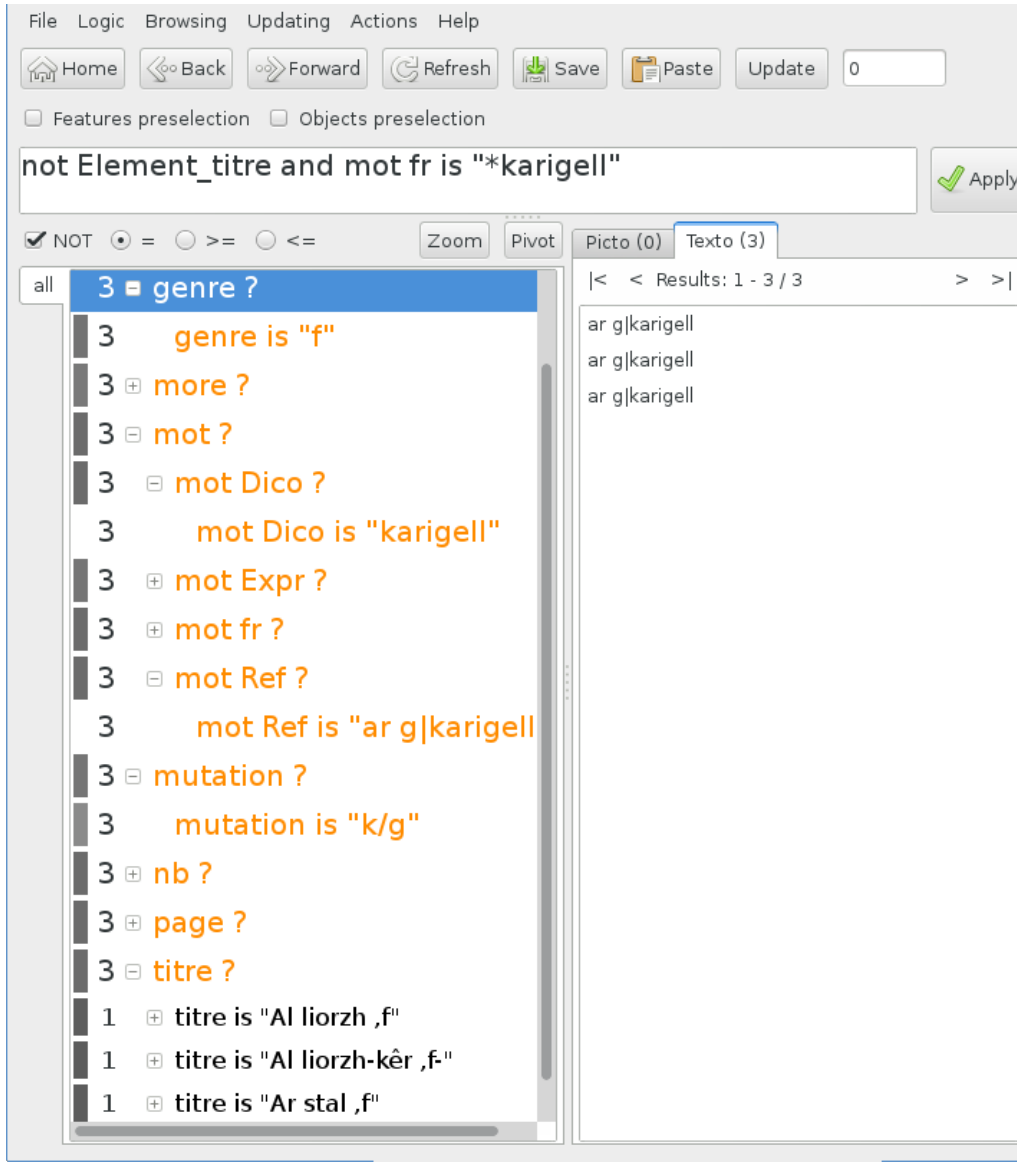
produces context rules

rule_extr (mot Dico is "kador") mot fr is "Chaise"



Multilingual context

Context br + fr
→ after new selection (left)
→ related to a breton word
ar "karigell"
belonging to several themes
but not found in Apertium



