Using formal concept analysis as a general method for processing websites

Bc. Aleš Sekanina

Master thesis
2010

Tomas Bata University in Zlín
Faculty of Applied Informatics

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Aleš SEKANINA**
Osobní číslo:	**A08492**
Studijní program:	**N 3902 Inženýrská informatika**
Studijní obor:	**Počítačové a komunikační systémy**
Téma práce:	**Využití formální konceptuální analýzy jako obecné metody pro zpracování internetových stránek**

Zásady pro vypracování:

1. **Uveďte základní pojmy a tvrzení teorie uspořádaných množin a teorie svazů.**
2. **Uveďte základní vlastnosti Galoisových konexí na uspořádaných množinách a úplných svazech a jejich souvislost s uzávěrovými operátory. Všechna tvrzení uvádějte bez důkazů s odkazem na doporučenou literaturu.**
3. **Zpracujte přehledně základy formální konceptuální analýzy a formulujte základní reprezentační větu v termínech Galoisových konexí. Uveďte konkrétní příklad kontextu a jeho konceptuálního svazu.**
4. **Popište obecné principy a metody tvorby internetových stránek.**
5. **Na základě doporučené literatury formulujte základní metody vývoje internetových stránek s využitím formalní konceptuální analýzy. Zdůrazněte jejich výhody.**

Rozsah diplomové práce:
Rozsah příloh:
Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **WILLE, R., GANTER, B. Formal Concept Analysis Mathematical Foundations. 1st ed. Springer, 1998. 284 s. ISBN 3-540-62771-5.**
2. **SCHRÖDER, B.S.W. Ordered Sets: an introduction,Birkhäuser Boston, 2003, 391 p.,ISBN 0-8176-41289.**
3. **HARZHEIM, E. Ordered Sets, Springer, 2005, 386 p., ISBN 0-387-24219.8.**
4. **KAUSHIK, A. Web analytics: An Hour a Day. Sybex, 2007. 480 p. ISBN 978-8126513284.**
5. **CLIFTON, B. Advanced Web Metrics with Google Analytics. Wiley Publishing, 2008. 384 s. ISBN 978-0470253120.**
6. **KRUG, S. Don't Make Me Think! A Common Sense Approach to Web Usability 2nd Edition. New Riders Press; 2nd edition, 2005. 216 p. ISBN 978-0321344755.**
7. **BELOHLAVEK, R., VYCHODIL, V. Adding background knowledge to formal concept analysis via attribute dependency formulas. ACM SAC 2008, pp. 938-943. Available from WWW: http://bingweb.binghamton.edu/árbelohla/publications/BeVy_Abkfcaadf.pdf.**
8. **COLE, R., EKLUND, P. Browsing Semi-structured Web texts using Formal Concept Analysis [online].**

Vedoucí diplomové práce: **RNDr. Jiří Klimeš, CSc.**
Ústav matematiky
Datum zadání diplomové práce: **19. února 2010**
Termín odevzdání diplomové práce: **7. června 2010**

Ve Zlíně dne 19. února 2010

UNIVERZITA TOMÁŠE BATI VE ZLÍNĚ
Fakulta aplikované informatiky
L.S.
①

prof. Ing. Vladimír Vašek, CSc.
děkan

prof. Ing. Karel Vlček, CSc.
ředitel ústavu

ABSTRAKT

Teoretická část diplomové práce se skládá ze dvou hlavních oddílů. První z nich poskytuje stručný přehled o teorii svazů, přičemž rozebírá vlastnosti jednotlivých typů uspořádaných množin. Druhá část teoretické části se zabývá formální konceptuální analýzou, jejími matematickými základy a způsoby grafické interpretace.

Praktická část pojednává o obecných metodách vývoje a návrhu internetových stránek a jejich zpracování pomocí webových analytik. Na tyto metody je aplikována formalní konceptuální analýza.

Klíčová slova: Teorie svazů, formální konceptuální analýza, webová analytika, vývoj internetových stránek, struktura internetových stránek

ABSTRACT

Theory of the master thesis consists two main sections. The first one provides a brief overview of the lattice theory and discusses the characteristics of different types of ordered sets. The second section of theory deals with the formal concept analysis, its mathematical foundations and methods of graphical interpretation.

Analysis discusses using formal concept analysis as general method for processing websites. There are described general methods for designing websites and its main structure. The next section shows application formal concept analysis on these methods.

Keywords: Lattice theory, formal concept analysis, web analytics, web development, website structure

ACKNOWLEDGEMENTS

My thank you belongs to my supervisior RNDr. Jiri Klimes, CSc. for leading and coaching to better implementation my gain theoretical knowledge to practical application. Another thank you belongs to doc. Ing. Anezka Lengalova, Ph.D. for advice with academic writing master thesis in English language.

I hereby declare that the print version of my Bachelor's/Master's thesis and the electronic version of my thesis deposited in the IS/STAG system are identical.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně ……………………

podpis diplomanta

CONTENTS

INTRODUCTION

Internet is today considered to be largest information medium. But it is not only information medium. Internet has become the best place for business and advertisement. With explosive growth of Internet grows the number of people using this service. The number of searched keywords, browsed websites, clicked links, shown pictures videos and downloaded files is increasing every day. Using Internet is needed by owners of websites, expecially visiting their sites. They must be careful about mystification their visitors – it is not allowed. Generally there is only one chance to give them the best. Other chance do not have to be next time.-It means lost potential clients, fans or users of their product. Owner of website must be sensitive towards his potencial visitors to adjust it for them. Web analytics employ oneself in understanding of Internet visitors and users.

Web analytics is the measurement, collection, analysis and reporting of internet data for purposes of understanding and optimizing web usage. This internet data is able to get by analytics tools. Analytic tools can measure and collect data of visitors from owner's websites. Owner gets all the collected data in the acceptable form. The next part of web analytics process is analyzing the data. It can be units, tens or thousands records. So it can be time-consuming process. At the end of the data analyzing by human can be everything different. Owner often analyzes data with his best decisions. As ~~a~~ result of conclusions have not to be good optimization for the websites.

This thesis show how to better analyzing with using mathematical model called Formal Concept Analysis. Formal Concept Analysis works with objects and their attributes e.g. data what we get with analytic tools. It is technice based on ordered lattice theory.

The aim of present paper is to present new method of analyzing and reporting in web analytics. It purposes better web personalization with using Formal Concept Analysis. First of all, it is described lattice theory to understanding. In the second part of this work is get general methods for design the website and mainly parts of website. Then will be applicated formal concept analysis to those methods, particularly to web analytics. Here we can see application of formal concept analysis on the practical examples. Every final conclusion is visualized in user-friendly graphs.

I. THEORY

1 LATTICE THEORY

This section contains some of the basic definitions of lattice theory and order theory [1,2,3].

1.1 Ordered set

An ordered set is a set equiped with a special type of binary relation. Recall that abstractly a ninary realtion on a set P is a just subset

$$R \subseteq P \times P = \{(p,q)\colon p,q \in P\}. p,q \in R$$

Simply means that "p is related to q under R". A binary relation R thus contains all the pairs of points that are related to each other under R. For any binary relation R is this text we will write pRq instead of $(p,q) \in R$ however p is related to q via R. The relations of most interest to us are order relations.

Definition 1.1 *An* **ordered set** *(or* **partially ordered set** *or* **poset***) is an ordered pair* $(P, \leq)$ *of a set* P *and a binary relation* $\leq$ *contained in* $P \times P$*, called the* **order** *(or the* **partial order***) on* P*, such that*

1. The relation $\leq$ *is reflexive. That is, each element is related to itself;*

$$\forall p \in P : p \leq p.$$

2. The relation $\leq$ *is antisymmetric. That is, if* p *is related to* q *and* q *is related to* p*, then* p *must equal* q*;*

$$\forall p,q \in P : [(p \leq q) \wedge (q \leq p)]\,) \Rightarrow (p = q).$$

3. The relation $\leq$ *is transitive. That is, if* p *is related to* q *and* q *is related to* r *, then* p *is related to ;*

$$\forall p,q,r \in P : [(p \leq q) \wedge (q \leq r)]\,) \Rightarrow (p \leq r).$$

The elements of P are called the points of the ordered set. Order relations introduce a hierarchy on the underlying set. The statement "$p \leq q$" is read "p is less than or equal to q" or "q is greater than or equal to p". The antisymmetry of the order relation ensures that there are no two-way ties ($p \leq q$ and $q \leq p$

for distinct p and q) in the hierarchy. The transitivity (in conjunction with antisymmetry) ensures that no cyclic ties ($p_1 \leq p_2 \leq \cdots \leq p_n \leq p_1$ for distinct $p_1, p_2, \ldots, p_n$) exist. We will also use the notation $q \geq p$ to indicate that $p \leq q$.

In keeping with the idea of a hierarchy, we will say that p, q 2 P are comparable and write $p \sim q$ iff $p \leq q$ or $q \leq p$. We will write $p < q$ for $p \leq q$ and $p \neq q$.

In this case we will say p is (strictly) less than q. An ordered set will be called finite (infinite) iff the underlying set is finite (infinite).

In many disciplines one investigates sets that are equipped with a structure. In cases where no confusion is possible it is customary to not mention the structure explicitly. Since mostly there will be no confusion possible, we will do the same and often not mention the order explicitly. A phrase such as “Let P be an ordered set” normally will mean that P carries an order that is usually denoted by $\leq$. In case we have several orders under consideration, they will be distinguished using subscripts or different symbols. When we have to talk about orders, we will automatically assume that a property of an order is defined in the same way as a property of the ordered set and vice versa.

1.2 Diagram

Having defined ordered sets as sets equipped with a certain type of relation, we are ready to investigate these entities. Yet it would be helpful to have a visual aid to work with ordered sets. A picture often says more than a thousand words. Such a visual aid is inspired by graph theory, so let us quickly review what a graph is.

Definition 1.2 *A* **graph** *G is a pair (V, E) of a set V (of* **vertices***) and a set $E \in \{\{a, b\} : a, b \in V\}$ (of* **edges***).*

This is a perfectly fine definition. However when working with graphs most people think not of the set theoretical entities of Definition 1.2.. Instead they visualize an entity such as shown in Figure 1. a). The connection is simple: For each vertex $v \in V$ we put a point in the plane (or in 3-space) to represent the vertex. For any two vertices $v, w \in V$ we join the corresponding points with a line (an edge) that does not touch any other points iff $\{v, w\} \in E$. In this fashion we have a good visual tool for the work with graphs and also a way to translate real life problems into mathematics (road networks for example can be modeled using graphs). We could now do the same thing for orders. Put points in the plane or 3-space and join related points with edges. Arrows could indicate the way in which points are related. This idea would have two shortcomings. First, the hierarchical structure of the order may be hard to detect and second, there will be many lines that can be considered superfluous because of transitivity. We shall tackle the second problem first.

Definition 1.3 *Let P be an ordered set. Then* $p \in P$ *is called a* **lower cover** *of* $q \in P$ *(and q is called an* **upper cover** *of p) iff* $p < q$ *and for all* $z \in P$ *we have that* $p \leq z \leq q$ *implies* $z \in \{p, q\}$*. In this case we write* $p \prec q$*. Points p and q that satisfy* $p \prec q$ *or* $q \prec p$ *will also be called* **adjacent**.

Example 1.4 To become familiar with the covering relation, consider.

1. In the power set P($\{1, \dots, 6\}$) the set $\{1, 3\}$ is a lower cover of $\{1, 3, 5\}$, but it is not a lower cover of $\{1, 2, 3, 4\}$.
2. In $\mathbb{Z}$ each number k has exactly one upper cover $(k + 1)$ and one lower cover $(k - 1)$.
3. Whether two elements are covers of each other depends on the surrounding universe. 2 is not an upper cover of 0 in $\mathbb{Z}$, but it is an upper cover of zero in the set of even numbers. Similarly $\{1, 2, 3, 4\}$ is an upper cover of $\{1, 3\}$ in the set of subsets of $\{1, \dots, 6\}$ that have an even number of elements.
4. In infinite ordered sets, elements may or may not have covers. Consider that

in ℝ and ℚ no two elements are covers of each other.

The covering relation carries no superfluous information. It is the smallest relation that carries all the information for a given finite order. To visually incorporate the hierarchy, we only have to impose an up-down direction. The resulting tool that is analogous to the sketch of a graph is the Hasse diagram. Its main use is, because of the difficulty indicated in Example 1.2.3, part 4, for finite ordered sets.

