Formal Concept Analysis
with
GaLicia
Galois Lattice Interactive Constructor

Petko Valtchev

Université de Montréal

Petko.Valtchev@UMontreal.CA

&

The GaLicia team

Galois Lattice Interactive Constructor

http://www.iro.umontreal.ca/~galicia/

Overview

- ***Formal concept analysis (FCA)***: "application of lattice theory to data analysis"
  - Theory:
    - Back to work by **O. Öre** and by **G. Birkhoff** in 40s,
    - M. Barbut & B. Monjardet, R. Wille, B. Ganter, V. Duquenne…
  - Practice:
    - *social sciences*: Duquenne, Wille,…
    - *information retrieval*: Godin, Carpineto and Romano,…
    - *software engineering*: Godin, Snelting,…
    - *data mining*: Missaoui & Godin, Lakhal,…
- Now:
  - rapidly growing community: "FCA" + "lattices" - couple of 10^3 hits with Google,
  - annual forums: 2 intl. conferences, 2+ workshops,
- Missing: a widely-shared software platform for FCA (ToscanaJ, ConExp, Galicia)

Outline of the Talk

- *FCA: Galois connections, closures, lattices, min. generators …*
- *Computational challenges*
- *Realization within Galicia + demo*

Formal Contexts and Galois Connections

K= (O, A, I)

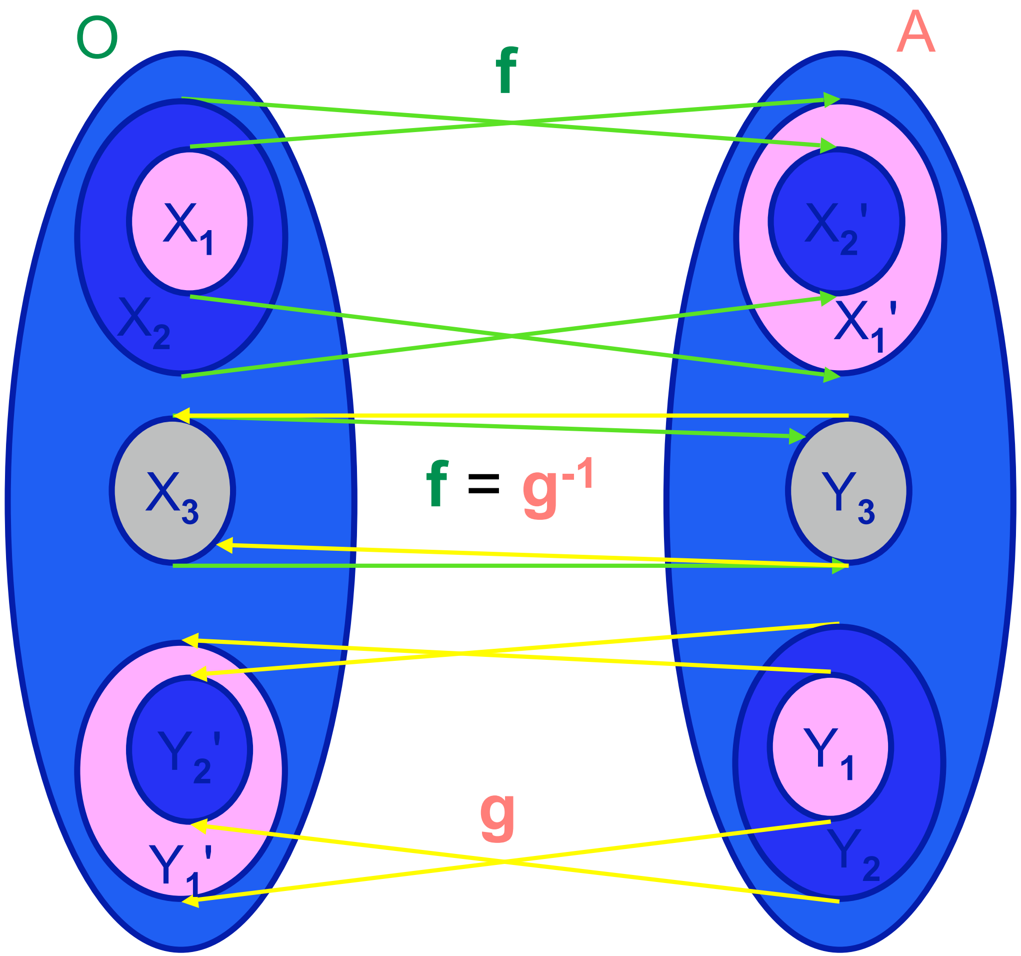
O

A	1	2	3	4	5	6	7	8
a	X	X	X	X	X	X	X	X
b	X	X	X		X	X		
c			X	X		X	X	X
d					X	X	X	X

$\{5, 7\}' = \{a, d\}$ $\{b, d\}' = \{5, 6\}$

$f(X) = X' = \{y \in A \mid \forall x \in X, (x, y) \in I\}$

$g(Y) = Y' = \{x \in O \mid \forall y \in Y, (x, y) \in I\}$



Galois connection $Y \subseteq f(X)$ iff $X \subseteq g(Y)$

closure operators

$X'' = g \circ f(X)$

$Y'' = f \circ g(Y)$

closed sets $X = X''$, $Y = Y''$

$\{a, d\}'' = \{a, d\}$

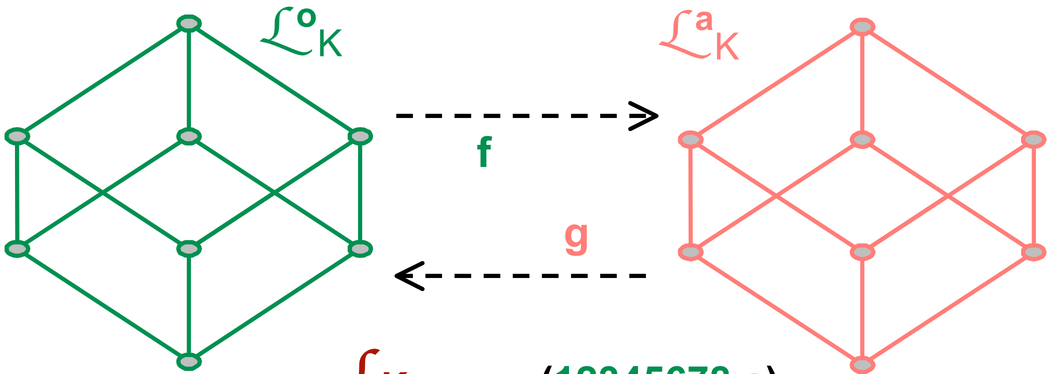
$\{5, 6\}'' = \{5, 6\}$ $\{b, d\}'' = \{a, b, d\}$

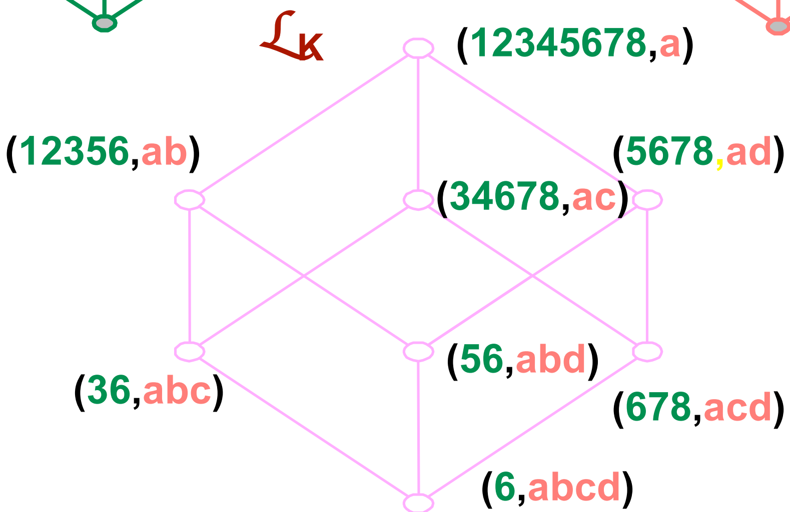
Lattices of Formal Concepts (« *de Galois* »)

Families of closed

$\mathcal{C}^{o}_{K} = \{ X \mid X \subseteq O, X'' = X\}$

$\mathcal{C}^{a}_{K} = \{ Y \mid Y \subseteq A, Y'' = Y \}$





lattice (anti-)isomorphism

$\mathcal{L}^{o}_{K} = (\mathcal{C}^{o}_{K}, \subseteq) \equiv \mathcal{L}^{a}_{K} = (\mathcal{C}^{a}_{K}, \supseteq)$

with f and g as *co-bijections*

formal concept (X,Y)

