Relational Scaling in Relational Semantic Systems*

Karl Erich Wolff

Mathematics and Science Faculty
Darmstadt University of Applied Sciences
Schoefferstr. 3, D-64295 Darmstadt, Germany
karl.erich.wolff@t-online.de
http://www.fbmn.fh-darmstadt.de/home/wolff

Abstract. In this paper two developments in Conceptual Knowledge Processing are combined, namely Contextual Logic introduced by Rudolf Wille and Temporal Concept Analysis introduced by the author. The basic structures connecting both theories are Relational Semantic Systems (RSS), each consisting of conceptual scales and a Relational Data Systems (RDS) for the representation of relational knowledge. We introduce the notion of a concept graph of a RSS. As opposed to the definition of a concept graph of a power context family the concepts used in the concept graph of a RSS are taken from the conceptual scales and not from the concept lattices of the contexts of k-ary relations. For the graphical representation of relational knowledge in information maps we modify tools from Temporal Concept Analysis and develop *relational trace diagrams*. Its usefulness is shown in a small example of a Relational Semantic System.

1 Introduction

In this paper relational conceptual structures are investigated with the purpose to develop practically successful methods for the representation, evaluation and visualization of relational structures. These investigations are based on the Theory of Conceptual Graphs as developed by J. Sowa [So84, So00] and the mathematization of concepts in Formal Concept Analysis [Wi82, GW99a]. For the purpose of representing relational knowledge R. Wille [Wi97] has introduced power context families. To represent judgments he has introduced the notion of a concept graph of a power context family which is now the main tool in Contextual Judgment Logic and Contextual Conclusion Logic where inferences between formal judgments are studied [Pr98, PW99]. To combine the knowledge representation in power context families with the advantages of many-valued contexts and the successful tools for conceptual scaling, as for example the program TOSCANAJ (see [BH05]), the notion of Relational Scaling has been introduced in [PW99] and connected to database theory in [He02]. These ideas had been

* This work is supported by DFG-RFBR project COMO, GZ: 436 RUS 113/829/0-1.

S. Rudolph, F. Dau, and S.O. Kuznetsov (Eds.): ICCS 2009, LNAI 5662, pp. 307–320, 2009.

continued in the paper *A Contextual-Logic Extension of TOSCANA* [EGSW00]. From its abstract we cite:

> As graphical representations we recommend, besides labelled line diagrams of concept lattices and Sowa's diagrams of conceptual graphs, particular information maps for utilizing background knowledge as much as possible.

In the following we contribute to these investigations by combining power context families and concept graphs with Conceptual Semantic Systems as introduced by the author in [Wo05b, Wo06, Wo07a]. A first step into that direction was the introduction of the notion of a *Relational Data System* (RDS) which, combined with a family of conceptual scales, yields the notion of a *Relational Semantic System* [Wo09]. These structures help to clarify the discussion about relational and conceptual scaling in power context families; they also improve the practical representation of relational knowledge, including the development of useful information maps.

1.1 Relational Trace Diagrams as Information Maps

An example of a nice information map about flights in Austria is shown in [EGSW00], Fig. 8, where surrounded by the border-line of Austria the Austrian towns with airport are represented as small circles which are connected by arrows indicating flight connections. Each arrowhead is connected by a dashed line with a small data table indicating further information about this flight connection, as for example the corresponding flight numbers, its departure and arrival times and its days.

The construction of a good information map needs a clever combination of several graphical tools, in this case the border-line of Austria, the correctly embedded points for the towns, the arrows and the small data tables. It is obvious that the geographical information about the border-line and the towns are *not* given in the data table on the Austrian flights as represented in Fig. 1 in [EGSW00].

It was shown by the author [Wo07a] how such geographical information can be used to construct an information map which may also include life tracks of moving objects (see Fig. 3 in [Wo07a]). The corresponding conceptual theory is based on the notion of *Temporal Conceptual Semantic Systems*. The ideas developed there seem to be relevant also for the discussion of relational conceptual structures in general. A first example has been shown by the author in [Wo09] where an animation of the Austrian flights [EGSW00] is explained using life tracks of flights.

In this paper we develop *relational trace diagrams* which are based on a new representation of objects which generalizes the broadly employed *objects as formal objects strategy* (OFOS) to represent objects from application domains as formal objects in many-valued contexts or formal contexts.

1.2 Representation of Objects: Not Necessarily as Formal Objects

Conceptual Semantic Systems have been introduced by the author [Wo04] with the purpose to understand the physical notions of "particles" and "waves" from a conceptual point of view. These notions are now well understood in that framework. The main idea was to study the ternary relation that "an object is at some time at some place" with respect to different granularities. That led to the notion of a *distributed object* as for example a wave or a wave packet. For that reason it was necessary to think about the role of formal objects and the consequences of representing objects (like particles) as formal objects. The main consequence of this representation is that each formal object has a unique object concept in each part of the derived context of the scaled many-valued context. That is nice if we represent a particle as a formal object and represent its place as an object concept in the concept lattice representing the space. Such an object "occupies a single point" in that space. But sometimes we wish to represent a particle, for example a ball or an electron, in such a way that it occupies a certain volume in space. For that purpose it is necessary to represent the objects not as formal objects; the mostly used alternative is to represent objects as values in the data table. That is for example the case when we represent several kinds of objects, like persons, days, places and some judgments of the kind that a person visited a place at some day. In the practice of many applications of FCA it is tried to choose a suitable kind of objects, for example the persons, to be represented as formal objects. Usually, the other objects are then represented as values in a many-valued context. It is well-known for all experts in FCA that the obligation to choose in some given application domain a suitable kind of objects to be represented as formal objects often yields no problems, but in some applications it is quite problematic. A typical example was our search for a good representation of temporal data where for example several persons at several points of time had to be represented conceptually. Should we take the persons as formal objects or the points of time? My actual solution to this problem, as given in the notion of a Conceptual Semantic System, is to take none of the many kinds of objects as formal objects. Instead, the formal objects g are interpreted as *basic judgments*, each representing the information given in the row of g in the data table of the given Conceptual Semantic System.

