

# Reconciling Expressivity and Usability in Information Access

from File Systems to the Semantic Web

Sébastien Ferré  
Team LIS, IRISA, Université de Rennes 1

habilitation defense  
6 November 2014, Rennes

INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTEMES ALÉATOIRES



# Motivation

- Information access comes with **more and more**
  - 1 **information** to be accessed
  - 2 **people** to access that information
  - 3 **questions** to ask about that information
- This requires **information systems** that combine
  - 1 **scalability**
  - 2 **usability**
  - 3 **expressivity**

*To which extent can we combine those requirements ?  
What level of trade-off is required ?*

# Overview

- 1 State of the Art
  - Information Access
- 2 Main Contributions
  - A Short History
  - Abstract Conceptual Navigation (ACN)
  - Contributions as Instances of ACN
- 3 Conclusion and Perspectives

# Plan

- 1 State of the Art
  - Information Access
- 2 Main Contributions
- 3 Conclusion and Perspectives

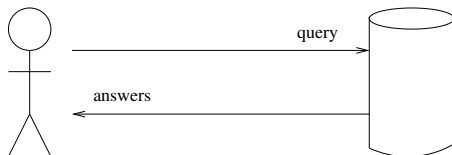
# Information Access: Paradigms

- There exists a **vast** range of approaches
  - ▶ we focus on structured data, like **tables**, **databases**, **XML**
- They can be grouped in 3 paradigms:
  - ▶ **Query Languages (QL)**  
*searching, querying*
  - ▶ **Navigation Structures (NS)**  
*navigating, browsing, exploring*
  - ▶ **Interactive Views (IV)**  
*interacting, dialoging, selecting, transforming*

# Information Access: Query Languages (QL)

## Definition

The user provides some input **query**, and the system returns some **answers** to the query.



- information retrieval (IR), search: keywords, forms
- formal query languages: SQL, XQuery, SPARQL
- natural language interfaces (NLI): IBM Watson

# QL Example: SPARQL

SPARQL is a standard QL for querying **Semantic Web** datasets

- Semantic Web  $\approx$  Web-scale **open** and **structured** database
- SPARQL is the SQL of the Semantic Web

query

```
SELECT ?x ?d WHERE {  
  ?x a dbo:Scientist .  
  ?x dbo:field  
    dbr:Computer_science .  
  ?x dbo:birthDate ?d .  
  FILTER (?d <= 1914-01-01) }  
ORDER BY ASC(?d)
```

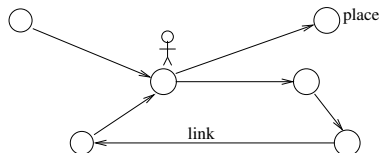
answers (DBpedia)

| ?x                | ?d         |
|-------------------|------------|
| Charles Babbage   | 1791-12-26 |
| Howard H. Aiken   | 1900-03-08 |
| John von Neumann  | 1903-12-28 |
| Konrad Zuse       | 1910-06-22 |
| Franz Alt         | 1910-11-30 |
| K. D. von Neumann | 1911-08-18 |
| Jean Kuntzmann    | 1912-06-01 |
| Alan Turing       | 1912-06-23 |
| ...               | ...        |

# Information Access: Navigation Structures (NS)

## Definition

The user navigates from **place** to place by traversing **links**.

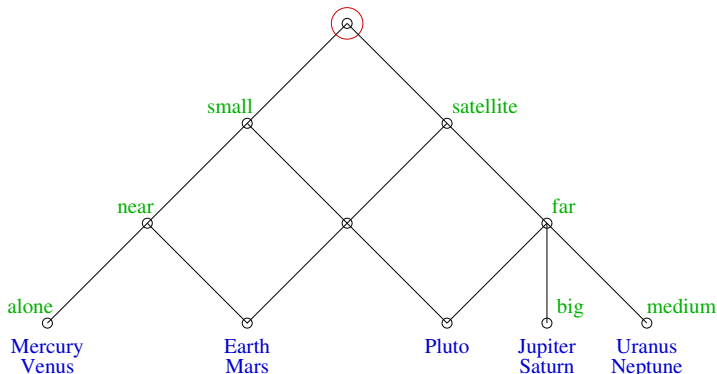


- trees (hierarchies): file systems, directories
- graphs: WWW, Wikipedia, Facebook
- concept lattices: Formal Concept Analysis (FCA) [Wille 82]



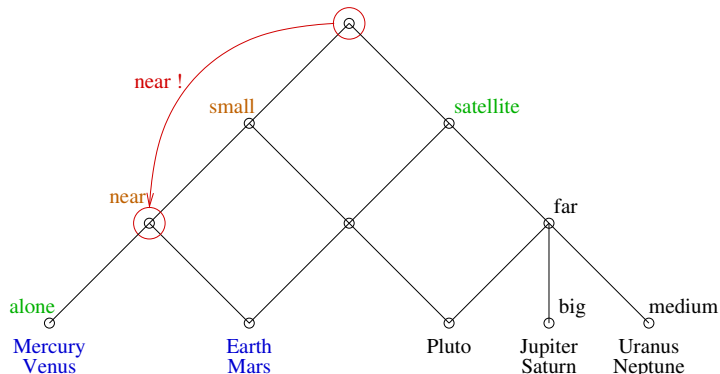
# NS Example: FCA Concept Lattice

- the **concept lattice** is automatically derived from a binary relation between objects and attributes (**formal context**)
- navigation places are concepts, and links are based on generalization ordering



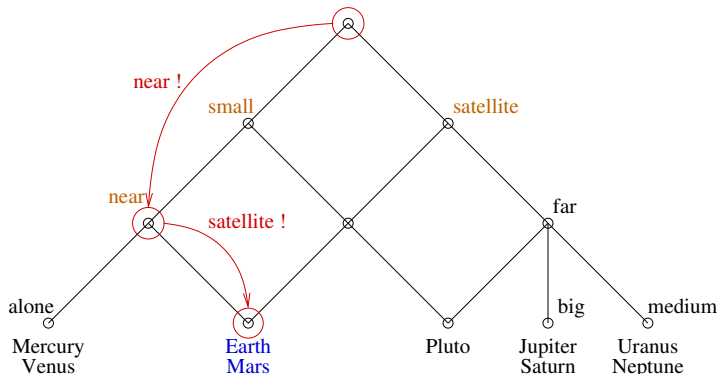
# NS Example: FCA Concept Lattice

- the **concept lattice** is automatically derived from a binary relation between objects and attributes (**formal context**)
- navigation places are concepts, and links are based on generalization ordering



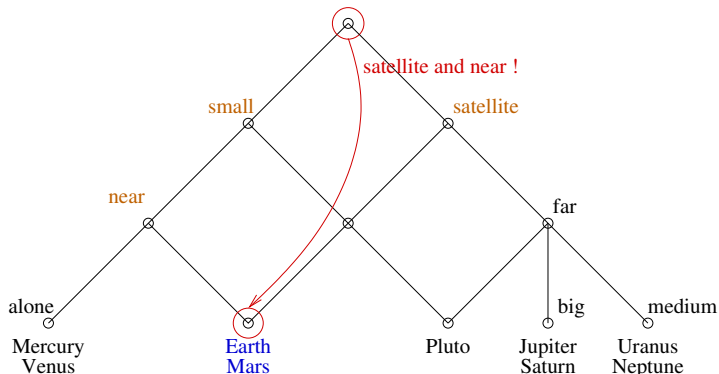
# NS Example: FCA Concept Lattice

- the **concept lattice** is automatically derived from a binary relation between objects and attributes (**formal context**)
- navigation places are concepts, and links are based on generalization ordering