$X \in \mathcal{C}^{o}_{K}$ (*extent*), $X = Y'$;

$Y \in \mathcal{C}^{a}_{K}$ (*intent*), $Y = X'$.

partial order

(sub-concept of)

$(X_1, Y_1) \leq (X_2, Y_2)$ **iff** $X_1 \subseteq X_2$

$(\Leftrightarrow Y_2 \subseteq Y_1)$

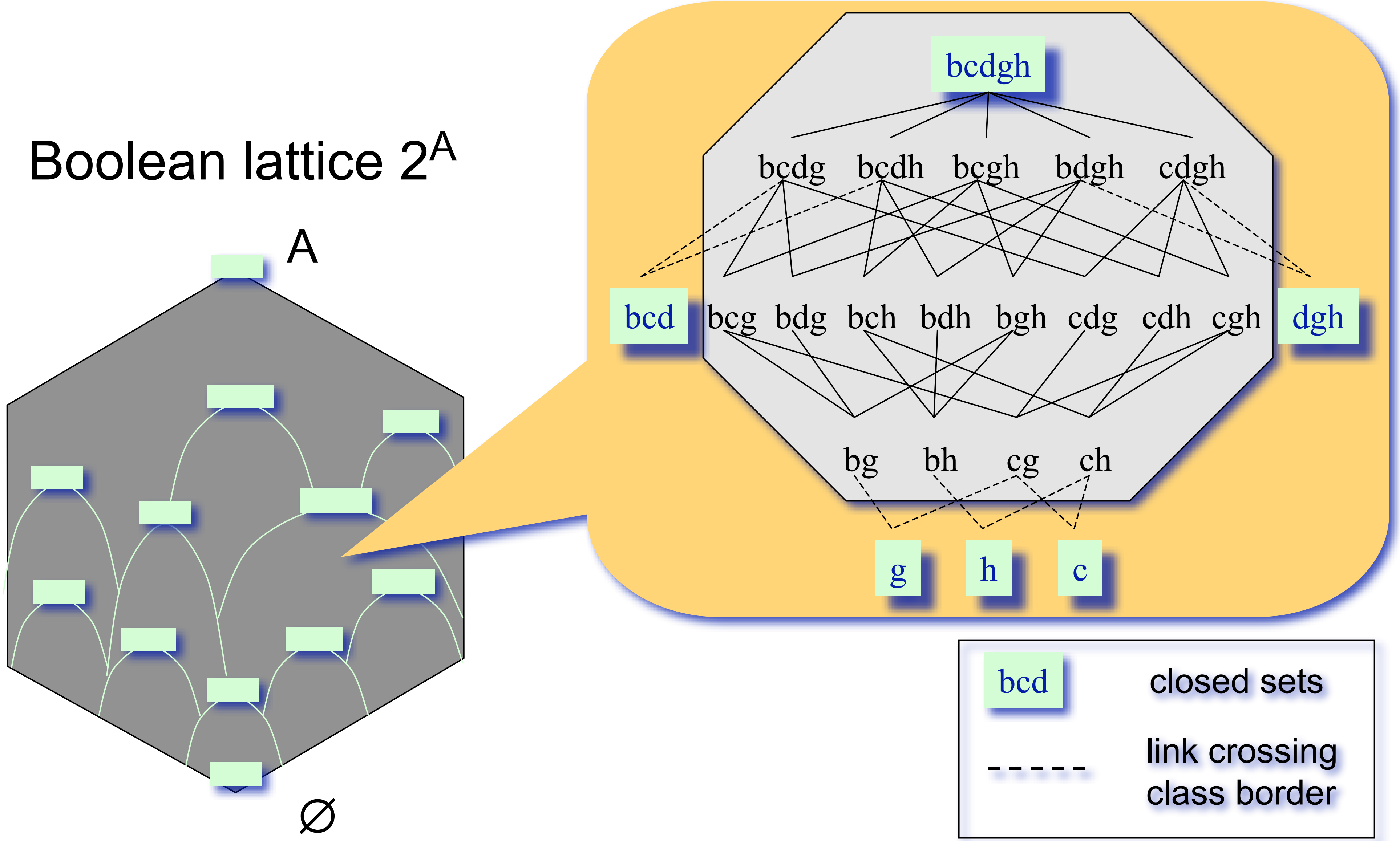
lattice operators

[Wille 82], [Barbut & Montjardet 70]

inf - $\bigcup_{j\in J}(X_j, Y_j) = (\bigcap_{j\in J} X_j, (\bigcup_{j\in J} Y_j)'')$

sup - $\bigcup_{j\in J}(X_j, Y_j) = ((\bigcup_{j\in J} X_j)'', \bigcap_{j\in J} Y_j)$

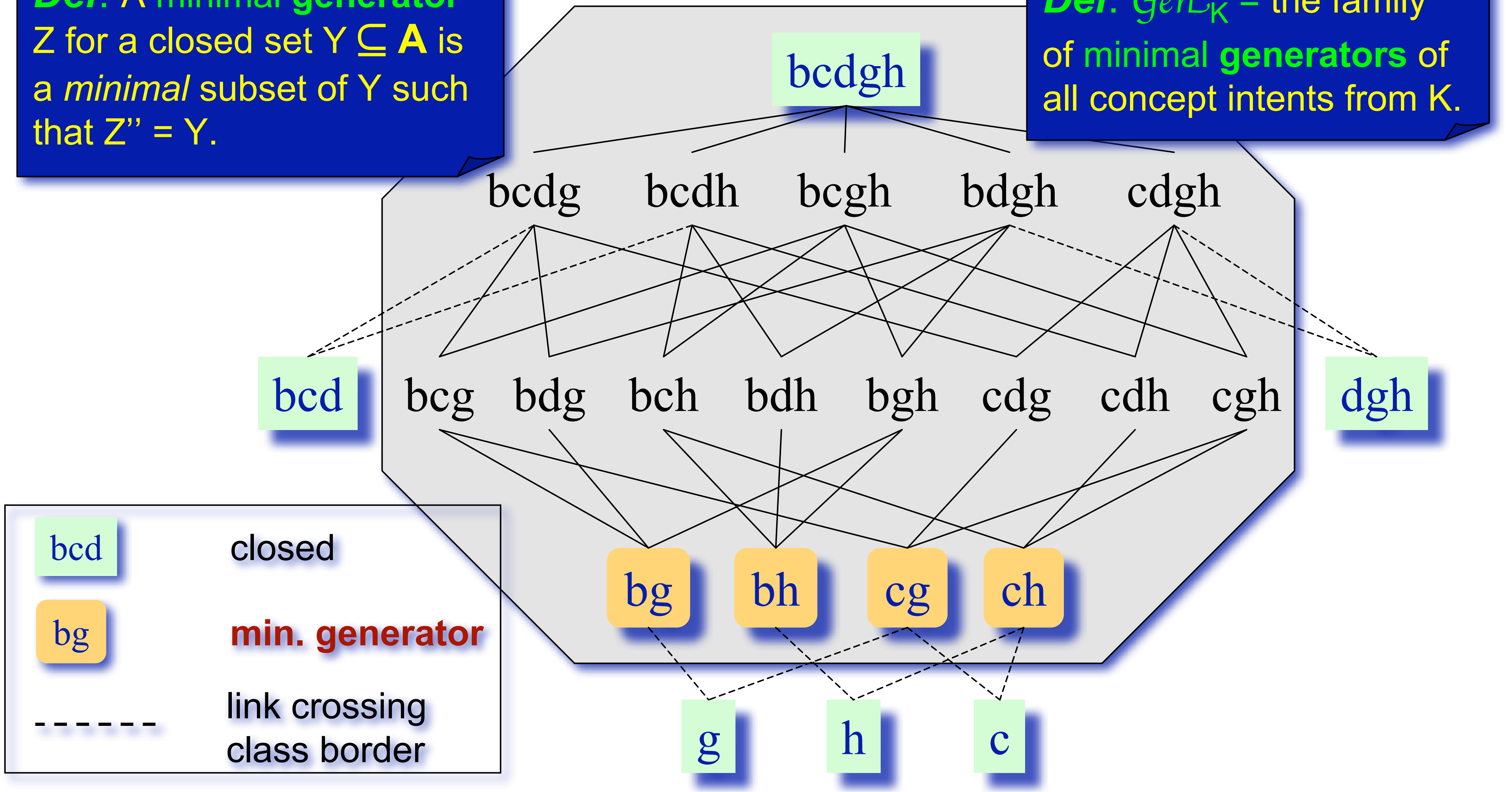
Equivalence Relation on 2^A Induced by $\mathcal{C}^a{}_K$



Minimal Generators

Def. A minimal **generator** Z for a closed set Y ⊆ **A** is a *minimal* subset of Y such that Z'' = Y.

