

Visualizing and Structuring Semantic Data

Teemu Mäenpää and Vesa Nyrhilä

Abstract—In this paper is proposed a method for representing semantic data and knowledge. The method is based two foundational concepts: semantic link network and adjacency model. The method allows graph presentations of semantic data and it preserves the semantic relationships between the concepts of the domain. Furthermore with the method it is possible to construct relational model of the semantically rich data.

Index Terms—Adjacency relation systems, semantic link network, data visualization, knowledge representation.

I. INTRODUCTION

The method proposed in this paper enables the preservation of semantic relationships between elements of the domain and it produces reasoning rules for the domain knowledge. The visualization can be done by using semantic link network [1] and adjacency relation system based graph [2]. Adjacency relation systems enable the restriction of search space and thus limit the search times used by queries. In addition adjacency relation systems can be also converted into relational model. In this paper the emphasis is on the construction of semantic link network and adjacency relation system based on the semantic link network.

The paper is organized as follows. In the first section the related research is discussed. In the second section the basic conversion method from semantic link network to adjacency relation system is introduced. In the third section visualization method is introduced by small example. Finally concluding remarks are given.

II. RELATED RESEARCH

A. Data models for Semistructured Data

Data models for unstructured or semistructured data were intensively studied topic from mid-90's to early 2000's. The research efforts aimed at integrating heterogeneous data as well developing an efficient structure for unstructured or semistructured data. Unstructured or semistructured data is defined as data which is not raw data but on the other hand it is not strictly typed or have precise schema[3],[4]. Typically semistructured data can be represented as trees with labeled edges. The Object Exchange Model, OEM, [5] is a graph representation of data where objects are depicted as vertices and edges are labeled with attributes. The deterministic data [6]-[11] model proposed for the semistructured data is based on the OEM model. The model is a rooted labeled graph where data of all types could be used as label. Researchers

also provided the basic syntax and semantics for constructing data [10]. Later research efforts focused on semistructured data contained in XML documents [12]. Abiteboul et al. [13] introduced a framework for dealing with incomplete information. The framework is used for constructing warehouses for XML data. The warehouse is called web house and it is based on abstracting XML documents as labeled data trees.

B. Semantic Link Network

Semantic link network (SLN) is a directed graph which consists of nodes and links connecting the nodes. A node can be for example text or a concept. Link between nodes is a labeled pointer. The label contains semantic properties which are derived from the domain [14]. Any semantic relationship between nodes is described by a property or by a combination of properties [14]. The formal definition for semantic link network can be stated as follows: $SLN = \langle NodeSet, LinkSet, RuleSet, OPSet \rangle$ [15]. *Node set* is a set of semantic nodes and *link set* is set of links between the nodes. A semantic link is denoted by $r \xrightarrow{\alpha} r'$, where r and r' represent nodes and α represents a link type.

According to Zhuge[14],[16] SLN provides semantic link primitives (Table I) which increase the networks expression and reasoning capabilities. The link primitives used in the SLN representing the domain are denoted by Ω . All of the domain's resources can be described by a property or the combinations of properties in Ω .

TABLE I: SEMANTIC LINK PRIMITIVES

Link type	Denoted as	Characteristics
Cause-effect	$r \xrightarrow{ce} r'$	Transitive link that indicates causality between two items.
Implication	$r \xrightarrow{imp} r'$	Transitive link which means that the semantics of predecessor implies to its successor.
Subtype	$r \xrightarrow{st} r'$	Transitive link that indicates that the successor is part of its predecessor.
Similar-to	$r \xrightarrow{sim} r'$	Intransitive link that describes similarity in semantics between successor and predecessor.
Instance	$r \xrightarrow{ins} r'$	Link showing that the successor is an instance of the predecessor.
Sequential	$r \xrightarrow{seq} r'$	Transitive link which indicates that the content of item r is a successor of the content of item r' . Also links can be connected in a sequential chain.
Reference	$r \xrightarrow{ref} r'$	Transitive link which means that the item r' is an explanation of the item r .
Equal-to	$r \xrightarrow{e} r'$	Link showing that two items are identical in meaning.
Empty	$r \xrightarrow{\emptyset} r'$	Link showing that two items are irrelevant to each other.
Null or unknown	$r \xrightarrow{N} r'$	Link that indicates unknown or uncertain relation between two items.
Non- α relation	$r \xrightarrow{\alpha^N} r'$	Link that shows that there is no semantic relationship between two items.
Reverse relation operation	$r \xrightarrow{\alpha^R} r'$	If there exists semantic relation from r to r' , then there exists also relation from r' to r .