# NS Example: FCA Concept Lattice

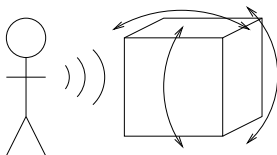
- the **concept lattice** is automatically derived from a binary relation between objects and attributes (**formal context**)
- navigation places are concepts, and links are based on generalization ordering



# Information Access: Interactive Views (IV)

## Definition

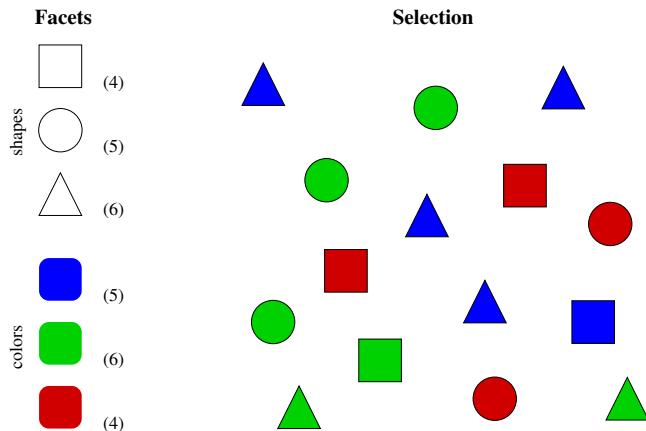
The user **interacts** with the system to define various **views** over data.



- **OLAP**: cubes, analytics, business intelligence [Codd 93]
- **Faceted Search (FS)**: item collections, filtering, e-commerce [Hearst 02]

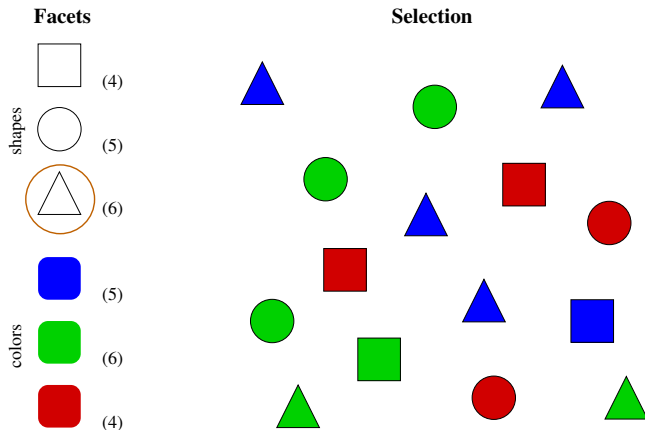
## IV Example: Faceted Search

- a collection of objects can be filtered by facet
- facet values reflect the current selection



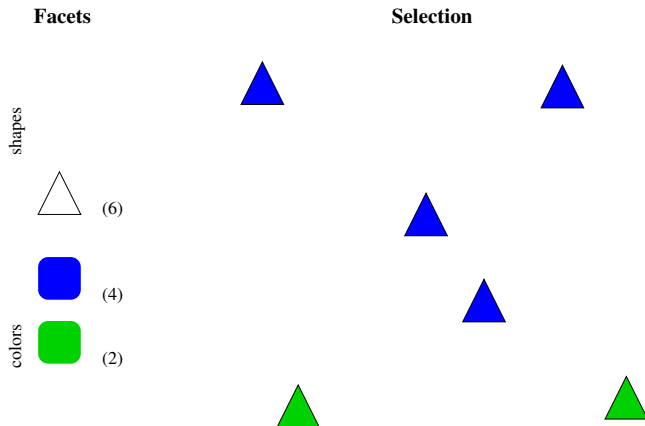
## IV Example: Faceted Search

- a collection of objects can be filtered by facet
- facet values reflect the current selection



## IV Example: Faceted Search

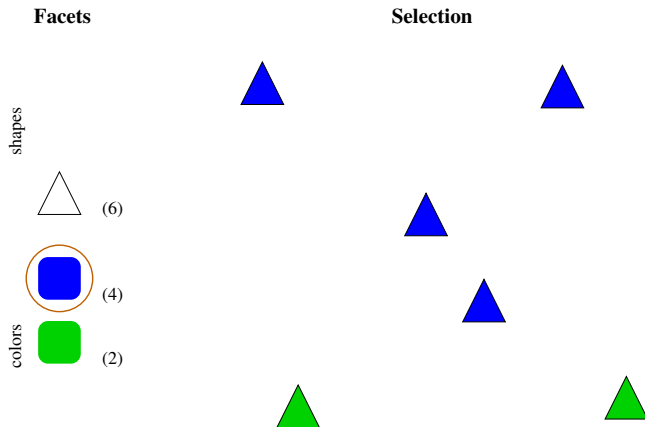
- a collection of objects can be filtered by facet
- facet values reflect the current selection





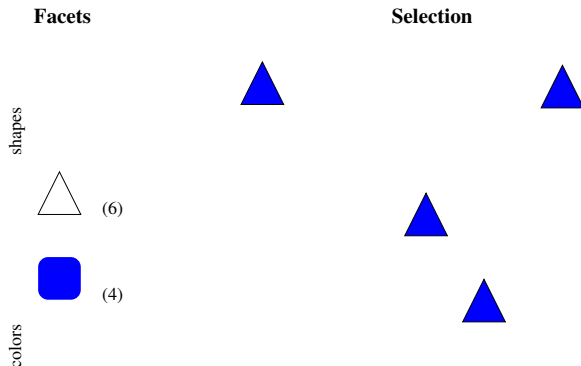
## IV Example: Faceted Search

- a collection of objects can be filtered by facet
- facet values reflect the current selection



## IV Example: Faceted Search

- a collection of objects can be filtered by facet
- facet values reflect the current selection



# Comparison Criteria for Information Access

- Power

- ▶ **expressivity**: *What range of questions can be answered?*
- ▶ **scalability**: *What amount of data can be accessed with acceptable response times?*

- Usability

- ▶ **guidance**: *Are users guided in their search?*
- ▶ **readability**: *How easy/difficult is it for users to understand user interface components and controls?*

# Comparison Criteria for Information Access

## ● Power

- ▶ **expressivity**: *What range of questions can be answered?*
  - ① **atom**: single entity/value as result
  - ② **list**: result lists, set operations (intersection, union, difference)
  - ③ **table**: result tables, data analytics
  - +/- **rel**: relational algebra (join, union, ...)
- ▶ **scalability**: *What amount of data can be accessed with acceptable response times?*
  - ① **Mega**: millions of facts (ex., spreadsheets)
  - ② **Giga**: billions of facts (ex., databases, Wikipedia)
  - ③ **Tera**: trillions of facts (ex., the Web)

## ● Usability

- ▶ **guidance**: *Are users guided in their search?*
- ▶ **readability**: *How easy/difficult is it for users to understand user interface components and controls?*

# Comparison Criteria for Information Access

## ● Power

- ▶ **expressivity**: *What range of questions can be answered?*
- ▶ **scalability**: *What amount of data can be accessed with acceptable response times?*

## ● Usability

- ▶ **guidance**: *Are users guided in their search?*
  - 1 none
  - 2 **schema-based guidance**: no syntactic/lexical error
  - 3 **instance-based guidance**: no empty results
- ▶ **readability**: *How easy/difficult is it for users to understand user interface components and controls?*
  - 1 **artificial**: SQL, SPARQL, technical English
  - 2 **natural**: NL, keywords, simple forms

# Comparison Criteria for Information Access

- Power

- ▶ **expressivity**: *What range of questions can be answered?*
- ▶ **scalability**: *What amount of data can be accessed with acceptable response times?*

- Usability

- ▶ **guidance**: *Are users guided in their search?*
- ▶ **readability**: *How easy/difficult is it for users to understand user interface components and controls?*

- *Other possible criteria*

- ▶ **inference power** (power)
- ▶ **learnability** (usability)
- ▶ **personalization** (usability)