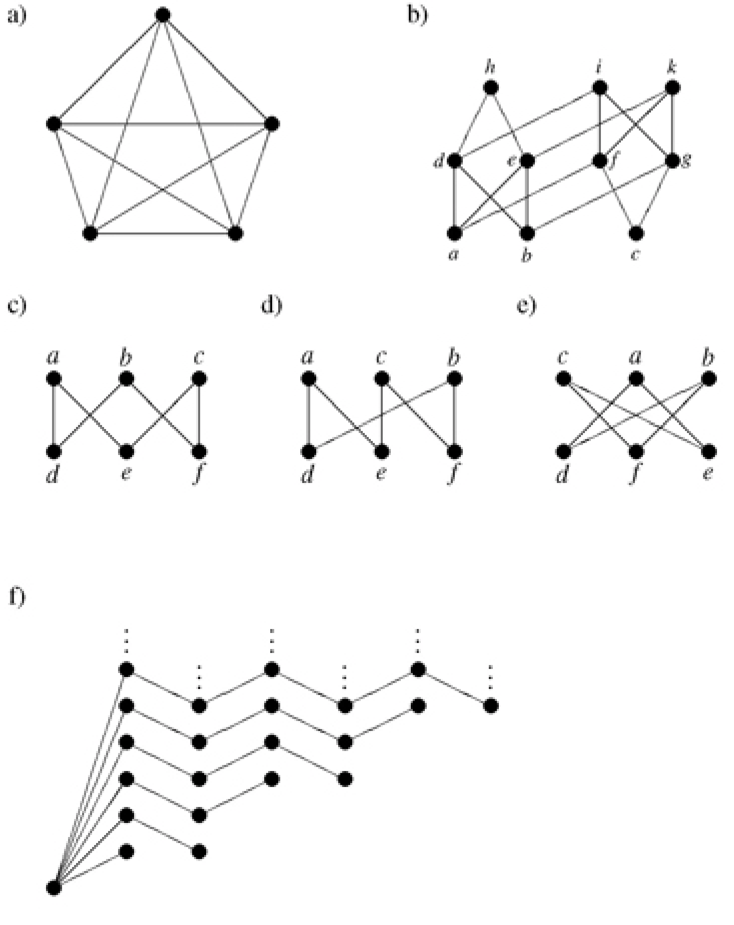


Figure 1 a) The complete graph with five vertices K_5, b) The diagram of an ordered set (the set P2 in [3]), c)–e) Three diagrams for the same ordered set (points to be identified by an isomorphism have the same letters), f) The “diagram” of an infinite ordered set (this “spider”, as Rutkowski named it, is taken from [5]).

Definition 1.5 *The* **(Hasse) diagram** *of a finite ordered set P is the ordered pair $(P, \prec)$, where $\prec$ is the lower cover relation as defined in Definition 1.3.*

Again a perfectly reasonable mathematical definition which gives rise to the following possibility for visualization (cf. Figure 1, b-e).

1. Draw the points of P in the plane (or in 3-space, or on some surface) such that if $p \leq q$, then the point for q has a larger y-coordinate (z-coordinate) than the point for p,
2. If $p \prec q$, draw a line or curve (an edge) between the points corresponding to p and q such that
   - The slope of the edge (or $\frac{\partial}{\partial z}$ if we are in 3-space and just assume that our connecting curves are somewhat smooth) does not change its sign and
   - The edge does not touch any points of P except those corresponding to p and q.

Do not join any other pairs of points by edges.

For purpose of illustration consider Figure 1, part b). We can easily read off the comparabilities in the diagram shown. For example, $a \leq d$, since there is an edge from a to d and a is lower than d.We also have $a \leq k$, since there is an edge going up from a to e and another going up from e to k. This means $a \prec e \prec k$ and by transitivity $a \leq k$. The knowledge that order relations are transitive allows us to avoid drawing some edges that would only clutter the picture. Just imagine all the edges (a, h), (a, i), (a, k), (b, h), (b, i), (b, k), (c, i), and (c, k) added to the picture. It would be quite confusing. Also note that from Figure 1 part b) we see that $c \not\leq h$. Indeed, even though there is an edge from c to , an edge from f to a, an edge from a to d and an edge from d to h, not all the edges are traversed in an upward direction in the trail just described. The edge from f to a is traversed downwards (from f to a), meaning that $c \prec f \succ a \prec d \prec h$. Transitivity cannot be applied to this sequence (and not to any other such sequence) and so $c \not\leq h$.

Drawing diagrams is not canonical. The same ordered set can be drawn in different ways according to the investigator's preferences or needs. As an example, consider Figure 1, parts c)–e). Each picture depicts the same ordered set, yet they do look distinctly different.

Diagrams can also be drawn for infinite ordered sets, but are then in need of explanation. The infinite ordered set depicted in Figure 1 f) consists of one "zigzag" with n elements for each $n \in \mathbb{N}$ such that all "zig-zags" have the same left endpoint. How to draw a diagram of a certain set is often a matter of taste. We will use diagrams extensively as visual tools.

From a relation-theoretic point-of-view the diagram is a certain subset of the order relation, formed according to the rule that only pairs (a, b) are selected for which there is no intermediate point i such that (a, i) and (i, b) are also in the relation. To formalize how to recover the original relation from the diagram we merely need to formulate the reading process indicated above as a mathematical construction.

1.3 Ideals and Filters

1.6 Definition *Let P be a lattice.*

1. *An* **ideal** *of P is a nonempty down-set in L that inherits finite joins, that is,*

- $a \in I, x \leq a \Longrightarrow x \in I$
- $a, b \in I \Longrightarrow a \vee b \in I$

In this case, we write $I \trianglelefteq L$. *A* **proper ideal**, *that is, an ideal* $I \neq L$, *is denoted by* $I \lhd L$. *The set of all ideals of* P *is denoted by* $\mathfrak{I}(L)$.

2. *Dually, a* **filter** *in* P *is a nonempty up-set in* P *that inherits finite meets, that is,*

- $a \in F, x \geq a \Longrightarrow x \in F$
- $a, b \in F \Longrightarrow a \wedge b \in F$

In this case, we write $I \trianglerighteq L$. *A* **proper filter**, *that is, a filter* $F \neq L$, *is*

denoted by $F \rhd L$. *The set of all filters of* P *is denoted by* $F(L)$.

Note that the properties of being an ideal and being a filter are *not* complementary.

Example 1.7 If L is a lattice and $a \in L$, then the principal ideal

$$\downarrow a = \{x \in L |\, x \leq a\}$$

is an ideal of L and the principal filter $\uparrow a$ is a filter.

The intersection $I \cap J$ of two ideals I and J of a lattice L is not empty, since if

$i \in I$ and $j \in J$, then $i \wedge j \in I \cap J$. Hence, $I \cap J$ is an ideal of L. On the other

hand, in the lattice $\mathbb{Z}$ of all integers under the usual order, the intersection of the

family $\{\downarrow n | n \in \mathbb{Z}\}$ of ideals is empty. This leaves us with two alternatives.

We can deal only with lattices with 0 for which $\mathfrak{I}(L)$ is an $\cap$-structure or we

can define

$$\mathfrak{I}_{\emptyset}(L) = \mathfrak{I}(L) \cup \{\emptyset\}$$

which is an intersection-structure for any lattice L.

Theorem 1.8 *Let* L *be a lattice.*

1. *The set* $\mathfrak{I}(L)$ *is a lattice, where meet is intersection and join is given by*

$$I \vee J = \{x \in L | x \leq i \vee j\}\ for\ some\ i \in I\ and\ j \in J$$

2. *2) The set* $\mathfrak{I}_{\emptyset}(L)$ *is an* $\cap$*-structure and therefore a complete lattice, where the join of a family* $\mathcal{F}$ *of ideals is given by*

$$\bigvee \mathcal{F} = \{x \in L | x \leq a_1 \vee \cdots \vee a_n, where\ a_1 \bigcup \mathcal{F}\ \ and\ n \geq 1\}$$

3. *The down map* $\emptyset_{\downarrow}: L \rightarrow \mathfrak{I}_{\emptyset}(L)$ *is a lattice embedding, that is, for all* $a, b \in L$,

   $$\downarrow a \cap \downarrow b = \downarrow (a \wedge b)\ and\ \downarrow a \vee \downarrow b = \downarrow (a \vee b)$$

   The image of $\emptyset_{\downarrow}$ *is the sublattice* $\mathcal{P}(L)$ *of* $\mathfrak{I}_{\emptyset}(L)$ *consisting of all principal ideals of L.*
4. *The union of any* directed *family of ideals in* L *is an ideal in* L.
5. *If L has the ACC, then all ideals are principal. Dually, if L has the DCC then all filters are principal.*

1.4 The Ideal Generated by a Subset

Every nonempty subset A of a lattice L is contained in a smallest ideal, namely, the intersection of all ideals of L containing A.

Definition 1.9 *Let* A *be a nonempty subset of a lattice L.*

1. *The* **ideal generated** *by* A, *denoted by* $(A]$, *is the smallest ideal of L containing* A.
2. *The* **filter generated** *by* A, *denoted by* $[A)$, *is the smallest filter of L containing* A.

Theorem 1.10 *Let L be a lattice and let $A \subseteq L$ be a nonempty subset of L.*

1.

$$(A] = \downarrow [A] = \downarrow \mathcal{T}_A$$

2.

$$(A] = \bigvee\{\downarrow a | a \in A\} = \{x \in L | x \leq a_1 \vee \cdots \vee a_n \;\; for \; a_i \in A\}$$

1.5 Complete poset

It is well known that the union of an arbitrary family of subgroups of a group G need not be a subgroup of G. However, the union of a *chain* of subgroups of G is a subgroup of G. More generally, the union of a *directed family* of subgroups of G is a subgroup of G. Similar statements can be made in the context of other algebraic structures, such as vector spaces or modules.

Actually, the statement about directed sets is *not* more general than the

statement about chains. We will prove that if every chain in a poset P has a join,

then every directed set also has a join.

Definition 1.11 *Let P be a poset.*

1) P is **chain-complete** *or* **inductive** *if every chain in P, including the empty*

chain, has a join. Thus, P has a smallest element.

2) P is **complete** *or a* **CPO** *if P has a smallest element and if every directed*

subset of P has a join.

We should mention that the adjective *complete* has a different meaning when

applied to posets than when applied to lattices. Complete lattices

are defined in a later chapter.

Theorem 1.12 *An infinite directed poset D is the union of a strictly increasing*

transfinite sequence

$$\mathcal{S} = \langle D_\alpha | \alpha < |D| \rangle$$

of directed subsets D_α for which $|D_\alpha| < |D|$. In particular, if α is finite, then

$|D_\alpha|$ *is finite and if* α *is infinite, then* $|D_\alpha| < |\alpha|$.

The finite part of the sequence is easy to define: At each step, just include the smallest element of D (under the well-ordering) not yet in the sequence, along with an upper bound (under the partial order) for what has come before.

Specifically, let $D_0 = \{d_0\}$ and let

$$D_{k+1} = D_k \cup \{e_{k+1}, u_{k+1}\}$$

where e_{k+1} is the least element not in D_k and u_{k+1} is any upper bound for the finite set $D_k \cup \{e_{k+1}\}$. It is clear that $D_k | k \in \mathbb{N} >$ is a strictly increasing sequence of finite directed (in fact, bounded above) sets. Moreover,

$$\{d_0, \dots, d_k\} \subseteq D_k$$

Theorem 1.13 *A poset* P *is complete if and only if it is chain-complete.*

Complete lattices

Definition Let P be a poset.

1) *P is a* ***complete lattice*** *if P has arbitrary meets and arbitrary joins.*

Example 1.14

1) A singleton set $L = \{a\}$ is a lattice under the only possible order on L. This is a trivial lattice. Any lattice with more than one element is a nontrivial lattice.

2) Any totally ordered set is a lattice, but not necessarily a complete lattice. For example, the set $\mathbb{Z}$ of integers under the natural order is a lattice, but not a complete lattice.

3) The set $\mathbb{N}$ of natural numbers is a lattice under division, where

$$a \wedge b = \gcd(a, b) \text{ and } a \vee b = \operatorname{lcm}(a, b)$$

4) If S is a nonempty set, then the power set $\wp(S)$ is a complete lattice under union and intersection.

5) If P is a poset, then the collection $\mathcal{O}(P)$ of down-sets of P is a complete

lattice, where meet is intersection and join is union.

6) If G is a group, then the family $S(G)$ of subgroups of G is a complete lattice under set inclusion, where meet is intersection. Similar statements can be made for other algebraic objects, such as the ideals (or subrings) of a ring, the submodules of a module or the subfields of a field.

1.6 Distributive lattices

This section describes the basic properties of distributive lattices. First, in a distributive lattice, maximal ideals are prime.

Theorem 1.15 *In a distributive lattice L, all maximal ideals are prime.*

In a distributive lattice, elements are join-irreducible if and only if they are joinprime.