Manuscript received January 31, 2013; revised March 8, 2013.

The authors are with the Department of Computer Science, Faculty of Technology, University of Vaasa, Vaasa, FI-65100 Finland (e-mail: teemu.maenpaa@uva.fi, vesa.nyrhila@uva.fi).

Semantic links can be transformed and composed with operations such as *reversal*, *addition* and *multiplication*. With these operations two semantic links can be converted to one semantic reasoning rule [14]. The *reversal* operation reversesthe semantic relation and it is the most essential operation inthe construction ofthe reasoning rules. The *addition* operation merges two semantic relations into one.

Consider semantic relationships α from r to r' and β from r' to r'' . The semantic relationship γ from r to r'' can be formed by reasoning process (see Fig. 1). The process is called *multiplication* operation denoted as $\alpha \times \beta = \gamma$. The rules presented in section IV are constructed with the reversal and multiplication operations. [17]

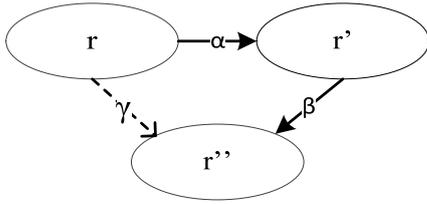


Fig. 1. Multiplication operation.

SLN enables and supports semantic reasoning. The semantic reasoning rules chain related semantic links in order to obtain a reasoned result [14]. The *ruleset* presents the set of rules on the *linkset*. The *ruleset* is always domain specific.

The *OPset* is a set of basic operations on SLN. The *OPset* defines operations for addition and deletion of nodes, links and rules [18].

Node types, semantic link definitions, and reasoning rules are described in the SLN schema. The SLN schema is a blueprint for constructing the SLN instances. The global SLN schema portrays the basic semantics of the domain. The SLN instances can be defined according to the global schema, or by defining a subschema according to the global schema. [19]

According to [17] the use of SLN can be justified as follows:

- It supports intelligent applications by assigning semantic indicators and rules to links and enabling relational, analogical, inductive, and complex reasoning.
- It explores the laws of semantic linking. It pursues diversity and user experience of linking and exploring rather than the correctness.
- It provides a light-weight semantic networking approach for peer-to-peer knowledge sharing.

C. Adjacency Model and Adjacency Relation System

Adjacency model (AM) is a model for data representation. It is based on the concepts of adjacency relation systems (ARS), ARS with adjacency defining sets (ARST) and the unique ARST and the valid ARST [20]. The early application utilizing the concept of adjacencies was introduced by Ni and Bloor [21]. Their work dealt with boundary structures built on relations between faces, edges and vertices. The concept of adjacency was then further developed by Wanne [2].

ARS is a structure by which graphs can be used to illustrate the adjacency of elements belonging to sets, which represent different entity types [22]. ARS is a pair (A, R) , where $A = \{A_1, A_2, \dots, A_n\}$, $n \geq 1$, is a set containing pairwise disjoint finite nonempty sets and $R = \{R_{ij} | i, j \in \{1, 2, \dots, m\}\}$

is a set of relations, where each R_{ij} is a relation on $A_i \times A_j$. If $(x, y_1), (x, y_2), \dots, (x, y_m) \in R_{ij}$ are all the pairs of relation R_{ij} having x as the first component, then each element $y_k (k = 1, 2, \dots, m)$ is adjacent to the element x , denoted by $Adj(x)$. ARS is said to be symmetric if for each pair $x \in A_i, y \in A_j$ holds that $x \in Adj_i(y)$ and also $y \in Adj_j(x)$ and for each $i, 1 \leq i \leq n$. [2]