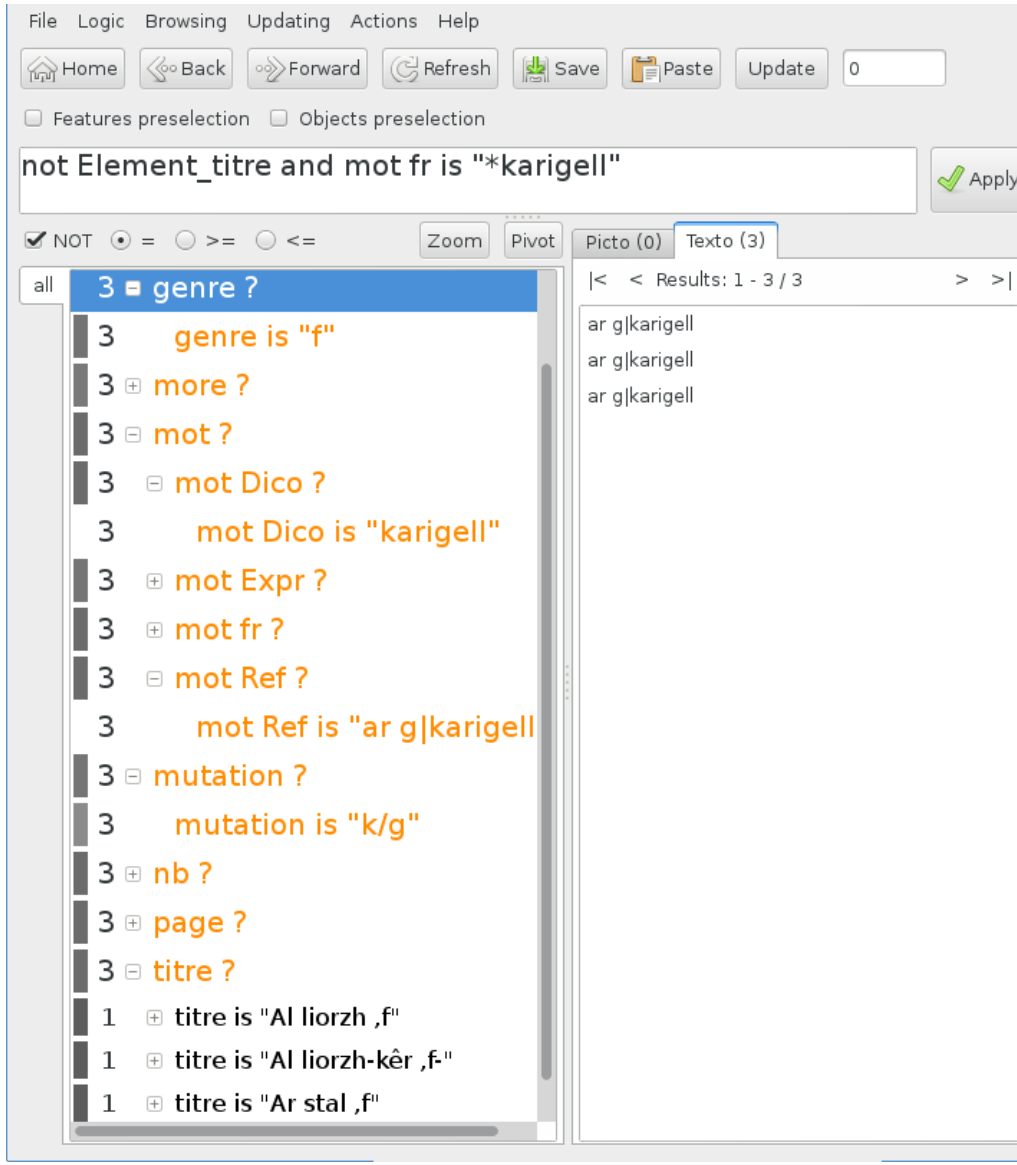
Multilingual context

Context br + fr
→ after new selection (left)
→ related to a breton word
ar "karigell"
belonging to several themes
but not found in Apertium

A finer analysis shows 100 words in the album without translation

Other problems, compound-words

Other multilingual findings
(involving br or not)



Categorial Grammars as LIS

- Categorial grammars (such as pregroups) can be considered as LIS both theoretically, and practically.
- Enabled by:
  - their lexicalized nature
  - the logical nature of this linguistic framework
- Logical Concept Analysis can provide:
  - better representations
  - better propagation of properties
- The combination, through the toolbox,
  - helps the creation, control and query of linguistic grammars
  - illustrates the strength and elegance of logic functors

Categorial grammars

definitions

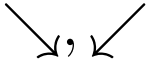
- Σ = alphabet for words of a natural language

  { Jean, court, nage, vite,. . .}

- Tp = types with constructors : $\backslash$ $/$ $\bullet$

  $\bullet$, $(.)^l$ and $(.)^r$, with $X^{ll} = X^{(-2)}$ etc. in pregroups (PG)

  Pr = primitive types : $(S, N, \cdots)$



- G = *categorial grammar* over Σ

  - associates types of Tp to words in Σ

  G_0 = { Jean $\mapsto N$, court, nage $\mapsto N \backslash S$, . . . }

- $\mathcal{L}(G)$ = set of strings generated by G

Jean	court	$\in \mathcal{L}(G_0)$
N	$N \backslash S$	$\vdash S$
j	$\lambda(x).court(x)$	$court(j)$

- G generates a string $c_1 \dots c_n \in \Sigma^+$ iff $\exists A_1, \dots, A_n \in Tp$:

