KNOWLEDGE SHARING IN ORGANIZATIONAL STRUCTURES[1]

IVO VONDRÁK, VÁCLAV SNÁŠEL AND JAN KOZUSZNÍK

The organizational structure is usually defined using the best experience and there is a minimum of formal approach involved. This paper shows the possibilities of the theory of concept analysis that can help to understand organizational structure based on solid mathematical foundations. This theory is extended by the concept of knowledge sharing and diversity that enables to evaluate the organizational structure. The alternative approach based on the hierarchical methods of cluster analysis is employed for the purposes of comparison.

Keywords: organizational structure modeling, concept lattice, knowledge diversity

AMS Subject Classification: 68N19, 68N30, 68U35

1. INTRODUCTION

Business processes represent the core of the company behavior. There are many possibilities how these processes can be defined. Usually all modeling tools are focused on various kinds of business process aspects based on what abstraction is considered as the main. From this point of view there are three basic approaches that can be employed [1] for the business process specification:

- *Functional View.* The functional view is focused on activities as well as on entities that flow into and out of these activities. This view is often expressed by Data Flow Diagrams.
- *Behavioral View.* The behavioral view is focused on when and/or under what conditions activities are performed. This aspect of the process model is often based on various kinds of State Diagrams or Interaction Diagrams. More sophisticated approaches based on the theory of Petri Nets are convenient for systems that may exhibit asynchronous and concurrent activities [8]. The behavioral view captures the control aspect of the process model. It means that the direction of the process is defined on current state of the system and event that occurs.

[1]The work has been partially supported by the Ministry of Education, Youth and Sports of the Czech Republic through the Research Grant VZ J17/98:272400013 "Modeling and Implementation of Distributed Processes".

- *Structural View.* The structural view is focused on the static aspect of the process. It captures objects that are manipulated and used by a process as well as the relationships that exist among them. These models are often based on the Entity-Relation Diagrams [2] or any of the Object Diagrams that are used by the various kinds of Object Oriented Methods.

Unfortunately, none of these views captures organization structure of roles implemented by human resources participating in processes being modeled. The next chapters will show how the theory of concepts might remove the gap between process models and organizational structure.

2. MOTIVATING EXAMPLE

Let us start with a toy example to demonstrate how the business process definition serves as a source of the organizational structure specification. Let us assume that we have a car sale company with a showroom that employs four people: *manager, salesman, technician* and *accountant.* Let us assume that we have only two business processes enacted: *car sale* and *car fleet purchase.* The first one reflects the situation when a customer wants to buy a car; the showroom performs the second one when a fleet of cars has to be purchased for demonstration and for immediate sale purposes. Car sale process starts with the activity of *offering* a car to a customer. Activity of *ordering* the chosen car from a manufacturer follows if the car is not available in the showroom. Employees of the showroom try to help the customer with *financing* afterwards and finally the payment from the customer is *checked* and the car is *handed over.* Fleet purchase process is started with the *selection* of the appropriate fleet, and then the selected cars are *ordered, paid* and *taken over* by the showroom (Figure 1 shows both processes using simple flowcharts).

It is obvious that the next logical step is to assign roles responsible for the specified activities. Based on that assignment it is possible to derive so called table of competencies that can be used for the purposes of the organization structure specification. Let us assume that in the car sale process for *offering* activity the *salesman* or *technician* is responsible. The showroom *manager* or *salesman* can realize the *ordering* activity while the *accountant* or *manager* takes care of the financial operations like help with *financing* and *checking* the payment. Finally for the activity car *hand over* the *technician* or *salesman* is responsible. The process of fleet purchase has to be assigned with its roles, too. Resulting tables (Table 1 and Table 2) of competencies are the following:

It is obvious that our showroom would have to implement some additional processes with more complex structure in a real life situation but for our purposes that are to demonstrate the potential of the theory of concepts this simplified example should be sufficient.