Def. $\mathcal{G}en_K$ = the family of minimal **generators** of all concept intents from K.



Why Are Min. Generators Interesting?

Minimal generators in…

- **…theory**:
  - related to ***minimal transversals*** in hypergraph theory [Berge 89]
  - **candidate keys** of the tables in a *relational database*
- **… practice**:
  - **minimal sets** of *tests/exams/questions* for a ***medical diagnosis***
- **…algorithmic** design:
  - ***canonical representatives*** for concept **intents**:
    - minimal generating *prefixes* in *NextClosure* [Ganter 84]
  - "***seeds***" for the computation of **intents**:
    - in general-purpose FCA algorithms: *Titanic* [Stumme *et al* 02]
    - in FCA-flavored *data mining* algorithms: *Close*, *Aclose* [Pasquier 00]

Implications

Given K = (O,A,I), Y, Z $\subseteq$ A,
Y $\rightarrow$ Z is an **implication** :
- Y ***premise***,
- Z ***conclusion***.
(aka ***functional dependency*** *in DB*)

Def. Y $\rightarrow$ Z ***valid*** in K if
$\forall$ o $\in$ O, Y $\subseteq$ o' forces Z $\subseteq$ o' (iff Z $\subseteq$ Y'').
Σ_K = all ***valid*** implications of K.

Ex. bd $\rightarrow$ af, ae $\rightarrow$ cd : ***valid***,
bc $\rightarrow$ agh : ***invalid*** (6 - ctr-ex.).

Σ_K is large and redundant!

	a	b	c	d	e	f	g	h	i
1	x	x					x		
2	x	x					x	x	
3	x	x	x				x	x	
4	x		x				x	x	x
5	x	x		x		x			
6	x	x	x	x		x			
7	x		x	x	x				
8	x		x	x		x			

Def. A maximally ***informative*** *rule*:
- *minimal* premise,
- *maximal* conclusion.

Ex. bd $\rightarrow$ af : ***infromative***
ae $\rightarrow$ cd : ***not*** (e $\rightarrow$ acd valid).

Inference Axioms and Covers

Def. **Armstrong's axioms** for entailment
$\models \subseteq \mathbf{2}^{\Sigma_K} \times \mathbf{2}^{\Sigma_K}$
inference model (calculi) over Σ_K

- $\emptyset \models Y \to Y$;
- $Y \to Z, U \to V \models Y \cup U \to Z \cup V$;
- $Y \to Z, U \to V, U \subseteq Z \models Y \to V$.

Ex.

bd $\to$ af, e $\to$ acd $\models$ bde $\to$ acdf

Def. **Cover** for a set of implications

For $\mathcal{I}, \mathcal{J} \subseteq \Sigma_K$, $\mathcal{I}$ is a **cover** of $\mathcal{J}$ iff $\mathcal{I} \models \mathcal{J}$

Pseudo-closed Sets and Canonical Basis

Def. $\mathcal{PC}_K \subseteq 2^A$: the *pseudo-closed sets* of K:
- $Y \neq Y''$,
- for all Z pseudo-closed, $Z \subset Y$ forces $Z'' \subset Y$.

Ex. acdef in $\mathcal{PC}_K$:

ae, af in $\mathcal{PC}_K$;

$ae''=acde \subset acdef$,

$af''=afd \subset acdef$.

Def. *(Duquenne & Guigues 86)*

Canonical basis of K, $\mathcal{B}_K = \{Z \rightarrow Z'' - Z \mid Z \in \mathcal{PC}_K\}$.

Prop. For all K, $\mathcal{B}_K$ is a **cover** of $\sum_K$ of a minimal size (***nb. of rules***).

Ex. The basis of the example

$adg \rightarrow bcefhi \quad acg \rightarrow h \quad ah \rightarrow g \quad \rightarrow a$

$acdef \rightarrow bghi \quad abd \rightarrow f \quad ae \rightarrow cd \quad af \rightarrow d$

$abcghi \rightarrow def \quad ai \rightarrow cgh$

	a	b	c	d	e	f	g	h	i
1	x	x					x		
2	x	x					x	x	
3	x	x	x				x	x	
4	x		x				x	x	x
5	x	x		x		x			
6	x	x	x	x		x			
7	x		x	x	x				
8	x		x	x		x			

Partial Implications and Further Bases

Def. Partial implication $X \rightarrow Y$ *(Luxenburger 92)*
Not valid to 100% (exists object o : $X \subseteq o'$, but $Y \not\subseteq o'$).

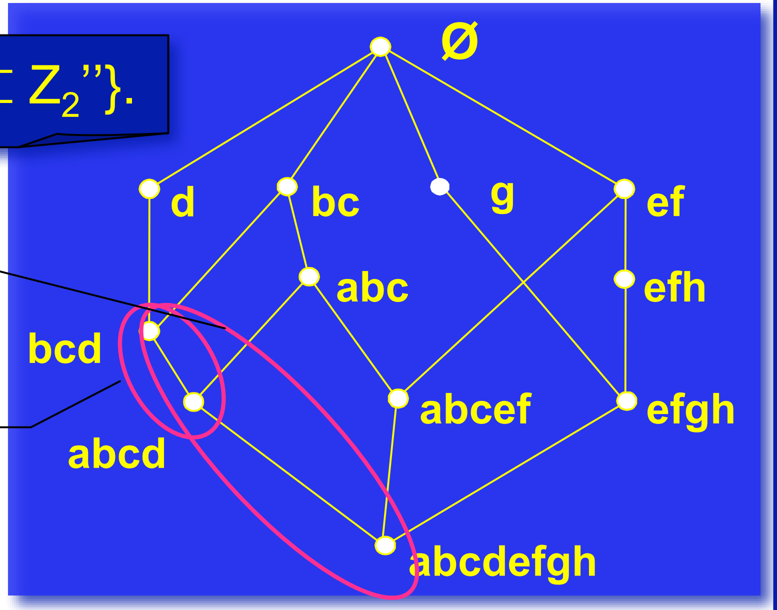
a.k.a ***association rules***

Two bases for partial implications, following the **lattice structure** [Luxenburger 92]

Def. Global basis : $\{Z_1'' \rightarrow Z_2'' - Z_1'' \mid Z_1'' \subset Z_2''\}$.

bcd → aefgh
Z = bcd, Y = abcdefgh

bcd → a
Z = bcd, Y = abcd



Def. Cover basis : $\{ Z'' \rightarrow Y'' - Z'' \mid$ *Z'' minimal closed subset of Y''*$\}$.

Why Study the Pseudo-closed?

Pseudo-closed in…

- **…theory**:
  - related to the precedence relation in the ***lattice of all closures*** on a ground set A [Caspard & Monjardet 03]
  - ***minimal covers*** for functional dependencies in relational databases [Maier 80]
- **…algorithmic** design:
  - ***alternative closure*** *computation mechanism* for intents:
    - helps restrict usage of extents in large datasets [Valtchev & Duquenne 03],
- **… practice**:
  - **non-redundant sets** of association rules in *data mining* [Kryszkiewicz 02]

Intriguing Properties

- Families not necessarily disjoint:
  - Only $\mathcal{C}_K \cap \mathcal{PC}_K = \varnothing$
  - $\mathcal{Gen}_K$ may share elements with both other families

Prop. ***$\mathcal{Gen}_K$ is an order ideal of the Boolean lattice 2^A :***

$Z \in \mathcal{G}_K$ forces $\forall\ Y \subseteq Z$, $Z \in \mathcal{Gen}_K$.

Prop. ***$\mathcal{PC}_K \cup \mathcal{C}_K$ is closed for intersection (closure space):***

$\mathcal{PC}_K \cup \mathcal{C}_K = (\mathcal{PC}_K \cup \mathcal{C}_K)^{\cap}$.

Prop. ***Individual elements of $\mathcal{PC}_K$ preserve the closure property:***

$\forall\ Y \in \mathcal{PC}_K$, $\forall\ Z \in \mathcal{C}_K$, $Y \cap Z \in \mathcal{C}_K \cup \{Y\}$.

