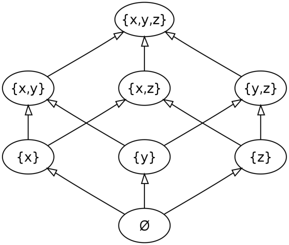
Knowledge Discovery with FCA

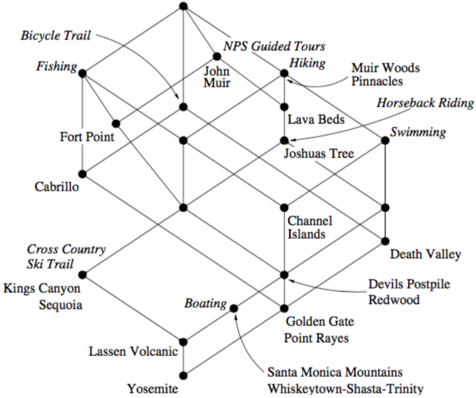
Lecture 4: Introduction to Formal Concept Analysis

Babeş-Bolyai University, Computer Science Department, Cluj-Napoca
csacarea@cs.ubbcluj.ro



Concept Lattice: Implications (Preview)

Def.: An *implication* $X \to Y$ *holds* in a context, if every object that has all attributes from X also has all attributes from Y.



Examples:

- $\{Swimming\} \to \{Hiking\}$
- $\{Boating\} \to \{Swimming, Hiking, NPS\ Guided\ Tours, Fishing, Horseback\ Riding\}$
- $\{Bicycle\ Trail, NPS\ Guided\ Tours\} \to \{Swimming, Hiking, Horseback\ Riding\}$

Concept Lattice

Recapitulation: Lattices

Def.: Let $(P, \leqslant)$ be a partial order and A a subset of P. A *lower bound* of A is an element ℓ of P with $\ell \leqslant a$ for all $a \in A$. An *upper bound* is defined dually. If there is a largest element in the set of all lower bounds of A, it is called the *infimum* of A and is denoted by inf A or $\bigwedge A$; dually, a least upper bound is called *supremum* and denoted by sup A or $\bigvee A$. If $A = \{x, y\}$, we also write $x \wedge y$ for inf A and $x \vee y$ for sup A. Infimum and supremum are frequently also called *meet* and *join*, respectively.

Def.: A partial order $(V, \leqslant)$ is a *lattice* if for any two elements x and y in V the infimum $x \wedge y$ and the supremum $x \vee y$ always exist. $(V, \leqslant)$ is called a *complete lattice* if the supremum $\bigvee X$ and the infimum $\bigwedge X$ exist for any subset X of V. Every complete lattice $(V, \leqslant)$ has a largest element $\bigvee V$, called the *unit element* of the lattice, denoted by 1_V. Dually, the smallest element 0_V is called the *zero element*.

Concept Lattice: The Basic Theorem on Concept Lattices

Def.: For an element v of a complete lattice $(V, \leqslant)$, we define $v_* := \bigvee\{x \in V \mid x < v\}$ and $v^* := \bigwedge\{x \in V \mid v < x\}$. We call v $\bigvee$-*irreducible*, if $v \neq v_*$, i.e., if v cannot be represented as the supremum of strictly smaller elements. In this case, v_* is the unique lower neighbour of v. Dually, we call v $\bigwedge$-irreducible, if $v \neq v^*$. $J(V, \leqslant)$ denotes the set of all $\bigvee$-irreducible elements and $M(V, \leqslant)$ the set of all $\bigwedge$-irreducible elements. A set $X \subseteq V$ is called *supremum-dense* in V, if every element from V can be represented as the supremum of a subset of X and, dually, *infimum-dense*, if $v = \bigwedge\{x \in X \mid v \leqslant x\}$ for all $v \in V$.

Def.: An *isomorphism* between two lattices $(V_1, \leqslant_1)$ and $(V_2, \leqslant_2)$ is a bijective mapping $\varphi : V_1 \to V_2$ such that for all $x, y \in V_1$ holds $x \leqslant_1 y$ if and only if $\varphi(x) \leqslant_2 \varphi(y)$. If such an isomorphism exists, we say that $(V_1, \leqslant_1)$ and $(V_2, \leqslant_2)$ are *isomorphic* and write $(V_1, \leqslant_1) \cong (V_2, \leqslant_2)$.