The adjacency between elements of entity types can be expressed by relations. If the adjacency of the elements depends on the definition of the relation it is said to be weak. The adjacency of elements is considered strong if it is defined with respect to a set of the entity types. The definition of adjacency defining sets is given next. [22]-[24]

In the adjacency relation system each element set $A_i, i = 1, 2, \dots, n$, represents a certain entity type denoted as T_i . In addition, associate with each index pair $i, j \in \{1, 2, \dots, n\}$ a set of indices $K \subseteq \{1, 2, \dots, n\} - \{i, j\}$ and also a set of the entity types $\tilde{T}_{ij} = \{T_k | k \in K\}$. The set \tilde{T}_{ij} gives the entity types which determine the adjacency between the elements of A_i and A_j . [2]

The adjacency defining set \tilde{T}_{ij} is defined as follows: Elements $x \in A_i, y \in A_j$ where $i, j \in \{1, 2, \dots, n\}$ and $x \neq y$, are considered to be adjacent with respect to a set of entity types $\tilde{T}_{ij} = \{T_k | k \in K\} = \emptyset$ if for each $k \in K$ there is an element $z \in A_k$ such that $x \in Adj_i(z)$ and $y \in Adj_j(z)$. [2],[22]

Example1. Let (A, R) be an asymmetric adjacency relation system where $A = \{A_1, A_2, A_3\}, A_1 = \{x_1, x_2\}, A_2 = \{y_1, y_2, y_3\}, A_3 = \{z_1, z_2\}$, and R consist of relations $R_{12} = R_{13} = R_{23} = \emptyset$
 $R_{11} = \{(x_1, \{x_2\})\}$
 $R_{21} = \{(y_1, \{x_1\}), (y_3, \{x_2\})\}$
 $R_{22} = \{(y_3, y_2)\}$
 $R_{31} = \{(z_1, \{x_2\})\}$
 $R_{32} = \{(z_2, \{y_3\})\}$
 $R_{33} = \{(z_2, \{z_1\})\}$

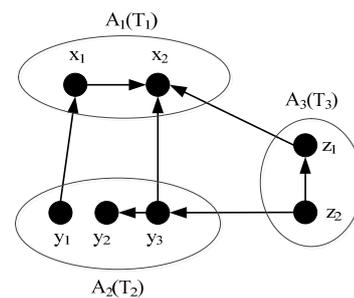


Fig. 2. Asymmetric ARS of example 1.

In Fig. 2 the adjacency defining set is $\tilde{T}_{23} = \{T_1\}$, y_3 and z_1 are adjacent with respect to \tilde{T}_{23} since $y_3 \in Adj_2(x_2)$ and $z_1 \in Adj_3(x_2)$. In Fig. 1 the adjacency between y_1 and x_1 is considered to be weak and the adjacency between y_2 and x_2 is strong if $\tilde{T}_{23} = \{T_1\}$. The concepts of weak and strong adjacency are combined together in the concept of unique adjacency [22].

The ARS with adjacency defining sets (ARST) can be denoted by (A, R, τ) , where τ means the set of adjacency defining sets. ARST is unique if for each pair $i, j = \{1, 2, \dots, n\}$ of integers the adjacency defining set \tilde{T}_{ij} is

nonempty for all elements $x \in A_i, y \in A_j$ and x and y are adjacent if and only if they are adjacent with respect to \tilde{T}_{ij} . [2]

Example 2. Consider the ARST shown in Fig.2. Let us add a new relation (dashed arrow in Fig. 3) $R_{23} = \{(y_3, z_1)\}$. Now we have the set of relations:

- $R_{12} = R_{13} = \emptyset$
- $R_{11} = \{(x_1, \{x_2\})\}$
- $R_{21} = \{(y_1, \{x_1\}), (y_3, \{x_2\})\}$
- $R_{22} = \{(y_3, y_2)\}$
- $R_{23} = \{(y_3, z_1)\}$
- $R_{31} = \{(z_1, \{x_2\})\}$
- $R_{32} = \{(z_2, \{y_3\})\}$
- $R_{33} = \{(z_2, \{z_1\})\}$

and the adjacency defining set

- $\tilde{T}_{23} = \{T_1\}$,
- $\tilde{T}_{11} = \tilde{T}_{12} = \tilde{T}_{13} = \tilde{T}_{21} = \tilde{T}_{22} = \tilde{T}_{31} = \tilde{T}_{32} = \tilde{T}_{33} = \emptyset$.