1.3 Formal Concepts as Values in a Many-Valued Context

At that point of the discussion we should notice that the usage of the notion of a "value" in a many-valued context does not fit really well with the accepted and basic doctrines of concepts, judgments, and conclusions [Wi00]. It seems natural to understand a "value" as a concept, and to represent it as a formal concept in some formal context which explains the meaning of the given "value".

Therefore, in the definition of a conceptual semantic system (CSS) [Wo04] we start with a family $(\mathbb{S}_m)_{m \in M}$ of formal contexts; their formal concepts are used as values of a many-valued context, described by a mapping $\lambda : G \times M \to W$ which satisfies $\lambda(g, m) \in \mathfrak{B}(\mathbb{S}_m)$ for all $g \in G$ and all $m \in M$.

Each element of $g \in G$ represents the information $(\lambda(g,m)_{m\in M})$ given in its row of the data table. For the theory and applications of CSS and temporal CSS the reader is referred to [Wo04, Wo05b, Wo06, Wo07a, Wo07b]. Useful construction methods for information maps based on (temporal) CSS have been demonstrated in these papers, for example in [Wo07a], Fig. 3 where a weather map with a moving high pressure zone is shown. It is well-known that such maps are valuable tools for the representation of relational knowledge in a suitably chosen granularity. The formal connection between these two important fields of research, namely the relational structures on one side and the granularity structures on the other side are not well understood until now. In this paper we do some steps to improve our understanding of the connections among these fields of research.

1.4 Relations and Granularity

For the purpose of combining relations and granularity in a mathematical theory I like to use well-established approaches, namely to ground on the philosophical doctrines of concepts, judgments, and conclusions as emphasized by R. Wille [Wi00]. I also agree with Wille's idea to represent relations as formal concepts, such that the extent of such a relation concept is a subset of a direct product of sets; hence the usual mathematical notion of a relation corresponds to the extent of a relation concept.

For the formal representation of statements like "ALICE works as a TEACHER in BERLIN" we introduce the relation " . works as a . in ." and formalize that statement as an *infon* in the sense of Devlin [De91] by (. works as a . in .; ALICE, TEACHER, BERLIN). In general, an infon has the form $(R; x_1, ..., x_k)$ where R denotes a k-ary relation and $(x_1, ..., x_k) \in R$.

For the contextual representation of a set of infons with possibly different arities of their relations we represent each infon in a row of a data table. The formal definition of such conceptual structures has been introduced by the author in [Wo09] under the name *Relational Semantic Systems* (RSS) since they are Conceptual Semantic Systems with an additional structure for the formal representation of the relational information. This additional relational structure is defined in the notion of a *Relational Data Systems* (RDS). The precise definitions will be recalled from [Wo09] in the following.

2 Relational Data Systems

The main idea for the introduction of a Relational Data System can be explained easily using the example in Table 1 where five infons labeled from 1,...,5 are represented.

To represent statements like "BOB lives in ENGLAND" or "ALICE works as a TEACHER in BERLIN" in a data table we have to represent some linear ordering on the set of words of the statement. One possibility is to use the spoken sequence of the words of the given statement, but we restrict ourselves

Table 1. A data table for relational information

infon	r*	$PERSON_1$	$PERSON_2$	PROFESSION	LOCATION
1	.lives in.	BOB			ENGLAND
2	.lives in.	BOB			LONDON
3	.works as a.in.		ALICE	TEACHER	BERLIN
4	.meets.in.	BOB	ALICE		PARIS
5	.is the native town of.	ALICE			PARIS

Table 2. The arity-position table

c	$\alpha(c)$	$PERSON_1$	$PERSON_2$	PROFESSION	LOCATION
.lives in.	2	1			2
.works as a.in.	3		1	2	3
.meets.in.	3	1	2		3
.is the native town of.	2	2			1

to statements of the form of an infon $(R; x_1, \ldots, x_n)$. The example "PARIS is the native town of ALICE" shows that we do not insist that the statements have to be read from left to right in the table; clearly, the set M of attributes of a RDS is not assumed to be ordered.

In general, we map each value c of the many-valued attribute r* (called the *relational attribute*) first to a non-negative integer $\alpha(c)$ which is interpreted as its arity, second to a subset of the set of many-valued attributes $\beta(c)$ which is called the *region* of c, and third, for $\beta(c) \neq \emptyset$, to a bijection π_c which assigns to each integer $i \in [1, \alpha(c)]$ a many-valued attribute $\pi_c(i) \in \beta(c)$, called the "$i-$th position of c". In our example the mappings α, β and π_c are given in Table 2.