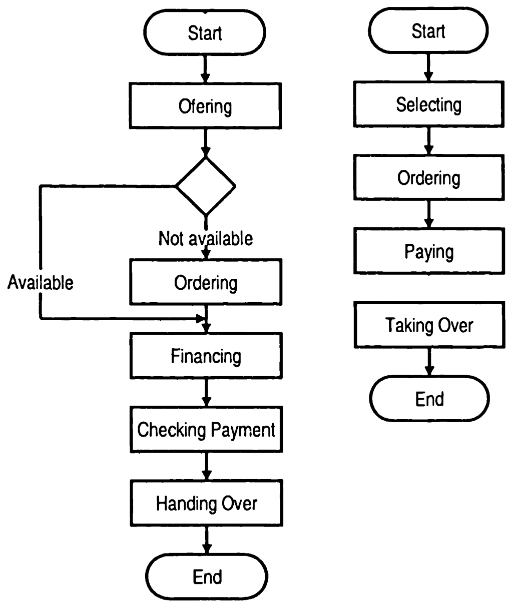

Fig. 1. Flowcharts of Car Sale (left) and Fleet Purchase (right) processes.

Table 1. Role Assignment for Car Sale process.

	Offering	Ordering	Financing	Checking P.	Handing Over
Manager		×	×	×	
Salesman	×	×			×
Technician	×				×
Accountant			×	×	

Table 2. Role Assignment for Fleet Purchase process.

	Selecting	Ordering	Paying	Taking Over
Manager	×	×		
Salesman		×		
Technician	×			×
Accountant			×	

3. CONCEPT ANALYSIS

Concept analysis theory can be used for grouping of *objects* that have common *attributes* [4]. Concept analysis begins with a binary relation, or boolean table, T between a set of objects O and set of attributes A. It means that $T \subseteq \mathsf{O} \times \mathsf{A}$. For any set of objects $O \subseteq \mathsf{O}$, their set of common attributes is defined as

$$\sigma(O) = \{a \in \mathsf{A} \mid \forall o \in O : (o,a) \in T\}. \tag{1}$$

For any set of attributes $A \subseteq \mathsf{A}$, their set of common objects is

$$\tau(A) = \{o \in \mathsf{O} \mid \forall a \in A : (o,a) \in T\}. \tag{2}$$

A pair (O, A) is called a *concept* if

$$A = \sigma(O) \wedge O = \tau(A). \tag{3}$$

The very important property is that all concepts of a given table form a *partial order* via

$$(O_1, A_1) \leq (O_2, A_2) \stackrel{\text{def}}{=} O_1 \subseteq O_2. \tag{4}$$

It was proven that such set of concepts constitutes a complete lattice called *concept lattice* $L(T)$. For two elements (O_1, A_1) and (O_2, A_2) in the concept lattice, their *meet* $(O_1, A_1) \wedge (O_2, A_2)$ is defined as

$$(O_1 \cap O_2, \sigma\tau(A_1 \cup A_2)) \tag{5}$$

and their *join* $(O_1, A_1) \vee (O_2, A_2)$ as

$$(\tau\sigma(O_1 \cup O_2), A_1 \cap A_2). \tag{6}$$

A concept $C = (O, A)$ has *extent* $e(C) = O$ and *intent* $i(C) = A$. More about concept analysis can be found in [4, 10, 14].

Concept lattice can be depicted by a standard form of the lattice diagram. It would however be too messy to label each concept by its extent and its intent. A much simpler *reduced labeling* is achieved if each object and each attribute is entered only once in the diagram. The name of object is attached to the lower half of the corresponding object concept

$$C = (\tau(\sigma(O), O) \tag{7}$$

while the name of attribute A is located at the upper half of the attribute concept

$$C = (\tau(A), \sigma(\tau(A)). \tag{8}$$

4. ORGANIZATIONAL STRUCTURE MODELLING

The tables of responsibilities specified in the previous chapter correspond with boolean tables described in concept analysis where objects of the relation are substituted by roles and attributes of objects are substituted by activities that the roles are responsible for. Before we construct the conceptual lattice describing roles and their responsibilities from our showroom example we have to join two tables of competencies defined for each process separately. The reason is that we want to have one organizational structure for the showroom as a whole not for each of the defined processes. The resulting Table 3 will have the following form:

Table 3. Role Assignment for all processes.