In Fig. 3 y_3 and z_1 are adjacent with respect to relation R_{23} and they are also adjacent according to $\tilde{T}_{23} = \{T_1\} \neq \emptyset$. If there are no other nonempty adjacency defining sets the ARST is said to be unique.

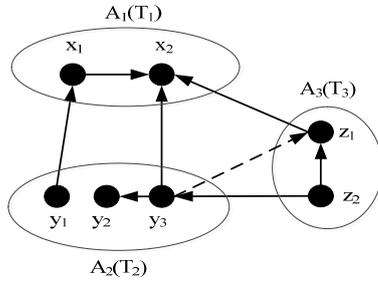


Fig. 3. Unique ARS of example 2.

The notation $T_i \rightarrow T_j$ for a relation type indicates that the relations R_{ij} are defined in $A_i \times A_j$. The set of relation types $\{T_i \rightarrow T_j | (i, j) \in S\}$, where $S \subseteq \{1, 2, \dots, n\}$, is called a relation combination. The relation combination for an ARST (A, R, τ) the restriction of R on a given relation combination is determined by $S \subseteq \{1, 2, \dots, n\} \times S \subseteq \{1, 2, \dots, n\}$ is denoted by $R|S$, i.e. $R|S = \{R_{ij} \in R | (i, j) \in S\}$. [2]

The relation combination is defined as follows. Given the entity types T_1, \dots, T_n , a set $S \subseteq \{1, 2, \dots, n\} \times S \subseteq \{1, 2, \dots, n\}$, and adjacency defining sets τ . A relation combination $T_i \rightarrow T_j | (i, j) \in S$ is said to be valid if for any unique ARST (A, R, τ) there is no unique ARST (A, R', τ) such that $R|S = R'|S$. Otherwise the relation combination is not valid. [2]

The valid and minimal relation combination makes it possible to derive all the relations and sets of a given data structure from the smallest set of stored relations. [2], [22], [25]

D. Accident Models

Accident models are in the key role when the designing of the information model is done. The accident models depict how the accident occurs. The accident models represent, and define all the implicit information about the accident. Typically accidents are represented with causal-sequence models [26].

Domino theory [27] is basic causal-sequence model that describes the accident as a chain of linked events occurred at the work place. In the ILCI (Internal Loss Control

Institute) model [28] is depicted same kind of events as in domino theory but the event descriptions are extended and categorized. Domino theory and ILCI model are criticized because they do not cover personal, organizational or managerial causality factors. The TRIPOD model [29] represents causal sequences similar to ILCI model but in addition it includes managerial involvement.

III. CONVERTING SLN INTO ARS

The procedure for converting SLN to ARS is quite straightforward. In ARS the nodes and links of the SLN are represented by sets and their relationships are represented as adjacencies (relations). Each node and link is members of the sets of the entity types.

Example 3. Consider a simple SLN with two nodes a and b connected by semantic link α , denoted as $a \xrightarrow{\alpha} b$. The SLN can be represented with a directed graph (Fig. 4).

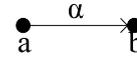


Fig. 4. A simple semantic link network.

The ARS representation of SLN (Fig. 5) is constructed as follows. The ARS is a pair (A, R) , where $A = \{A_1, A_2, A_3\}$ and $A_1 = \{a\}$, $A_2 = \{b\}$ and $A_3 = \{\alpha\}$. R consists of relations $R_{13} = \{(a, \alpha)\}$, $R_{32} = \{(\alpha, b)\}$.

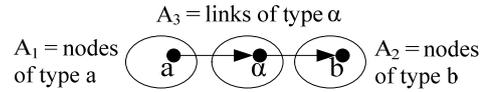


Fig. 5. SLN based ARS.