  $G : c_i \mapsto A_i \ (1 \leq i \leq n)$ and $A_1, \dots, A_n \vdash S$

  - with derivation rules on types

$\backslash_e : B, B \backslash A \vdash A$

$/_e : A / B, B \vdash A$

a few more for L or CCG

$MP : B, B \to A \vdash A$

$/_e : A \bullet B^l, B \vdash A$

for PG

Lexicon

Language

Logic

Categorial grammars

definitions

- Σ = alphabet for words of a natural language

  { Jean, court, nage, vite,. . .}

- Tp = types with constructors : $\backslash \quad / \quad \bullet$

  $\bullet$, $(.)^l$ and $(.)^r$, with $X^{ll} = X^{(-2)}$ etc. in pregroups (PG)

  Pr = primitive types : $(S, N, \cdots)$

Lexicon

- G = *categorial grammar* over Σ

  - associates types of Tp to words in Σ

  G_0 = { Jean $\mapsto N$, court, nage $\mapsto N \backslash S$, . . . }

- $\mathcal{L}(G)$ = set of strings generated by G

Jean	court	$\in \mathcal{L}(G_0)$
N	$N \backslash S$	$\vdash S$
j	$\lambda(x).court(x)$	$court(j)$

Language

- G generates a string $c_1 \dots c_n \in \Sigma^+$ iff $\exists A_1, \dots, A_n \in Tp$:

  $G : c_i \mapsto A_i \ (1 \leq i \leq n)$ and $A_1, \dots, A_n \vdash S$

Logic

- with derivation rules on types

$\backslash_e : B, B \backslash A \vdash A$ $\qquad$ $/_e : A / B, B \vdash A$

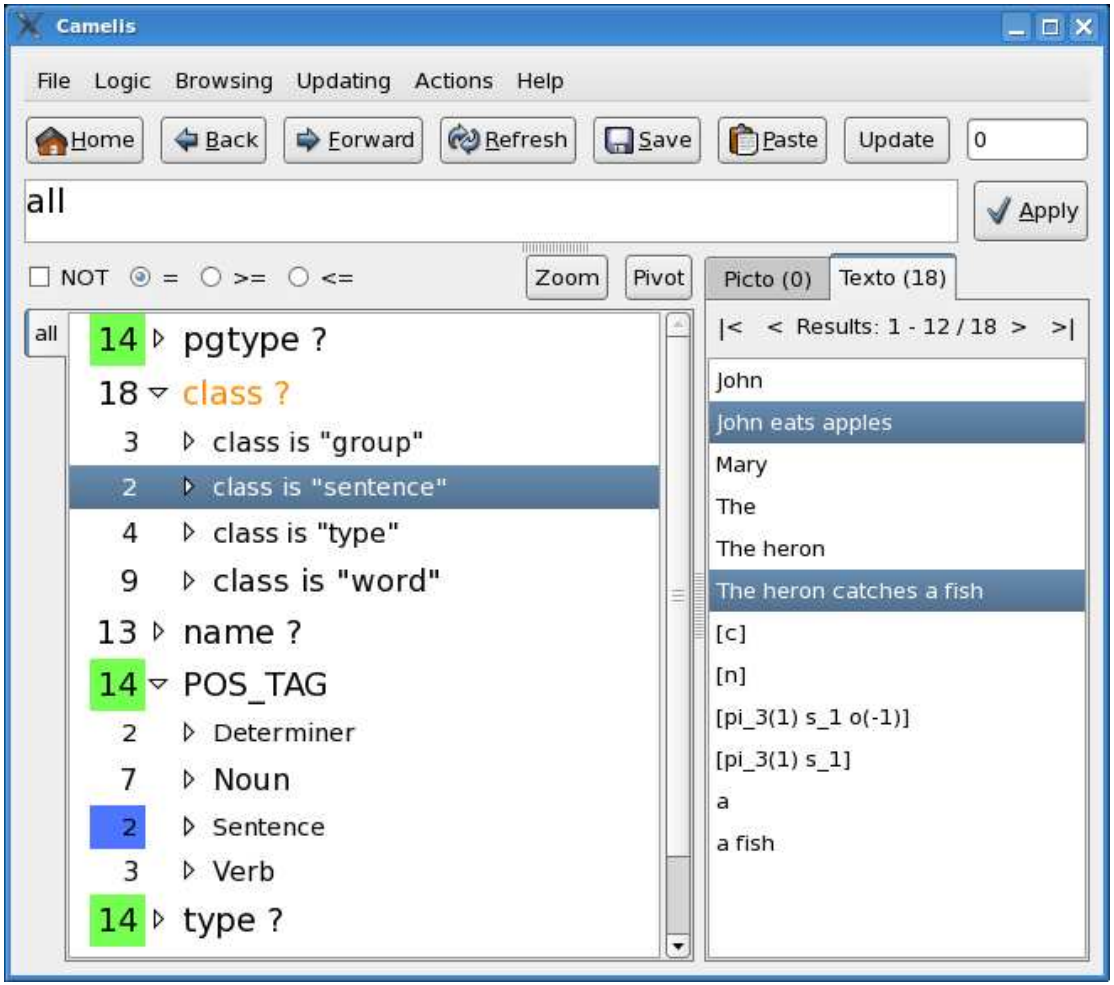
a few more for L or CCG

$MP : B, B \to A \vdash A$ $\qquad$ $/_e : A \bullet B^l, B \vdash A$

for PG

- G k-valued iff at most k types per symbol (rigid : $k = 1$)