Theorem 1.16 *Let L be a distributive lattice.*

1. *The following are equivalent for $x \in L$:*
    - *a) x is join-irreducible*
    - *b) x is join-prime*
    - *c) $\uparrow x$ is a prime filter, or equivalently, $(\uparrow x)^{c}$- is a prime ideal.*
2. *Dually, the following are equivalent for $x \in L$:*
    - *a) x is meet-irreducible*
    - *b) x is meet-prime*
    - *c) $\downarrow x$ is a prime ideal.*

We have seen that the down map $\emptyset_{\downarrow}$ is an order embedding and so its restriction $\emptyset_{\downarrow}: \mathcal{M}(L) \hookrightarrow \text{Spec}\ (L)$ is an order embedding of meet-irreducibles into prime ideals. Similarly, the up map is an anti-embedding and so the *not-up map* $\emptyset_{\neg\uparrow}: \mathcal{J}(L) \to \text{Spec}\ (L)$ defined by

$$\emptyset_{\neg\uparrow}(a) = (\uparrow a)^{c}$$

an order embedding of join-irreducibles into prime ideals. More can be said if L has a chain condition.

Theorem 1.17 *Let L be a distributive lattice.*

1. The not-up map $\emptyset_{\neg\uparrow}: \mathcal{J}(L) \to \mathrm{Spec}\ (L)$*defined by*

$$\emptyset_{\neg\uparrow}(a) = (\uparrow a)^c$$

is an order embedding. If L has the DCC, then all prime ideals have the

form $(\uparrow a)^c$ *for* $a \in \mathcal{J}(L)$ *and so*

$$\emptyset_{\neg\uparrow}: \mathcal{J}(L) \approx \mathrm{Spec}\ (L)$$

is an order isomorphism.

2) *The down map* $\emptyset_{\downarrow}: \mathcal{M}(L) \to \mathrm{Spec}\ (L)$ *defined by*

$$\emptyset_{\downarrow}(m) = \downarrow m$$

is an order embedding. If L has the ACC, then

$$\emptyset_{\downarrow}: \mathcal{M}(L) \approx \mathrm{Spec}\ (L)$$

is an order isomorphism.

3) *If L is finite, then*

$$\emptyset_{\downarrow} \circ \emptyset_{\neg\uparrow}: \mathcal{J}(L) \approx \mathcal{M}(L)$$

and so $\mathcal{J}(L)$ and $\mathcal{M}(L)$ are isomorphic.

Distributive lattices also have nice properties with respect to finiteness.

Theorem 1.18 *For a distributive lattice L, the following are equivalent:*

1) *L is finite*

2) *L has finite length*

3) *L has no infinite chains* (*equivalently, L has both chain conditions*).

1.7 Algebraic lattices

The finitely-generated subalgebras of an algebra A have a special property,

namely, if a finitely-generated subalgebra S is contained in the join of an arbitrary family $\mathcal{F}$ of subalgebras, then S is contained in the join of a finite subfamily of $\mathcal{F}$, that is,

$$S \subseteq \bigvee \mathcal{F} \implies S \subseteq \bigvee \mathcal{F}_0$$

for some finite subfamily $\mathcal{F}_0$ of $\mathcal{F}$. This motivates the following definitions.

Definition 1.19 *Let L be a complete lattice and let $a \in L$.*

1) A **join-cover** *of a is a subset B of L for which*

$$a \leq \bigvee B$$

2) The element a is **compact** *if every join-cover of a has a finite join-subcover, that is,*

$$a \leq \bigvee B \Rightarrow a \leq \bigvee B_0$$

for some finite subset $B_0 \subseteq B$

The set of compact elements of P is denoted by $\mathcal{K}(L)$.

Theorem 1.20 *Let L be a complete lattice. An element $a \in L$ is compact if and only if every directed join-cover of a has a finite (therefore singleton) join subcover, that is,*

$$a \leq \overset{\longrightarrow}{\bigvee} D \implies a \leq d \; for \; some \; d \in D$$

Example 1.21

1) All elements of a finite lattice are compact.

2) The compact elements of $\wp(X)$ are the finite subsets of X.

3) The compact elements of the subgroup lattice $\mathcal{S}(G)$ of a group G are the finitely-generated subgroups.

4) The compact elements of the subspace lattice $\mathcal{S}(V)$ of a vector space V are the finite-dimensional subspaces.

Theorem 1.22 *Let L be a complete lattice.*

1) $0 \in L$ *is compact.*

2) *The compact elements* $\mathcal{K}(L)$ *inherit finite joins from* L *and so* $\mathcal{K}(L)$ *is a join-semilattice with* 0*.*

3) *The ideals of* $\mathcal{K}(L)$*are precisely the sets of the form* $\mathcal{K}_{\downarrow}(\mathrm{a})$ *for* $a \in L$*.*

Definition 1.23 *A complete lattice* L *is* **algebraic**, *or* **compactly-generated** *if the compact elements of* L *are join-dense in* L*.*

Note that not all authors require completeness in the definition of an algebraic lattice.

All finite lattices are algebraic. We have just seen that the subalgebra lattice $\mathrm{Sub}(A)$ of an algebra is algebraic and we will show that every algebraic lattice is the subalgebra lattice for some algebra.

1.7.1 $\bigcap \overrightarrow{\bigcup}$-structures

Subalgebra lattices are ∩-structures but, in general, they do not inherit arbitery unions from $\wp(X)$. However, they do inherit *directed* unions. This follows from the finitary nature of the operations in an algebra. For if $\mathcal{F} = \{S_i | i \in I\}$ is a directed subfamily of $\mathcal{S}$ and if $U = \bigcup^{\rightarrow} \mathcal{F}$, then

$$x_1, \dots x_n \in U \Longrightarrow x_1, \dots x_n \in F \text{ for some } F \in \mathcal{F}$$

and so $f(x_1, \dots x_n) \in F \subseteq U$ for all $f \in \Omega$. Hence, U is a subalgebra of X.

The following definition is motivated by this example.

Definition 1.24 *If a nonempty subset* $\mathcal{M}$ *of a power set* $\wp(X)$ *inherits arbitrary meets and directed joins, we call* $\mathcal{M}$ *an* $\bigcap \overrightarrow{\bigcup}$***-structure****.*

Some authors refer to $\bigcap \overrightarrow{\bigcup}$-structures as **algebraic intersection-structures**, but we prefer to avoid this terminology, since intersection-structures are komplete lattices and the term *algebraic* has a different definition when applied to complete lattices (as we will see a bit later).

It is not hard to characterize the compact subsets of an $\bigcap \overrightarrow{\bigcup}$-structure $\mathcal{M}$ in X

and to show that an $\bigcap \overrightarrow{\bigcup}$-structure is an algebraic lattice. Recall that the $\mathcal{M}$ - closure of a subset S of X is defined by

$$\langle S\rangle = \bigcap\{M \in \mathcal{M} | S \subseteq M\}$$

Definition 1.25 *Let* $\mathcal{M}$ *be an* $\cap\ \vec{\cup}$*-structure in* X*. A set in* $\mathcal{M}$ *that is the* $\mathcal{M}$*-closure* $\langle S\rangle$ *of a finite subset* S *of* X *is called a* **finitely-generated** *element of* $\mathcal{M}$*.*

Any finitely-generated element $\langle F\rangle$ of $\mathcal{M}$ is compact, for if

$$\langle F\rangle \subseteq \overrightarrow{\bigcup}\{M_i | M_i \in \mathcal{M}, i \in I\}$$

then F is also contained in this directed union and so $F \in M_j$ for some $j \in I$.

Hence, $\langle F\rangle \subseteq M_j$. Conversely, suppose that K is compact in $\mathcal{M}$. The family of all $\mathcal{M}$ -closures $\langle F\rangle$ of finite subsets of K is directed, for if F_1 and F_2 are finite, then $\langle F_1\rangle, \langle F_1\rangle \subseteq \langle F_1 \cup F_2\rangle$. Hence,

$$K \subseteq \overrightarrow{\bigcup}\{\langle F\rangle | F \subseteq K, F \text{ finite}\}$$

finite

and so there is a finite subset F of K for which $K \subseteq \langle F\rangle$. Since the reverse inclusion is clear, we have $K = \langle F\rangle$.

Theorem 1.26 *Let* $\mathcal{M}$ *be an* $\cap\ \vec{\cup}$*-structure in* $\wp(X)$*.*

1) The compact elements of $\mathcal{M}$ *are the finitely-generated elements of* $\mathcal{M}$*.*

2) $\mathcal{M}$ *is an algebraic lattice.*

Theorem 1.27 *An algebraic lattice* L *is isomorphic to the ideal lattice of a joinsemilattice with* 0*. In particular, if* $\mathcal{K} = \mathcal{K}(L)$*, then the* $\mathcal{K}$*-down map*

$$\emptyset_{\mathcal{K},\downarrow}: L \approx \mathfrak{I}(\mathcal{K}(L))$$

defined by

$$\emptyset_{\mathcal{K},\downarrow}(a) = \mathcal{K}_{\downarrow}(a)$$

is an isomorphism. Moreover, $\mathfrak{I}(\mathcal{K}(L))$ *is an* $\cap\ \vec{\cup}$*-structure in* $\wp(\mathcal{K}(L))$ *and so* *every algebraic lattice is isomorphic to an* $\bigcap \overrightarrow{\bigcup}$*-structure.*

1.8 Galois connections

Definition 1.28 *Let* P *and* Q *be posets. A* **Galois connection** *on the pair* (P, Q) *is a pair* (Π, Ω) *of maps* $\Pi: P \rightarrow Q$ *and* $\Omega: Q \rightarrow P$*, where we write* $\Pi(p) = p^*$ *and* $\Omega(q) = q^*$ *with the following properties:*

2) (**Order-reversing**) *For all* $a, b \in P$ *and* $r, s \in Q$,

$$p \leq q \Longrightarrow p^* \geq q^* \quad r \leq s \Longrightarrow r^{'} \geq s^{'}$$

3) (**Extensive**) *For all* $p \in P, q \in Q$,

$$p \leq p^{*'} \Longrightarrow q \leq q^{'*}$$

Example 1.29 If P is a poset, then the maps $u, \ell: \wp(P) \longrightarrow \wp(P)$ defined by

$$u(A) = A^u \;\; and \;\; \ell(A) = A^\ell$$

for any $A \subseteq P$ form a Galois connection on the power set $\wp(P)$, that is, the pair $(u, \ell)$is a Galois connection for $(\wp(P), \wp(P))$.

Example 1.30 Let $X$and Y be nonempty sets and let $R \subseteq X \times Y$ be a relation on $X \times Y$. Then the maps

$$S \subseteq X \mapsto S^* = \{y \in Y | xRy \; for \; all \; x \in S\}$$

and

$$T \subseteq Y \mapsto T^{'} = \{x \in X | xRy \; for \; all \; y \in T\}$$

form a Galois connection on $(\wp(X), \wp(Y))$.

Example 1.31 Let F and E be fields with $F \subseteq E$. Let $G = G(E|F)$ be the group of all automorphisms of l that fix each element of F. The group G is called the **Galois group** of the field extension $F \leq E$. The most famous example of a Galois connection is the **Galois correspondence** between the intermediate fields $F \subseteq K \subseteq E$ and the subgroups H of the Galois group G. This correspondence is given by

$$F \mapsto G(E/K) := \{\sigma \in G | \sigma x = x \text{ for all } x \in K\}$$

and

$$H \mapsto fix(H) \coloneqq \{x \in E | \sigma x = x \text{ for all } x \in H\}$$

Example 1.32 Let $n > 1$ and let F be a field. Let $F[x_1, \dots, x_n]$ denote the ring of polynomials in n variables over F and let F^n denote the ring of all n-tuples over F. Let

$$P = \wp(F[x_1, \dots, x_n]) \text{ and } Q = \wp(F^n)$$

Let $\Pi: F[x_1, \dots, x_n] \longrightarrow \mathcal{P}(F^n)$ be defined by

$$\Pi(S) = \text{Set of all common roots of the polynomials in}$$

$$= \{x \in F^n |\, p(x) = 0 \text{ for all } p \in S\}$$

and let $\Omega: \mathcal{P}(F^n) \longrightarrow F[x_1, \dots, x_n]$ be defined by

$$\Omega(\mathrm{T}) = \text{Set of all polynomials whose root set includes } T$$

$$= \{p \in F[x_1, \dots, x_n] |\, p(t) = 0 \text{ for all } t \in T\}$$

The pair (Π, Ω) is a Galois connection on (P, Q).

There are two closure operators associated with a Galois connection.

Theorem 1.33 *Let* (Π, Ω) *be a Galois connection on* (P, Q)*. Then*

$$p^{*'*} = p^* \textit{ and } q^{'*'} = q^{'}$$

for all $p \in P$ *and* $q \in Q$*. It follows that the composite maps*

$$p \rightarrow p^{*'} \text{ and } q \rightarrow q^{'*}$$

are closure operators on P *and* Q*, respectively. Furthermore,*

1) p^* *is closed for all* $p \in P$.