# Best Deals: Strengths and Limits

- QL/Query builders (QB) [SemanticCrystal]
  - ▶ **high expressivity**: table+rel
  - ▶ **low readability**: formal languages
  - ▶ **schema-based guidance**: no immediate feedback, empty results
- QL/Controlled Natural Language (CNL) + auto-completion
  - ▶ good expressivity: list+rel [Ginseng, AceWiki]
  - ▶ good readability: controlled NL
  - ▶ **limited guidance**: like QB + only rightmost expansion of queries
- QL/Natural Language Interfaces (NLI) [Aqualog, CASIA]
  - ▶ high readability: spontaneous NL input
  - ▶ **low effective expressivity**: because NL understanding is hard
- IV/Semantic Faceted Search (SFS) [Ontogator,gFacet,SemFacet]
  - ▶ instance-based guidance: immediate feedback, no empty results
  - ▶ **limited expressivity**: list (set operations) + limited relational algebra

# Best Deals: Strengths and Limits

- QL/Query builders (QB) [SemanticCrystal]
  - ▶ **high expressivity**: table+rel
  - ▶ **low readability**: formal languages
  - ▶ **schema-based guidance**: no immediate feedback, empty results
- QL/Controlled Natural Language (CNL) + auto-completion [Ginseng, AceWiki]
  - ▶ **good expressivity**: list+rel
  - ▶ **good readability**: controlled NL
  - ▶ **limited guidance**: like QB + only rightmost expansion of queries
- QL/Natural Language Interfaces (NLI) [Aqualog, CASIA]
  - ▶ **high readability**: spontaneous NL input
  - ▶ **low effective expressivity**: because NL understanding is hard
- IV/Semantic Faceted Search (SFS) [Ontogator,gFacet,SemFacet]
  - ▶ **instance-based guidance**: immediate feedback, no empty results
  - ▶ **limited expressivity**: list (set operations) + limited relational algebra



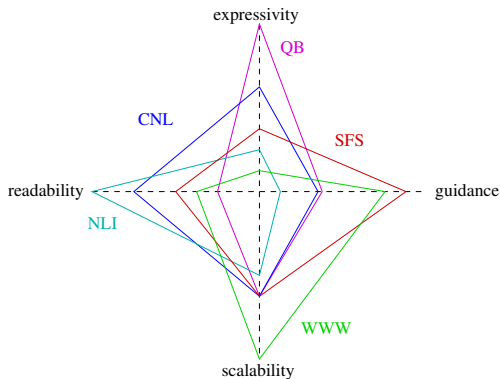
# Best Deals: Strengths and Limits

- QL/Query builders (QB) [SemanticCrystal]
  - ▶ **high expressivity**: table+rel
  - ▶ **low readability**: formal languages
  - ▶ **schema-based guidance**: no immediate feedback, empty results
- QL/Controlled Natural Language (CNL) + auto-completion [Ginseng, AceWiki]
  - ▶ **good expressivity**: list+rel
  - ▶ **good readability**: controlled NL
  - ▶ **limited guidance**: like QB + only rightmost expansion of queries
- QL/Natural Language Interfaces (NLI) [Aqualog, CASIA]
  - ▶ **high readability**: spontaneous NL input
  - ▶ **low effective expressivity**: because NL understanding is hard
- IV/Semantic Faceted Search (SFS) [Ontogator,gFacet,SemFacet]
  - ▶ **instance-based guidance**: immediate feedback, no empty results
  - ▶ **limited expressivity**: list (set operations) + limited relational algebra

# Best Deals: Strengths and Limits

- QL/Query builders (QB) [SemanticCrystal]
  - ▶ **high expressivity**: table+rel
  - ▶ **low readability**: formal languages
  - ▶ **schema-based guidance**: no immediate feedback, empty results
- QL/Controlled Natural Language (CNL) + auto-completion [Ginseng, AceWiki]
  - ▶ **good expressivity**: list+rel
  - ▶ **good readability**: controlled NL
  - ▶ **limited guidance**: like QB + only rightmost expansion of queries
- QL/Natural Language Interfaces (NLI) [Aqualog, CASIA]
  - ▶ **high readability**: spontaneous NL input
  - ▶ **low effective expressivity**: because NL understanding is hard
- IV/Semantic Faceted Search (SFS) [Ontogator,gFacet,SemFacet]
  - ▶ **instance-based guidance**: immediate feedback, no empty results
  - ▶ **limited expressivity**: list (set operations) + limited relational algebra

# Comparing Information Access Paradigms



- negative correlation between **usability** and **expressivity**
  - ▶ *expressivity is lost in usability !*
- CNL and SFS are the more balanced

# Plan

1 State of the Art

2 Main Contributions

- A Short History
- Abstract Conceptual Navigation (ACN)
- Contributions as Instances of ACN

3 Conclusion and Perspectives

# Research Methodology

Agile methodology with relatively short cycles:

- ① **theories** formalizing my approach
  - ▶ to precisely **compare** with different approaches
  - ▶ to **prove** desired properties rather than to rely on test only
    - ★ e.g., **expressivity level, guidance safeness and completeness**
- ② **softwares** implementing the theories
  - ▶ to **test and refine** theories
  - ▶ to enable **practical** use and evaluation
- ③ **applications** to real data and use cases
  - ▶ to quickly **identify limits** of my approach

# A Short History and Main Contributions

from File Systems to the Semantic Web

- 1999** conceptual **navigation** (FCA) + logic **expressivity** [Ridoux]  
 ▶ Logical Concept Analysis (LCA) [ICCS'00, IP&M'04]  
 ▶ CAMELIS on **files, photos, software** [Padioleau]
- 2007** + faceted search (FS) **usability** [Sacco, Tzitzikas]  
 ▶ contribution to the **formalization of FS** [FIND'07-08, IJGS'09, book]  
 ▶ GEOLIS, ABILIS on **geographical data** [Quesseveur, Bedel, Allard]
- 2009** + SPARQL **expressivity** (relational data)  
 ▶ Query-based Faceted Search (QFS) [ISWC'11, IJMSO'12]  
 ▶ SEWELIS on **genealogy, comics** [Hermann, Guérin]
- 2012** + CNL **readability**  
 ▶ SQUALL: a CNL syntax for SPARQL 1.1 [CNL'12, NLDB'13, DKE'14]  
 ▶ SQUALL2SPARQL on **DBpedia** [QALD challenge]
- 2013** + **scalability** and **portability** [Guyonvarc'h, Ducasse]  
 ▶ QFS on top of SPARQL endpoints [ISWC'14]  
 ▶ SPARKLIS on **DBpedia, bioinformatics** [GenOuest]

# Abstract Conceptual Navigation (ACN)

a generic framework to reconcile expressivity and usability

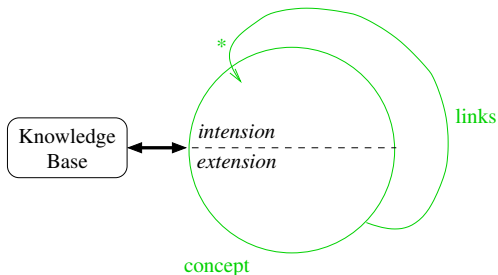
ACN merges the three information access paradigms (QL, NS, IV):

Knowledge  
Base

# Abstract Conceptual Navigation (ACN)

a generic framework to reconcile expressivity and usability

ACN merges the three information access paradigms (QL, NS, IV):