A toy grammar, with other information



using PG axioms

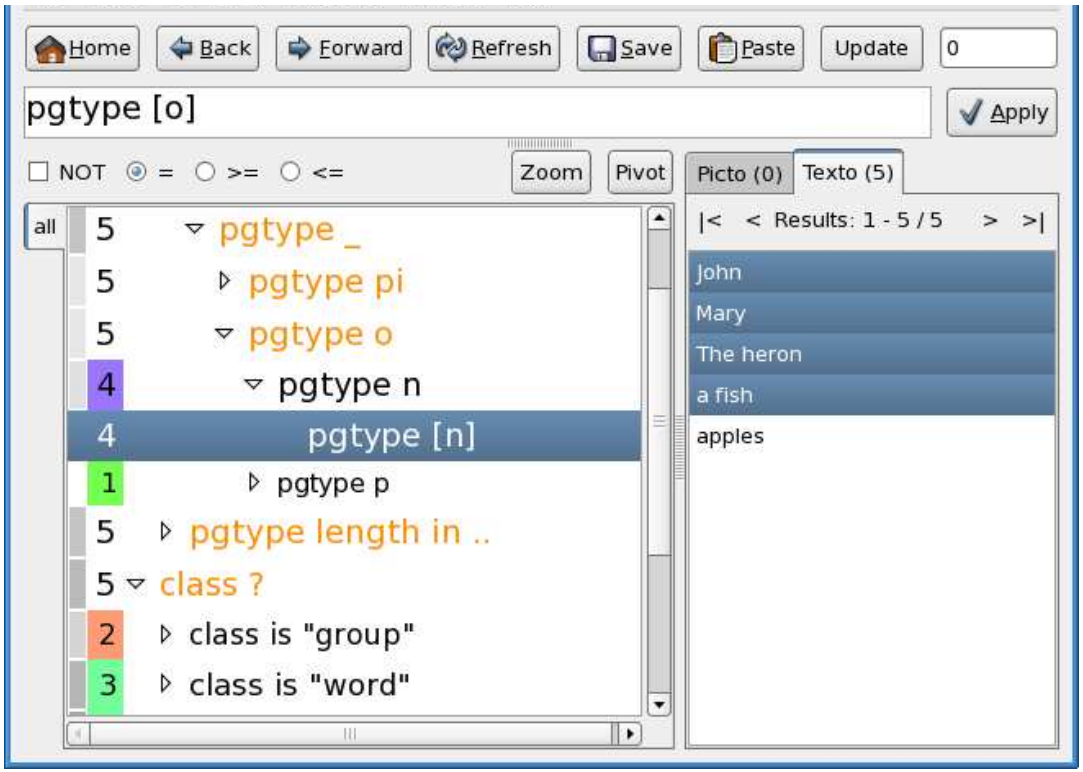