For example, for the relation ".lives in." the arity is 2, the region β(.lives in.) = {$PERSON_1$, LOCATION} and the first position of ".lives in." is $PERSON_1$, its second position is LOCATION. From Table 1 we see that there are two rows in which the relation ".lives in." occurs, giving the statements "BOB lives in ENGLAND" and "BOB lives in LONDON".

In the following mathematical definition of a Relational Data System we do not represent any specific interpretation of the values as for example the interpretation that a value "is a relation". Therefore we do not use here the dots occuring in the relation names.

2.1 Definition of a Relational Data System

The following definition of a Relational Data System has been introduced by the author in [Wo09] with the purpose to represent any given power context family, and any given concept graph, and to combine them with the possibility for conceptual scaling which will be done in Relational Semantic Systems later.

Definition 1. *"Relational Data System"*
Let G, M, W be sets, $\lambda : G \times M \to W$. Then $\mathfrak{R} := (G, M, W, \lambda, r^*, A, \alpha, \beta, \pi)$ is called a Relational Data System (RDS) if

- $r^* \in M$
- $A \subseteq W_{r^*} := \{\lambda(g, r^*) \mid g \in G\}$
- $\alpha : W_{r^*} \to \mathbb{N} := \{1, 2, \ldots\}$
- $\beta : W_{r^*} \to \mathfrak{P}(M \setminus \{r^*\}) := \{X \mid X \subseteq M \setminus \{r^*\}\}$
- *π is a mapping which maps each $c \in W_{r^*}$ with $\beta(c) \neq \emptyset$ to a bijection $\pi_c : [1, \alpha(c)] \to \beta(c)$.*

For $\beta(c) \neq \emptyset$ and $1 \leq i \leq \alpha(c)$ the many-valued attribute $\pi_c(i)$ is called the $i-$th position of c. For $c \in W_{r^*}$ the integer $\alpha(c)$ is called the arity of c. For $c \in W_{r^*}$ let $D_c := \{g \in G \mid \lambda(g, r^*) = c\}$, and for $\beta(c) \neq \emptyset$ and $g \in D_c$ let $\vec{c}(g) := (\lambda(g, \pi_c(i)))_{1 \leq i \leq \alpha(c)}$; $\vec{c}(g)$ is called the tuple of c at g. The set $A \subseteq W_{r^*}$ is called the set of artificial relations. The set $A \subseteq W_{r^*}$ will be used for the representation of a normed power context family $(\mathbb{K}_k)_{k \in S}$ by an RDS; then the *isolated* formal objects of $\mathbb{K}_k$, namely the elements of $\{g \in G_k \mid g^\uparrow = \emptyset\}$ will be collected in an artificial relation k^*. Then $\beta(c) \neq \emptyset$ for all $c \in A$ and the set $\tau(A) := \{\vec{c}(g) \mid c \in A,\, g \in D_c\}$ is disjoint from $\tau(W_{r^*} \setminus A)$.

3 Relational Data Systems and Power Context Families

Power context families have been introduced by Wille [Wi97] using the following definition.

> A *power context family* is a sequence $\vec{\mathbb{K}} := (\mathbb{K}_0, \mathbb{K}_1, \mathbb{K}_2, \ldots)$ of formal contexts $\mathbb{K}_k := (G_k, M_k, I_k)$ with $G_k \subseteq (G_0)^k$ for $k = 1, 2, \ldots$. The formal concepts of $\mathbb{K}_k$ with $k = 1, 2, \ldots$ are called *relation concepts*, because they represent k-ary relations on the object set G_0 by their extents.

For our purposes it is not necessary to distinguish between the formal contexts $\mathbb{K}_0$ and $\mathbb{K}_1$. Furthermore, we would like to make explicit the dots "..." in the previous definition by introducing a set S of indices.

Definition 2. *"Normed Power Context Family of Type S"*
$\vec{\mathbb{K}} := (\mathbb{K}_k)_{k \in S}$ is called a normed power context family of type S if

- *$S \subseteq \mathbb{N} := \{1, 2, \ldots\}$, $1 \in S$, and*
- *$\mathbb{K}_k = (G_k, M_k, I_k)$ is a formal context for all $k \in S$ such that $G_k \subseteq (G_1)^k$ for $k > 1$.*

$\vec{\mathbb{K}}$ is called a normed power context family (NPCF) if $\vec{\mathbb{K}}$ is a normed power context family of type S for some set S.

Clearly each power context family $\vec{\mathbb{K}} := (\mathbb{K}_0, \mathbb{K}_1, \mathbb{K}_2, \ldots)$ in the sense of the definition in [Wi97] can be represented by a normed power context family, simply by deleting $\mathbb{K}_0$ and replacing $\mathbb{K}_1$ by $(G_0, M_0 \cup M_1, I_0 \cup I_1)$ assuming M_0 and M_1 disjoint.

It has been proven by the author in [Wo09] that each normed power context family can be faithfully represented by a RDS in the following sense.