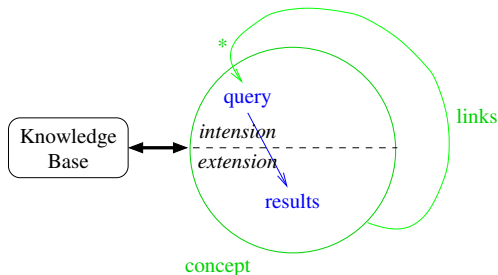




# Abstract Conceptual Navigation (ACN)

a generic framework to reconcile expressivity and usability

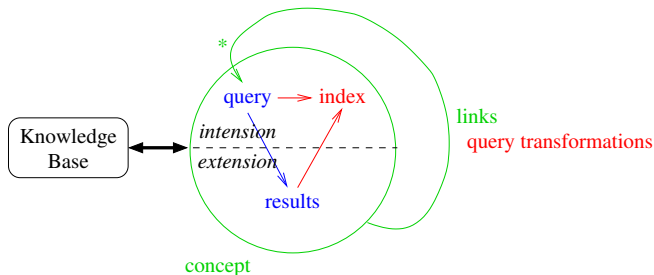
ACN merges the three information access paradigms (QL, NS, IV):



# Abstract Conceptual Navigation (ACN)

a generic framework to reconcile expressivity and usability

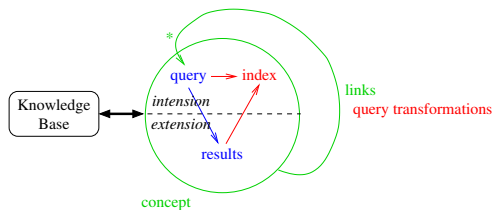
ACN merges the three information access paradigms (QL, NS, IV):



## Abstract Conceptual Navigation (ACN)

a generic framework to reconcile expressivity and usability

ACN merges the three information access paradigms (QL, NS, IV):



### Difficulties arising from combining queries and navigation:

- **safe guidance**: avoiding empty results (instance-based guidance)
- **complete guidance**: reaching all queries (effective expressivity)
- **readability** of the query and index
- **scalability** of index computation

# ACN Definitions

Abstract components, and their **instanciation for LCA**:

- **knowledge base** ( $K$ ): data, facts, rules, domain knowledge, etc.
  - ▶ in LCA: logical context  $K = (\mathcal{O}, \mathcal{L}, d)$ 
    - ★  $\mathcal{O}$ : collection of **objects**
    - ★  $\mathcal{L} = (L, \sqsubseteq)$ : **logic** = partial order of object descriptors
    - ★  $d \in \mathcal{O} \rightarrow \mathcal{L}$ : **description** function
    - ★ ex:  $d(o) = \text{date} = 6 \text{ nov } 2014, \text{ location is IRISA, title is "habilitation defense"}$
- **query language** ( $Q$ ): expressible queries
- **extensions** ( $E$  and  $ext \in Q \rightarrow E$ ): query results
- **indexes** ( $I$  and  $index \in Q \times E \rightarrow I$ ): suggestions, feedback
- **links** ( $links \in Q \times E \times I \rightarrow 2^Q$ ): navigation links

# ACN Definitions

Abstract components, and their **instanciation for LCA**:

- **knowledge base** ( $K$ ): data, facts, rules, domain knowledge, etc.
- **query language** ( $Q$ ): expressible queries
  - ▶ in LCA: Boolean closure of  $\mathcal{L}$  (and, or, not)
  - ▶ ex:  $q = \text{title matches "defense" and (location is Rennes or location is Lannion)}$
- **extensions** ( $E$  and  $\text{ext} \in Q \rightarrow E$ ): query results
- **indexes** ( $I$  and  $\text{index} \in Q \times E \rightarrow I$ ): suggestions, feedback
- **links** ( $\text{links} \in Q \times E \times I \rightarrow 2^Q$ ): navigation links

# ACN Definitions

Abstract components, and their **instanciation for LCA**:

- **knowledge base** ( $K$ ): data, facts, rules, domain knowledge, etc.
- **query language** ( $Q$ ): expressible queries
- **extensions** ( $E$  and  $ext \in Q \rightarrow E$ ): query results
  - ▶ in LCA: sets of objects  $E = 2^{\mathcal{O}}$
  - ▶ in LCA: inductive definition over queries

$$\begin{aligned}
 ext(q \in \mathcal{L}) &= \{o \in \mathcal{O} \mid d(o) \sqsubseteq q\} \\
 ext(q_1 \text{ and } q_2) &= ext(q_1) \cap ext(q_2) \\
 ext(q_1 \text{ or } q_2) &= ext(q_1) \cup ext(q_2) \\
 ext(\text{not } q_1) &= \mathcal{O} \setminus ext(q_1)
 \end{aligned}$$

- **indexes** ( $I$  and  $index \in Q \times E \rightarrow I$ ): suggestions, feedback
- **links** ( $links \in Q \times E \times I \rightarrow 2^Q$ ): navigation links

# ACN Definitions

Abstract components, and their **instanciation for LCA**:

- **knowledge base** ( $K$ ): data, facts, rules, domain knowledge, etc.
- **query language** ( $Q$ ): expressible queries
- **extensions** ( $E$  and  $ext \in Q \rightarrow E$ ): query results
- **indexes** ( $I$  and  $index \in Q \times E \rightarrow I$ ): suggestions, feedback
  - ▶ in LCA: **frequency of logical features** ( $X \subseteq \mathcal{L}$ ) over an extension
  - ▶ in LCA:  $I = X \rightarrow \mathbb{N}$
  - ▶ in LCA:  $index(q, e) = \{x \mapsto n \mid x \in X, n = \#(e \cap ext(x))\}$
  - ▶ ex: { date in 2013  $\mapsto$  12,  
       date in 2014  $\mapsto$  15,  
       title contains "habilitation"  $\mapsto$  5, ... }
- **links** ( $links \in Q \times E \times I \rightarrow 2^Q$ ): navigation links

# ACN Definitions

Abstract components, and their **instanciation for LCA**:

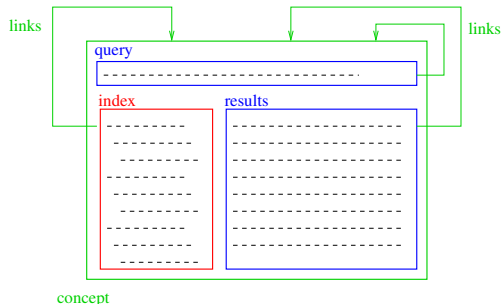
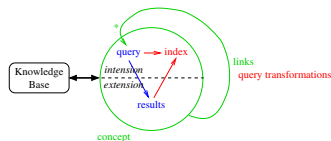
- **knowledge base** ( $K$ ): data, facts, rules, domain knowledge, etc.
- **query language** ( $Q$ ): expressible queries
- **extensions** ( $E$  and  $ext \in Q \rightarrow E$ ): query results
- **indexes** ( $I$  and  $index \in Q \times E \rightarrow I$ ): suggestions, feedback
- **links** ( $links \in Q \times E \times I \rightarrow 2^Q$ ): navigation links
  - ▶ in  $links(q, e, i)$ , we assume  $e = ext(q)$ , and  $i = index(q, e)$
  - ▶ in LCA: different kinds of moves in concept lattice (query transformations)
    - ★ **downward**: adding a feature, replacing more general features  
 $France \rightsquigarrow France \text{ and } \mathbf{Building}$
    - ★ **upward**: removing a feature, or replacing it with a more general one  
 $France \text{ and } Building \rightsquigarrow \mathbf{Europe} \text{ and } Building$
    - ★ **sideward**: downward+upward or upward+downward  
 $Europe \text{ and } Building \rightsquigarrow Europe \text{ and } \mathbf{Landscape}$