Outline of the Talk

- *FCA: Galois connections, closures, lattices, min. generators …*
- *Computational challenges*
- *Realization within Galicia*

Algorithmic Problems in FCA

			Target structure			
			Concept set $\mathcal{C}_K$	**Concept set + precedence** $\mathcal{L}_K = (\mathcal{C}_K, \leq)$	**Min. generators** $Gen_{\mathcal{L}_K}$	**Canonical basis** $\mathcal{B}_K$
Mode	**batch**		*NextClosure* [Ganter 84], [Chein 69]	[Bordat 86], [Nourine & Raynaud 99]	Titanic [Stumme *et al* 02], [Pfalz & Taylor 02]	NextClosure for PC [Ganter 84]
	on-line	**O**	[Norris 78]	[Godin *et al* 95], [Carpinetto & Romano 96], [Valtchev *et al* 02, 03]	[Valtchev *et al* 04],	
		A		[Nehme *et al* 05]	[Nehme *et al* 05]	[Ob'edkov & Duquenne 03]
	merge	**O**		[Valtchev & Missaoui 01]	[Frambourg *et al*, ***submitted***]	
		A	[Valtchev & Duquenne 03]	[Valtchev *et al* 02]		[Valtchev & Duquenne 03]

NextClosure

- **Reference algorithm in FCA**: [Ganter 84]
- Typical combinatorial generation (listing) procedure:
  - Search for **closed attribute sets** throughout the Boolean lattice 2^A ,
  - Attribute set A **totally ordered**,
  - **Closures** of candidate sets computed,
  - Closed sets listed in a **lexicographic** order:
    - Implicit ***tree structure***
  - Looking for a **canonical representative** for each closed set:
    - a minimal generating prefix = minimal prefix including a **minimal generator**
    - pruning the search tree
  - Uses **no memory**:
    - moves from one candidate to the next one in the **lexicographic** order,
    - hence suitable for **large lattices**,

On-line Maintenance of Lattices & Co.

Why?

- Natural **evolution** in a dataset:
  - organizations feed new data to their databases on a regular basis,
  - reuse of current analysis results instead of computing the new ones from scratch,
- **Explorative** analysis:
  - adding/removing input data elements,
  - tracking the changes in the result,
- **Potential efficiency** gains:
  - ***Incremental*** mode: much faster than batch reconstruction from scratch,
  - ***Batch*** mode: provably faster for sparse data,

On-line Lattice Maintenance

	a	b	c	d	e	f	g
1			x	x			x
2	x	x	x				
3				x	x	x	x
4							x
5					x	x	
6	x	x	x				
7		x	x	x			
8	x			x			

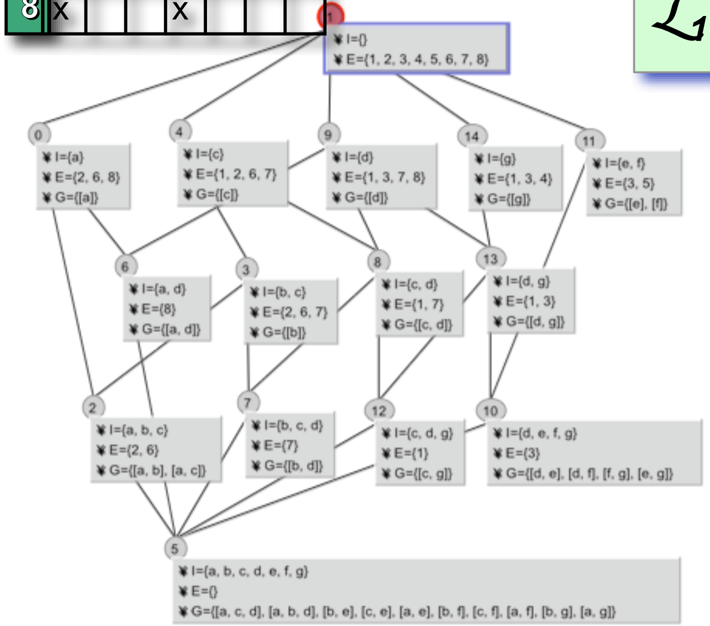
$K_1 = (O, A, I)$

$K_2 = (O, A \cup \{a\}, I \cup a \times a')$

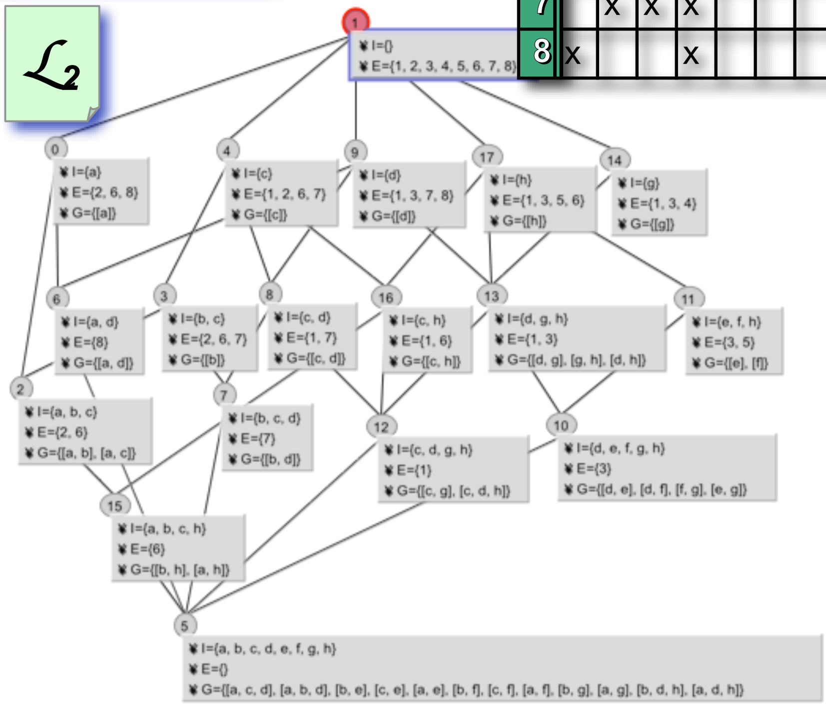
	a	b	c	d	e	f	g	h
1			x	x			x	x
2	x	x	x					
3				x	x	x	x	x
4							x	
5					x	x		x
6	x	x	x					x
7		x	x	x				
8	x			x				

Problem: Given $\mathcal{L}_1$ and (a, a'), transform the ***data structure*** representing $\mathcal{L}_1$ into an equivalent for $\mathcal{L}_2$.