In [Wo09] two operators $\mathbf{R}$ and $\vec{\mathbf{K}}$ have been introduced; $\mathbf{R}$ maps each normed power context family $\vec{\mathbb{K}}$ to a Relational Data System $\mathbf{R}(\vec{\mathbb{K}})$, while $\vec{\mathbf{K}}$ maps each Relational Data System $\mathfrak{R}$ to a normed power context family $\vec{\mathbf{K}}(\mathfrak{R})$; in [Wo09], Proposition 1 says that for any normed power context family $\vec{\mathbb{K}}$ (with non-empty contexts) we get $\vec{\mathbf{K}}(\mathbf{R}(\vec{\mathbb{K}})) = \vec{\mathbb{K}}$; and Proposition 2 shows that for any RDS $\mathfrak{R}$ (with non-empty contexts in its power context family) we get $\vec{\mathbf{K}}(\mathbf{R}(\vec{\mathbf{K}}(\mathfrak{R}))) = \vec{\mathbf{K}}(\mathfrak{R})$, but $\mathbf{R}(\vec{\mathbf{K}}(\mathfrak{R}))$ is not necessarily equal to $\mathfrak{R}$.

4 Relational Semantic Systems

4.1 Definition of a Relational Semantic Systems

In this section we recall from [Wo09] the notion of a Relational Semantic System (RSS) which covers the notions of a Conceptual Semantic System (CSS) [Wo05b, Wo06] and the notion of a Relational Data System. That makes explicit the idea that we list in each row of a data table an infon $(R; c_1, \ldots, c_k)$. According to traditional philosophical logic with its doctrines of concepts, judgements, and conclusions (cf. [Wi00]) we start with concepts $c_1, \ldots, c_k$ which are combined with a relational concept R to build an infon $(R; c_1, \ldots, c_k)$. Therefore, we represent the concepts $R, c_1, \ldots, c_k$ as formal concepts of formal contexts $\mathbb{S}_m$ $(m \in M)$. These formal contexts will play the role of conceptual scales of the many-valued context described in the following definition of a Relational Semantic System.

Definition 3. *"Relational Semantic System"*
Let $\mathfrak{R} := (G, M, W, \lambda, r^*, A, \alpha, \beta, \pi)$ be a Relational Data System and for each $m \in M$ let $\mathbb{S}_m := (G_m,\ N_m,\ I_m)$ be a formal context and $\underline{\mathfrak{B}}(\mathbb{S}_m)$ its concept lattice. If $\lambda : G \times M \to W$ satisfies $\lambda(g, m) \in \mathfrak{B}(\mathbb{S}_m)$ for all $g \in G$ and all $m \in M$, then the pair $(\mathfrak{R}, (\mathbb{S}_m)_{m \in M})$ is called a Relational Semantic System (RSS).

Remark: If $(\mathfrak{R}, (\mathbb{S}_m)_{m \in M})$ is a RSS, then $(G, M, (\underline{\mathfrak{B}}(\mathbb{S}_m))_{m \in M}, \lambda)$ is a CSS.

4.2 Example of a Relational Semantic Systems

We construct a RSS $(\mathfrak{R}_1, (\mathbb{S}_m)_{m \in M})$ from the
RDS $\mathfrak{R}_1 := (G, M, W, \lambda, r^*, A, \alpha, \beta, \pi)$ given in Table 1 and Table 2, where $G := \{1, 2, 3, 4, 5\}$, $M := \{r^*, PERSON_1, PERSON_2, PROFESSION, LOCATION\}$, $W := \bigcup_{m \in M} \mathfrak{B}(\mathbb{S}_m)$; to define the scales we denote for any

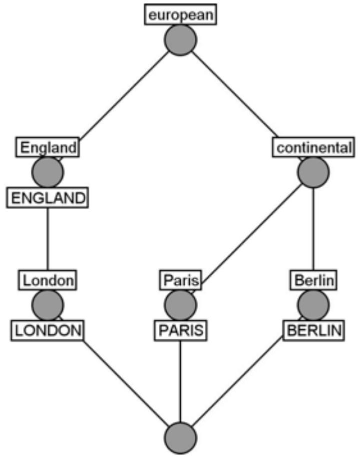

Fig. 1. The scale for LOCATION

set X by $N(X) := (X, X, =)$ the nominal scale on X.
$\mathbb{S}_{r^*} := N(\{$.lives in., .works as a.in, .meets.in., .is the native town of.$\})$,
$\mathbb{S}_{PERSON_1} := \mathbb{S}_{PERSON_2} := N(\{$BOB, ALICE$\})$,
$\mathbb{S}_{PROFESSION} := N(\{$TEACHER$\})$,
$\mathbb{S}_{LOCATION}$ is given by the line diagram of its concept lattice in Fig. 1.

For $g \in G$ and $m \in M$ let $\lambda(g, m) := \gamma_m(h)$ be the object concept in $\mathbb{S}_m$ of the formal object h which occurs in row g and column m in Table 1. For example, the fact that the name "BOB" occurs in Table 1 in row 1 and column "PERSON$_1$" is understood as an abbreviation for $\lambda(1, \text{PERSON}_1) = \gamma_{PERSON_1}(BOB)$. For short, each value in Table 1 is a formal object in the scale of its column and represents its object concept in that scale.

In other examples the tabular description of the mapping λ may not use the object names for the representation of object concepts, for example if some λ-values are not object concepts. To continue the definition of the RDS $\mathfrak{R}_1$ we mention that r^* is the relational attribute, that $A := \emptyset$, and that α, β, π can be seen from the arity-position table in Table 2.