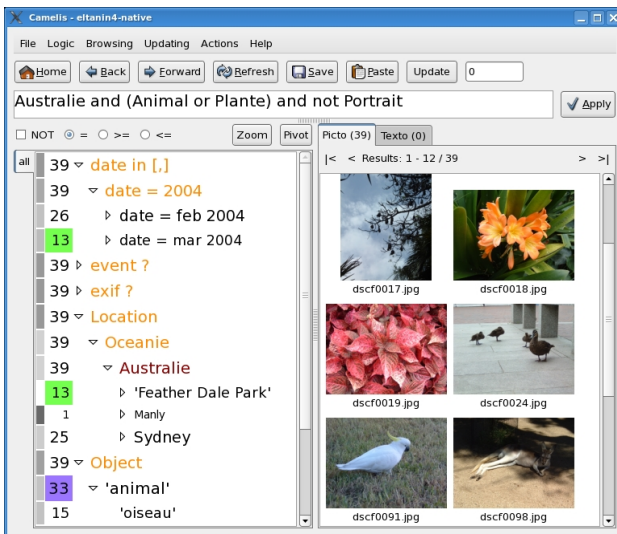
# ACN from the User Point of View

## Natural mapping of ACN to a User Interface (UI)

- ACN can be used as a **UI design pattern**
- UI widgets: **query**, **results**, **index**
- UI controls: **navigation links**
  - ▶ e.g., selection of index elements, buttons for applying query transformations



# CAMELIS screenshot: illustration on photos



# Other ACN Instances

- 1 Cubes of Concepts
- 2 Query-based Faceted Search (QFS)
- 3 Update Through Interaction (UTILIS)
- 4 Possible World Explorer (PEW)

# Instance 1: Cubes of Concepts

[PhD Allard, Ridoux, 2008-11]

- objective: OLAP-like analytics in LCA
  - ▶ example question: total sale per product and per year
  - ▶ expected result: a 2D table of numbers with products as rows, and years as columns (OLAP cube)
- contributions: [ICFCA'12]
  - ▶ in LCA: cubes of concepts as navigation places (ACN query), and cubes of values as results (ACN extension)
  - ▶ in OLAP: multi-valued attributes (ex., paper authors)
  - ▶ in OLAP: all attributes can be used for selection (query), as dimension, as measure, and all at once
- implementation: in ABILIS (Web version of CAMELIS)
- applications:
  - ▶ interactive discovery of functional dependencies and association rules [EGC'10,CLA'10]
  - ▶ group decision: publication plan of a research team [CLA'11]

# Instance 1: Cubes of Concepts

[PhD Allard, Ridoux, 2008-11]

- objective: OLAP-like analytics in LCA
  - ▶ example question: total sale per product and per year
  - ▶ expected result: a 2D table of numbers with products as rows, and years as columns (OLAP cube)
- contributions: [ICFCA'12]
  - ▶ in LCA: cubes of concepts as navigation places (ACN query), and cubes of values as results (ACN extension)
  - ▶ in OLAP: multi-valued attributes (ex., paper authors)
  - ▶ in OLAP: all attributes can be used for selection (query), as dimension, as measure, and all at once
- implementation: in ABILIS (Web version of CAMELIS)
- applications:
  - ▶ interactive discovery of functional dependencies and association rules [EGC'10,CLA'10]
  - ▶ group decision: publication plan of a research team [CLA'11]

## Instance 2: Query-based Faceted Search (QFS)

- objective: handle **relationships** between objects
  - ▶ LCA has only unary predicates
  - ▶ **n-ary predicates** are key to the expressivity of SQL/SPARQL
- difficulty: the “linguistic subject” of the query becomes ambiguous
  - ▶ ex: *person*( $x$ )  $\wedge$  *father*( $x, y$ )
  - ▶ *Is the query about the person  $x$  or the father  $y$  ?*
- contributions: [ICFGA'10, ISWC'11, IJMSO'12]
  - ▶ query focus: specifies the current “linguistic subject” (**ACN query**)
    - ★ determines **ACN extension** and **ACN index**
    - ★ can be changed, serves as an insertion point (**ACN links**)
  - ▶ guidance proved safe and complete over a large fragment of SPARQL (graph patterns including cycles, union, negation)
- implementation: SEWELIS, as an evolution of CAMELIS
  - ▶ applied to genealogy, films, ...

## Instance 2: Query-based Faceted Search (QFS)

- objective: handle **relationships** between objects
  - ▶ LCA has only unary predicates
  - ▶ **n-ary predicates** are key to the expressivity of SQL/SPARQL
- difficulty: the “linguistic subject” of the query becomes ambiguous
  - ▶ ex: *person*( $x$ )  $\wedge$  *father*( $x, y$ )
  - ▶ *Is the query about the person  $x$  or the father  $y$  ?*
- contributions: [ICFCA'10, ISWC'11, IJMSO'12]
  - ▶ **query focus**: specifies the current “linguistic subject” (**ACN query**)
    - ★ determines **ACN extension** and **ACN index**
    - ★ can be changed, serves as an **insertion point** (**ACN links**)
  - ▶ guidance **proved safe and complete** over a large fragment of SPARQL (graph patterns including cycles, union, negation)
- implementation: **SEWELIS**, as an evolution of CAMELIS
  - ▶ applied to **genealogy**, **films**, ...

# Instance 3: Update Through Interaction (UTILIS)

[PhD Hermann, Ducassé, 2009-12]

- objective: QFS-like instance-based guidance for **updating RDF graphs**
  - ▶ because writing **RDF descriptions** is as difficult as writing SPARQL queries for users
  - ▶ idea: use description as query, and results as models
- difficulty: empty results because most objects are unique
- solution: query relaxation to find similar objects [KCAP'11,EKAW'12]
  - ▶ based on edit distance and dynamic programming (**ACN extension**)
- implemented in SEWELIS, reusing the same UI
- applied and evaluated on describing comics panels (characters, bubbles, ...)
  - ▶ UTILIS enables more consistency and reuse than PROTÉGÉ



# Instance 3: Update Through Interaction (UTILIS)

[PhD Hermann, Ducassé, 2009-12]

- objective: QFS-like instance-based guidance for **updating RDF graphs**
  - ▶ because writing **RDF descriptions** is as difficult as writing SPARQL queries for users
  - ▶ idea: use description as query, and results as models
- difficulty: empty results because most objects are unique
- solution: **query relaxation** to find **similar objects**  
[KCAP'11,EKAW'12]
  - ▶ based on **edit distance** and **dynamic programming** (**ACN extension**)
- implemented in SEWELIS, reusing the same UI
- applied and evaluated on describing **comics panels** (characters, bubbles, ...)
  - ▶ UTILIS enables more **consistency** and reuse than PROTÉGÉ

# Instance 4: Possible World Explorer (PEW)

[Rudolph's visit, 2012]

- objective: explore and complete **ontologies**
  - ▶ understanding in terms of **possible worlds**
    - ★ ex: a pizza must have a base and toppings
  - ▶ adding **missing axioms** to exclude unexpected worlds
    - ★ ex: a pizza can also be a country!
- difficulty on the notion of **object**
  - ▶ ontologies often miss completely concrete objects
  - ▶ there is an **infinite set of worlds**
- solution: use a reasoner instead of a query evaluator [EKAW'12]
  - ▶ **ACN extension** = existence of possible worlds (satisfiability)
  - ▶ guidance proved safe and complete for simple OWL class expressions (**ACN queries**)
- implementation: PEW, adapted from SEWELIS
- applied to the pizza ontology
  - ▶ We found that Veggie pizza can contain meat or fish !