$\mathcal{L}_1$

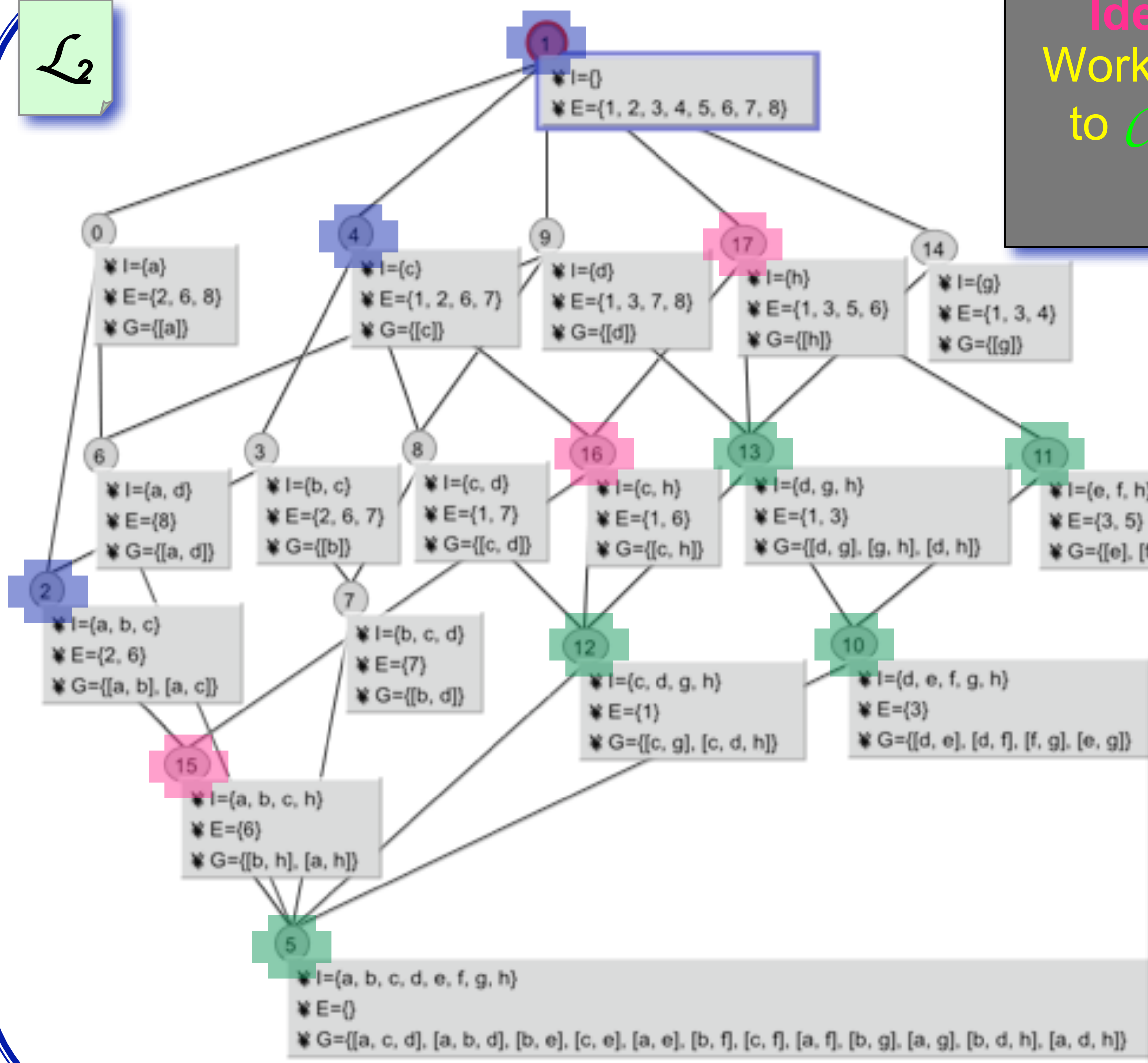


$\mathcal{L}_2$



The Approach Foundations

$\mathcal{L}_2$



Idea: Object dimension stable. Work amounts to add a new extent to $\mathcal{C}_1^o$ and close the result by $\cap$:

$$\mathcal{C}_2^o = (\mathcal{C}_1^o \cup a')^{\cap}$$

New *extents:* $\mathcal{C}_2^o - \mathcal{C}_1^o$

$\Rightarrow$ *new concepts* : $N_2(a)$

Existing *extents:* $\mathcal{C}_2^o \cap \mathcal{C}_1^o$

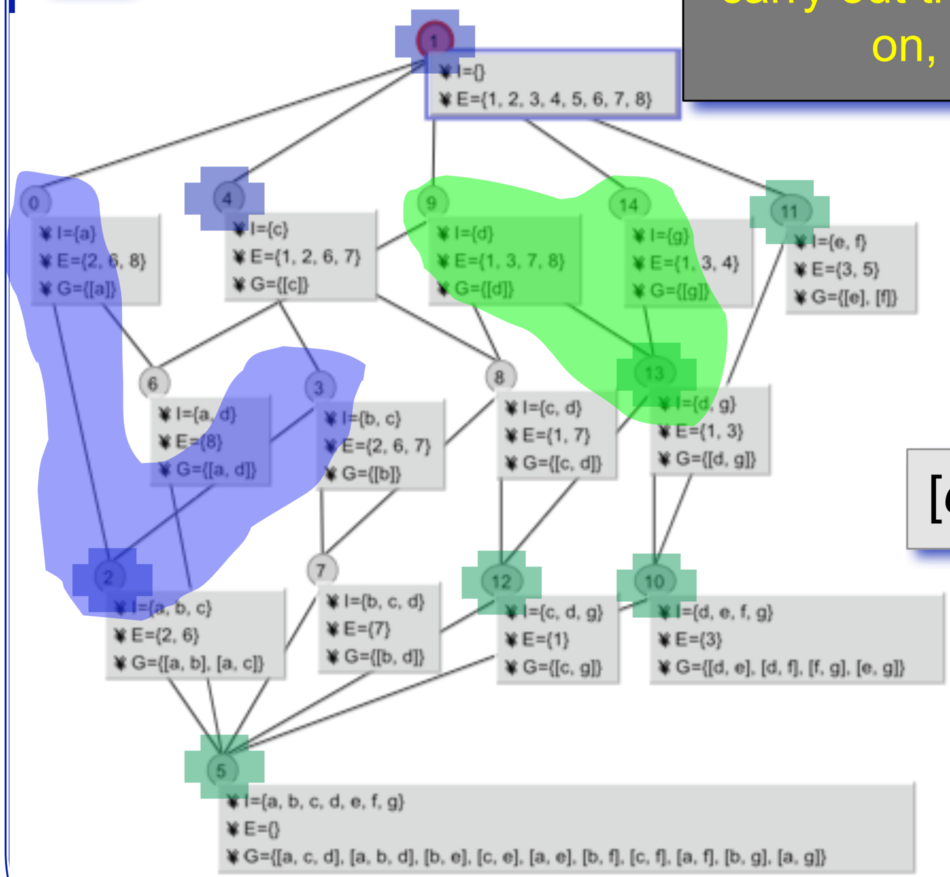
L_1 -> L_2 transition	*old*	*genitor*	*modified*
intent	same	same	**change**
extent	same	same	same
lower cov.	same	**change**	same
upper cov.	same	same	**change**
notation	$U_2(a)$	$G_2(a)$	$M_2(a)$

The Approach Foundations (cont'ed)

$\mathcal{L}_1$

Idea: Find the homologous concepts of **genitors** and **modified** in $\mathcal{L}_1$ and carry out the restructuring from them on, up to obtaining $\mathcal{L}_2$.

	genitor	*modified*
L_1	$G_1(a)$	$M_1(a)$
L_2	$G_2(a)$	$M_2(a)$



Equivalence relation *on* $\mathcal{L}_1$*, by extent intersection with a':*

$$[c]_a = \{\underline{c} \in C_1 \mid ext(\underline{c}) \cap a' = ext(c) \cap a'\}$$

Characterization *of* $G_1(a)$ *and* $M_1(a)$*: Minima in their equivalence. classes* $[\ ]_a$

$$c \in G_1(a) \cup M_1(a) \Leftrightarrow c = min([c]_a)$$

Lattice Update Method: Attribute-wise

Procedure Add-Attribute(
Input: $\mathcal{L}$ a lattice, *a* an attribute;
Output: $\mathcal{L}$ a lattice, *updated*)

for each $c = (X, Y)$ in $\mathcal{L}$
$E \leftarrow X \cap a'$
if *c* ***minimal*** for *E* then
if $X = E$ then // ***modified***
Update(*c*)
else // ***genitor***
cc $\leftarrow$ ***new-concept***(E, $Y \cup \{o\}$)
$\mathcal{L} \leftarrow \mathcal{L} \cup \{cc\}$
UpdateOrder(*c*, ***cc***)

Problem$_1$: Fit min. generator $\mathcal{Gen}_K$ computation to Add-Attribute($\mathcal{L}$,*a*).

See [Nehme *et al.* 05]

Problem$_2$: Fit pseudo-closed $\mathcal{PC}_K$ computation to Add-Attribute($\mathcal{L}$,*a*).

See [Ob'edkov & Duquenne 03]

Merge of Lattices & Co.

Why?

- looking for the **interactions among subsets of descriptors** in a dataset:
  - **split** the descriptor set,
  - **process** the resulting subsets:
    - first independently (**factor** lattices),
    - then as a whole (**global** lattice),
  - **map** the **factor** lattices into the **global** one,
  - **merge-based** construction = last two steps carried out ***simultaneously***.