5 Concept Graphs of a Relational Semantic System

5.1 Definition of a Concept Graph of a Relational Semantic System

To define the notion of a *concept graph of a Relational Data System* we recall the definition of a relational graph ([Wi04]).

A *relational graph* is a structure (V, E, ν) consisting of two disjoint sets V and E together with a map $\nu : E \to \bigcup_{k=1,2,\ldots} V^k$; the elements of V and E are called *vertices* and *edges*, respectively, and $\nu(e) = (v_1, \ldots, v_k)$ is read: $v_1, \ldots, v_k$ are the *adjacent vertices* of the *k-ary edge* e ($|e| := k$ is the *arity* of e; the arity of a vertex is defined to be 0). Let $E^{(k)}$ be the set of all elements of $V \cup E$ of arity k ($k = 0, 1, 2, \ldots$).

In the following definition of a concept graph of a RSS we modify the definition of a concept graph of a power context family.

Definition 4. *"Concept Graph of a Relational Semantic System"*
Let $(\mathfrak{R}, (\mathbb{S}_m)_{m\in M})$ *be a RSS and* $\mathfrak{R} := (G, M, W, \lambda, r^*, A, \alpha, \beta, \pi)$. *A concept graph of* $(\mathfrak{R}, (\mathbb{S}_m)_{m\in M})$ *is a structure* $\mathfrak{G} := (V, E, \nu, \kappa, \rho)$ *for which*

- (V, E, ν) *is a relational graph, where the arity of an edge* $e \in E$ *is* $|e| := k$, *if* $\nu(e) \in V^k$
- $\kappa\colon V \cup E \to \bigcup_{m\in M} \mathfrak{B}(\mathbb{S}_m)$
- $\rho\colon V \to \mathfrak{P}(\bigcup_{m\in M\setminus\{r^*\}} \mathfrak{B}(\mathbb{S}_m))\setminus\{\emptyset\}$

such that for $v \in V$ *and* $e \in E$

1. $\kappa(v) \in \bigcup_{m\in M\setminus\{r^*\}} \mathfrak{B}(\mathbb{S}_m)$
2. $\kappa(e) \in \mathfrak{B}(\mathbb{S}_{r^*})$
3. *if* $\nu(e) = (v_1, \ldots, v_k)$, *then* $\alpha(\kappa(e)) = k = |e|$ *and for* $1 \le i \le k$
4. $\kappa(v_i) \in \mathfrak{B}(\mathbb{S}_{p(e,i)})$, *where* $p(e, i) := \pi_{\kappa(e)}(i)$ *is the i-th position of* $\kappa(e)$ *and*
5. $\rho(v_i) \subseteq \mathfrak{B}(\mathbb{S}_{p(e,i)})$ *such that*
6. $d \le \kappa(v_i)$ *for all concepts* $d \in \rho(v_i)$ *and*
7. *for all* $(d_1, \ldots, d_k) \in \rho(v_1) \times \ldots \times \rho(v_k)$ *there is a* $g \in G$ *such that* $\lambda(g, r^*) = \kappa(e) =: c$ *and* $\vec{c}(g) = (d_1, \ldots, d_k)$.

This definition of a concept graph of a RSS differs from the definition of a concept graph of a power context family in a remarkable point, namely in the choice of the employed concept lattices: for a concept graph of a RSS the concept lattices $\mathfrak{B}(\mathbb{S}_m)$ are employed, while for a concept graph of a power context family $(\mathbb{K}_0, \mathbb{K}_1, \mathbb{K}_2, \ldots)$ the concept lattices $\mathfrak{B}(\mathbb{K}_k)$ are used. Either of the two series of concept lattices are needed: the concept lattices $\mathfrak{B}(\mathbb{S}_m)$ represent the meaning of the formal concepts which occur as values in a given RSS $(\mathfrak{R}, (\mathbb{S}_m)_{m\in M})$; the concept lattice $\mathfrak{B}(\mathbb{K}_k)$ of the power context family $\vec{\mathbf{K}}(\mathfrak{R})$ represents the intersections of all chosen $k-$ary relations.

An example of a concept graph of a Relational Semantic System will be shown in the next subsection.

5.2 Example of a Concept Graph of a Relational Semantic System

Fig. 2 shows a graphic representing a concept graph of the Relational Semantic System $(\mathfrak{R}_1, (\mathbb{S}_m)_{m\in M})$. This graphic is drawn according to J. Sowa's convention for drawing conceptual graphs [So84]. This concept graph represents a judgment consisting of the first four infons in Table 1.

The graphic in Fig. 2 visualizes the relational graph with three edges e_1, e_2, e_3, drawn as ellipses, and six vertices $v_1, \ldots, v_6$, drawn as boxes. The ellipse of an edge e is connected to the boxes of its adjacent vertices by straight lines which are labeled by the integers $1, \ldots, |e|$ where $|e|$ is the arity of e. For example, $\nu(e_2) = (v_1, v_4, v_3)$ is the tuple of adjacent vertices of e_2. Now it is easily seen that Fig. 2 represents a relational graph.