# Instance 4: Possible World Explorer (PEW)

[Rudolph's visit, 2012]

- objective: explore and complete **ontologies**
  - ▶ understanding in terms of **possible worlds**
    - ★ ex: a pizza must have a base and toppings
  - ▶ adding **missing axioms** to exclude unexpected worlds
    - ★ ex: a pizza can also be a country!
- difficulty on the notion of **object**
  - ▶ ontologies often miss completely concrete objects
  - ▶ there is an **infinite set of worlds**
- solution: use a **reasoner** instead of a query evaluator [EKAW'12]
  - ▶ **ACN extension** = existence of possible worlds (**satisfiability**)
  - ▶ guidance **proved safe and complete** for **simple OWL class expressions** (**ACN queries**)
- implementation: PEW, adapted from SEWELIS
- applied to the **pizza ontology**
  - ▶ We found that **Veggie pizza can contain meat or fish !**

# Applications

- file systems RIDOUX, PADIOLEAU [CL'00, *USENIX'03*]
- software engineering
  - ▶ retrieval of **software components** by type  
SIGONNEAU, RIDOUX [*MSR'06*, *TSI'06*]
  - ▶ **fault localization** with data mining  
CELLIER, DUCASSÉ, RIDOUX [ICFCA'08, *SEKE'09'11*]
- geographical information systems  
BEDEL, ALLARD, RIDOUX, QUESSEVEUR [*SAGEO'06*, ICFCA'08, *RIG'08*]
- linguistic data
  - ▶ **lexicalized grammars** FORET [ICFCA'10, *ICJ'10*]
  - ▶ mined **linguistic patterns** CELLIER, CHARNOIS [ICCS'11, *CICLing'12*]
- group decision and negotiation DUCASSÉ, CELLIER [ICCS'08, *GDN'14*]
- bioinformatics
  - ▶ LCA-based machine learning KING [FI'05, *JCB'06*]
  - ▶ workflow composition BA, DUCASSÉ [ESWC'13, *DEXA'14*]
- comics HERMANN, DUCASSÉ [EKAW'12, *Guérin'14*]

\* *I am not an author of references in italic.*

# Collaborations

- co-supervised PhD students (by defense year)

2015 Mouhamadou Ba (Ducassé): guided composition of workflows

2012 Alice Hermann (Ducassé): update through interaction

2011 Pierre Allard (Ridoux): cubes of concepts

2009 Olivier Bedel (Ridoux): geographical information systems

2008 Peggy Cellier (Ducassé, Ridoux): fault localization in programs

- Academics

2011-14 MUMIA COST Action: multifaceted interactive information access

2012 S. Rudolph (Karlsruhe, now Dresden): Possible World Explorer

2011 C. Guérin, K. Bertet (La Rochelle): comics in SEWELIS

2011 L. Spagnolo (Milano): user interaction

2005- E. Quesseveur (Rennes 2): geographical information systems

2002-04 R.D. King (post-doc in Wales): bioinformatics

- Industrials

- ▶ ARIADNEXT: ANR project IDFRAud on ID documents

- ▶ MEDIADONE: enriched interactive Web TV

# Plan

- 1 State of the Art
- 2 Main Contributions
- 3 Conclusion and Perspectives**

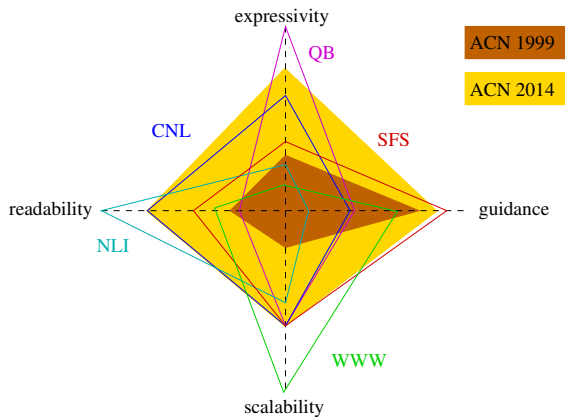
# Conclusion

## Achievement and Limits (Semantic Web)

- **Expressivity:** covers a large part of SPARQL 1.1
  - ▶ cyclic graph patterns, UNION/NOT/OPT, aggregations, ordering
    - ★ completeness theorem: every covered query is reachable
    - ★ the query focus is a key element
  - ▶ missing: expressions, transitive property paths, reified graphs
- **Usability:**
  - ▶ similar guidance to faceted search
    - ★ safeness theorem: no empty results
  - ▶ similar readability to controlled NL
  - ▶ user studies performed on CS students
    - ★ complex queries OK, except scope of negation and co-references
    - ★ users quickly improve in correctness and response time
    - ★ need for more user studies, in particular non-IT people
- **Scalability:**
  - ▶ same complexity as query languages for computing results
  - ▶ same complexity as faceted search for computing the index
  - ▶ responsive on DBpedia (more than 2G triples), not Web-scale

# Conclusion

ACN evolution from 1999 to 2014



- ACN combines the **strengths of SFS and CNL**
- ACN has room for improvement on each criteria
- *There are **other criteria** that contribute to power and usability!*



# Perspectives w.r.t. Expressivity

- **Computations**: SPARQL expressions, and nested aggregations
  - ▶ for more powerful **data analytics**
    - ★ ex: the median per social category of the average income per capita in households
  - ▶ to cover **spreadsheet** features: e.g., formula cells
- **Exploring the immaterial**: when there are no concrete objects
  - ▶ models of a logical theory (e.g. PEW)
  - ▶ frequent patterns over a dataset (data-mining)
  - ▶ solutions of a set of constraints
- **Scientific workflows**: guided composition of tasks
  - ▶ *PhD Mouhamadou Ba (Ducassé), 2012-2015*
  - ▶ queries are **programs**, and results are **static evaluations** (e.g., typing)
  - ▶ application to **bioinformatics**

# Perspectives w.r.t. Usability

- **Visualization**: for a better **readability**
  - ▶ in both results and the index
    - ★ ex: **charts**, **timelines**, **maps**
  - ▶ Allard's PhD provides some initial ideas and results
- **Ontology lexicons**: for a better NL verbalization
  - ▶ **lexicons**: mappings between URIs and nouns/verbs/adjectives...
  - ▶ can be defined manually or learned from a corpus
- **Intelligent guidance**: for a better exploration
  - ▶ guidance **personalization** based on past choices
  - ▶ **multi-step** guidance based on planning and strategies
- **Dynamic refactoring**: for a better collaborative authoring
  - ▶ in collaborative bottom-up representation of **shared knowledge**
  - ▶ to avoid **irreversible** decisions and **divergent** representations

# Motivating Application: MEMOLIS [LIS team]

## Collaborative **Dynamic** and **Semantic** Information Systems

- in the line of Memex [Bush 1945] or MyLifeBits [Gemmell et al. 2006]
- features:
  - ▶ **Semantic** Collective Memory (all in **RDF**)
  - ▶ User-centered search, exploration, analytics, data mining + **sharing**
  - ▶ **Dynamic** behaviour based on **reactive workflows**
  - ▶ **Collaborative** authoring of facts, rules, workflows + curation & harmonization + **group decision**
- users: teams, organizations, individuals
- application: knowledge capitalization, lifelogging, data analytics, introspection, decision making, anticipation, collaboration

# The End

Thank you for your attention!

*Thanks to all former and present members of the LIS team for their collaboration on many aspects of this work.*

- *colleagues: Olivier RIDOUX, Mireille DUCASSÉ, Annie FORET, Peggy CELLIER, Yves BEKKERS, Véronique ABILY*
- *PhD students: Yoann PADIOLEAU, Benjamin SIGONNEAU, Olivier BEDEL, Pierre ALLARD, Alice HERMANN, Mouhamadou BA, Soda CISSÉ*
- *MSc students: Soazig BARS, Étienne ANDRÉ, Joris GUYONVARC'H*
- *visitors: Sebastian RUDOLPH, Luigi SPAGNOLO, Clément GUÉRIN*

Questions?