- **visualization** (related to previous topic):
  - present the global lattice as "projected" into the *direct product* of the factors,

- **potential efficiency gains**: take advantage of distributed/parallel architecture
  - split the work into sub-problems,
  - deal with them separately,
  - put together the partial results,

Fragmentation of Contexts

Apposition =

recompose a context
after a *split*

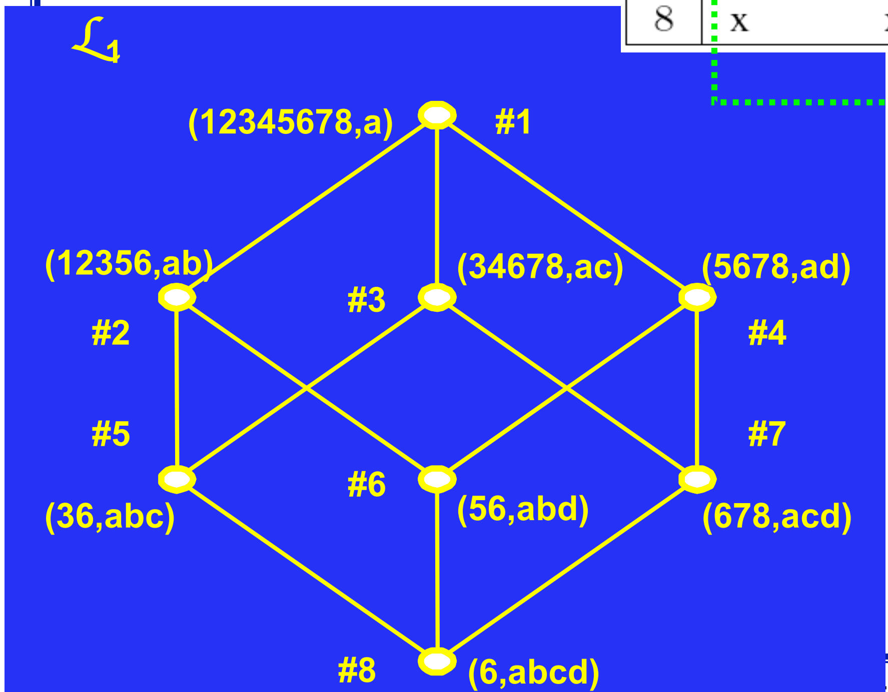
$K = K_1 \mid K_2$

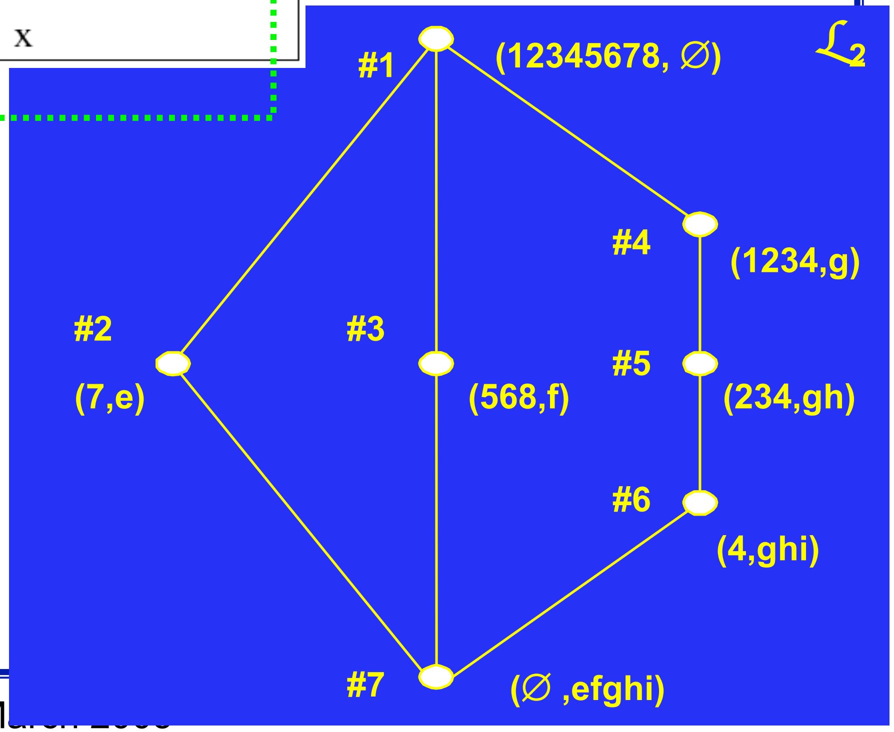
$K = (O, A, I)$

$K_1 = (O, A_1, I \cap O \times A_1)$

$K_2 = (O, A_2, I \cap O \times A_2)$

	A_1				A_2				
	a	b	c	d	e	f	g	h	i
1	x	x					x		
2	x	x					x	x	
3	x	x	x				x	x	
4	x		x				x	x	x
5	x	x		x		x			
6	x	x	x	x		x			
7	x		x	x	x				
8	x		x	x		x			





Lattice Merge
The Problem

Notations:

- ***Contexts***: ***factors*** K_1, K_2, ***global*** $K_3 = K_1 \mid K_2$.
- ***Closures***: ***operators*** $\_^{ii}$ (i=1,2,3).
- ***Lattices, canonical bases, generators***:
  - ***factors*** $\mathcal{L}_i$ / $\mathcal{B}_i$ / $Gen_{\mathcal{L}_i}$ (i=1,2),
  - ***global*** $\mathcal{L}_3$ / $\mathcal{B}_3$ / $Gen_{\mathcal{L}_3}$,
  - ***direct product*** $\mathcal{L}_{1,2}$ / $\mathcal{B}_{1,2}$.

Given:

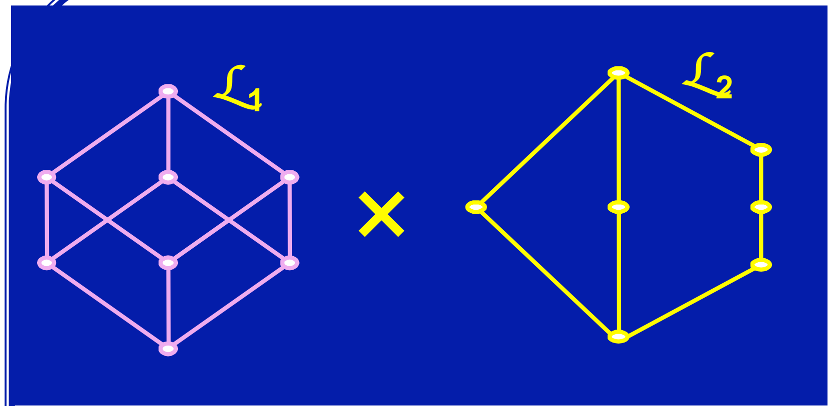
- ***Factor lattices***: $\mathcal{L}_1$, $\mathcal{L}_2$
- ***(OPT) canonical bases of factors***: $\mathcal{B}_1$, $\mathcal{B}_2$
- ***(OPT) min. generator families of factors***: $Gen_{\mathcal{L}_1}$, $Gen_{\mathcal{L}_2}$

Find:

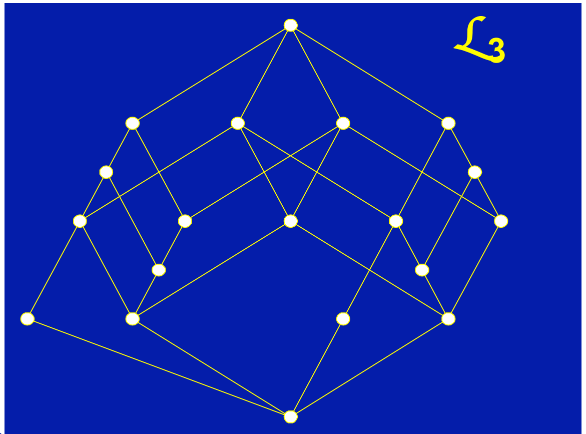
- ***Global lattice***: $\mathcal{L}_3$
- ***(OPT) global canonical base***: $\mathcal{B}_3$
- ***(OPT) global min. generator family***: $Gen_{\mathcal{L}_3}$