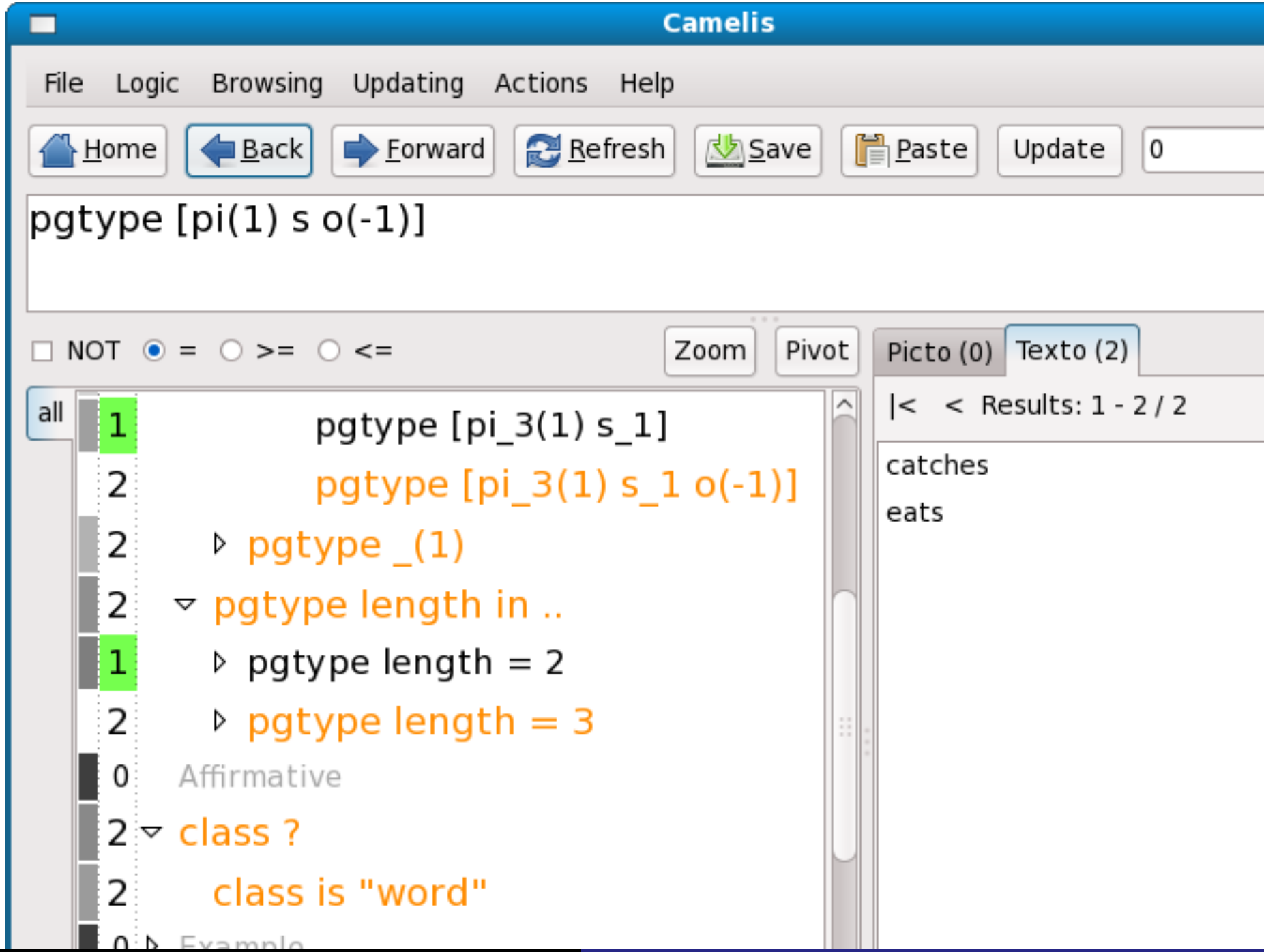
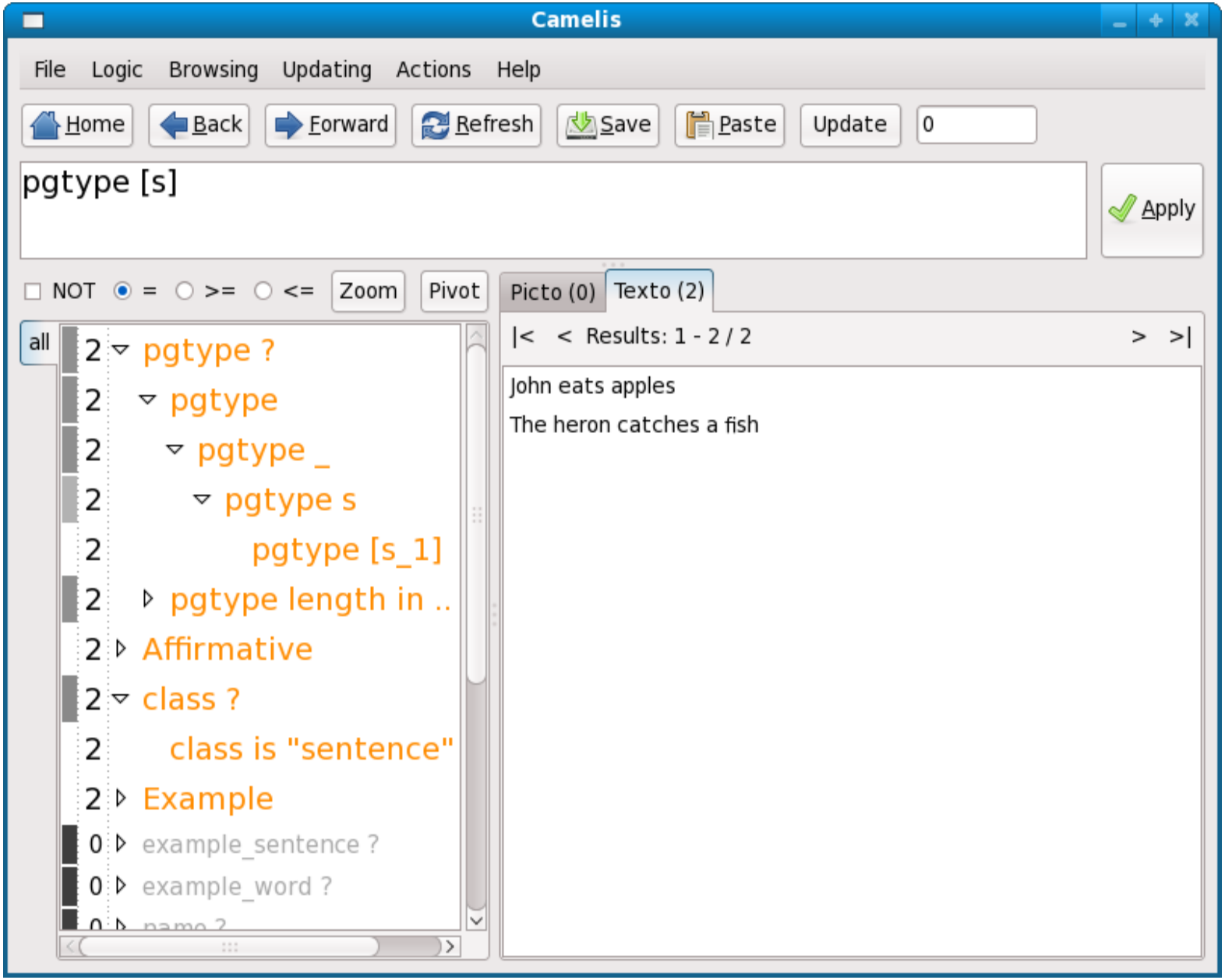