IV. DATA VISUALIZATION – CASE ESAW

In this section a simplified example of data visualization method is given. The concepts discussed in this section are based on European Statistics on Accidents at Work methodology (ESAW) [30]. ESAW -methodology provides a framework for statistical accident information reporting. The SLN presented in this section is a graph covering the factors associated with a hazardous situation at work. The concepts of the graph are given in Table II. The semantic links and their explanations are shown in Table III.

TABLE II: EXPLANATIONS OF ESAW VARIABLES

ESAW variable	Explanation
Working process (wp)	The description of the main type of work or task before the accident.
Specific physical activity (spa)	Describes what victim was doing at the time of accident.
Deviation (dev)	Description of an event deviating from the normal working process that triggers the accident.
Contact (con)	Describes the contact that injured the victim.
Material agents (ma _n)	Describes tools or objects associated with specific physical activity, deviation and contact.
Injury type (inj)	Describes the physical consequences (e.g. wound, bone fracture) to the victim.

TABLE III: EXPLANATIONS OF SEMANTIC LINKS IN FIG. 5

SLN links	Explanation
Reference (ref ₁)	Specific physical activity is a part of working process performed by the victim before the accident.
Reference (ref ₂)	Material agent(ma ₁) which is involved in Specific physical activity.
Sequential (seq ₁)	Expresses the chronological order of events.
Reference (ref ₃)	Material agent(ma ₂) which is involved in the deviating event.
Sequential (seq ₂)	Expresses the chronological order of events.
Reference (ref ₄)	Material agent(ma ₃) which is involved with the contact.
Cause-effect (ce ₁)	Contact causes the type of injury.

The visualization method consists of four steps:

Step 1. Construct SLN depicting the domain concepts (Table II).

Step 2. Define link types between concepts – link type definitions are based on Zhuge's [16] definitions (see Table I). In Table III the explanations of semantic links are given.

TABLE IV: ADDITIONAL RULES OF ESAW-GRAPH

No	Rules	Rule summary
1	$wp \xrightarrow{ref_1} spa, spa \xrightarrow{ref_2} ma_1 \Rightarrow wp \xrightarrow{ref_5} ma_1$	$ref_1 \cdot ref_2 \Rightarrow ref_5$
2	$wp \xrightarrow{ref_1} spa, spa \xrightarrow{seq_1} dev \Rightarrow wp \xrightarrow{ref_9} dev$	$ref_1 \cdot seq_1 \Rightarrow ref_9$
3	$spa \xrightarrow{seq_1} dev, dev \xrightarrow{ref_3} ma_2 \Rightarrow spa \xrightarrow{ref_6} ma_2$	$seq_1 \cdot ref_3 \Rightarrow ref_6$
4	$spa \xrightarrow{seq_1} dev, dev \xrightarrow{seq_2} con \Rightarrow spa \xrightarrow{seq_3} con$	$seq_1 \cdot seq_2 \Rightarrow seq_3$
5	$dev \xrightarrow{seq_2} con, con \xrightarrow{ref_4} ma_3 \Rightarrow dev \xrightarrow{ref_7} ma_3$	$seq_2 \cdot ref_4 \Rightarrow ref_7$
6	$dev \xrightarrow{seq_2} con, con \xrightarrow{ce_1} inj \Rightarrow dev \xrightarrow{ce_2} inj$	$seq_2 \cdot ce_1 \Rightarrow ce_2$
7	$ma_1 \xrightarrow{ref_2^R} spa, spa \xrightarrow{seq_1} dev \Rightarrow ma_1 \xrightarrow{ref_{10}} dev$	$ref_2^R \cdot seq_1 \Rightarrow ref_{10}$
8	$ma_2 \xrightarrow{ref_3^R} dev, dev \xrightarrow{seq_2} con \Rightarrow ma_2 \xrightarrow{ref_{11}} con$	$ref_3^R \cdot seq_2 \Rightarrow ref_{11}$
9	$ma_3 \xrightarrow{ref_4^R} con, con \xrightarrow{ce_1} inj \Rightarrow ma_3 \xrightarrow{ref_8} inj$	$ref_4^R \cdot ce_1 \Rightarrow ref_8$

Step 3. Define additional rules (dashed lines) for the SLN.