2) $q^{'}$ *is closed for all* $q \in Q$.

3) The maps $\Pi\colon \mathrm{P} \rightarrow \mathrm{Cl(Q)}$ *and* $\Omega\colon \mathrm{Q} \rightarrow \mathrm{Cl(P)}$ *are surjective and the restricted maps* $\Pi\colon \mathrm{Cl(P)} \rightarrow \mathrm{Cl(Q)}$ *and* $\Omega\colon \mathrm{Cl(Q)} \rightarrow \mathrm{Cl(P)}$ *are inverse bijections. Thus,*

$$Cl(P) = \{q^{'} |\, q \in Q\} \textit{ and } Cl(Q) = \{p^* |\, p \in P\}$$

Definition 1.34 *A closure operator on a poset* P *is a unary operator* cl *on* P *with the following properties:*

1) (**extensive**)

$$p \leq \mathrm{cl}(p)$$

2) (**monotone**)

$$p \leq q \implies \mathrm{cl}(p) < \mathrm{cl}(q)$$

3) (**idempotent**)

$$\mathrm{cl}\big(\mathrm{cl}(p)\big) = \mathrm{cl}(p)$$

The element $\mathrm{cl}(p)$ is called the **closure** of p. An element $p \in P$ is **closed** if it is equal to its own closure, that is, if $p = \mathrm{cl}(p)$. We denote the closed elements in P by $\mathrm{Cl}(P)$. A closure operator on a power set $\wp(X)$ is often referred to as a closure operator **on** X.

1.9 Properties of Lattices

We now resume our discussion of the properties of lattices.

1.9.1 Lattices as Algebraic Structures

Since a lattice L is a meet-semilattice and a join-semilattice, Theorem 1.35 implies that meet and join are associative, commutative and idempotent operations. The link between the two operations is provided by the **absorption laws**

$$a \wedge (a \vee b) = a$$

and

$$a \vee (a \wedge b) = a$$

In fact, these properties characterize lattices.

Theorem 1.35 *The following properties hold in a lattice L:*

L1) (**associativity**)

$$(a \vee b) \vee c = a \vee (b \vee c)$$

$$(a \wedge b) \wedge c = a \wedge (b \wedge c)$$

L2) (**commutativity**)

$$a \vee b = b \vee a$$

$$a \wedge b = b \wedge a$$

L3) (**idempotency**)

$$a \vee a = a$$

$$a \wedge a = a$$

L4) (**absorption**)

$$a \vee (a \wedge b) = a$$

$$a \wedge (a \vee b) = a$$

Conversely, if P is a nonempty set with two binary operations $\wedge$ and $\vee$ satisfying L1)*–L4*)*, then P is a lattice where meet is $\wedge$, join is $\vee$ and the order relation is given by*

$$a \leq b \text{ if } a \vee b = b$$

or equivalently,

$$a \leq b \text{ if } a \wedge b = a$$

Moreover, since the set of axioms L1)*–L4*) *is self-dual, it follows that if a statement holds in every lattice, then any dual statement holds in every lattice.*

1.9.2 Varieties of Lattices

Theorem 1.35 shows that lattices can be defined *equationally*, that is, by identities. More generally, a **variety** or **equational class** of lattices is a family of lattices defined by a set of lattice identities. Theorem 1.35 says that the family of all lattices is an)-based variety of lattices, since there are) identities in.

Theorem 1.35

As another example, *distributive lattices* are those lattices L that satisfy the two distributive identities

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$$

$$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$

for all $a, b, c \in L$. Thus, the distributive lattices form a 10-based variety of lattices.

1.9.3 Tops and Unbounded Sequences

If a lattice L has no largest element, then L must be infinite and it seems reasonable that L should have some form of strictly increasing infinite semence with no upper bound. This is easy to see when L is countably infinite. For if $L = (a_1, a_2, \ldots)$ has no largest element, the sequence of finite joins

$$\mathcal{S} = \langle s_k \rangle = \langle a_1 \vee \cdots \vee a_k \rangle$$

has no upper bound, since $\mathcal{S} \leq u$ implies that $L \leq u$. Removing all duplicates from $\mathcal{S}$ gives a strictly increasing infinite sequence with no upper bound.

For an arbitrary infinite lattice L with no largest element, we can well-order L

and then Theorem 2.20 implies that L contains a cofinal transfinite subsequence

$$\mathcal{S} = \langle b_\alpha | \alpha < \delta \rangle$$

compatible with the order of L, that is,

$$b_\alpha < b_\beta \implies b_\alpha \leq b_\beta \text{ or } b_\alpha \parallel b_\beta$$

Let us assume that all duplicates have been removed from this subsequence and so

$$b_\alpha < b_\beta \implies b_\alpha < b_\beta \text{ or } b_\alpha \parallel b_\beta$$

To eliminate the problem of parallel elements, let $\mathcal{K}$ be a maximal chain in $\mathcal{S}$

(with respect to the order of L), which must exist by the Hausdorff maximality

principle. Then $\mathcal{K}$ also has no upper bound in L. For if $\mathcal{K} \leq u$ for some $u \in L$,

then u must be the largest element of L. Otherwise, there is an $a \in L$ for which

$u < a$ or $u \parallel a$. In either case, $\mathcal{K} \leq u < u \vee a \leq s$ and the cofinality of S in L

implies that there is an $s \in S$ for which $\mathcal{K} < u \vee a \leq s$. Hence, $\mathcal{K} \cup \{s\}$ is a

chain in S, which contradicts the maximality of $\mathcal{K}$. Thus, $\mathcal{K}$ is a strictly

increasing transfinite sequence in L with no upper bound.

Theorem 1.36

1) A countably infinite lattice L has no largest element if and only if L has a *nonempty strictly increasing sequence with no upper bound in L.*

2) *An infinite lattice L has no largest element if and only if it has a nonempty strictly increasing transfinite sequence*

$$\mathcal{B} = \langle b_\alpha | \alpha < \mathcal{K} \rangle$$

for some ordinal $\varkappa$, with no upper bound in L.

Of course, the previous theorem has a dual, which we leave to the reader to formulate.

1.9.4 Bounds for Subsets

On a related matter, given a nonempty subset of a lattice L, can we find a strictly increasing sequence of elements of L that has the same set of upper bounds as A? This is trivial if A is a finite set: Just take the one-element sequence consisting of the join $\bigvee A$.

For a countably infinite set A, we can sequentially order A to get $A = (a_1, a_2, \dots)$, then take the sequence of finite joins as before

$$\mathcal{S} = \langle s_k \rangle = \langle a_1 \vee \cdots \vee a_k \rangle$$

Removing all duplicate entries gives a sequence that has the same upper bounds as A.

If A is an infinite set of arbitrary cardinality $|A| = \Sigma$, then we can well-order E to get a transfinite sequence

$$\mathcal{A} = \langle a_\alpha | \alpha < \Sigma \rangle$$

of length Σ. Since Σ is an initial ordinal, if $\epsilon < \Sigma$, then $|\epsilon| < |\Sigma|$. Thus, to mimic the proof for the countably infinite case, we assume that every nonempty subset of A of cardinality less than Σ has a join. Then for any $\epsilon < \Sigma$, the transfinite subsequence

$$\mathcal{A}_\epsilon = \langle a_\alpha | \alpha < \epsilon \rangle$$

of $\mathcal{A}$ has a join

$$u_\epsilon = \bigvee \langle a_\alpha | \alpha < \epsilon \rangle$$

Moreover, the transfinite sequence of joins

$$\mathcal{B} = \langle u_\epsilon | \epsilon < \Sigma \rangle$$

is nondecreasing and $\mathcal{B}^u = A^u$. As before, we can cast out any duplicate entries in $\mathcal{B}$ to get a strictly increasing sequence.

Theorem 1.37 *Let L be a lattice.*

1) If

$$A = \{a_1, a_2, \dots\}$$

is a countably infinite subset of L, then there is a strictly increasing sequence $\mathcal{S}$ in L for which $\mathcal{S}^u = A^u$.

2) If A is an infinite subset of L of cardinality $|A| = \Sigma$ and if every nonempty subset of A of cardinality less than Σ has a join in L, then there is a strictly increasing transfinite sequence

$$\mathcal{S} = \langle s_\alpha | \alpha < \delta \rangle$$

in L for which $\mathcal{S}^u = A^u$.

2 FORMAL CONCEPT ANALYSIS

Formal concept analysis (FCA) is a mathematical theory for concepts and concept hierarchies that reflects an understanding of “concept” which is first mentioned explicitly in the Logic of Port Royal in 1668. FCA explicitly formalises extension and intension of a concept, their mutual relationships, and the fact that increasing intent implies decreasing extent and vice versa. Based on lattice theory, it allows to derive a concept hierarchy from a given dataset. FCA complements thus the usual ontology engineering approach, where the concept hierarchy is modeled manually.[4]

2.1 Basic definitions

Definition 2.1 A (formal) context *is a triple* $\mathbb{K} := (G, M, I)$, *where* G *is a set whose elements are called* objects, M *is a set whose elements are called* attributes, *and* I *is a binary relation between* G *and* M *(i.e.* $I \subseteq G \times M$*).* $(g, m) \in I$ *is read “the object g* has *the attribute* m*”.*

Example 2.2 The left part of Figure 2 shows a formal context developed for an educational movie about living beings and water[4]. The object set G comprises the eight living beings that were discussed in the movie, and the attribute set M lists features that distinguish the living beings. The binary relation I is given by the cross table and describes which living being has which of the attributes.

For a given context (G, M, I), we can define two *derivation operators*, both denoted by the _ symbol. They are used for defining formal concepts.

Definition 2.3 *For* $A \subseteq G$, *let* $A' := \{m \in M \mid \forall g \in A\colon (g, m) \in I\}$ *and, for* $B \subseteq M$, *let* $B' := \{g \in G \mid \forall m \in B\colon (g, m) \in I\}$. A (formal) concept *of a formal context* (G, M, I) *is a pair* (A, B) *with* $A \subseteq G, B \subseteq M, A' = B$ *and* $B^{'} = A$. *The sets* A *and* B *are called the* extent *and the* intent *of the formal concept* (A, B), *respectively. The* subconcept–superconcept relation *is*

	Needs water to live	Lives in water	Lives on land	Needs chlorophyll to produce food	Two seed leaves	One seed leaf	Can move around	Has limits	Suckless its offspring
Leech	X	X					X		
Bream	X	X					X	X	
Frog	X	X	X				X	X	
Dog	X		X				X	X	X
Spike-weed	X	X		X		X			
Reed	X	X	X	X		X			
Bean	X		X	X	X				
Maize	X		X	X		X			

Table 1 A formal context about living beings.

	Two seed leave	One seed leaf
0		
1		X
2	X	

Table 2 A conceptual scale for the many-valued attribute “Number of seed leaces”

formalised by $(A_1, B_1) \leq (A_2, B_2) :\Longleftrightarrow A_1 \subseteq A_2 (\Longleftrightarrow B_1 \supseteq B_2)$. *The set of all formal concepts of a context* $\mathbb{K}$ *together with this order relation is always a complete lattice, called the* concept lattice *of* $\mathbb{K}$ *and denoted by* $\underline{\mathfrak{B}}(\mathbb{K})$.

Theorem 2.4 *The concept lattice* $\underline{\mathfrak{B}}(\mathbb{K})$ *of a context* $\mathbb{K} := (G, M, I)$ *is a complete lattice*[6] *in which infimum* $(\bigwedge)$ *and supremum* $(\bigvee)$ *of a set* $\{(A_t, B_t) | t \in T\}$ *of concepts (where* T *is any index set) are given by*

$$\bigwedge_{t\in T}\langle A_t, B_t\rangle = \langle \bigcap_{t\in T} A_t\,, \left(\bigcap_{t\in T} A_t\right)'\rangle = \langle \bigcap_{t\in T} A_t\,, \left(\bigcup_{t\in T} B_T\right)''\rangle,$$

A complete lattice V *is isomorphic to* $\underline{\mathfrak{B}}(\mathbb{K})$ *if and only if there are mappings* $\tilde{\gamma} : G \rightarrow \mathbb{V}$ *and* $\tilde{\mu}: M \rightarrow \mathbb{V}$ *such that* $\tilde{\gamma}(G)$ *is supremum-dense in* $\mathbb{V}$ *[i.e. each element in* $\mathbb{V}$ *is supremum of some subset of* $\tilde{\gamma}(G)$*] and* $\tilde{\mu}(M)$ *is infimum-dense in* $\mathbb{V}$. *In particular,* $\mathbb{V} \cong \underline{\mathfrak{B}}(\mathbb{K})\ for\ \mathbb{K} = (\mathbb{V}, \mathbb{V}, \leq)$.