Fig. 3. A toy grammar as a LIS context:
`pgtype [o]` vs `pgtype [n]`
($n \leq o$ in the pregroup grammar)


Camelis
File Logic Browsing Updating Actions Help
Home Back Forward Refresh Save Paste Update 0
pgtype [pi(1) s o(-1)]
NOT = >= <=
Zoom Pivot
Picto (0) Texto (2)
|< < Results: 1 - 2 / 2
catches
eats
all
1 pgtype [pi_3(1) s_1]
2 pgtype [pi_3(1) s_1 o(-1)]
2 pgtype _(1)
2 pgtype length in ..
1 pgtype length = 2
2 pgtype length = 3
0 Affirmative
2 class ?
2 class is "word"
0 Example


Camelis
File Logic Browsing Updating Actions Help
Home Back Forward Refresh Save Paste Update 0
pgtype [s]
Apply
NOT = >= <= Zoom Pivot
Picto (0) Texto (2)
all
2 pgtype ?
2 pgtype
2 pgtype _
2 pgtype s
2 pgtype [s_1]
2 pgtype length in ..
2 Affirmative
2 class ?
2 class is "sentence"
2 Example
0 example_sentence ?
0 example_word ?
|< < Results: 1 - 2 / 2 > >|
John eats apples
The heron catches a fish

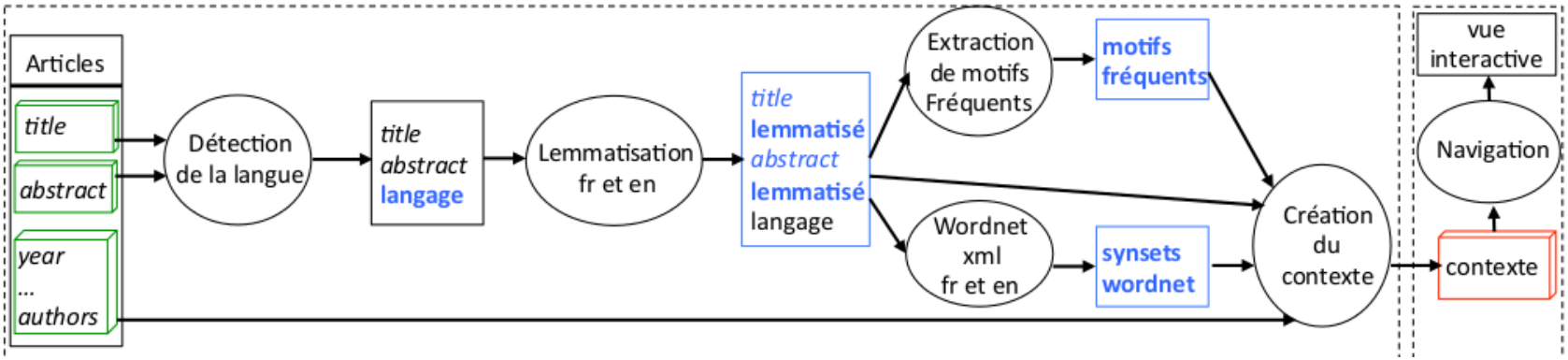
The EGC Challenge - task overview

Data
1 text file extracted by the RNTI editor
1937 lines (1 line = 1 scientific article)
8 fields/line (missing values on some lines)

series	*booktitle*	*year*	*title*	*abstract*	*authors*	*pdf1page*	*pdfarticle*
Revue des Nouvelles Technologies de l'Information	EGC	2015	Qualité et complexité en évaluation des mesures d'intérêt	Remplacer des hypothèses sur le modèle de données par des informations mesurées sur les données réelles est l'une des forces de la fouille de données. […]	Bruno Crémilleux, Arnaud Giacometti, Arnaud Soulet	http://editions-rnti.fr/render_pdf.php?p1&p=1002094	
...							