# Main Publications by Topic

## Concept Analysis

- S. Ferré, P. Allard, and O. Ridoux. **Cubes of Concepts: Multi-dimensional Exploration of Multi-valued Contexts**. *Int. Conf. Formal Concept Analysis (ICFCA)*, 2012. Springer.
- S. Ferré. **Camelis: a logical information system to organize and browse a collection of documents**. *Int. J. General Systems*, 2009.
- P. Cellier, S. Ferré, O. Ridoux and M. Ducassé. **A Parameterized Algorithm to Explore Formal Contexts with a Taxonomy**. *Int. J. Foundations of Computer Science (IJFCS)*, 2008.
- S. Ferré and O. Ridoux. **An Introduction to Logical Information Systems**. *Information Processing & Management*, 2004.

## Faceted Search

- (5 chapters in) G. M. Sacco and Y. Tzitzikas (Eds.). **Dynamic Taxonomies and Faceted Search: Theory, Practice, and Experience**, volume 25 of *The Information Retrieval Series*. Springer, 2009.

## Semantic Web

- S. Ferré. **Expressive and Scalable Query-Based Faceted Search over SPARQL Endpoints**. *Int. Semantic Web Conf., LNCS*, pages 438-453, 2014. Springer.
- S. Ferré and Alice Hermann. **Reconciling faceted search and query languages for the Semantic Web**. *Int. J. Metadata, Semantics and Ontologies*, 7(1):37-54, 2012.
- S. Ferré and Sebastian Rudolph. **Advocatus Diaboli - Exploratory Enrichment of Ontologies with Negative Constraints**. In A. ten Teije et al., editor, *Int. Conf. Knowledge Engineering and Knowledge Management (EKAW)*, LNAI 7603, pages 42-56, 2012. Springer.
- A. Hermann, S. Ferré, and Mireille Ducassé. **An Interactive Guidance Process Supporting Consistent Updates of RDFS Graphs**. In A. ten Teije et al., editor, *Int. Conf. Knowledge Engineering and Knowledge Management (EKAW)*, LNAI 7603, pages 185-199, 2012. Springer.

## Controlled Natural Language

- S. Ferré. **SQUALL: The expressiveness of SPARQL 1.1 made available as a controlled natural language**. *Data & Knowledge Engineering*, 2014. To appear.





# ACN Instances for Information Access

Guided information access with increasing expressivity:

- Logical Concept Analysis (LCA) [1999-]
  - ▶ FCA + logical formulas instead of sets of attributes
  - ▶ people: Olivier Ridoux, Yoann Padioleau, Olivier Bedel, Benjamin Sigonneau, Véronique Abily, Yves Bekkers
  - ▶ softwares: CAMELIS, LISFS, GEOLIS, ABILIS, PORTALIS
- Cubes of concepts [2008-2012]
  - ▶ LCA + OLAP analytical queries + visualization
  - ▶ PhD of Pierre Allard, with O. Ridoux, implemented in ABILIS
- Query-based Faceted Search (QFS) [2009-]
  - ▶ FS + SPARQL queries + query focus
  - ▶ software: SEWELIS (a.k.a. CAMELIS2)



# ACN Instances for Information Authoring

ACN also applies to guided information authoring, where normal query results would be empty:

- **Update Through Interaction (UTILIS)** [2009-2012]
  - ▶ QFS for the guided update of RDF graphs
    - ★ query = object description
    - ★ extension = objects having a similar description
  - ▶ PhD of **Alice Hermann**, with M. Ducassé, implemented in **SEWELIS**
- **Possible World Explorer (PEW)** [2012]
  - ▶ motivation: ontology exploration and completion (no object)
  - ▶ QFS on OWL ontologies + axiom assertions
    - ★ query = OWL class expression
    - ★ extension = logical models of that class (*possible worlds*)
  - ▶ with **Sebastian Rudolph (Karlsruhe)**, implemented in **PEW**

*This shows that ACN components “queries” and “extension” should be interpreted in a liberal way.*

# CAMELIS: definitions (1/2)

- **K**: logical context  $(\mathcal{O}, \mathcal{L}, d)$ 
  - ▶  $\mathcal{O}$ : collection of objects
  - ▶  $\mathcal{L}$ : a **logic** (partial order of object descriptors) – **knowledge** valued attributes, taxonomies, intervals of numbers/dates, string patterns, and domain specific descriptors (e.g., DNA sequences, linguistic types, Java method signatures)  
a toolbox of components (*logic functors*) that can be composed
  - ▶  $d \in \mathcal{O} \rightarrow \mathcal{L}$ : **description** function – **facts**
- **Q**: Boolean closure of  $\mathcal{L}$  (and, or, not)
- $ext(q \in Q) \in E$ : set of objects whose description “matches”  $q$ 
  - ▶  $ext(q \in \mathcal{L}) = \{o \in \mathcal{O} \mid d(o) \sqsubseteq q\}$
  - ▶  $ext(q_1 \text{ and } q_2) = ext(q_1) \cap ext(q_2)$
  - ▶  $ext(q_1 \text{ or } q_2) = ext(q_1) \cup ext(q_2)$
  - ▶  $ext(\text{not } q_1) = \mathcal{O} \setminus ext(q_1)$

# CAMELIS: definitions (1/2)

- **K**: logical context  $(\mathcal{O}, \mathcal{L}, d)$ 
  - ▶  $\mathcal{O}$ : collection of objects
  - ▶  $\mathcal{L}$ : a **logic** (partial order of object descriptors) – **knowledge** valued attributes, taxonomies, intervals of numbers/dates, string patterns, and domain specific descriptors (e.g., DNA sequences, linguistic types, Java method signatures)  
a toolbox of components (*logic functors*) that can be composed
  - ▶  $d \in \mathcal{O} \rightarrow \mathcal{L}$ : **description** function – **facts**
- **Q**: Boolean closure of  $\mathcal{L}$  (and, or, not)
- $ext(q \in Q) \in E$ : set of objects whose description “matches”  $q$ 
  - ▶  $ext(q \in \mathcal{L}) = \{o \in \mathcal{O} \mid d(o) \sqsubseteq q\}$
  - ▶  $ext(q_1 \text{ and } q_2) = ext(q_1) \cap ext(q_2)$
  - ▶  $ext(q_1 \text{ or } q_2) = ext(q_1) \cup ext(q_2)$
  - ▶  $ext(\text{not } q_1) = \mathcal{O} \setminus ext(q_1)$

# CAMELIS: definitions (2/2)

- $index(q, e) \in I$ : similar to FCA for a partially ordered vocabulary of properties
- $links(q, e, i) \subseteq Q$ : different kinds of moves in concept lattice
  - ▶ **downward**: adding an index element, replacing more general query elements  
`France  $\rightsquigarrow$  France and Building`
  - ▶ **upward**: removing a query element, or replacing it with a more general descriptor  
`France and Building  $\rightsquigarrow$  Europe and Building`
  - ▶ **sideward**: downward+upward or upward+downward  
`Europe and Building  $\rightsquigarrow$  Europe and Landscape`

# CAMELIS: illustration on photos

Camelis - eltanin4-native

File Logic Browsing Updating Actions Help


Home Back Forward Refresh Save Paste Update 0

Australie and (Animal or Plante) and not Portrait ☒ Apply

☐ NOT ☒ = ☐ >= ☐ <= Zoom Pivot Picto (39) Texto (0)

all 39 ▾ date in [,]  
 39 ▾ date = 2004  
 26 ▸ date = feb 2004  
 13 ▸ date = mar 2004  
 39 ▸ event ?  
 39 ▸ exif ?  
 39 ▾ Location  
 39 ▾ Oceanie  
 39 ▾ Australie  
 13 ▸ 'Feather Dale Park'  
 1 ▸ Manly  
 25 ▸ Sydney  
 39 ▾ Object  
 33 ▾ 'animal'  
 15 'oiseau'

|< < Results: 1 - 12 / 39 > >|



dscf0017.jpg dscf0018.jpg  
 dscf0019.jpg dscf0024.jpg  
 dscf0091.jpg dscf0098.jpg

# CAMELIS: evaluation

- **expressivity:**

- ▶ extensible and unlimited for **object descriptors** (custom logics)
- ▶ **no relation** between objects
- ▶ **not all Boolean combinations** reachable by navigation
- ▶ **no analytics** (grouping, aggregations, ...)