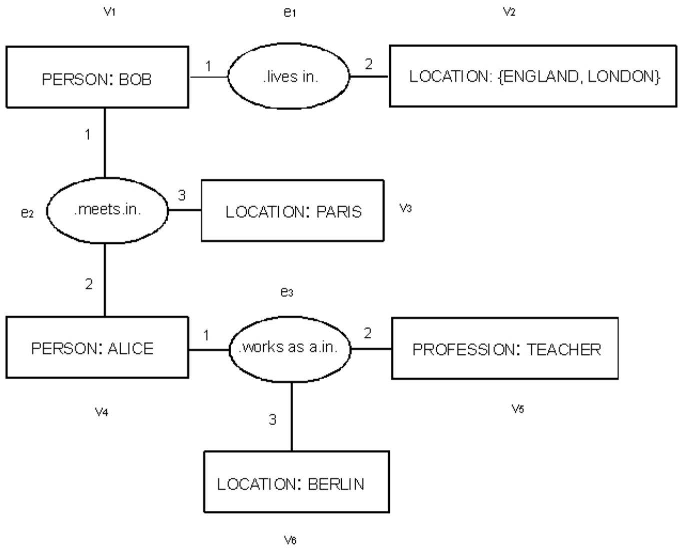

Fig. 2. A concept graph of a Relational Semantic System

To define the mappings κ and ρ using Fig. 2 we mention that the name in the ellipse of an edge e denotes the formal concept $\kappa(e)$; for example, $\kappa(e_1)$ is the object concept of the formal object ".lives in." in the scale $\mathbb{S}_{r^*}$. The first name in the box of a vertex v denotes the formal concept $\kappa(v)$, the second name denotes its reference set $\rho(v)$. In Fig. 2 the first name in a box denotes the top concept of the corresponding scale; for example, $\kappa(v_2) = LOCATION$ is understood here as the top concept in the concept lattice of $\mathbb{S}_{LOCATION}$. Therefore, one can easily check that κ and ρ are mappings as demanded.

Conditions (1.) to (3.) in Def. 4 are obviously satisfied, for example, the arity $\alpha(\kappa(e_1)) = \alpha(\text{.lives in.}) = 2 = |e_1|$. Condition (4.) is satisfied; for example, in the tuple $\nu(e_2) = (v_1, v_4, v_3)$ the vertex v_4 as the *second* entry in that tuple satisfies that $\kappa(v_4)$ is the top concept of the scale $\mathbb{S}_{PERSON_1} = \mathbb{S}_{PERSON_2}$, hence it is an element of $\mathfrak{B}(\mathbb{S}_{PERSON_2})$ and $PERSON_2 = p(\kappa(e_2), 2)$ is the *second* position of $\kappa(e_2)$, the object concept of ".meets.in.". Condition (5.) is also satisfied; as an example we choose edge e_1 and its *second* vertex v_2. Its reference set is $\rho(v_2) = \{\gamma_{LOCATION}(ENGLAND), \gamma_{LOCATION}(LONDON)\}$ which is a subset of $\mathfrak{B}(\mathbb{S}_{p(e_1,2)}) = \mathfrak{B}(\mathbb{S}_{LOCATION})$. Obviously, condition (6.) is satisfied; we discuss condition (7.) for the edge e_1; for each of the two 2-tuples in the set $\rho(v_1) \times \rho(v_2)$ there is an infon satisfying the condition, namely infon 1 and infon 2, see Table 1. Condition (7.) is satisfied obviously also for the other edges.

6 Relational Trace Diagrams

Trace diagrams have been introduced by the author [Wo07a]. The main idea for the construction of trace diagrams stems from usual weather maps where

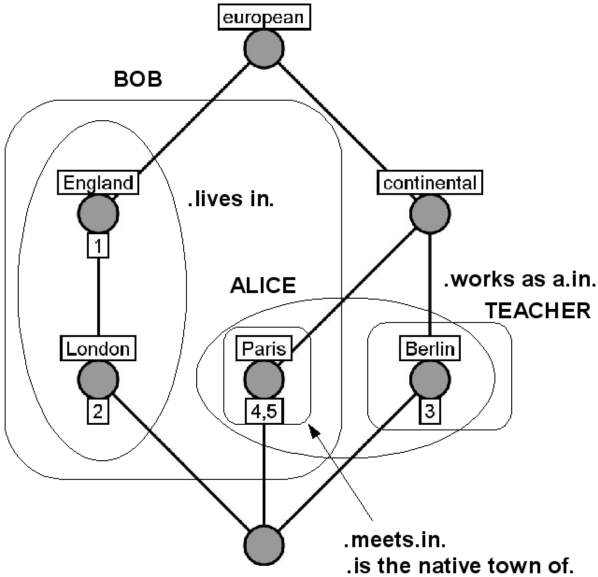

Fig. 3. An information map of the location view

information about many entities like pressure or temperature is embedded into a single map. In this section we introduce *relational trace diagrams* of Relational Semantic Systems using a small example.

6.1 Example of a Relational Trace Diagram

Fig. 3 shows a relational trace diagram of the Relational Semantic System of the previous example.