Concept Lattice: The Basic Theorem on Concept Lattices

Theorem

The concept lattice $\mathfrak{B}(G, M, I)$ *is a complete lattice in which infimum and supremum are given by*

$$\bigwedge_{t\in T}(A_t, B_t) = \left(\bigcap_{t\in T} A_t, \left(\bigcup_{t\in T} B_t\right)''\right) \text{ and } \bigvee_{t\in T}(A_t, B_t) = \left(\left(\bigcup_{t\in T} A_t\right)'', \bigcap_{t\in T} B_t\right)$$

A complete lattice $(V, \leqslant)$ *is isomorphic to* $\mathfrak{B}(G, M, I)$ *if and only if there are mappings* $\tilde{\gamma} : G \to V$ *and* $\tilde{\mu} : M \to V$ *such that*

- $\tilde{\gamma}(G)$ *is supremum-dense in* $(V, \leqslant)$,
- $\tilde{\mu}(M)$ *is infimum-dense in* $(V, \leqslant)$, *and*
- gIm *is equivalent to* $\tilde{\gamma}(g) \leqslant \tilde{\mu}(m)$ *for all* $g \in G$ *and all* $m \in M$.

In particular, $(V, \leqslant) \cong \mathfrak{B}(V, V, \leqslant)$.

Concept Lattice: The Duality Principle

- Let $(V, \leqslant)$ be a (complete) lattice. Then $(V, \geqslant)$ is also a (complete) lattice.
- (cf. with the definition of the dual context)
- If a theorem holds for (complete) lattices, then the 'dual theorem' also holds, i.e., the theorem where all occurrences of $\leqslant, \cap, \cup, \wedge, \vee, \mathbf{1}_V, \mathbf{0}_V$, etc. have been replaced by $\geqslant, \cup, \cap, \vee, \wedge, \mathbf{0}_V, \mathbf{1}_V$, etc.

Computing all concepts

There are several algorithms to compute all concepts:

- naive approach
- intersection method
- Next-Closure (Ganter 1984)
- Titanic (Stumme et al. 2001)
- Inclose family
- and several incremental algorithms

Computing All Concepts: Naive Approach

Theorem

Each concept of a context (G, M, I) has the form (X'', X') for some subset $X \subseteq G$ and (Y', Y'') for some subset $Y \subseteq M$.
Conversely, all such pairs are concepts.

Algorithm

Determine for every subset Y of M the pair (Y', Y'').

Computing All Concepts: Naive Approach

Theorem

Each concept of a context (G, M, I) has the form (X'', X') for some subset $X \subseteq G$ and (Y', Y'') for some subset $Y \subseteq M$.
Conversely, all such pairs are concepts.

Algorithm

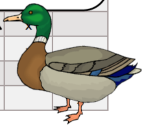
Determine for every subset Y of M the pair (Y', Y'').

Inefficient! (Too) many concepts are generated multiple times.

Calculating Formal Concepts

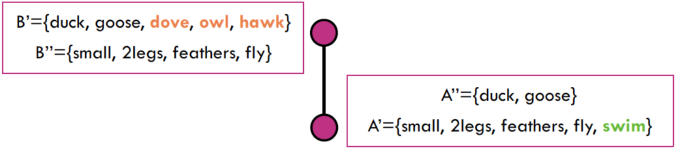
𝕂	small	medium	big	2legs	4legs	feathers	hair	fly	hunt	run	swim	mane	hooves
dove	x			x		x		x					
hen	x			x		x							
duck	x			x		x		x			x		
goose	x			x		x		x			x		
owl	x			x		x		x	x				
hawk	x			x		x		x	x				
lion			x		x		x		x	x		x	
horse			x		x		x			x			
zebra			x		x		x			x			
cow			x		x		x						

1. Pick a set of objects: A = {duck}
2. Derive attributes: A' = {small, 2legs, feathers, fly, swim}
3. Derive objects: (A')' = {small, 2legs, feathers, fly, swim}' = {duck, goose}
4. Formal concept: (A'',A') = ({duck, goose}, {small, 2legs, feathers, fly, swim})



Ordering Concepts

The **formal concept** (A",A')=({duck, goose}, {small, 2legs, feathers, fly, swim}) is represented in the line diagram as a node:



Consider another **formal concept** (B',B")=({duck, goose, dove, owl, hawk},{small, 2legs, feathers, fly}).

The **formal concept** (B',B") is a superconcept of (A",A') and (A",A') is a subconcept of (B',B"), because A" is a subset of B'.

So (B',B") is drawn above (A",A') and connected to it by a line.