Concept lattices can be visualised as *line (Hasse) diagrams*. Line diagrams follow the conventions for the visualisation of hierarchical concept systems. In a line diagram, each node represents a formal concept. A concept c_1 is a subconcept of a concept c_2 if and only if there is a path of descending edges from the node representing c_2 to the node representing c_1.

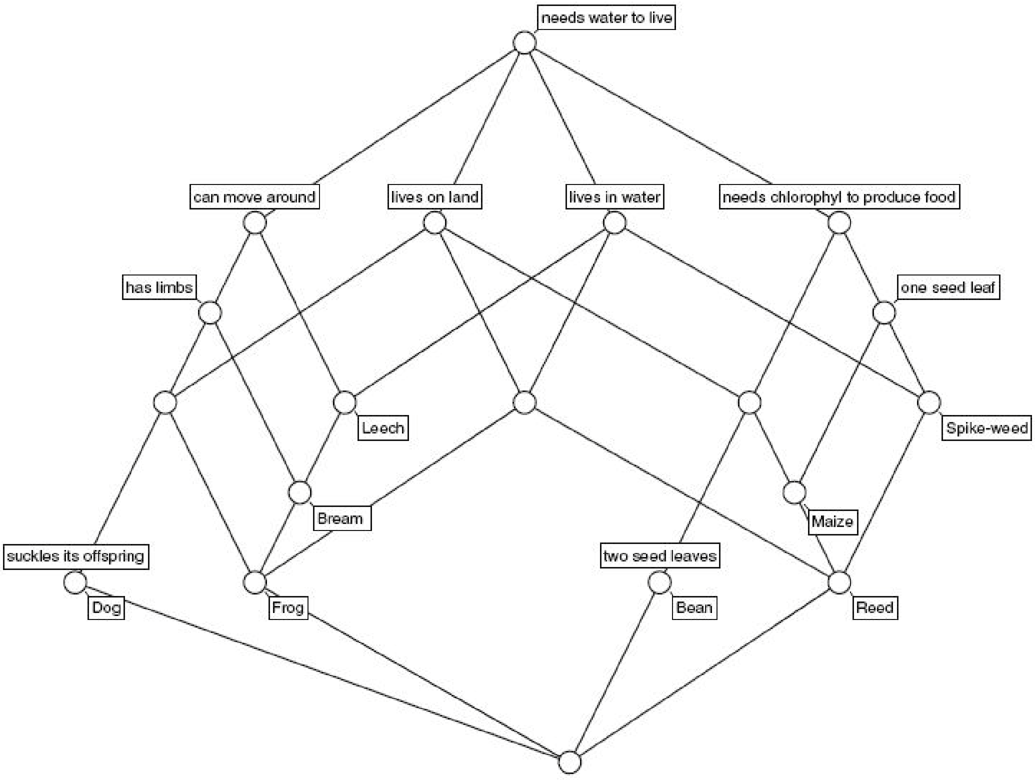


Figure 2[4] The concept lattice of the context in Table 2

Example 2.5 Figure 2 shows the concept lattice of the context in Table 2.1 as a line diagram.

The second part of Theorem 2.4 provides an efficient visualisation of concept lattices via line diagrams, as it states that a diagram is unambiguous even if each object name and each attribute name is displayed only once: in the line diagram, the name of an object $g \in G$ is attached to the node representing the object concept $\gamma(g) := (\{g\}'', \{g\}')$, and the name of an attribute $m \in M$ is attached to the node representing the attribute concept $\mu(m) := (\{m\}', \{m\}'')$. This means that the name of an object g is always attached to the node representing the smallest concept with g in its extent; dually, the name of an attribute m is always attached to the node representing the largest concept with m in its intent. We can read the context relation from the diagram because an object g has an attribute m if and only if the concept labeled by g is a subconcept of the one labeled by m. The extent of a concept consists of all objects whose labels are attached to subconcepts, and, dually, the intent consists of all attributes attached to superconcepts.

Example 2.6 For example, the concept in the very middle of Figure 3 has {Frog, Reed} as extent, and {lives on land, lives in water, needs water to live} as intent. It is a direct subconcept of the two concepts ({lives on land}, {Dog, Frog, Maize, Reed, Bean}) an d ({lives in water}, {Leech, Bream, Frog, Spikeweed, Reed}).

The top concept of the diagram always has all objects in its extent. In this case, its intent is non-empty, as it contains the attribute "needs water to live". This indicates that all living beings addressed in the movie depend on water. We see also that the diagram – and thus the set of objects – can be decomposed into two parts: the animals are grouped under the attribute "can move around", while all plants "need chlorophyll to produce food".

Dependencies between attributes can be described by implications.

Definition 2.7 *For* $X, Y \subseteq M$*, the* implication $X \rightarrow Y$ holds *in the context, if each object having all attributes in* X *also has all attributes in* Y .

Example 2.8 The implication {can move around, lives on land} → {has limbs} holds in this context. It can be read directly in the line diagram: the largest concept having both "can move around" and "lives on land" in its intent (i.e. the infimum of μ(can move around) and μ(lives on land), which is the concept that is second-most to the left) also has "has limbs" in its intent.

Note that these implications hold only for those objects that are listed in the context. If one is interested in implications that "hold globally", the context needs thus to contain sufficiently many "typical" objects.

II. ANALYSIS

3 METHODS AND PRINCIPLES OF WEB DESIGN

The purpose of the phase called *design* is to model all important aspects of the future Web. Forcing us before the start of development to address some important aspects that would later identify the project more expensive. The proposal is therefore important output, which makes use of the supplier Web site to create, the very process by which we arrive to the exit - he should help us to choose the best possible solutions from those that are available to critically look at them and tune them to the highest perfection[6,7].

- **The importance of adversarial**

The proposal should continuously through an opposition proceeding, as well as a good designer can work on some aspects neglected, and it is uncomfortably apparent later in the project. The other option is going in the design of two people - very fast reveal possible mistakes and dead ends without enter into details and this project will increase the speed and quality design.

- **Outputs**

Outputs from this phase (the specifications) should be formalized so that above them in the project to communicate all the people involved, and according to their supplier to determine the price for web site creation, and according to them to create - at least in basic outline. Details may be given in the documentation, can be discussed successively in the development site for the following reasons:

1. The sponsor site, author and documentation consultants can change the view to implementing some of the sections, pages and features at a time when they see part of the site live. Usually, people are not able to imagine the site perfectly, until they sees, and even the best design methods do not help them out. Description of detail in the documentation is then unnecessary, rather it should contain only the foundation stones on which site stands.
2. Details greatly increase the overall size and complexity of the documentation. As the sponsor and supplier site then do not pay its enough attention, and just need them to explain the details.

3. According to the selected technical solution can often tell which features a section of the web will be realized. Too detailed specification is then unnecessary extra work.
4. Even with extensive and detailed tender documents need to work with all parties to communicate in the course of the project work site, and due inquiry and testing site.

Basic documentation for the vendor Web site map (sitemap) and wire-frames (sketches pages) described in more detail those steps in *the design*, which are formed.

3.1 Web Strategy

The strategy is "creative conclusion" from the analysis. Logically, it falls to the analysis yet, but given its creativity is already encompasses to the proposal.

3.1.1 A competitive advantage

It is necessary to determine at this point what the project site should offer new or better to its users than the competition - what should be its unique competitive advantage (USP) and what should the market break. For this step is key information about the objectives, target groups and competitors from the *analysis*. To find a competitive advantage is also possible to use brainstorming or focus group. More on these methods will be shown below. After the entire site should be designed and constructed about competitive advantages.

3.1.2 Future Development Site

Part of the strategy should also look at what direction should develop the site into the future. There is any precise planning very difficult in today's fast time. It is always good to have prepared some ideas for future development based on various assumptions about the future, otherwise the project (the company in general) will not develop anywhere and allowed to drift environment.

3.1.3 How many to create Web sites

Web site set here think having the same graphic design, logo, name, domain and navigation. The basic rule for this step is that the project should be divided into several different websites, if not design an information architecture for its supposed contents, so that was for users to understand and easy to navigate. This situation arises especially when we have a very wide range, range along which at first sight unrelated or range designed for very different audiences. But usually it is the default option to create just one site because it is easier in this case to build its brand, maintain and develop it.

3.1.4 Methods of promotion

In this step, we should think about what methods of promotion website should in future use. This question is of course very complex and deals it with the entire field of internet marketing - especially during the operation site. The purpose of this step is not substitute for this course, but realize the promotional methods, which are for future consideration, and taken into account when designing the Web. It is important, especially in the marketing (SEO and PPC), which is good for Web readiness crucial.

3.2 Content and functionality of future site

3.2.1 Assembly of all content and functionality

In this step we need to assemble a list of all content and functionality for future site. Documentation of this step is very difficult, therefore it is usually not treated. It is just as important in particular, how content is distributed by sections and pages of our site. It can be based on user scenarios in this step, if well handled. We should also consider how third-party content is available if we want a similar use of content on the web. There are very appropriate interviewing potential users, focus group or internal brainstorming to get ideas for future content and functionality of the website.

3.2.2 Prioritization

Here we should prioritize the list of content and functionality from the previous step. Priorities are to serve two purposes:

1. The succession of site creation. It may be reasonable to divide the implementation of the site into several parts in some cases, which starts gradually - first to prepare the most important part of the site and run with them, then prepare the other components. This procedure is suitable for larger sites where are need to have fast re-

sults or for projects that carry greater risk of failure (in this case using the basic version of the web are trying, if you grab the project, and then if necessary, complementing other content and functionality).

2. Greater preference for critical content within a site. In this case, the priorities are part of the content into a proposal for the structure and site design content and layout of each page.

3.2.3 Taking into account the return on investment (ROI)

For each of the content we should consider whether we will really enough to invest back into it. Each part of course content does not involve direct cash income (or otherwise meet the main target site), may be paid in the event that attracts many visitors or meet other intermediary target.

3.2.4 Keyword analysis

In this section will be described only specific conclusions from the analysis for the design of the Web:

1. When we conducted an analysis of key words previously (in the old version of the website or during the analysis of demand), we first extend it to the questions relating to the content, which we did not expect in such analysis.
2. As the main questions for the appropriate search engine optimization we choose those with high relevance and commercial potential, as far as wanted and competitiveness in both the lowest.
3. Next we add the questions that are of high relevance and commercial potential and high competitiveness in the Wanted. The following questions are for search engine optimization goes well, but not in the first place.
4. The next step, we identify the list of those questions that have less commercial potential and are well sought. These questions are called *lateral*. They are interesting because they can often quite cheaply attract visitors on our web site, which can then be persuaded to buy or an equivalent action (depends on the objectives of the site). These visitors may also refer to our website and recommend it.

5. Now all the selected questions are organized into groups. Each group should present a single page, which will seek to attract visitors from search engines to questions located in the group. We should make groupings at least the most important questions.

3.3 Web structure

The aim of this step is to propose a structure (hierarchy) site, i.e. the one in which categories and how the content (which we have collected previously) group. In conclusion of this step is the content of navigation. Thus, we describe the different types of navigation, then I will deal with best practices for design and navigation methods chosen for their design, and finally ways of navigating documentation.

3.3.1 Types of Navigation

Basic navigation **division** is as following [8]:

- **Global navigation**

Section contains the main site, usually located on all pages (in a vertical or horizontal stripe). If a site optimized for search engines, should the majority of introductory pages of each section represent the most wanted and most competitive relevant keywords. Global navigation is in addition to facilitating the movement of the web further, equally important function - it should give an overview of the range of themed Web site to users.

- **Local navigation**

Navigation within each main section includes links to individual subsections. It is displayed only when the user is located in the section (either as part of a global navigation or separately, again vertically or horizontally). Along with the global navigation are sometimes collectively known as the *basic navigation.*

- **Ancillary Navigation**

Links to pages outside the main navigation hierarchy, linked from every page - for example *Recommend site Sitemap* or the *Terms.*

- **Gallery**

Gallery is a special site where main contents are only links to other sites - for example, listing of articles, products, or extract the search results.

- **Facety classification**

Many types of content are needed to sort by multiple criteria. Eg.wine can be sorted by color, national origin, variety, etc. These criteria called *facets* and the resulting navigation is called *facety classification.*

- **Breadcrumb**

This type of navigation as part of the web structure is usually not dealt with. The menu is mostly located at the top of each page and shows what pages and sections are in the hierarchy take precedence over the current page. Primarily this menu is used for users to navigate, especially when visitors coming from search engines or another site.

- **Contextual links**

They are usually presented through the context menu (related links). Those linking sites that are located elsewhere in the structure of pages in other sections, but in some way linked together thematically. It is usually possible to better adapt to web users represented by user journey scenarios use contextual links.