	Off.	Ord.	Fin.	Check	Hand.	Sel.	Pay.	Tak.
Manager		×	×	×		×		
Salesman	×	×			×			
Technician	×				×	×		×
Accountant			×	×			×	

The set of concepts that can be derived from the joined table of competencies consists of:

$$
\begin{aligned}
C_{MSTA} &= (\{\text{Man., Sal., Tech., Acc.}\}, \{\}) \\
C_{MS} &= (\{\text{Man., Sal.}\}, \{\text{Ord.}\}) \\
C_{MT} &= (\{\text{Man., Tech.}\}, \{\text{Sel.}\}) \\
C_{MA} &= (\{\text{Man., Acc.}\}, \{\text{Fin., Check.}\}) \\
C_{ST} &= (\{\text{Sal., Tech.}\}, \{\text{Off., Hand.}\}) \\
C_{M} &= (\{\text{Man.}\}, \{\text{Ord., Fin., Check., Sel.}\}) \\
C_{S} &= (\{\text{Sal.}\}, \{\text{Off., Ord., Hand.}\}) \\
C_{T} &= (\{\text{Tech.}\}, \{\text{Off., Hand., Sell., Tak.}\}) \\
C_{A} &= (\{\text{Acc.}\}, \{\text{Fin., Check., Pay.}\}) \\
C_{\emptyset} &= (\{\}, \{\text{Off., Ord., Fin., Check., Hand., Sel., Pay., Tak.}\})
\end{aligned}
$$

Concept lattice (Figure 2) can be constructed from the set of described concepts using following rules defining a structure of the graph:

- Graph nodes represent concepts and arcs their ordering.
- The top-most node is a concept with the biggest number of roles in its extent (C_{MSTA} in our case).
- Concept node is labeled with an activity if it is the largest concept with this activity in its intent.

- Concept node is labeled with role if it is the smallest concept with this role in its extent (reduced labeling).

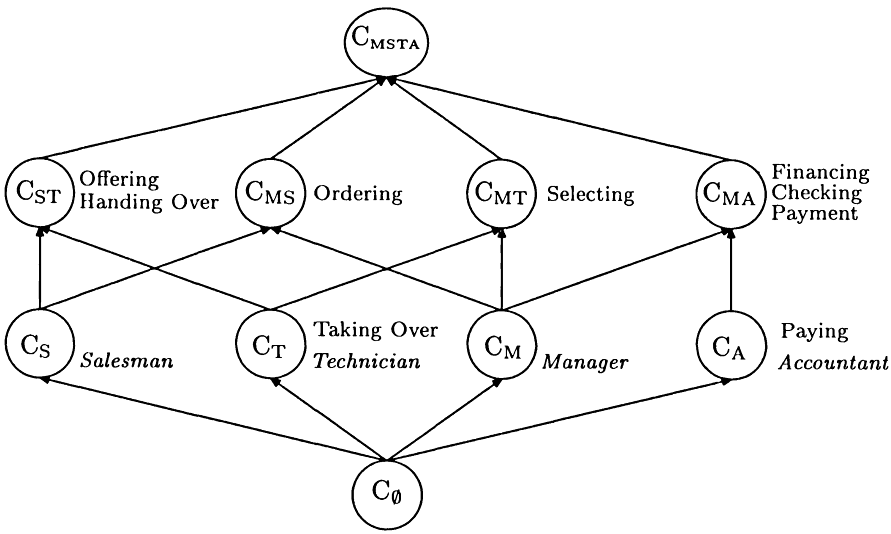

Fig. 2. Concept lattice of the organizational structure.

The resulting graph provides alternate views on the information contained in the above-described table. In other words, the concept lattice enables to visualize the structure "hidden" in the binary relation. In our example we can see that the technician is the only one who can take over delivered cars but he/she can also select a fleet of cars as well as the manager or to offer and hand over car like the salesman. Obviously, the more complex is the table of responsibilities the more difficult is to understand who is responsible for what.

5. KNOWLEDGE SHARING AND DIVERSITY

The nodes in our picture of concept lattice can be considered as a potential source of how the organizational units can be defined. For example, concept node labeled as C_{MA} unifies roles responsible for financial operations. Two of them *Financing* and *Checking Payment* are common for both roles *Manager* and *Accountant.* On the other hand *Paying* is the activity that only the *Accountant* can be responsible for. The question is how to evaluate identified concept from point of view if they should or should not be the source of organizational units? In other words, is it appropriate to put together these roles with the common set of activities or not? Let us assume that we would like to have in one organizational unit activities that have something in common. This "something in common" we would call *Knowledge Sharing*[2] and

[2]In our case Knowledge Sharing represents a coefficient that reflects common background for activities. The Knowledge Sharing is also used in a more general scope for sharing and reuse of knowledge bases and knowledge based systems (Knowledge Sharing Effort consortium) see [3, 9].

we can it formally defined as

$$K_{\text{share}}(a_i, a_j) = 1 \tag{9}$$

for activities a_i and a_j that share the knowledge,

$$K_{\text{share}}(a_i, a_j) = 0 \tag{10}$$

otherwise. It is obvious that this relation is symmetric and reflexive, i. e.

$$K_{\text{share}}(a_i, a_j) = K_{\text{share}}(a_j, a_i) \tag{11}$$

and

$$K_{\text{share}}(a_i, a_j) = 1 \text{ for } i = j. \tag{12}$$

Based on that Table 4 of the knowledge sharing among activities can be defined as follows:

Table 4. Knowledge Sharing.