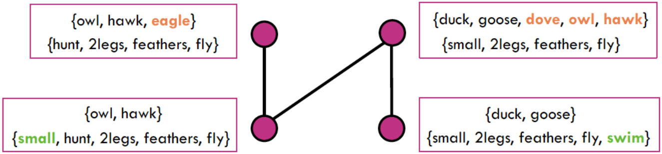
Ordering Concepts

We extend the diagram by adding more **formal concepts**

({owl, hawk}, {feathers, 2legs, small, fly, hunt})

({owl, hawk, eagle}, {feathers, 2legs, fly, hunt})

... and **subconcept relations**:

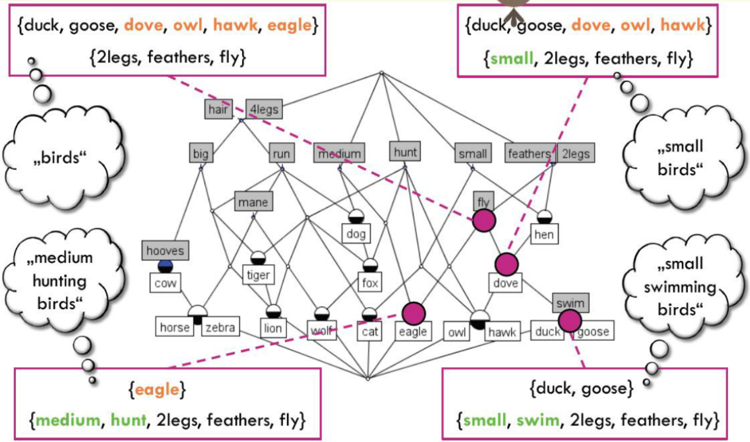


... and so on.

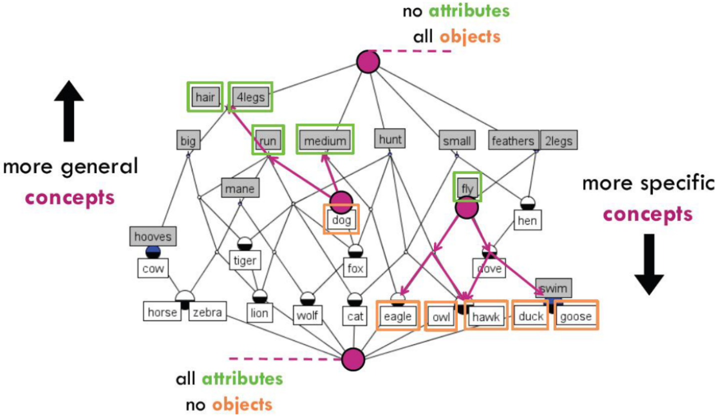
Several methods exist to derive all **formal concepts**:

Cut over extents, Ganter's algorithm etc.

Concept Lattice – Formal Concepts



Concept Lattice – Top and Bottom



Computing All Concepts: Intersection Method

- Suitable for manual computation (Wille 1982)
- Best worst-case time complexity (Nourine, Raynoud 1999)
- Based on the following

Theorem

Every extent is the intersection of attribute extents. (I.e., the closure system of all extents is generated by the attribute extents.)

Which intersections of attribute extents should we take?

Computing All Concepts: Intersection Method

How to determine all formal concepts of a formal context:

1. For each attribute $m \in M$ compute the attribute extent $\{m\}'$.
2. For any two sets in this list, compute their intersection. If it is not yet contained in the list, add it.
3. Repeat until no new extents are generated.
4. If G is not yet contained in the list, add it.
5. For every extent A in the list compute the corresponding intent A'.

Triangles				
abbreviation	coordinates			diagram
T1	(0,0)	(6,0)	(3,1)	
T2	(0,0)	(1,0)	(1,1)	
T3	(0,0)	(4,0)	(1,2)	
T4	(0,0)	(2,0)	$(1,\sqrt{3})$	
T5	(0,0)	(2,0)	(5,1)	
T6	(0,0)	(2,0)	(1,3)	
T7	(0,0)	(2,0)	(0,1)	

Attributes	
symbol	property
a	equilateral
b	isoceles
c	acute angled
d	obtuse angled
e	right angled

	a	b	c	d	e
$T1$		×		×	
$T2$		×			×
$T3$			×		
$T4$	×	×	×		
$T5$				×	
$T6$		×	×		
$T7$					×

Drawing Concept Lattices

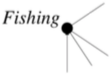
How to draw a concept lattice by hand:

1. Draw a small circle for the extent G at the top.
2. For every extent (starting with the one's with the most elements) draw a small circle and connect it with the lowest circle(s) whose extent contains the current extent.
3. Every attribute is written slightly above the circle of its attribute extent.
4. Every object is written slightly below the circle that is exactly below the circles that are labeled with the attributes of the object.