- **usability:**

- ▶ very similar to **faceted search**
- ▶ **Boolean operators** may be an issue for some users
- ▶ some query elements have still to be **entered manually**

- **efficiency:** biggest context on a recent computer

- ▶ 100,000 files  $\times$  70 descriptors/file
- ▶ **5 min for loading + saving** (including metadata extraction)


# *Cubes of concepts [with Pierre Allard]*

An extension of LCA that supports **OLAP-like analytics**

- $K$  is a **multi-valued context**
  - ▶ i.e., an object can have **zero, one or several values** for each attribute
  - ▶ **LCA logics** can be used for each **value domain**
- queries (i.e., selections of objects) are extended with
  - ▶ **dimensions** to **group** selected objects (cube axes)
    - ★ dimension = attribute
  - ▶ **measures** to **aggregate** groups of objects (cube cells)
    - ★ measure = attribute + **aggregation operator**
- extensions are cubes of (LCA) extensions
  - ▶ hence, concepts are **cubes of (LCA) concepts**
- indexes remain the same
- additional links for changing dimensions and measures



# Cubes of concepts: illustration on bibliography



*In which years (since 2000) were journal and conference papers published per author and per type?*


abiLIS  Logged in as Anonymous guest.

File Options Logic Settings Context admin About Log out

(type is "article" or type is "inproceedings") and year = 2k


Ok Home author ? (0)   2 dim. array

type ? (0)   --dependent--

Measure : year ?  No aggregation Tag cloud

Features Actions zoom





pivot








☐ Not ☐ = ☐ <= ☐ >= 




- ☐ address ? (117)
- ☐ annotate ? (2)
- ☐ author ? (115)
- ☐ authors ? (115)
- ☐ booktitle ? (94)
- ☐ editor ? (57)
- ☐ journal ? (21)
- ☐ keyword ? (49)
- ☐ lis\_author ? (115)
- ☐ Misc ? (115)
- ☐ note ? (19)
- ☐ publisher ? (62)
- ☐ series ? (23)
- ☐ title ? (115)
- ☐ type ? (115)
  - ☐ type is "article" (21)
  - ☐ type is "inproceedings" (94)
- ☐ year ? (115)
  - ☒ year = 2k (115)

Filter: Clear

Objects actions Filter in current page: Clear

115 accessible objects Select: All None Inverse << < - 1/3 - >> >> view all    

|  | article    | inproceedings   |
|--|---|--|
| lis_author Allard   |   | 2008 (1, 50.0%)<br>2010 (1, 50.0%)   |
| lis_author Bars     |   | 2002 (1, 100.0%)   |
| lis_author Bedel    | 2008 (1, 100.0%)  | 2006 (2, 50.0%)<br>2007 (1, 25.0%)<br>2008 (1, 25.0%)  |
| lis_author Cellier  | 2008 (1, 100.0%)  | 2008 (2, 40.0%)<br>2009 (1, 20.0%)<br>2006 (1, 20.0%)<br>2007 (1, 20.0%)   |
| lis_author Ducasse  | 2008 (2, 25.0%)<br>2004 (1, 12.5%)<br>2001 (1, 12.5%)<br>2010 (1, 12.5%)<br>2000 (1, 12.5%)<br>2002 (1, 12.5%)<br>2009 (1, 12.5%) | 2002 (6, 17.1%)<br>2000 (5, 14.2%)<br>2007 (4, 11.4%)<br>2005 (4, 11.4%)<br>2009 (3, 8.5%)<br>2003 (3, 8.5%)<br>2001 (3, 8.5%)<br>2004 (3, 8.5%)<br>2008 (2, 5.7%) |

Links   



# Query-based Faceted Search (QFS): definitions (1/2)

- **K**: a **RDF graph** = a set of triples (labeled links) = a set of binary relations = a set of formal contexts
  - ▶ similar to context families of RCA [Rouane et al. 2007]
- **Q**: LISQL, a mild-syntax fragment of SPARQL
  - ▶ queries are **graph patterns** over RDF data

a person

birth :

year : (1601 or 1649)

place : ?X and part of England

father : birth : place : not ?X

- ▶ **query focus**: *What are results and index about?*

- ★ a person: *"people born in 1601 or 1649 somewhere in England, and whose father was not born there"*
- ★ ?X and part of England: *"places in England where a person was born in 1601 or 1649, and where his father was not born"*
- ★ used as an **insertion point** for user selections

# Query-based Faceted Search (QFS): definitions (2/2)

- $ext(q) \in E$ : extents are still **sets of objects** (RDF nodes)
  - ▶ but membership of an object depends on its **relationships** to other objects
  - ▶ can be defined by translating LISQL into SPARQL
- $index(q, e) \in I$ : similar to FCA for a **partially ordered vocabulary**
  - ▶ **vocabulary**: classes (**person**), properties (**place** :, **part of**), entities (**England**), values (**1601**), co-references (**?X**)
  - ▶ **partial order**: class hierarchy, property hierarchy
  - ▶ the extension is included in the index !
- $links(q, e, i)$ : focus-centered **query transformations**
  - ▶ from index: insertion/removal of index elements
  - ▶ from query: focus changes, insertion of Boolean operators

# QFS: illustration on genealogy (SEWELIS)

The screenshot shows the Sewelis web application interface with the following components:

- Query + focus:** A query input area on the left containing the query:
 

```
a person
birth :
  year :
    1601
    or 1649
  place :
    ?X
    opt part of England
father : birth : place : not ?X
```

 Below the query is a "Create" button.
- Query transformations (or, not, ?X, ...):** A vertical toolbar in the center containing buttons for:
  - Focus Up
  - P
  - P ?
  - or ?
  - not ?
  - not \_
  - { \_ }
  - ?X...?Z
  - Describe
  - Select
  - Delete
  - Reverse
  - ...
- Facet hierarchy (classes and properties):** A list of suggested features for 1 object, including:
  - 1 ▶ a person
  - 1 ▼ ancestor : ?
  - 1 ▼ parent : ?
  - 1 ▶ father : ?
  - 1 ▶ mother : ?
  - 1 ▶ ancestor of ?
  - 1 ▶ birth : ?
  - 1 ▶ child of ?
  - 1 ▶ death : ?
  - 1 ▶ firstname : ?
- Selection (query answers):** A list of query answers on the right, including:
  - 1 ▶ Joseph /Ball/ [I09]
- Values per facet (resources and literals):** A list of values for the selected facet, including:
  - parent : @
  - opt ancestor : @
  - 1 ▶ William /Ball/ [I11]
  - 1 ▶ Hannah /Artherold/ [I13]

# PEW [with Sebastian Rudolph]

## An adaptation of QFS for OWL ontologies

- *K*: a set of OWL axioms, possibly knowledge only (no facts)
  - ▶ example: the **Pizza ontology**
- *Q*: situations
  - ▶ OWL class expressions, only atomic negation (subset of LISQL)
  - ▶ a **Pizza and topping : Tomato**
- *ext(q)*: the **satisfiability** of the query (a Boolean!)
  - ▶ a **reasoning engine** is used instead of a query engine
- *index(q, e)*: similar to QFS but satisfiability instead of non-zero frequency
- *links(q, e, i)*: subset of QFS
- Application: ontology understanding and completion
  - ▶ exploration of **possible worlds** (satisfiable class expressions)
  - ▶ identification of unexpected worlds
  - ▶ exclusion of unexpected worlds by asserting new axioms
  - ▶ e.g., **we found that a Veggie pizza could contain meat or fish!**