- **Alternative and Complementary navigational charts**

It is good to consider after the primary form of navigation design if it is not appropriate to disclose the contents of the site's features and still other types of navigation - for example, according to other characteristics of the product or target groups. Is it possible to ponder the less frequent use of navigational charts, such as tagging.

- **Other types of navigation**

Among the navigation is sometimes classified as searching within a site, site map (a page listing all the Web pages, or at least the main sections), indexes (the page with a list of keywords that occur on the site with links to individual instances) and guides (user particular process by page sequence performing, such an order in online store). **Other possible types of** navigation are as follows:

▪ Themes - the topic content is sorting criterion.

▪ Target groups - for example, Web site's section for companies and entrepreneurs.

▪ According to the tasks users - according to what they want to reach users on the Web, based on user scenarios.

- **Recommended procedures for designing the structure of the Web**

▪ Design of the Web come out of the two basic inputs:

▪ A structure of content that is needed to present the web – eg. the articles may be heading, brief description, the date, author, keywords and full text.

▪ Perception of content by users, ie, what contents are important to them, or how much matter is how different parts of the contents are understood and perceived. Eg. the article will be very important for the user title and description, and you should therefore certainly be located at the gallery. Author and date may not be located in some cases.

At designing a website is needed the two primary inputs (views on the content) to balance between appropriate.

▪ It is very important to proper names of individual items at designing the structure of the web (sections, subsections, pages, etc.), but also there are often the most difficult tasks. The description should be always interpreted to the widest group of users. It is advisable often not to be afraid to write more names to better describe the content of each item.

▪ The hierarchy is generally more recommended wider rather than deeper - that is rather less levels, but with more items in each level. User has to click more in deeper hierarchy, which is time consuming for him (assuming it does when passing through the hierarchy error and subsequently corrected).

▪ There should be a maximum of about seven items on each level of the hierarchy. Around this number varies the maximum number of options that one is able to simultaneously compare. This rule obviously does not apply if the option is "mindless", such as the choice of language users.

▪ Each hierarchy always should be formed only by one criterion, it should be more of them combine together.

- To a certain level of hierarchy should always include all of the content (eg sites, products or articles) that are available. It should not happen that some content is only available at a higher level of hierarchy.

If such site offers twenty red computers, twenty blue computers, two silver computers and one black computer, it should contain only subcategories "Red PC" and "Computer Blue", with the silver and black computer to be found only in the parent category "Computers ". The correct solution is the introduction of sub-"Silver Computer 'and' Black Computer" or in one sub-category "Other" (if the computer long enough, and the users do not seek so much).

- We are looking for context links at the structure by walking through the main hierarchy for each item we will ask the following questions: Would the user find the item elsewhere? What items related to it? What items are thematically related? For what other items a user might find this item?

- Designer may be inspired by competing webs at creating web structure. In addition to the inspiration itself more easily to our web site, including the logic on which users are already familiar from other sites.

- In addition to the logic of the structure of the site should respect the priorities of individual pieces of content. Ie. If it is not entirely logical to add an item to the highest level of basic navigation, we should put it there if it is commercially or otherwise important.

- The structure of the web should use the links appropriately internally distributed PageRank (evaluation of the quality of search engine links page). Important sites should have the most links from other web sites (it usually arranges for the fact that they are located up in the hierarchy), and variability is important words in the text links to individual pages.

3.4 Methods for designing the structure of the Web

The following methods can help us at designing the structure of the web:

3.4.1 Card sorting (sorting cards)

With this method, different parts of the content written on paper cards and let the hired testers (future users of the web with current capabilities using the Internet) to categorize into groups (which are either ready, or you yourself create testers). We communicate with the tester in sorting into groups, which serves us a good insight into the thinking of users (and thus facilitate the subsequent structure design better). Another possibility is to calculate the various statistical indicators to indicate how often the cards are together in a group.

3.4.2 Testing by tasks / user scenarios

We simulate users by themselves in this test - we are using the proposed structure to perform the tasks that will be on site to meet users. We usual find the problems and ambiguities. If we prepare user scenarios, we are coming from them.

3.4.3 Users testing navigation

This test is the same as the previous method, only the tasks are performed by outside hired testers (ordinary Internet users).

3.4.4 Controlled vocabularies

Controlled vocabularies are a tool for picking dates, relationships and hierarchies in a particular field.

3.4.5 Documentation of the proposed site structure (sitemap)

Documentation for web structure is a set of diagrams, called *Visual vocabulary for describing information architecture and interaction design* that is widely supported by manufacturers of software and web proffesionals recognition.

Creating this documentation is quite lengthy, it is usually necessary to prepare a draft structure of a pencil on paper (otherwise we are constrained applications charts they use). Therefore, he mostly used for documentation web structure specialized software to create mind maps (mind maps), namely FreeMind (but the market is in many other products). FreeMind[9] enables easy capture of web structure without the need to address given final map (it deals with the software itself).

3.5 Distribution of content to the site

This step specifies the content of Web sites - usually not all (it would not be effective), but a few typical pages. Examples are the homepage, the homepage of the main section, page listing of products, product detail page.

3.5.1 Details of the specification page

At each site we should at least specify:

- Page Title - is usually used as the HTML (HyperText Markup Language) *title element,* but may also use the page title, breadcrumb, or in internal links (We often choose different names for the page to get the highest variability of keywords).

- URL of the page.

- Key words - each page should be optimized for only a few phrases (typically on a group of phrases, keyword analysis). Keywords for this page should be prioritized and respected in the text content and text links to a page within the site and elsewhere on the Internet (where it can affect).

- Content sites - a list of content elements which will be located on the site. Each content area should be designated a priority that should be taken into account later when creating web graphics.

3.5.2 Documentation

We can use one of these options at design documentation for each site or combination thereof:

- Wire-frames - sketches site. These are sketches of sites without any graphics. There Are only visible page content, its layout and priorities of individual elements. Wire-frames can be created using chart programs (eg. MS Visio) or manually with pencil and paper (and subsequent scanning).

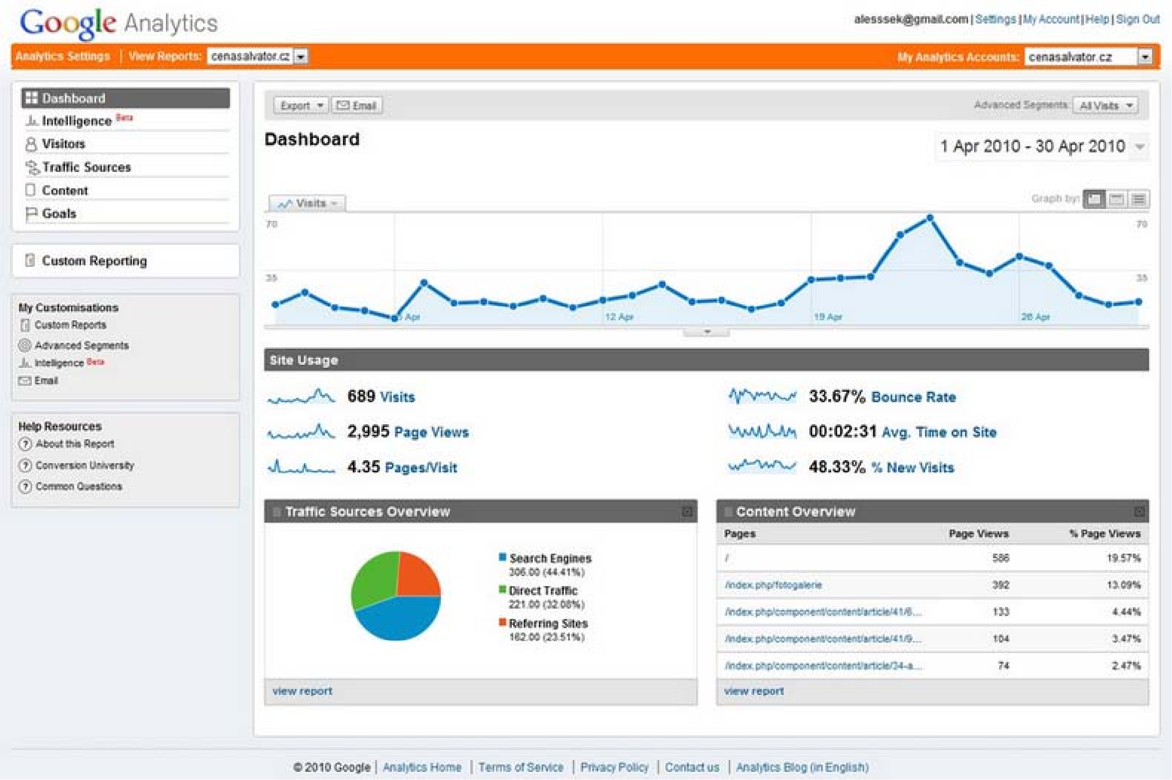
- Text labels of pages.

- HTML prototype – is a set of linked HTML pages that mimic the behavior of the future Web. Pages are no graphics and only include content elements diagrammatically.

If the site contains many different elements of content, clarity usually create wire-frames. If the site is so complicated, or if you want to keep graphic artist more freedom, create text-only site descriptions.

4 USING FCA IN WEB ANALYTICS

Web analytics is closely related to design and redesign of the site. Brings with it useful data that is processed later. Based on the processing of such data is identifying further ways of development and restructuring the Web. Web analytics recorded user behavior on our site. They give us lots of information, such as where to get a visitor came to the key words used to find our website, how much time they spend on each page of our Web site, etc. With the consistency of the data to better understand visitor behavior and focused on target groups that visit our site. Web analytics give us a large amount of data that we process for further development. This data will show the tools created specifically for web analytics. The best known and most comprehensive tool is Google Analytics. Google analytics gives us all the reports in a simple and understandable form[10].



689 Visits

2,995 Page Views

4.35 Pages/Visit

33.67% Bounce Rate

00:02:31 Avg. Time on Site

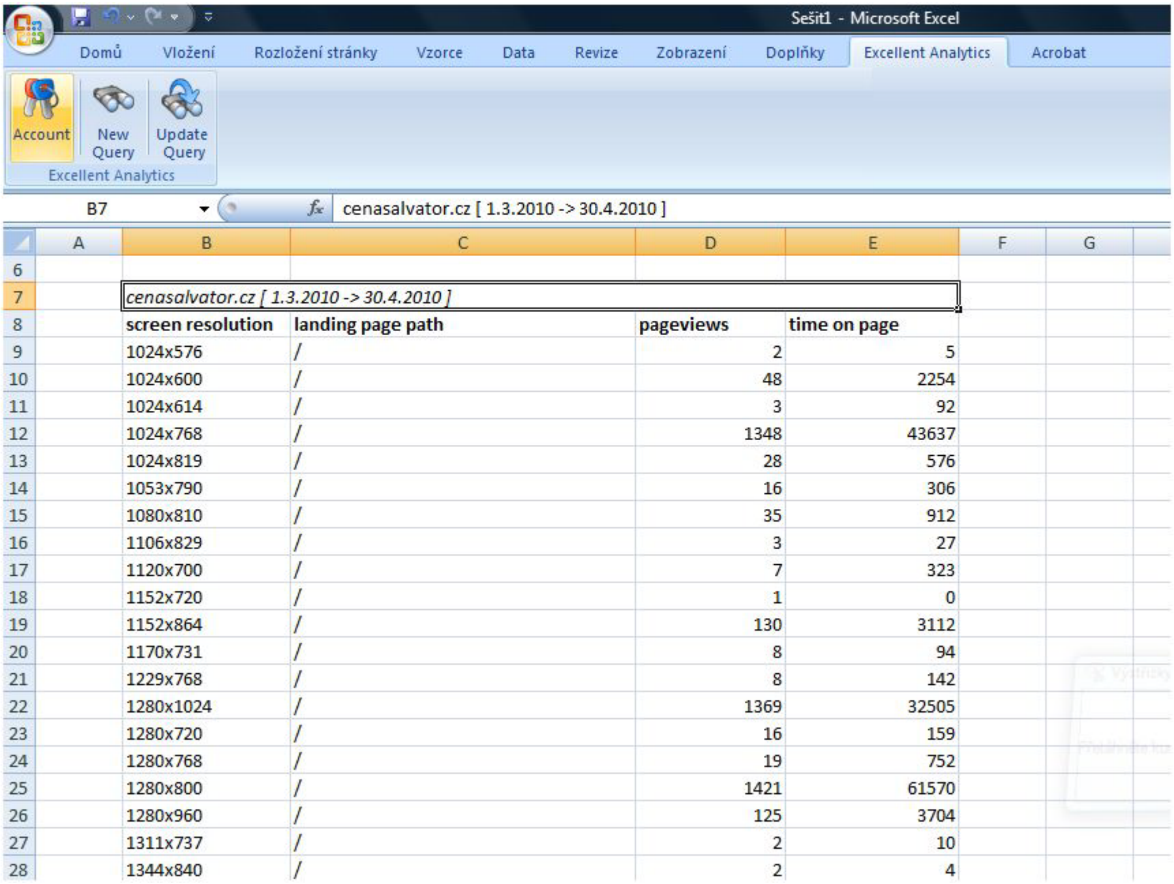
48.33% % New Visits

Pages	Page Views	% Page Views
/	586	19.57%
/index.php/fotogalerie	392	13.09%
/index.php/component/content/article/41/6...	133	4.44%
/index.php/component/content/article/41/9...	104	3.47%
/index.php/component/content/article/34-a...	74	2.47%