Drawing Concept Lattices

How you can check the drawn diagram:

1. Is it really a lattice? (that's often skipped)
2. Is every concept with exactly one upper neighbor labeled with at least one attribute?



3. Is every concept with exactly one lower neighbor labeled with at least one object?

4. Is for every $g \in G$ and $m \in M$ the label of the object g below the label of the attribute m iff $(g, m) \in I$ holds?

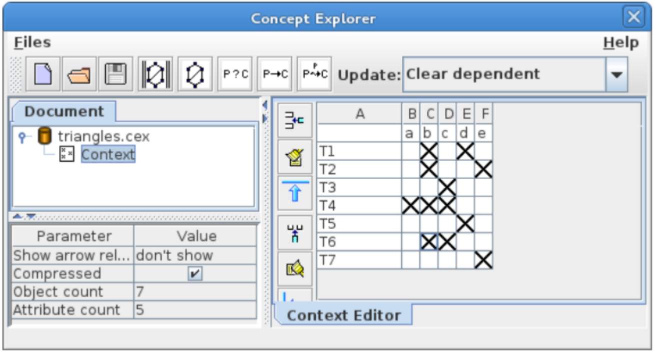
Clarifying and Reducing a Formal Context

Def.: A formal context (G, M, I) is called *clarified* if for every $g_1, g_2 \in G$ with $\{g_1\}' = \{g_2\}'$ holds $g_1 = g_2$ and for every $m_1, m_2 \in M$ with $\{m_1\}' = \{m_2\}'$ holds $m_1 = m_2$.
An object $g \in G$ is called *irreducible* if the *object concept* $(\{g\}'', \{g\}')$ is $\bigvee$-irreducible in $\underline{\mathfrak{B}}(G, M, I)$. Likewise, an attribute $m \in M$ is called *irreducible* if the *attribute concept* $(\{m\}', \{m\}'')$ is $\bigwedge$-irreducible in $\underline{\mathfrak{B}}(G, M, I)$.
A context is called *reduced*, if all its objects and attributes are irreducible.

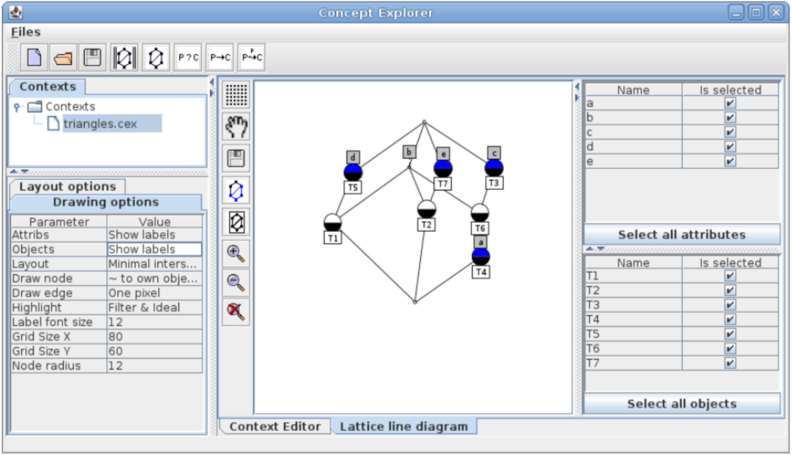
Theorem

A finite context and its reduced context have isomorphic concept lattices. For every finite lattice L there is (up to isomorphism) exactly one reduced context, the concept lattice of which is isomorphic to L, namely its standard context.