Data enrichment (after cleaning) with symbolic information,
Browsing facilities by 2 (LIS) tools
Camelis [Ferré09] Sparklis [Ferré14]

Overview

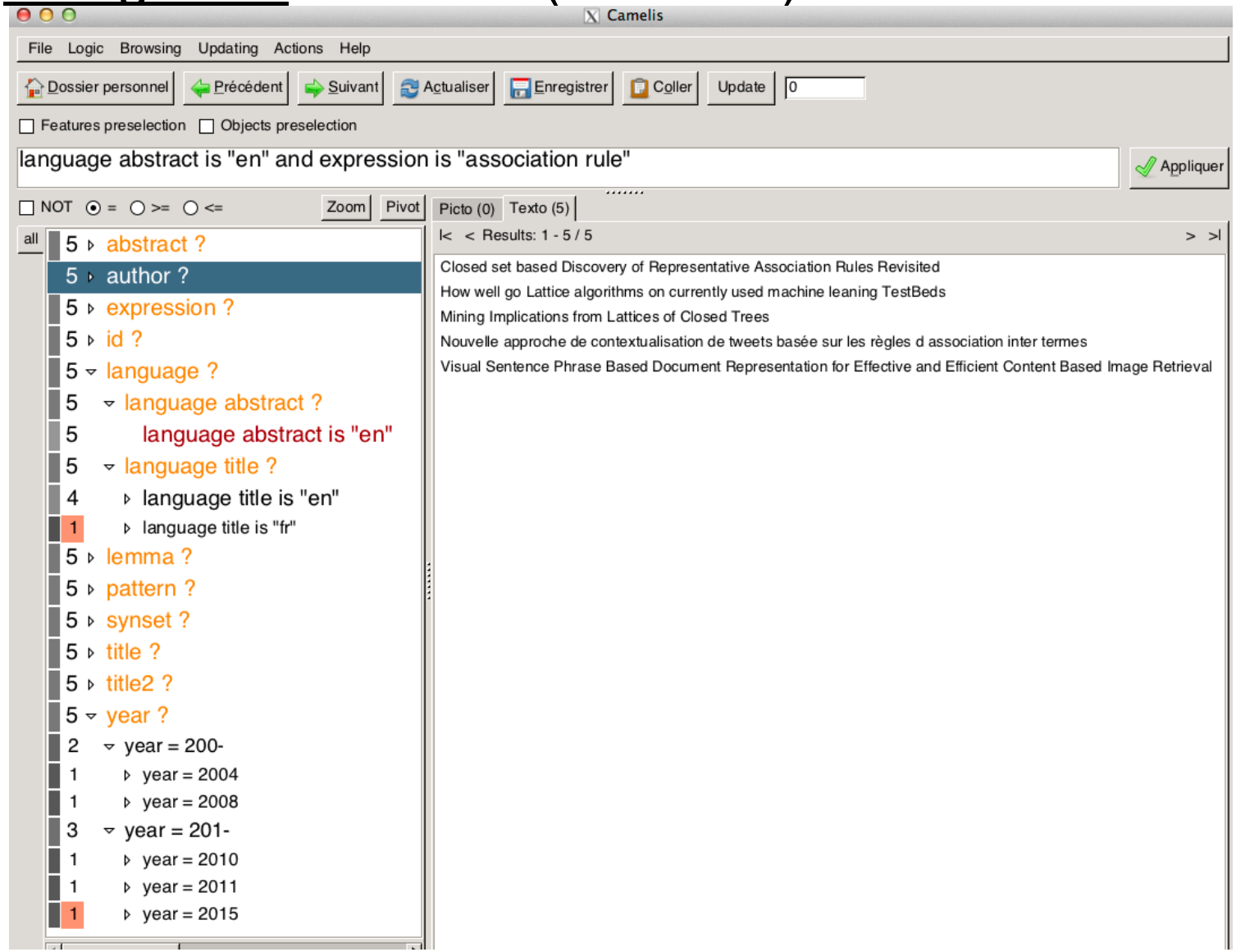


Context = enriched data

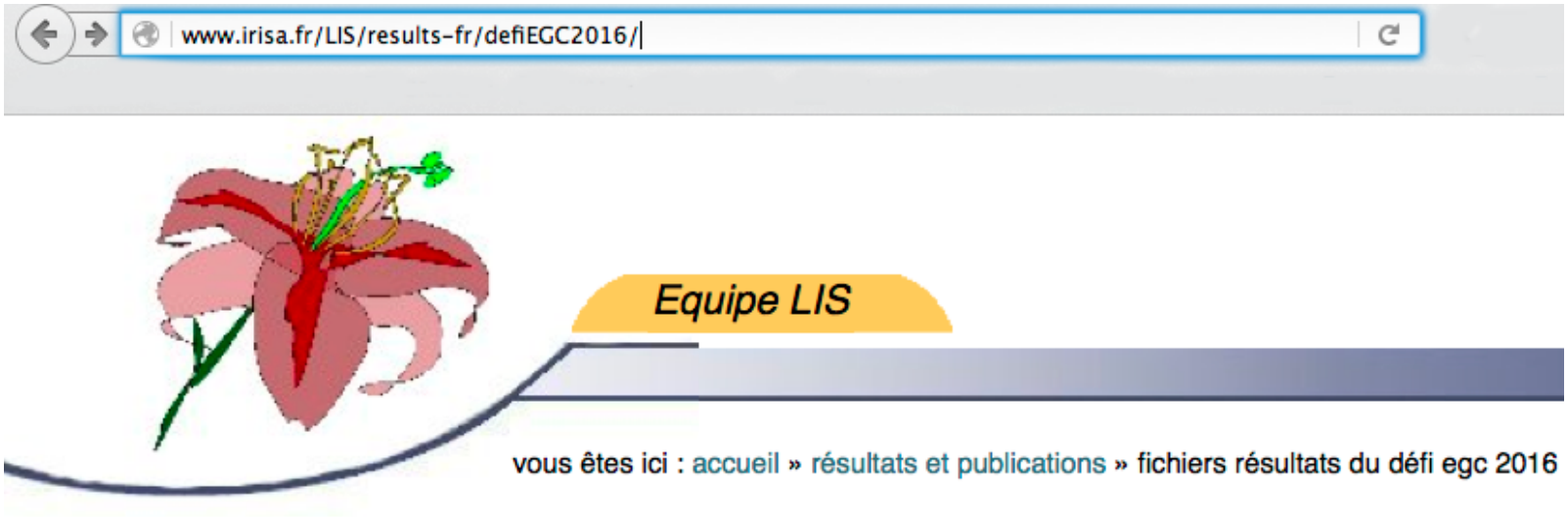
For each article

- *Year*
- *Title*
- *Lemmatized Title*
- *Synsets (Wordnet) on title*
- *Summary*
- *Lemmatized summary*
- *Synsets on summary*
- *Authors*
- *Languages (english, french)*
- *Frequent patterns (extracted keywords)*

Navigation : Camelis (interface)



Access to data and tools



Conclusion and perspectives

Logical concept analysis (LCA) framework and tools could be beneficial for linguistic resource and knowledge management or browsing

LCA methods provide guidance and serendipity (cf experiment)

- to explore the information content : linguistic information, in a flexible way, without a priori knowledge,
- to get data quality feedbacks improvements suggested (in the navigation tree, counts and colors for concepts, ...)
- to ease other treatments, e.g. a selected sub-context can be exported as text and generate a documentation

Extensibility

- by adding objects without confusion between sources (since a property indicating the source is associated with the object) ; e.g. new (compliant) linguistic info. (personal, enterprise, ...)
- by adding properties to expand the browsing possibilities ; e.g. from additional data, Wordnet ... compare, enrich
- by adding triggered actions on objects ; e.g. for linguistic processing parse syntactic head, etc.