It is constructed as follows: starting from the RSS $(\mathfrak{R}_1, (\mathbb{S}_m)_{m\in M})$ in the previous example we take its underlying CSS $(G, M, (\underline{\mathfrak{B}}(\mathbb{S}_m))_{m\in M}, \lambda)$ where $\mathbb{S}_m = (G_m, N_m, I_m)$ $(m \in M)$ and construct its (semantically) derived context [Wo07a]

$$\mathbb{K} := (G, N, J) \text{ where } N := \{(m, n) \mid m \in M, n \in N_m\} \text{ and}$$
$$gJ(m, n) :\Longleftrightarrow n \in int(\lambda(g, m)).$$

Remark: Roughly speaking, each formal concept $\lambda(g, m) \in \mathfrak{B}(\mathbb{S}_m)$ is represented in the derived context by its intent.

Then we select a *view* Q which is defined as a subset $Q \subseteq N$ and construct its corresponding subcontext $\mathbb{K}_Q := (G, Q, J \cap (G \times Q))$, called the $Q-$*part* of $\mathbb{K}$. For the concept lattice in Fig. 3 the view is chosen as the set of the LOCATION-attributes in N. The concept lattice of the $Q-$part is shown in Fig. 3. To understand the role of the infons 1,...,5 in this $Q-$part we mention, that for each many-valued attribute $m \in M$ and each formal object (infon) $g \in G$ in the $m-$part $\mathbb{K}_m := (G, \{m\} \times N_m, J \cap (G \times (\{m\} \times N_m)))$ of the (semantically) derived context $\mathbb{K} := (G, N, J)$ we have

$$g^J = \{(m,n) | n \in B\} \text{ in } \mathbb{K}_m \Longleftrightarrow B \text{ is the intent of } \lambda(g,m) \text{ in } \mathbb{S}_m$$

by definition of the semantically derived context.

For example, for the many-valued attribute LOCATION and the infon 2 the set $B = \{\text{London, England, european}\}$ is the intent of $\lambda(2, LOCATION) = \mu_{LOCATION}(London)$, the attribute concept of *London* in the scale $\mathbb{S}_{LOCATION}$.

To explain the representation of relational knowledge in Fig. 3 we use the object representation and the traces of objects in CSS as introduced in [Wo07a, Wo07b].

6.2 Object Representation and Traces of Objects

For the formal definition of the notion of an *object* in Conceptual Semantic Systems and the definition of a trace of an object in some view Q the reader is referred to [Wo07a, Wo07b]. To present the main ideas quickly we use our example of the RSS $(\mathfrak{R}_1, (\mathbb{S}_m)_{m \in M})$ where we represent the object "BOB" not as a formal object, but as the formal concept $\mu_{PERSON_1}(BOB)$ of the scale $\mathbb{S}_{PERSON_1}$. Then we select the set $S := \{g \in G | \lambda(g, PERSON_1) = \mu_{PERSON_1}(BOB)\}$ and construct the set of object concepts $\gamma_{LOCATION}(S)$ which is a special trace of the formal concept $\mu_{PERSON_1}(BOB)$ in the LOCATION-view. This set of three object concepts is visualized in Fig. 3 by the rectangle labeled "BOB".

Clearly, in the same way we can represent traces of relation concepts in $\mathfrak{B}(\mathbb{S}_{r^*})$. They are shown in Fig. 3 for all four relation concepts. The same can be done for subtuples of an infon; for example, we might be interested in which locations Alice works as a teacher, for short denoted by the *question* "ALICE works as a TEACHER in ?"; then we would find only the object concept of 3 in the LOCATION-view, and that is the attribute concept of BERLIN.

The construction of such traces can be supported in the computer program TOSCANAJ choosing the scale of the view Q as the last in the priority list for the nested line diagrams. For example, for the question "ALICE works as a TEACHER in ?" one should choose the priority list (PERSON_2, r^*, PROFESSION, LOCATION). Clicking on ALICE, .works as a.in., TEACHER in their scales yields only those object concepts which satisfy all the conditions. Hence the trace of the object tuple (ALICE, .works as a.in., TEACHER) consists only of the object concept of 3, hence we get as answer the location "BERLIN".

7 Conclusion and Future Work

In this paper two developments in Conceptual Knowledge Processing are combined, namely Contextual Logic introduced by Rudolf Wille and Temporal Concept Analysis introduced by the author. The basic structures which can now serve for both theories are Relational Semantic Systems. They are defined using the notion of a Relational Data System which has been introduced recently by the author for the representation of relational knowledge [Wo09].

A Relational Semantic System $(\mathfrak{R}, (\mathbb{S}_m)_{m \in M})$ contains the conceptual scales $\mathbb{S}_m$ as well as the relational scales, namely the formal contexts of its power context family $\vec{\mathbf{K}}(\mathfrak{R})$. The concept graphs of a Relational Semantic System represent formal concepts of the conceptual scales, while the concept graphs of a power context family represent the formal concepts of the relational scales.

For the graphical representation of relational structures we have employed the object representation by tuples and the visualization of traces of objects as developed by the author in Temporal Concept Analysis. That yields trace diagrams for the representation of relational knowledge. Its usefulness is shown in a small example of a Relational Semantic System.

Future work has to combine the rich theory in Contextual Logic with Relational Semantic Systems. One of the most challenging problems is the development of a Conclusion Logic in Relational Semantic Systems. In many problems in practice the relational structures change with time, therefore temporal Relational Semantic Systems should be developed.