Figure 3 Google Analytics dashboard view

4.1 Web analytics with FCA

Google Analytics [11] gives us figures and graphs for a certain period, then we need to work with this data as a whole. We can not the data from Google Analytics to analyze for each user. Thus we could not use the FCA. Here we must use a tool Excellent Analytics, which is the extension of Microsoft Excel [12]. It can export the data necessary for us to each visitor individually.



cenasalvator.cz [1.3.2010 -> 30.4.2010]

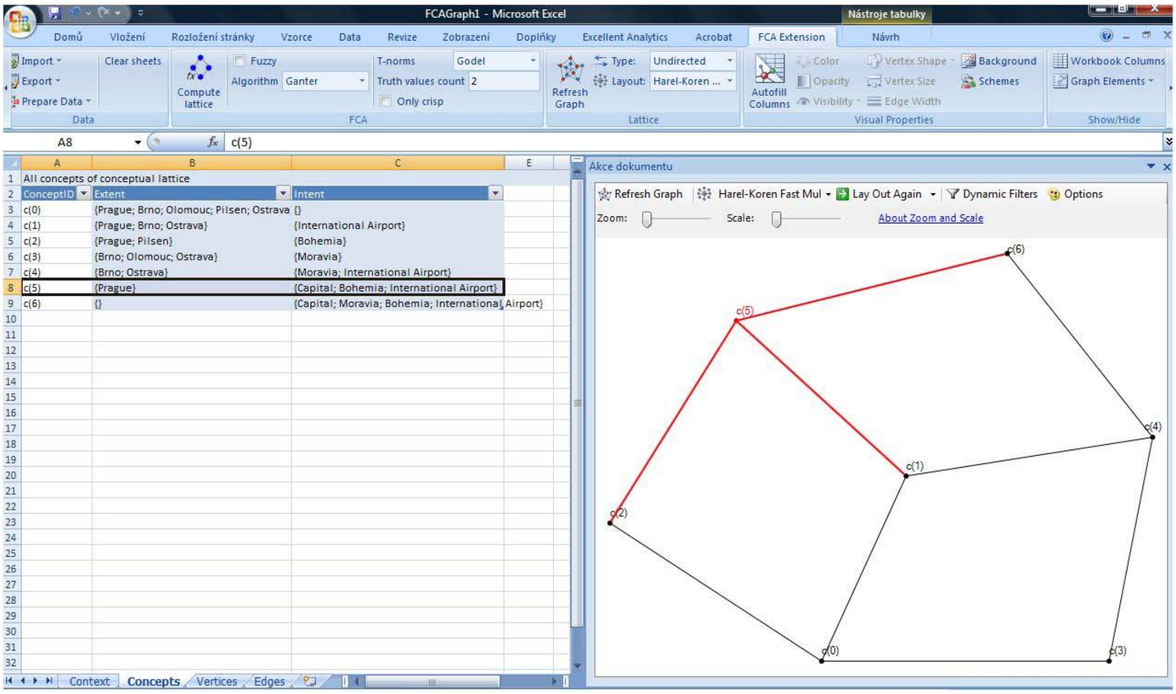
screen resolution	landing page path	pageviews	time on page
1024x576	/	2	5
1024x600	/	48	2254
1024x614	/	3	92
1024x768	/	1348	43637
1024x819	/	28	576
1053x790	/	16	306
1080x810	/	35	912
1106x829	/	3	27
1120x700	/	7	323
1152x720	/	1	0
1152x864	/	130	3112
1170x731	/	8	94
1229x768	/	8	142
1280x1024	/	1369	32505
1280x720	/	16	159
1280x768	/	19	752
1280x800	/	1421	61570
1280x960	/	125	3704
1311x737	/	2	10
1344x840	/	2	4

Figure 4 Excellent Analytics in use

Excellent Analytics tool get outputs that can acquire a large number of rows. From these data we can accurately determine the behavior of visitors to our website. To find the connection of user behavior on web pages, we must apply the FCA.

4.1.1 FCA Excel Template

An extension of Microsoft Excel called FCA Excel Temlate can compute and represent graphs from context table. Authors of this tool are Radvanský and Martin Sklenář [13]. The tool can compute concepts. Here we can see which objects have the same attributes specifically connection between them. Each concept is represent by edges.An advantage is the other color of selected edge. So we can see connections into the details.



Algorithm: Ganter

T-norms: Godel

Truth values count: 2

Type: Undirected

Layout: Harel-Koren ...

A8 c(5)

All concepts of conceptual lattice

ConceptID	Extent	Intent
c(0)	{Prague; Brno; Olomouc; Pilsen; Ostrava	{}
c(1)	{Prague; Brno; Ostrava}	{International Airport}
c(2)	{Prague; Pilsen}	{Bohemia}
c(3)	{Brno; Olomouc; Ostrava}	{Moravia}
c(4)	{Brno; Ostrava}	{Moravia; International Airport}
c(5)	{Prague}	{Capital; Bohemia; International Airport}
c(6)	{}	{Capital; Moravia; Bohemia; International Airport}

Figure 5 FCA Excel Template

Example 4.1 We need to specify categories for article types at the development of the web sites. We have types of articles and their attributes. Visitors are useful to find out the same types of information in the same category.

	Text	Link	Information	In list	Legislation	Image	Products	Ads
Article	1		1			1	1	1
Standart	1	1	1	1	1			
Law	1	1	1	1	1			
Government Regulation	1	1		1	1			
Decree	1	1		1	1			
Directive	1	1						
Judgment	1	1	1					
Research	1	1	1					
Events	1	1						1
Answers&Questions	1		1					
Reviews	1	1	1				1	1
Professional publications	1	1	1					
Statistics	1		1	1				
Vocabulary	1		1	1				

Table 3 Context of article types and attributes

{Article; Standart; Law; Government Regulation; Decree; Directive; Judgment; Research; Events; Answers&Questions; Reviews; Professional publications; Statistics; Vocabulary}	{text}
{Article; Events; Reviews}	{text; ads}
{Standart; Law; Government Regulation; Decree; Statistics; Vocabulary}	{text; list}
{Article; Standart; Law; Judgment; Research; Answers&Questions; Reviews; Professional publications; Statistics; Vocabulary}	{text; information}
{Article; Reviews}	{text; information; products; ads}
{Article}	{text; information; image; products; ads}
{Standart; Law; Statistics; Vocabulary}	{text; information; list}
{Standart; Law; Government Regulation; Decree; Directive; Judgment; Research; Events; Reviews; Professional publications}	{text; link}
{Events; Reviews}	{text; link; ads}
{Standart; Law; Government Regulation; Decree}	{text; link; list; legislation}
{Standart; Law; Judgment; Research; Reviews; Professional publications}	{text; link; information}
{Reviews}	{text; link; information; products; ads}
{Standart; Law}	{text; link; information; list; legislation}
{}	{text; link; information; list; legislation; image; products; ads}

Table 4 Concepts of the Table 3.

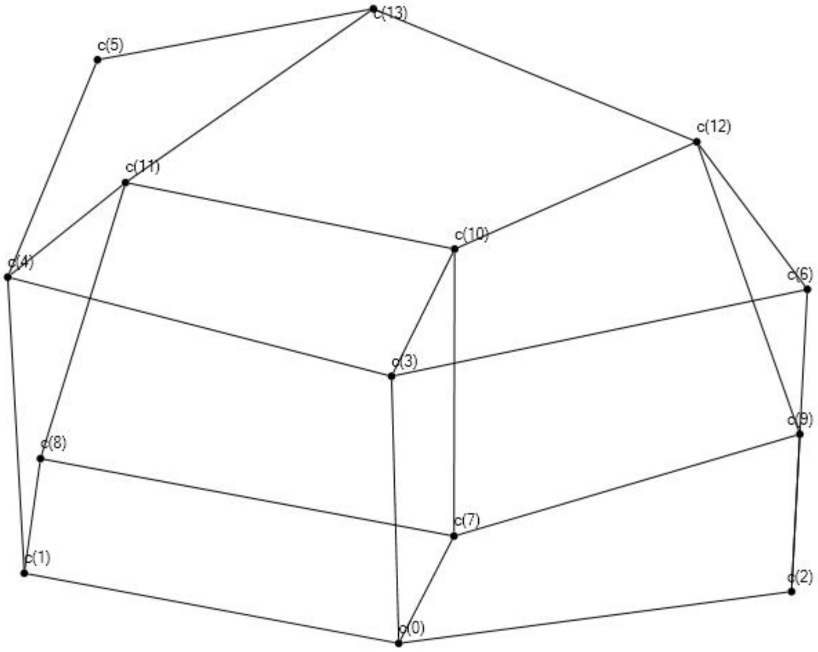


Figure 6 Graph with concepts of the Table 3.

Here we can see a part of solutions oin the graph. Types of articles goes through the edges in the same side. For example c(2) conatins {Standart; Law; Government Regulation; Decree; Statistics; Vocabulary} with the same attributes {text; list}.

Example 4.2 We need to find out if visitors with a small resolution of their display leave our websites early than visitors with a large resolution. We have got bounce rates, user resolution and time on site from Google Analytics.

	Resolution < 1024x768	Bounce rate > 40%	Time on site < 10sec
Visitor 1	1		1
Visitor 2	1		
Visitor 3	1		
Visitor 4	1		
Visitor 5			
Visitor 6			
Visitor 7		1	
Visitor 8		1	1
Visitor 9			
Visitor 10		1	
Visitor 11			
Visitor 12			
Visitor 13			
Visitor 14		1	
Visitor 15			
Visitor 16			
Visitor 17			
Visitor 18		1	
Visitor 19			
Visitor 20			
Visitor 21			
Visitor 22			
Visitor 23			
Visitor 24		1	
Visitor 25			
Visitor 26			
Visitor 27			
Visitor 28			
Visitor 29			
Visitor 30			
Visitor 31			
Visitor 32			
Visitor 33			
Visitor 34			
Visitor 35			
Visitor 36		1	
Visitor 37			
Visitor 38			
Visitor 39			1
Visitor 40		1	1
Visitor 41			
Visitor 42			
Visitor 43			
Visitor 44		1	
Visitor 45			
Visitor 46	1	1	1
Visitor 47	1		
Visitor 48	1	1	1
Visitor 49	1	1	
Visitor 50	1		
Visitor 51	1		
Visitor 52	1	1	
Visitor 53	1	1	1
Visitor 54	1		
Visitor 55	1		1
Visitor 56	1	1	1
Visitor 57	1		
Visitor 58	1		
Visitor 59	1		
Visitor 60	1	1	1

Table 5 Context of visitors

{Visitor 1; Visitor 2; Visitor 3; Visitor 4; Visitor 5; Visitor 6; Visitor 7; Visitor 8; Visitor 9; Visitor 10; Visitor 11; Visitor 12; Visitor 13; Visitor 14; Visitor 15; Visitor 16; Visitor 17; Visitor 18; Visitor 19; Visitor 20; Visitor 21; Visitor 22; Visitor 23; Visitor 24; Visitor 25; Visitor 26; Visitor 27; Visitor 28; Visitor 29; Visitor 30; Visitor 31; Visitor 32; Visitor 33; Visitor 34; Visitor 35; Visitor 36; Visitor 37; Visitor 38; Visitor 39; Visitor 40; Visitor 41; Visitor 42; Visitor 43; Visitor 44; Visitor 45; Visitor 46; Visitor 47; Visitor 48; Visitor 49; Visitor 50; Visitor 51; Visitor 52; Visitor 53; Visitor 54; Visitor 55; Visitor 56; Visitor 57; Visitor 58; Visitor 59; Visitor 60}	{}
{Visitor 1; Visitor 8; Visitor 39; Visitor 40; Visitor 46; Visitor 48; Visitor 53; Visitor 55; Visitor 56; Visitor 60}	{time on site}
{Visitor 7; Visitor 8; Visitor 10; Visitor 14; Visitor 18; Visitor 24; Visitor 36; Visitor 40; Visitor 44; Visitor 46; Visitor 48; Visitor 49; Visitor 52; Visitor 53; Visitor 56; Visitor 60}	{Bounce rate}
{Visitor 8; Visitor 40; Visitor 46; Visitor 48; Visitor 53; Visitor 56; Visitor 60}	{Bounce rate; time on site}
{Visitor 1; Visitor 2; Visitor 3; Visitor 4; Visitor 46; Visitor 47; Visitor 48; Visitor 49; Visitor 50; Visitor 51; Visitor 52; Visitor 53; Visitor 54; Visitor 55; Visitor 56; Visitor 57; Visitor 58; Visitor 59; Visitor 60}	{small resolution}
{Visitor 1; Visitor 46; Visitor 48; Visitor 53; Visitor 55; Visitor 56; Visitor 60}	{small resolution; time on site}
{Visitor 46; Visitor 48; Visitor 49; Visitor 52; Visitor 53; Visitor 56; Visitor 60}	{small resolution; Bounce rate}
{Visitor 46; Visitor 48; Visitor 53; Visitor 56; Visitor 60}	{small resolution; Bounce rate; time on site}

Table 6 Concepts of visitors

These concepts show a small number of visitors with lower screen resolution and high bounce rate. Now we can say that websites are readable on the small computers like netbooks or mobile phones.