- (combined) we could also consider inverse properties (such as translation) and enrich the context with these objects.

Some ongoing work (in the SemLIS team)

more automation (from data to LIS and back)
corpus, treebank (Sequoia, CoNLL, CCG)
programming comments
semantic web
collaborative platform

fundational

Sébastien Ferré and Olivier Ridoux.

2004.

Introduction to logical information systems.

Inf. Process. Manage., 40(3):383–419.

Bernhard Ganter and Rudolf Wille.

1999.

Formal concept analysis - mathematical foundations.

Springer.

Peggy Cellier, Sébastien Ferré, Annie Foret and Olivier Ridoux.

2016

Exploration des Données du Défi EGC 2016 à l'aide d'un Système d'Information Logique.

In Defi EGC 2016.

Annie Foret.

2015

A logical information system proposal for browsing terminological resources.

In Terminology and Artificial Intelligence (TIA) 2015.

Annie Foret, Valérie Bellynck and Christian Boitet

2015

Akenou, un projet de plate-forme valorisant des ressources et outils informatiques et linguistiques pour le breton

In Traitement Automatique des langues régionales (TALARE), TALN workshop, 2015.

Annie Foret and Sébastien Ferré.

2010.

On categorial grammars as logical information systems.

In Léonard Kwuida and Baris Sertkaya, editors, *Formal Concept Analysis, 8th International Conference, ICFCA 2010, Agadir, Morocco, March 15-18, 2010. Proceedings*, volume 5986 of *Lecture Notes in Computer Science*, pages 225–240. Springer.

Remarks

not limited to browsing

- actions : included in this proposal
- updates : partial in this proposal (not yet back and forth)