	Off.	Ord.	Fin.	Check.	Hand.	Sel.	Pay.	Tak.
Off.	×	×				×		
Ord	×	×				×		
Fin.			×	×			×	
Check.			×	×			×	
Hand.					×			×
Sel.	×	×				×		
Pay.			×	×			×	
Tak.					×			×

The knowledge sharing among activities can be used to evaluate each concept from point of view how wide knowledge is required by a group of roles common to this concept (potentially organizational unit) to cover all its activities. Let us introduce the new notion of *Knowledge Diversity* that reflects the width of knowledge required by the concept (O, A) and that is formally defined as

$$K_{\text{div}}(O, A) = 1 - \frac{\sum_{(a_i,a_j)\in A\times A} K_{\text{share}}(a_i, a_j)}{|A|^2} \tag{13}$$

where $|A|$ is a cardinality of the set of attributes related to a given concept. The highest possible knowledge diversity has value 1 and the lowest one is equal to 0. The knowledge diversity computed for each of our concepts identified in the previous chapter has the following values:

$$K_{div}(\{Man., Sal.\}, \{Ord.\}) = 0$$
$$K_{div}(\{Man., Tech.\}, \{Sel.\}) = 0$$
$$K_{div}(\{Man., Acc.\}, \{Fin., Check.\}) = 0$$
$$K_{div}(\{Sal., Tech.\}, \{Off., Hand.\}) = 0.5$$
$$K_{div}(\{Man.\}, \{Ord., Fin., Check., Sel.\}) = 0.625$$
$$K_{div}(\{Sal.\}, \{Off., Ord., Hand.\}) = 0.445$$
$$K_{div}(\{Tech.\}, \{Off., Hand., Sel., Tak.\}) = 0.5$$
$$K_{div}(\{Acc.\}, \{Fin., Check., Pay.\}) = 0$$
$$K_{div}(\{\}, \{Off., Ord., Fin., Check., Hand., Sel., Pay., Tak.\}) = 0.656$$

Obviously, the highest knowledge diversity has the concept with all activities associated. In our case the whole showroom as the biggest organizational unit represents such concept. Values of knowledge diversity can add third dimension to our graph of organizational units represented by the darkness of each node. The higher is the knowledge diversity the darker is the node representing the concept (Figure 3).

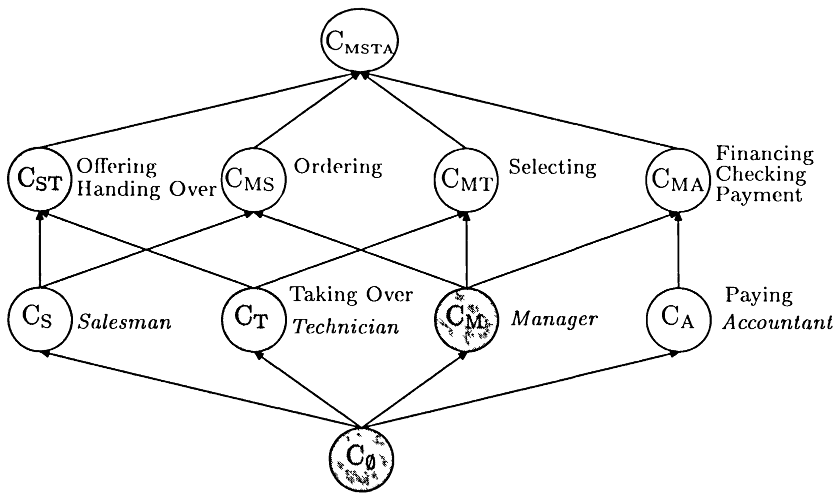

Fig. 3. Concept lattice with associated knowledge diversity.