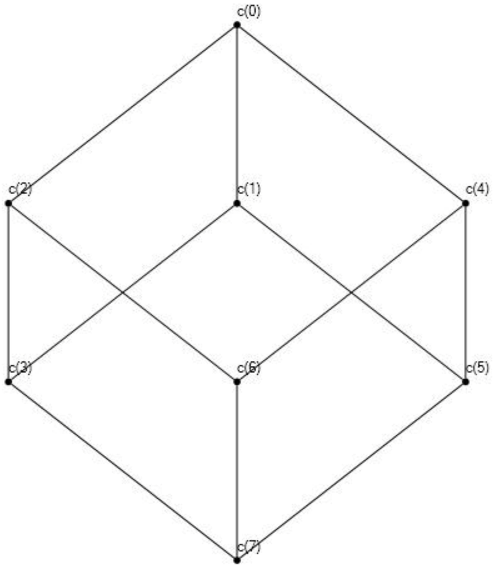


Figure 7 Graph of visitors from Example 6

Example 4.3 In development of banner system on our sites we need to know what time our visitors come to our sites. By these data we can select payment rules for a displaying banners on our sites.

	8-10	**11-12**	**13-14**	**15-16**	**17-18**	**19-20**	**21-22**	**time on site > 500sec**
Visitor 1						1		1
Visitor 2				1				1
Visitor 3	1							
Visitor 4			1					
Visitor 5			1					
Visitor 6						1		
Visitor 7		1						
Visitor 8				1				
Visitor 9			1					
Visitor 10	1							
Visitor 11	1							
Visitor 12							1	
Visitor 13						1		1
Visitor 14					1			
Visitor 15	1							
Visitor 16		1						
Visitor 17			1					1
Visitor 18							1	1
Visitor 19					1			1
Visitor 20							1	1
Visitor 21							1	
Visitor 22					1			
Visitor 23						1		
Visitor 24		1						1
Visitor 25							1	1
Visitor 26		1						1
Visitor 27						1		1
Visitor 28							1	1
Visitor 29						1		1
Visitor 30	1							1
Visitor 31							1	1
Visitor 32					1			
Visitor 33					1			
Visitor 34								
Visitor 35	1							1
Visitor 36		1						1
Visitor 37		1						1
Visitor 38			1					
Visitor 39				1				
Visitor 40					1			
Visitor 41						1		
Visitor 42							1	1
Visitor 43	1							1
Visitor 44					1			
Visitor 45	1							1
Visitor 46				1				
Visitor 47					1			
Visitor 48					1			
Visitor 49						1		
Visitor 50							1	
Visitor 51	1							
Visitor 52			1					
Visitor 53			1					
Visitor 54				1				
Visitor 55				1				
Visitor 56				1				
Visitor 57					1			
Visitor 58							1	
Visitor 59	1							
Visitor 60	1							

Table 7 Context with times of visits

As you can see in the table, we need hours of the day and spend time on site from Google Analytics. Thanks to those data we can say when is Prime Time on our sites.

{Visitor 1; Visitor 2; Visitor 3; Visitor 4; Visitor 5; Visitor 6; Visitor 7; Visitor 8; Visitor 9; Visitor 10; Visitor 11; Visitor 12; Visitor 13; Visitor 14; Visitor 15; Visitor 16; Visitor 17; Visitor 18; Visitor 19; Visitor 20; Visitor 21; Visitor 22; Visitor 23; Visitor 24; Visitor 25; Visitor 26; Visitor 27; Visitor 28; Visitor 29; Visitor 30; Visitor 31; Visitor 32; Visitor 33; Visitor 34; Visitor 35; Visitor 36; Visitor 37; Visitor 38; Visitor 39; Visitor 40; Visitor 41; Visitor 42; Visitor 43; Visitor 44; Visitor 45; Visitor 46; Visitor 47; Visitor 48; Visitor 49; Visitor 50; Visitor 51; Visitor 52; Visitor 53; Visitor 54; Visitor 55; Visitor 56; Visitor 57; Visitor 58; Visitor 59; Visitor 60}	{}
{Visitor 1; Visitor 2; Visitor 13; Visitor 17; Visitor 18; Visitor 19; Visitor 20; Visitor 24; Visitor 25; Visitor 26; Visitor 27; Visitor 28; Visitor 29; Visitor 30; Visitor 31; Visitor 35; Visitor 36; Visitor 37; Visitor 42; Visitor 43; Visitor 45}	{time on site > 500}
{Visitor 12; Visitor 18; Visitor 20; Visitor 21; Visitor 25; Visitor 28; Visitor 31; Visitor 42; Visitor 50; Visitor 58}	{21-22}
{Visitor 18; Visitor 20; Visitor 25; Visitor 28; Visitor 31; Visitor 42}	{21-22; time on site > 500}
{Visitor 1; Visitor 6; Visitor 13; Visitor 23; Visitor 27; Visitor 29; Visitor 41; Visitor 49}	{19-20}
{Visitor 1; Visitor 13; Visitor 27; Visitor 29}	{19-20; time on site > 500}
{Visitor 14; Visitor 19; Visitor 22; Visitor 32; Visitor 33; Visitor 40; Visitor 44; Visitor 47; Visitor 48; Visitor 57}	{17-18}
{Visitor 19}	{17-18; time on site > 500}
{Visitor 2; Visitor 8; Visitor 39; Visitor 46; Visitor 54; Visitor 55; Visitor 56}	{15-16}
{Visitor 2}	{15-16; time on site > 500}
{Visitor 4; Visitor 5; Visitor 9; Visitor 17; Visitor 38; Visitor 52; Visitor 53}	{13-14}
{Visitor 17}	{13-14; time on site > 500}
{Visitor 7; Visitor 16; Visitor 24; Visitor 26; Visitor 36; Visitor 37}	{11-12}
{Visitor 24; Visitor 26; Visitor 36; Visitor 37}	{11-12; time on site > 500}
{Visitor 3; Visitor 10; Visitor 11; Visitor 15; Visitor 30; Visitor 35; Visitor 43; Visitor 45; Visitor 51; Visitor 59; Visitor 60}	{8-10}
{Visitor 30; Visitor 35; Visitor 43; Visitor 45}	{8-10; time on site > 500}
{}	{8-10; 11-12; 13-14; 15-16; 17-18; 19-20; 21-22; time on site > 500}

Table 8 Concepts of context for Example 4.3

The concepts show that visitors often come to our websites in the morning and in the evening. We have almost similar numbers between 8-10 hours and 21-22 hours. It is not important to have as many visitors, but also the longer time on site.

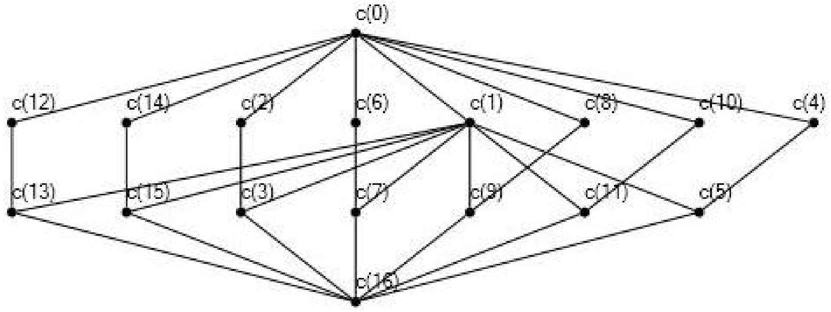


Figure 8 Conceptual lattice of visit times on the websites

We follow verticles connected to vertical c(1) in the figure. C(1) is spent longer time on site.

In the other example when we need to know which time is the best time for sending our newsletter, is needed to add attribute *day of visit*.

CONCLUSION

Master thesis deals with the using of formal concept analysis as a general method for processing websites. It is important theme in design of websites. In the theory there are introduced the basic concepts of the lattice theory, which is fundamental for a formal concept analysis. There are presented the basic concepts and assertions of a formal concept analysis, which are demonstrated by the appropriate examples. The objective of this theory is a graphical representation of concept lattices in the form of Hasse diagrams.

Analysis concerns general methods for designing websites and its main structure. Next section deals the most popular method for processing websites called web analytics. With web analytics is presented tool for processing data Google Analytics.

This thesis is to show formal concept analysis used to design websites and their processing. We can find its advantage in compared with the general method in mathematical terms. Unlike the general methods that use statistical analysis (frequency), we obtain the relationships between objects and their characteristics. We understand each user as object which attributes are metrics when visiting websites. Other objects may be types of goods, articles and more. General methods use intuitive design based on the data obtained. Formal concept analysis allows visualization of the relationships derived by conceptual relations. The result of a formal concept analysis is a clearer visualization of obtained relations and easier workflow for webdesigners. Visitors give more relevant information to their query. I have implemented these methods in the design of an information portal for personal protective equipment[14].

BIBLIOGRAPHY

1. **Schröder, B.S.W.** *Ordered Sets: an introduction.* Boston : Birkhäuser Boston, 2003. p. 191. ISBN 0-8176-41289.

2. **Harzheim, Egbert.** *Ordered Sets.* New York : Springer, 2005. ISBN 0-387-24219.8.

3. **Staab, Steffen and Rudi, Studer.** *Handbook on Ontologies (International Handbooks on Information Systems).* 2. Berlin : Springer Berlin Heidelberg, 2009. p. 811. ISBN 978-3-540-92673-3.

4. **Rutkowski, A.** The fixed point property for small sets. 1989, Vol. Order 6, pp. 1-14.

5. **Walker, J.W.** *Isotone relations and the fixed point property for posets.* 1984. pp. 275-288. Vol. Discrete Math. 48.

6. **Kaushik, A.** *Web analytics: An Hour a Day.* s.l. : Sybex, 2007. p. 410. ISBN 978-8126513284.

7. **Krug, S.** *Don't Make Me Think! A Common Sense Approach to Web Usability 2nd Edition.* s.l. : New Riders Press, 2005. p. 216. ISBN 978-0321344755.

8. Website Navigation. [Online] [Cited: 5 12, 2010.] Available from WWW:<http://www.webpagemistakes.ca/website-navigation/>.

9. FreeMind - free mind mapping software. [Online] [Cited: 5 12, 2010.] Available from WWW: <http://freemind.sourceforge.net/wiki/index.php/Main_Page>.

10. **Clifton, B.** *Advanced Web Metrics with Google Analytics.* s.l. : Wiley Publishing, 2008. p. 384. ISBN 978-0470253120.

11. Microsoft Office Excel 2007. [Online] [Cited: 5 14, 2010.] Available from WWW: <http://www.microsoft.com/cze/office/programs/excel/highlights.mspx>.

12. Google Analytics - Official website. [Online] [Cited: 5 13, 2010.] Available from WWW: <http://www.google.com/analytics/>.

13. **Radvansky, M., Sklenář, M.** FCA applications - News. [Online] [Cited: 5 17, 2010.] Available from WWW: <http://fca.radvansky.net/news.php>.

14. **Ochranky.info.** Ochranky.info - Informační portál osobních ochranných pracovních prostředků. [Online] [Cited: 6 1, 2010.] Available from WWW: <http://www.ochranky.info>.

LIST OF ABBREVIATIONS

ACC	Asceding chain condition.
Ads	Advertisement on the page.
CPO	Complete Poset.
FCA	Formal concept analysis.
HTML	HyperText Markup Language.
PC	Personal computer.
PPC	Pay per click.
ROI	Return On Investments.
SEO	Search Engine Optimization
URL	Uniform Resource Locator

LIST OF FIGURES

LIST OF TABLES