By following the steps 1 to 3 the SLN shown in Fig. 6 is constructed.

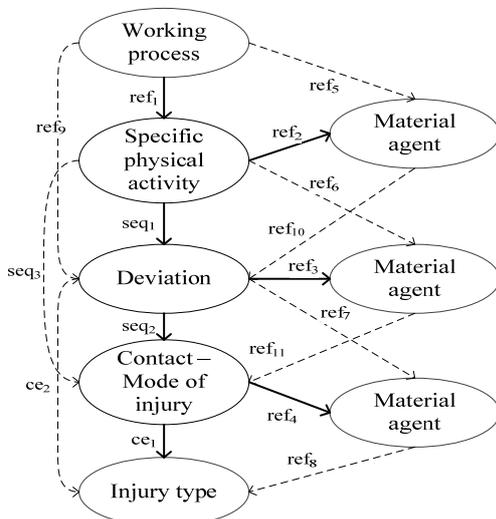


Fig. 6. ESAW-SLN, dashed arrows represent the additional rules.

Based on the semantic links derived from the domain additional rules are defined (Fig. 6). The rules 1-9 (Table IV) have been defined by using *reversal* and *multiplication* operations (see [1] and Fig. 1).

Step 4. Transform the SLN into ARS (Fig. 7).

- 4.1. Define the type sets and attach elements to the type sets.
- 4.2. Define relationships between the elements.
- 4.3. Define additional relations – the additional relations are based on the rules defined in step 3.
- 4.4. Define the adjacency defining sets.
- 4.5. If possible define the relation combinations for the ARST.

The rules defined in step 3 are added to ARS as in Fig. 7. The rule relations are marked with dashed arrows.

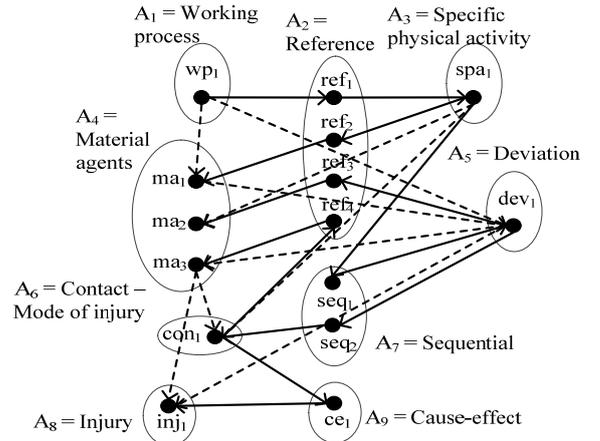


Fig. 7. ESAW-ARS with the rule relations.

Now ESAW-ARS is a pair (A, R) , where $A = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9\}$ and R consists of relations

$$\begin{aligned}
 R_{12} &= \{(wp_1, \{ref_1\})\} \\
 R_{23} &= \{(ref_1, \{spa_1\})\} \\
 R_{24} &= \{(ref_2, \{ma_1\}), (ref_3, \{ma_2\}), (ref_4, \{ma_3\})\} \\
 R_{37} &= \{(spa_1, \{seq_1\})\} \\
 R_{52} &= \{(dev_1, \{ref_3\})\} \\
 R_{57} &= \{(dev_1, \{seq_2\})\} \\
 R_{62} &= \{(con_1, \{ref_4\})\} \\
 R_{69} &= \{(con_1, \{ce_1\})\} \\
 R_{76} &= \{(seq_2, \{con_1\})\} \\
 R_{98} &= \{(ce_1, \{inj_1\})\}
 \end{aligned}$$

Added relations are defined as follows:

$$\begin{aligned}
 R_{14} &= \{(wp_1, \{ma_1\})\} \\
 R_{15} &= \{(wp_1, \{dev_1\})\} \\
 R_{34} &= \{(spa_1, \{ma_2\})\} \\
 R_{36} &= \{(spa_1, \{con_1\})\} \\
 R_{45} &= \{(ma_1, \{dev_1\})\} \\
 R_{46} &= \{(ma_2, \{con_1\})\} \\
 R_{48} &= \{(ma_3, \{inj_1\})\} \\
 R_{54} &= \{(dev_1, \{ma_3\})\} \\
 R_{58} &= \{(dev_1, \{inj_1\})\}
 \end{aligned}$$