Interlude: ConExp

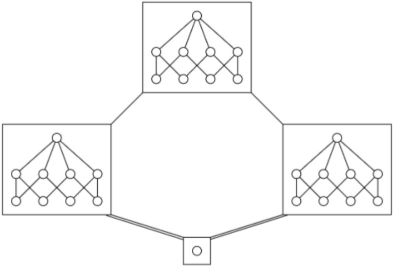


Interlude: ConExp

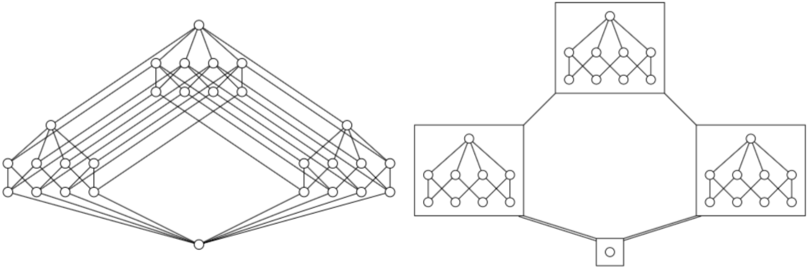


Nested Line Diagrams: Motivation and Idea

- readability of line diagrams often lost for many concepts ($\gtrapprox 50$)
- *nested line diagrams* allow us to go further
- and: support the visualization of changes caused by the addition of further attributes
- *basic idea:* cluster parts of an ordinary diagram and replace bundles of parallel lines between these parts by one line each



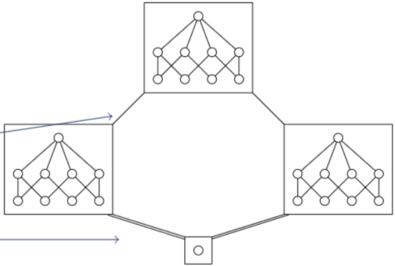
Nested Line Diagrams: Example



The previous concept lattice as ordinary line diagram and as nested line diagram. (For simplification, object and attribute labels have been omitted.)

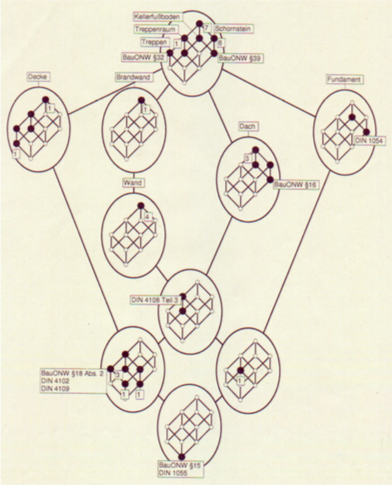
Nested Line Diagrams

- a nested line diagram consists of *boxes* which contain parts of the ordinary diagram and which are *connected by lines*
- simplest case: two boxes that are connected by a line are congruent → corresponding circles are direct neighbors
- double lines between two boxes: every element of the upper box is larger than every element of the lower box



Nested Line Diagrams

- two boxes connected by a single line need not be congruent but contain a part of two congruent figures
- the two congruent figures are drawn as “background structure” into the boxes
- elements are drawn as bold circles if they are part of the respective substructure
- the line connecting both boxes indicates that the respective pairs of elements of the background shall be connected with each other



Nested Line Diagrams: Drawing Example

Die Ducks. Psychogramm einer Sippe.

	generation			sex		financial status		
	older	middle	younger	♂	♀	rich	carefree	indebted
Tick			×	×			×	
Trick			×	×			×	
Track			×	×			×	
Donald		×		×				×
Daisy		×			×		×	
Gustav		×		×			×	
Dagobert	×			×		×		
Annette	×				×		×	
Primus v. Quack	×			×			×	

Taken from: Grobian Gans: *Die Ducks. Psychogramm einer Sippe.* Rowohlt, Reinbek bei Hamburg 1972, ISBN 3-499-11481-X

Nested Line Diagrams: Construction

1. split the attribute set: $M = M_1 \cup M_2$
   (needs not be disjoint, more important: both sets bear meaning)
2. draw the line diagrams of the subcontexts

$$\mathbb{K}_i := (G, M_i, I \cap G \times M_i), i \in \{1, 2\}$$

   and label them with with objects and attributes, as usual

3. sketch a nested diagram of the product of the concept lattices $\underline{\mathfrak{B}}(\mathbb{K}_i)$
   1. draw a large diagram of $\underline{\mathfrak{B}}(\mathbb{K}_1)$ where the concepts are large boxes
   2. draw a copy of $\underline{\mathfrak{B}}(\mathbb{K}_2)$ into each box

Nested Line Diagrams: Labeling

- if a list of elements of $\underline{\mathfrak{B}}(G, M, I)$ exists, enter them according to their intents
- otherwise, enter the object concepts (whose intents can be read off directly from the context) and form all suprema

This gives us another method for determining a concept lattice by hand:

- split up the attribute set as appropriate
- determine the (small) concept lattices of the subcontexts
- draw their product as nested line diagram
- enter the object concepts and close against suprema

This is particularly advisable in order to arrive at a useful diagram quickly.