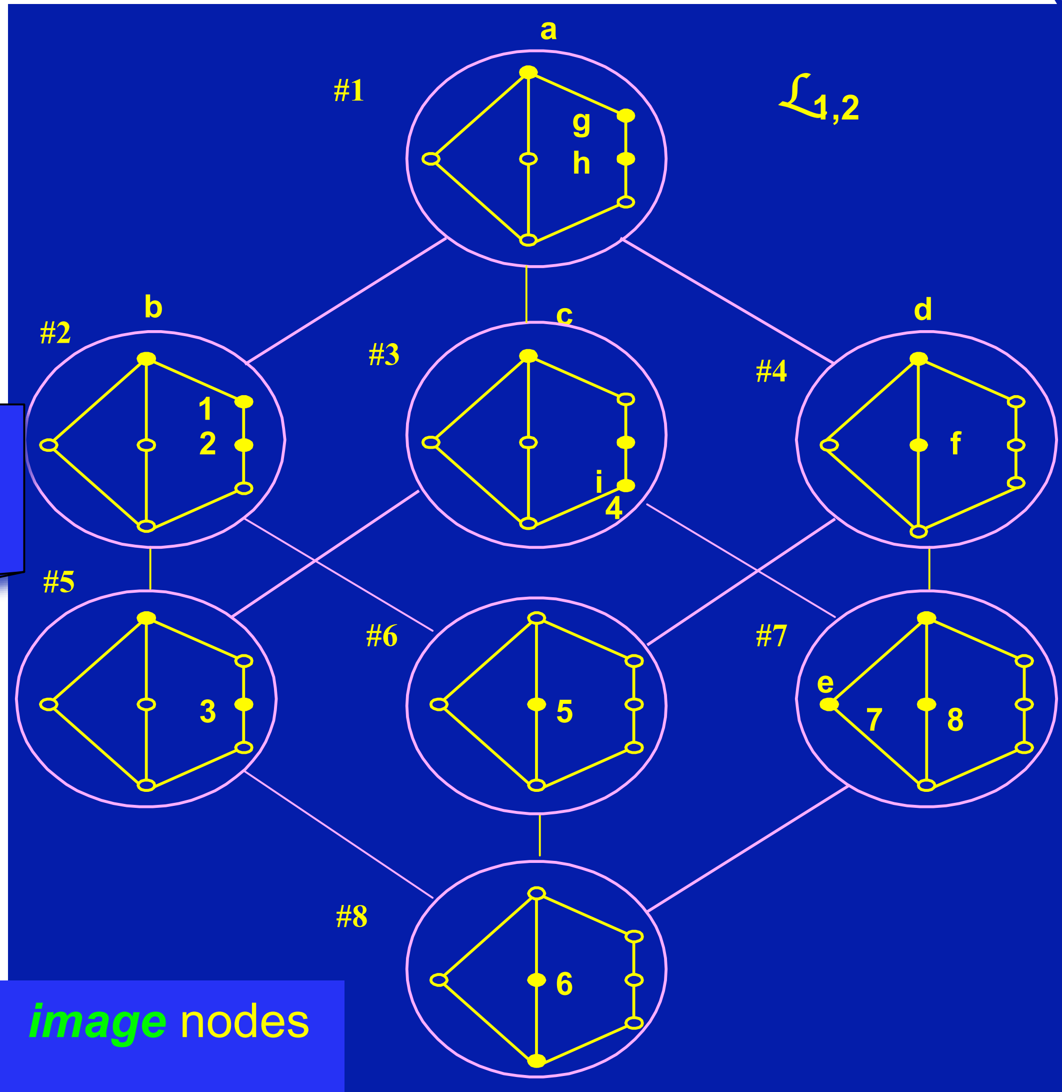
Visualization Tool

The Nested Line Diagram of $L_{1,2}$ *[Wille 82]*



Prop. $\mathcal{L}_3$ is a sub-semi-lattice of $\mathcal{L}_{1,2}$
hence may be embedded into it.





- ● ***image*** nodes
- ○ ***void*** nodes

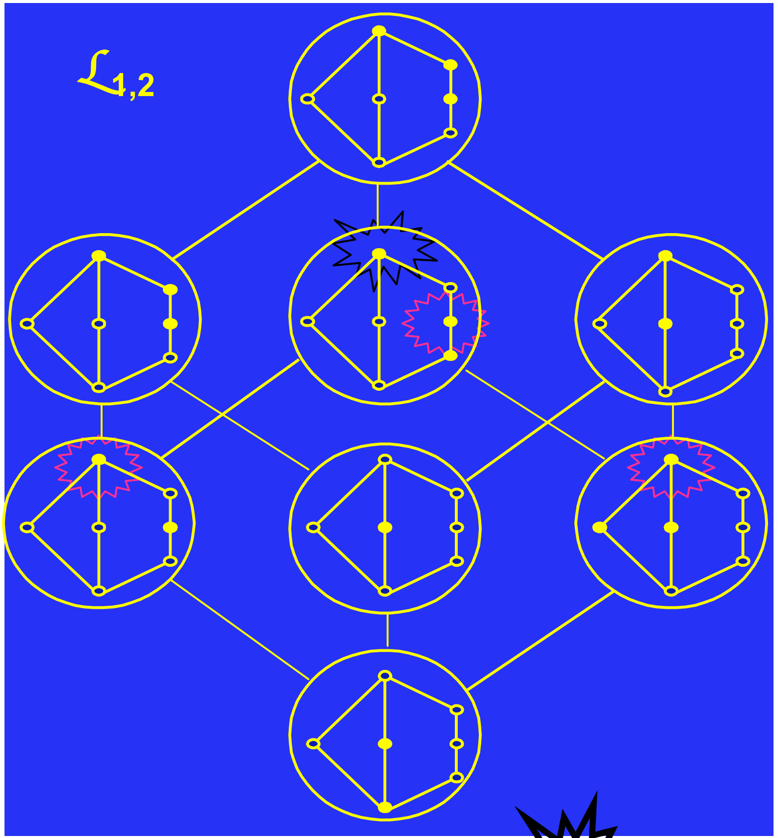
Approaching the Merge

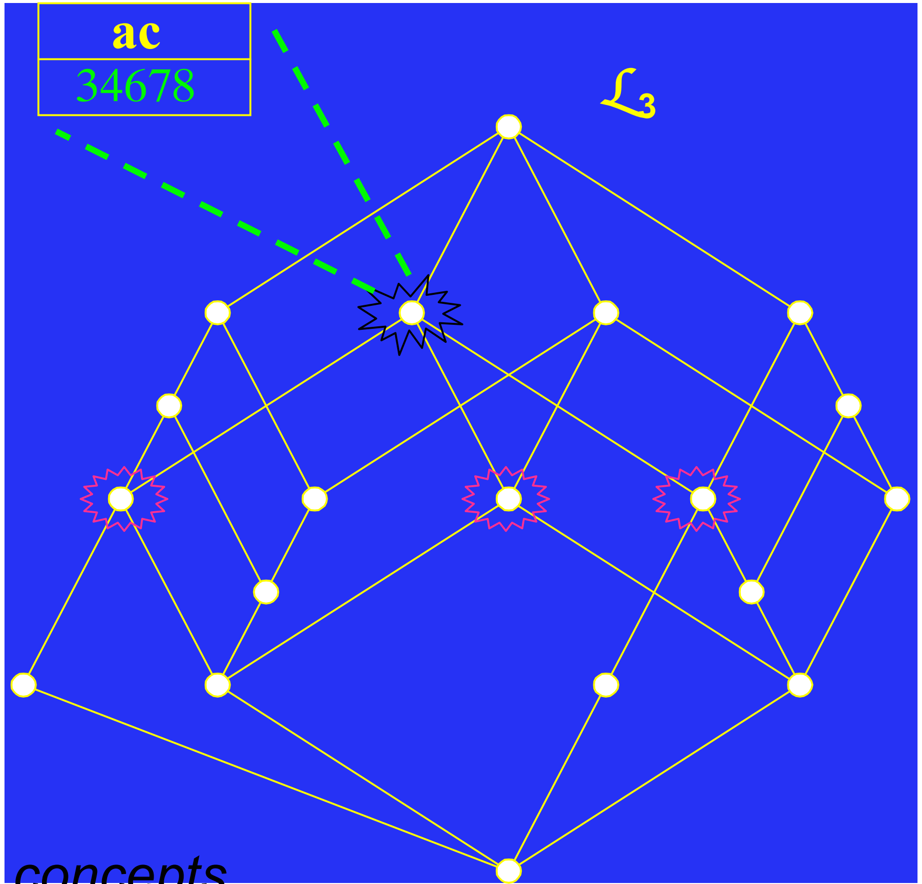
Complete lattice merge, i.e., **concepts** and **order**

Key ideas:

- ***Mixture of extent families***: $\mathcal{C}^o{}_3$ = all pair-wise intersections on $\mathcal{C}^o{}_1 \times \mathcal{C}^o{}_2$.
- *Each global **extent (3-extent) Y** : generated by a **set of pairs**.*
- ***Canonical element*** of $\mathcal{C}^o{}_1 \times \mathcal{C}^o{}_2$:
  - the **minimum** of all pairs $(\ddot{Y}_1, \ddot{Y}_2)$ from $\mathcal{C}^o{}_1 \times \mathcal{C}^o{}_2$ generating a 3-extent Y.
- Completing the concept (Y, Y^3): the ***intent*** Y^3 is the union of canonical *intents*:
  - $Y^3 = \ddot{Y}^1{}_1 \cup \ddot{Y}^2{}_2$.