Visualization of the organizational structures opens new possibilities to its reengineering. The concept lattice shows that the *accountant* or the *technician* are responsible for *paying* or *taking over* activities and thus cannot be substituted by anybody else. On the other hand, the *technician* in case of *offering* a car and *handing over* activity can substitute the *salesman* as well as the *manager* can substitute the salesman in the *ordering* activity. Since the *manager* provides functionality that requires wider knowledge than the *salesman* the second will be removed. On the other hand the *manager* has four activities that he/she is responsible for. If we remove his/her responsibility for *checking payment* activity then we obtain simplified organizational structure.

6. ALTERNATIVE APPROACH

Concept lattice is not the only one approach how to model and analyze organizational structures. Another possibility is to use cluster analysis for the same purposes. The method based on *hierarchical aggregation* seems to be the right one because its output shows clearly how the organizational structure should look.

Hierarchical aggregation is based on a similarity of objects (roles in our case). There are many options how the metrics of such similarity can be defined. We chose the *association coefficients* between two objects {roles} O_i, and O_j defined through so called *association table* (Table 5) to quantify this metrics.

Table 5. Association table.

		O_j	
		1	0
O_i	1	a	b
	0	c	d

In this table a represents a number of positive correspondences (attributes associated with both roles). Coefficient d represents a number of negative correspondences (attributes that are not associated with none of both roles). A number of differences where an attribute is associated with O_i and it is not associated with O_j reflects b and a number of differences where an attribute is not associated with O_i and it is associated with O_j is represented by c. The association coefficient can be computed from the following formula:

$$\mathrm{Assoc}(O_i, O_j) = \frac{2(a+d)}{2(a+d)+b+c}. \tag{14}$$

The value of Assoc belongs to interval $\langle 0,1 \rangle$: the more similar are two roles then the higher is the value of the association. For couple of roles that are identical this value is equal to 1. More practical for purpose of hierarchical aggregation is to use a coefficient of dissimilarity defined as follows:

$$\mathrm{Diss}(O_i, O_j) = 1 - \mathrm{Assoc}(O_i, O_j). \tag{15}$$

Based on that rules the association table can be built for every couple of roles. The attributes are represented by activities that these roles have in common. The resulting table is shown in Table 6.

Table 6. Association table for Car Sale process.

	Manager		Salesman		Technician		Accountant	
Manager	4	0	1	3	1	3	2	2
	0	4	2	2	3	1	1	3
Salesman	1	2	3	0	2	1	0	3
	3	2	0	5	2	3	3	2
Technician	1	3	2	2	4	0	0	4
	3	1	1	3	0	4	3	1
Accountant	2	1	0	3	0	3	3	0
	2	3	3	2	4	1	0	5

The association coefficient can be computed for all roles based on the above described formula. Table 7 and Table 8 show the resulting association coefficients and dissimilarities.

Table 7. Association coefficients of roles.

	Manager	Salesman	Technician	Accountant
Manager	1.0	0.545	0.4	0.769
Salesman	0.545	1.0	0.769	0.4
Technician	0.4	0.769	1.0	0.222
Accountant	0.769	0.4	0.222	1.0

Table 8. Table of dissimilarities.

	Manager	Salesman	Technician	Accountant
Manager	0	0.455	0.6	0.231
Salesman	0.445	0	0.231	0.6
Technician	0.6	0.231	0	0.778
Accountant	0.231	0.6	0.778	0

Next the *hierarchical agglomerative method* [5, 6] was used to build clusters of similar roles:

1. At the beginning there are only aggregates containing just one role.
2. Then two aggregates with the lowest dissimilarity are taken and they are unified into a new aggregate.

3. The step number 2 is repeated until there is only one aggregate containing the whole set of roles.

Since the later built aggregates contain more than one role it is necessary to define a *dissimilarity coefficient* D between such two aggregates e. g. A and B. For our purposes the coefficient called as *farthest neighbor* was chosen [5, 6]:

$$A \neq B \Rightarrow D(A, B) = \max_{O_i \in A, O_j \in B} \{\text{Diss}(O_i, O_j)\} \tag{16}$$

$$D(A, A) = 0. \tag{17}$$

Dendogram is a graphical representation of the resulting hierarchical aggregation (connect points based on similarity see [6]). It shows aggregation of roles based on what activities they have in common (Figure 4).