Nested Line Diagrams: Example

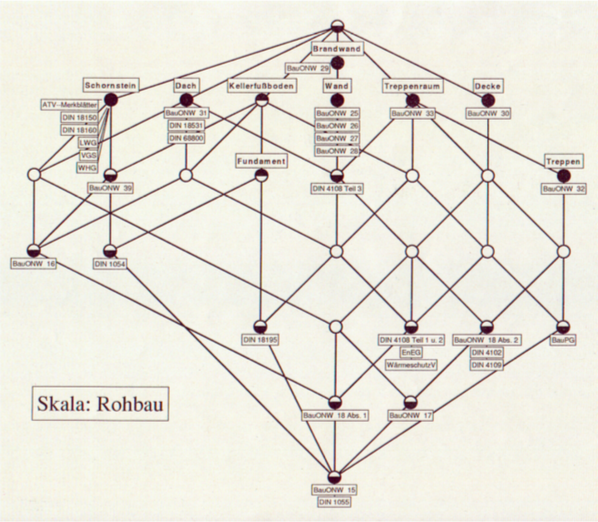
Baurecht in Nordrhein-Westfalen

Taken from: D. Eschenfelder, W. Kollewe, M. Skorsky, R. Wille: *Ein Erkundungssystem zum Baurecht: Methoden der Entwicklung eines TOSCANA-Systems*. In: G. Stumme, R. Wille (Eds.): Begriffliche Wissensverarbeitung – Methoden und Anwendungen. Springer 2000

	Dach	Decke	Wand	Brandwand	Treppen	Treppenraum	Fundament	Kellerfußboden	Schornstein
BauONW 15	×	×	×	×	×	×	×	×	×
BauONW 16	×							×	×
BauONW 17	×	×	×	×	×	×			×
BauONW 18 Abs. 1	×	×	×	×		×		×	×
BauONW 18 Abs. 2	×	×	×	×	×	×			
BauONW 25			×	×					
BauONW 26			×	×					
BauONW 27			×	×					
BauONW 28			×	×					
BauONW 29				×					
BauONW 30		×							
BauONW 31	×								
BauONW 32					×	×			
BauONW 33						×			
BauONW 36									
BauONW 39								×	×
BauONW 40									
BimSchG									
BauPG		×			×	×		×	
EnEG	×	×	×	×		×		×	
WHG									×
LWG									×
WärmeschutzV	×	×	×	×		×		×	
HeizAnlV									
BImSchV									
VGS									×
DIN 1054							×	×	×
DIN 1055	×	×	×	×	×	×	×	×	×
DIN 4102	×	×	×	×	×	×			
DIN 4108 Teil 1 u. 2	×	×	×	×		×		×	
DIN 4108 Teil 3	×		×	×		×			
DIN 4109	×	×	×	×	×	×			
DIN 18150									×
DIN 18160									×
DIN 18195	×		×	×		×	×	×	
DIN 18531	×								
DIN 68800	×								
DIN-Normen für Feuerungsanlagen									
DIN-Normen für Entwässerung									
ATV-Merkblätter									×

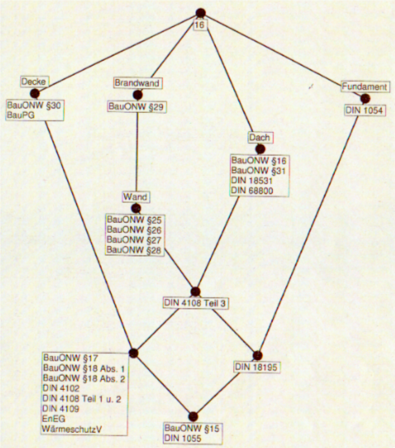
Nested Line Diagrams: Example

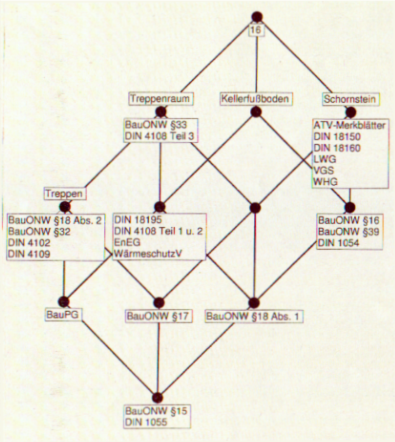
Baurecht in Nordrhein-Westfalen



Nested Line Diagrams: Example

Baurecht in Nordrhein-Westfalen





Nested Line Diagrams: Example

Baurecht in Nordrhein-Westfalen