The adjacency defining sets for the ARS are

$$\begin{aligned}
 \tilde{T}_{12} &= \tilde{T}_{23} = \tilde{T}_{25} = \{T_4\} \\
 \tilde{T}_{14} &= \{T_5\} \\
 \tilde{T}_{34} &= \tilde{T}_{37} = \tilde{T}_{47} = \{T_6\} \\
 \tilde{T}_{45} &= \tilde{T}_{49} = \tilde{T}_{59} = \{T_8\}
 \end{aligned}$$

The ARST constructed in step 4 is not considered to be unique. For example besides the adjacency defining set $\tilde{T}_{12} = \{T_4\}$ there should exist a relation between the elements wp_1 and ref_2 . Thus by definition the ARST cannot have valid relation combination.

V. CONCLUSION

The method proposed in this paper enables the visualization of the data and the knowledge. It combines the semantic capabilities of the semantic link network with the efficient structures of adjacency model. With semantic link network it is possible to construct a network representing key concepts and their relationships within the domain. Furthermore semantic link network can represent semantic rules for the domain and thus support the logical reasoning process.

The adjacency model provides graph based visualization of the domain concepts. With the adjacency model the concepts and semantic links connecting them can be represented as adjacency relations. The adjacency model gives a foundation for search space restrictions, query optimizations and eventually the construction of the relational database which preserves the semantics of the domain.

The method proposed in this paper is the first step towards an efficient semantic database. The future research efforts include the following areas: domain specific semantic link networks schema definitions and the conversion process from the adjacency model to the relational model. The semantic link network presented in this paper is based on the global semantic link network schema. In the future studies the domain specific schema for accident reporting variables will be defined.

There exists a preliminary version of the method for converting the adjacency model into the relational model. The conversion requires further research work on the definitions of the adjacency defining sets and the relation combinations.