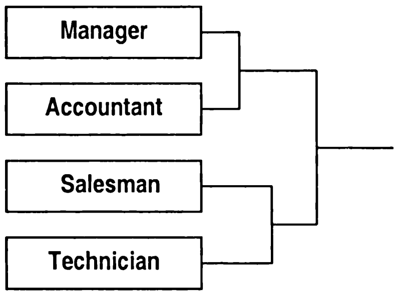

Fig. 4. Dendogram of Car Sale process.

Manager and *Accountant* are the most similar roles in our organizational structure. The second group of similar roles consists of *Salesman* and *Technician*.

The dendogram seems to be more illustrative than above mentioned concept lattice represent by Hasse diagram but it has one important disadvantage. There is no visualized information that would give reasons for such grouping. We do see on the diagram of concept lattice what activities have group of roles in common. The dendogram shows only that have they had something in common. Another kind of problems is the selection of the similarity metrics (farthest neighbor in our case). The different metrics can lead to different dendogram. It means that the resulting organizational structure is closely related to the individual selection of how the similarity is measured. This could not happen if concept theory is used for the organizational structure analysis.

7. CONCLUSION

The presented method of concept analysis provides exact and formally well defined way how the organizational structure can be analyzed and re-designed. The examples used in our paper were simplified but they demonstrated sufficiently the potential of concept lattices and the way that this theory can be adopted for purposes of organizational structure analysis. The problem is how to identify organizational structure itself. For that purpose the use of hierarchical aggregation seems to be a better tool because as well as the organizational structures they both employ hierarchy as the main abstraction. For measure knowledge sharing we going to use the dissimilarity metrics see [7, 13]. On the other hand the theory of concept shows better why the roles are grouped together and thus it serves as a better tool for understanding how the knowledge is shared among roles. We consider both approaches as complementary to each other and the future research is going to be focused on how they can be integrated together.

(Received October 31, 2003.)

REFERENCES

[1] A. Christie: Software Process Automation. Springer–Verlag, Berlin – Heidelberg – New York 1995.

[2] T. DeMarco: Structured Analysis and System Specification. Prentice-Hall, Englewood Cliffs, NJ 1979.

[3] T. R. Gruber: The Role of Standard Knowledge Representation for Sharing Knowledge-Based Technology. Technical Report No. KSL-90-53, Knowledge Systems Laboratory, Stanford University, August 1990.

[4] B. Ganter and R. Wille: Formal Concept Analysis, Mathematical Foundation. Springer–Verlag, Berlin – Heidelberg – New York 1999.

[5] A. Jain and R. Dubes: Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs, NJ 1988.

[6] A. Lukasová and J. Šarmanová: Methods of Cluster Analysis (in Czech). SNTL – Nakladatelství technické literatury, Praha 1985.

[7] K. Nehring and C. Puppe: A theory of diversity. Econometrica *70* (2002), 1155–1198.

[8] J. L. Peterson: Petri nets. ACM Computing Surveys *9* (1977), 3, 223–251.

[9] R. S. Patil, R. S. Fikes, P. F. Patel-Schneider, D. McKay, T. Finin, T. R. Gruber, and R. Neches: The DARPA knowledge sharing effort: progress report. In: Principles of Knowledge Representation and Reasoning: Proc. Third Internat. Conference (Charles Rich, B. Nebel, and W. Swartout, eds.), Cambridge, Morgan Kaufmann, San Francisco, CA 1992.

[10] G. Snelting and F. Tip: Reengineering Class Hierarchies Using Concept Analysis. Research Report No. RC 21164 (94592) 24APR97, IBM Research Division, U. S. A 1997.

[11] I. Vondrák and V. Snášel: Using concept lattices for organizational structure analysis. In: Proc. European Concurrent Engineering Conference ECEC 02, Modena, SCS, Ghent 2002, pp. 52–55.

[12] I. Vondrák and V. Snášel: Organizational Structures Modeling. In: Proc. ISM'02, Rožnov pod Radhoštěm, MARQ, Ostrava, Czech Republic 2002, pp. 11–18.

[13] M. Weitzman: On diversity. Quart. J. Econom. *107* (1992), 363–405.

[14] R. Wille: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Ordered Sets (I. Rival, ed.), Reidel, Dordrecht – Boston 1982, pp. 445–470.

Ivo Vondrák, Václav Snášel and Jan Kozušzník, Department of Computer Science, VŠB–Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava–Poruba. Czech Republic.

e-mails: ivo.vondrak, vaclav.snasel, jan.kozusznik@fei.vsb.cz