References

[BH05] Becker, P., Hereth Correia, J.: The ToscanaJ Suite for Implementing Conceptual Information Systems. In: Ganter, B., Stumme, G., Wille, R. (eds.) Formal Concept Analysis. LNCS (LNAI), vol. 3626, pp. 324–348. Springer, Heidelberg (2005)

[De91] Devlin, K.: Logic and Information. Cambridge University Press, Cambridge (1991)

[EGSW00] Eklund, P., Groh, B., Stumme, G., Wille, R.: A contextual-logic extension of TOSCANA. In: Ganter, B., Mineau, G.W. (eds.) ICCS 2000. LNCS (LNAI), vol. 1867. Springer, Heidelberg (2000)

[GW99a] Ganter, B., Wille, R.: Formal Concept Analysis: mathematical foundations. Springer, Heidelberg (1999); German version: Springer, Heidelberg (1996)

[He02] Hereth, J.: Relational Scaling and Databases. In: Priss, U., Corbett, D.R., Angelova, G. (eds.) ICCS 2002. LNCS (LNAI), vol. 2393, pp. 62–76. Springer, Heidelberg (2002)

[Pr98] Prediger, S.: Kontextuelle Urteilslogik mit Begriffsgraphen. Ein Beitrag zur Restrukturierung der mathematischen Logik. Dissertation, TU Darmstadt 1998. Shaker, Aachen (1998)

[PW99] Prediger, S., Wille, R.: The lattice of concept graphs of a relationally scaled context. In: Tepfenhart, W.M. (ed.) ICCS 1999. LNCS (LNAI), vol. 1640, pp. 401–414. Springer, Heidelberg (1999)

[So84] Sowa, J.F.: Conceptual structures: information processing in mind and machine. Addison-Wesley, Reading (1984)

[So00] Sowa, J.F.: Knowledge representation: logical, philosophical, and computational foundations. Brooks Cole Publ. Comp., Pacific Grove (2000)

[Wi82] Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival, I. (ed.) Ordered sets, pp. 445–470. Reidel, Dordrecht (1982)

[Wi97] Wille, R.: Conceptual Graphs and Formal Concept Analysis. In: Delugach, H.S., Keeler, M.A., Searle, L., Lukose, D., Sowa, J.F. (eds.) ICCS 1997. LNCS (LNAI), vol. 1257, pp. 290–303. Springer, Heidelberg (1997)

[Wi00] Wille, R.: Contextual Logic summary. In: Stumme, G. (ed.) Working with conceptual structures: Contributions to ICCS 2000, pp. 265–276. Shaker-Verlag, Aachen (2000)

[Wi04] Wille, R.: Implicational Concept Graphs. In: Wolff, K.E., Pfeiffer, H.D., Delugach, H.S. (eds.) ICCS 2004. LNCS (LNAI), vol. 3127, pp. 52–61. Springer, Heidelberg (2004)

[Wo01] Wolff, K.E.: Temporal Concept Analysis. In: Mephu Nguifo, E., et al. (eds.) ICCS 2001 International Workshop on Concept Lattices-Based Theory, Methods and Tools for Knowledge Discovery in Databases, Stanford University, Palo Alto (CA), pp. 91–107 (2001)

[Wo02a] Wolff, K.E.: Transitions in Conceptual Time Systems. In: Dubois, D.M. (ed.) International Journal of Computing Anticipatory Systems, CHAOS 2002, vol. 11, pp. 398–412 (2002)

[Wo04] Wolff, K.E.: 'Particles' and 'Waves' as Understood by Temporal Concept Analysis. In: Wolff, K.E., Pfeiffer, H.D., Delugach, H.S. (eds.) ICCS 2004. LNCS (LNAI), vol. 3127, pp. 126–141. Springer, Heidelberg (2004)

[Wo05a] Wolff, K.E.: States, Transitions, and Life Tracks in Temporal Concept Analysis. In: Ganter, B., Stumme, G., Wille, R. (eds.) Formal Concept Analysis. LNCS, vol. 3626, pp. 127–148. Springer, Heidelberg (2005)

[Wo05b] Wolff, K.E.: States of Distributed Objects in Conceptual Semantic Systems. In: Dau, F., Mugnier, M.-L., Stumme, G. (eds.) ICCS 2005. LNCS (LNAI), vol. 3596, pp. 250–266. Springer, Heidelberg (2005)

[Wo06] Wolff, K.E.: Conceptual Semantic Systems - Theory and Applications. In: Goncharov, S.S., Downey, R., Ono, H. (eds.) Mathematical Logic in Asia. Proceedings of the 9th Asian Logic Conference, pp. 288–301. World Scientific, New Jersey (2006)

[Wo07a] Wolff, K.E.: Basic Notions in Temporal Conceptual Semantic Systems. In: Kuznetsov, S.O., Schmidt, S. (eds.) ICFCA 2007. LNCS, vol. 4390, pp. 97–120. Springer, Heidelberg (2007)

[Wo07b] Wolff, K.E.: Applications of Temporal Conceptual Semantic Systems. In: Zagoruiko, N.G., Palchunov, D.E. (eds.) Knowledge - Ontology - Theory. Russian Academy of Sciences. Sobolev Institute for Mathematics. Novosibirsk, vol. 2, pp. 3–16 (2007)

[Wo09] Wolff, K.E.: Relational Semantic Systems, Power Context Families, and Concept Graphs. In: The Proceedings of the International Conference on Formal Concept Analysis 2009, Darmstadt (accepted for publication, 2009)