REFERENCES

- [1] H. Zhuge, "Active e-document framework ADF: model and tool," in *Information & Management*, vol. 41, no. 1, pp. 87-97, October 2003.
- [2] M. Wanne, "Adjacency relation systems," *Acta Wasaensia*, no. 60, Vaasa: University of Vaasa, Finland, 1998.
- [3] S. Abiteboul, "Querying semi-structured data," in *Proceedings of 6th International Conference Database Theory*, 1997, pp. 1-8.
- [4] D. Suciu, "An overview of semistructured data," in *ACM SIGACT News*, vol. 29 no. 4, pp. 28-38, December 1998.
- [5] Y. Papakonstantinou, H. Garcia-Molina, and J. Widom, "Object exchange across heterogeneous information sources," in *Proceedings of the 11th International Conference on Data Engineering*, 1995, pp. 251-260.
- [6] P. Buneman, S. Davidson, and D. Suciu, "Programming constructs for unstructured data," in *Proceedings of the 1995 International Workshop on Database Programming Languages*, 1995, pp. 1-12.
- [7] P. Buneman, "Semistructured data," in *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, 1997, pp. 117-121.
- [8] P. Buneman, S. Davidson G. Hillebr, and D. Suciu, "A query language and optimization techniques for unstructured data," in *Proceedings of the 1996 ACM SIGMOD international conference on Management of data*, 1996, pp. 505-516.
- [9] P. Buneman, S. Davidson, M. Fernandez, and Dan Suciu, "Adding structure to unstructured data," in *Proceedings of 6th International Conference Database Theory*, 1997, pp. 336-350.
- [10] P. Buneman, A. Deutsch, and W-C. Tan, "A deterministic model for semistructured data," in *Proc. Workshop on Query Processing for Semistructured Data and Non-Standard Data Formats*, 1998.
- [11] P. Buneman, W. Fan, and S. Weinstein, "Query optimization for semistructured data using path constraints in a deterministic data model," in *Proceedings of International Workshop on Database Programming Languages*, 1999, pp. 208-223.
- [12] S. Abiteboul, "Semistructured data: from practice to theory," in *Proc. 16th Annual IEEE Symposium on Logic in Computer Science*, pp. 379-386, June 2001.
- [13] S. Abiteboul, L. Segoufin, and V. Vianu, "Representing and querying XML with incomplete information," *ACM Transactions on Database Systems*, vol. 31, no. 1, pp. 208-254, March 2006.
- [14] H. Zhuge, *The Knowledge Grid*, 1st ed. Singapore, Singapore: World Scientific, 2004.
- [15] J. Chen, "A survey on the semantic linked network," in *Proc. 2012 Eighth International Conference on Semantics, Knowledge and Grids (SKG)*, Beijing, China, 2012, pp. 205-208.
- [16] H. Zhuge, "Semantic linking through spaces for cyber-physical-socio intelligence: A methodology," *Artificial Intelligence*, vol. 175, no. 5-6, pp. 988-1019, April 2011.
- [17] H. Zhuge, Y. Sun, R. Jia, and J. Liu, "Algebra model and experiment for semantic link network," *International Journal of High Performance Computing and Networking*, vol. 3, no. 4, pp. 227-238, December 2005.
- [18] Q. Zeng, X. Jiang, and H. Zhuge, "Adding logical operators to treepattern queries on graph-structured data," *PVLDB*, vol 5, no. 8, pp. 728-739, 2012.
- [19] H. Zhuge and Y. Sun, "The schema theory for semantic link network," *Future Generation Computer Systems*, vol. 26, no. 3, pp. 408-420, March 2010.
- [20] J. Töyli, Modeling semistructured data by the adjacency model, Vaasa: University of Vaasa, Finland, 2002.
- [21] X. Ni and S. Bloor, "Performance evaluation of boundary data structures," *IEEE Computer Graphics and Applications*, vol. 14, no. 6, pp. 66-77, November 1994.
- [22] M. Wanne and M. Linna, "A general model for adjacency," *Fundamenta Informaticae*, vol. 38, no. 1-2, pp. 39-50, April 1999.
- [23] J. Töyli, M. Linna, and M. Wanne, "Modeling relational data by the adjacency model," in *Proceedings of the Fourth International Conference on Enterprise Information Systems*, Universidad de Castilla-La Mancha, Ciudad Real, Spain, 2002, pp. 301-306.
- [24] J. Töyli, M. Linna, and M. Wanne, "Modeling semistructured data by the adjacency model," in *Proceedings of the Fifth Joint Conference on Knowledge-Based Software Engineering*, University of Maribor, Maribor, Slovenia, 2002, pp. 282-290.
- [25] J. Töyli, "AdSchema - a schema for semistructured data," *Acta Wasaensia*, no. 157, Vaasa: University of Vaasa, Finland, 2006.
- [26] U. Kjellén, *Prevention of accidents Through Experience Feedback*, 1st ed. England, London: Taylor & Francis, 2000.
- [27] H. W. Heinrich, *Industrial Accident Prevention - A Scientific Approach*, 4th ed. USA, New York: McGraw-Hill, 1959.
- [28] F. E. Bird and G. L. Germain, *Practical Loss Control Leadership*. USA, Georgia, Loganville: Division of International Loss Control Institute, 1985.
- [29] J. Reason, *Managing the Risk of Organizational Accidents*. 1st ed. England, Hampshire, Ashgate, 1991.
- [30] European Comission, *European Statistics on Accident at Work Methodology (ESAW)*, Eurostat, 2001.



Teemu Mäenpää received his M.Sc degree in economics and now is a doctoral student at University of Vaasa, Faculty of technology, department of computer science. He is a member of data modeling research group, DatMo at University of Vaasa. His research interests include: adjacency relation systems, data structures, data modeling, and knowledge management.



Vesa Nyrhilä received his M.Sc degree in economics and now is a doctoral student at University of Vaasa, Faculty of technology, department of computer science. He is a member of data modeling research group, DatMo at University of Vaasa. His research interests include: semantic link network, semantic data modeling, and knowledge management at the field of accident prevention management.