Merge: 3-step Construction Procedure





1	***Identify***	*concepts*
2	***Compute***	*intents & extents*
3	***Detect***	*precedence links*

Outline of the Talk

- *FCA: Galois connections, closures, lattices, min. generators …*
- *Computational challenges*
- *Realization within Galicia*

Goals of the *Galicia* project

Develop a tool set to support :

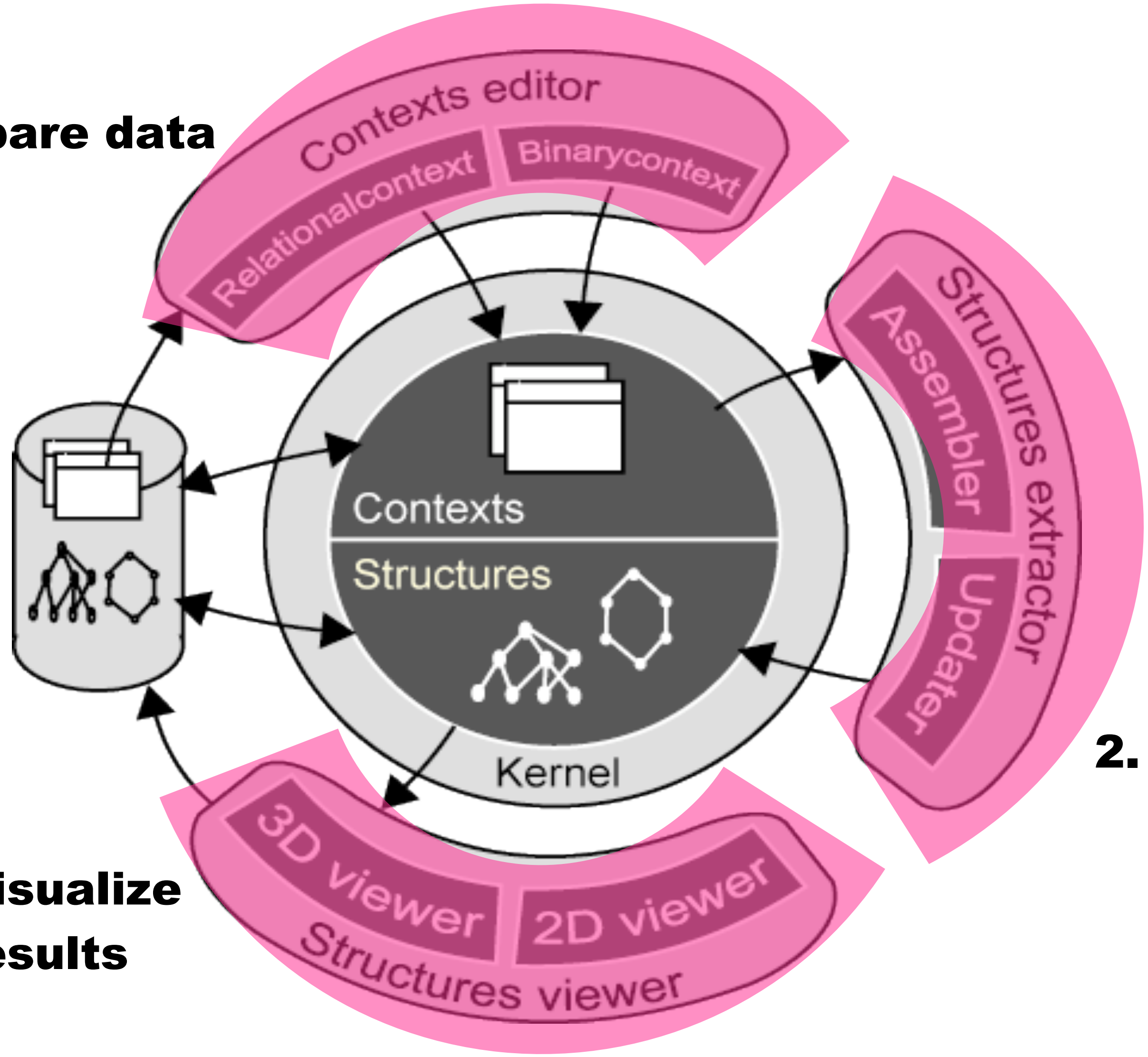
- **Research** on FCA theory and algorithms for the analysis of:
  - **structured** data formats (*data* and *meta-data*):
    - » relational DB, UML models, image meta-data, etc.
  - **semi-structured** data formats (*data* and *meta-data*):
    - » OWL, RDF(S), XMI, etc.
  - volatile datasets,
  - large databases,
- **Practical applications** of FCA techniques to:
  - Data analysis and mining in:
    - » Software engineering,
    - » Bioinformatics,
    - » Image retrieval and mining,
    - » Ontology construction.

Member Teams

- Université de Montréal (Qc, CA)
  - P. Valtchev (Assist. Prof.),
- **U**niversité du **Q**uébec **à** **M**ontréal (CA)
  - R. Godin (Prof.)
- **U**niversité du **Q**uébec en **O**utaouais (CA)
  - R. Missaoui (Prof.)
- LIRMM, Montpellier (FR)
  - M. Huchard (Prof.)
- Université de la Réunion (FR)
  - D. Grosser (Assist. Prof.)
- LORIA, Nancy (FR)
  - A. Napoli (Sen. Res.)

Life-cycle of a Lattice/Rule Set

1. Prepare data



2. Construct concept hierarchy/ rule set

3. Visualize results

The *Galicia* Platform

Rich set of tools for lattices, semi-lattices, general posets, rule bases, etc. :

- **Open-source**

  http://www.iro.umontreal.ca/~galicia (Home Page of the platform)

  https://sourceforge.net/projects/galicia/ (Home Page of the SF project)

- **Portable**: developed in Java,
- **Generic**: abstract types, implementations easily exchangeable.

- Supports different input data formats:
  - Binary data
  - Categorical data
  - Relational Context Families: entities + relations

Key Functions of *Galicia*

Context import/export and edition:

- binary,
- *relational* and *multi-valued*

Construction of **lattices** and derived **structures**:

- ***Lattice*** construction:
  - » **Batch** mode
  - » **Incremental**: object- and attribute-wise
  - » **Merge-based**: object- and attribute-wise
- Galois sub-hierarchies
- **Iceberg** lattices

Association rule extraction from the lattice of intents:

- **Exact rules (valid implications):** *Duquenne-Guigues* basis [Guidues & Duquenne 86], *generic* basis [Pasquier *et al.* 99].
- **Approximate rules (partial implications):** *Luxenburger* bases [Luxenburger 92], *informative* basis [Pasquier *et al.* 99].

Exploration of FCA Results

Structure **visualization** and **navigation** services:

- **Diagram types**:
  - Standard Hasse diagrams,
  - *Nested Line Diagrams (work **in progres**)*,
- **Layout mechanisms** for layered diagrams:
  - Static/dynamic formatting,
  - Layered,
  - Magnetism (attraction - repulsion model).
- **Views**: 2D, 3D, 3D + rotation.
- **Navigation**:
  - hierarchy overview.

I/O operations for various formats:

- dedicated data formats: SLF (*in-house*), IBM,
- XML-based formats: XML DTDs for input data and posets, RCF (*in-house*).

Demo
of

GaLicia

Galois Lattice Interactive Constructor

Research Perspective

On-going research projects:

- ***Relational*** FCA: bring FCA and conceptual data models (UML, E-R, etc.) closer:
  - ***Recursive*** and ***circular*** links in data,
  - **Co-definition** of concepts on different sorts of objects:
    - » *Ex.* Customer, Transaction, Product,
  - Iterative (fixed-point) construction of a set of related lattices
  - … and a bunch of unresolved problems…

- *Evolution* of association rule bases:
  - Merge of **factor bases** along:
    - » the **object** dimension,
    - » the attribute dimension,
  - Decomposition of lattices/posets along different operators

Application Perspective

On-going application projects involving ***Galicia***:

- **Re-engineering** of software **analysis-level models**: extracting high level abstractions from existing conceptual models described in UML;
- **Image retrieval and mining**: lattice products to detect and visualize interactions between lower level and higher level image characteristics,
- **Information** (text) **retrieval**: query analysis and expansion
- **Bio-informatics**: mining 3D structure of proteins (***initial stage***),