A Finite State Model for On-Line Analytical Processing in Triadic Contexts

Gerd Stumme

Chair of Knowledge & Data Engineering, Department of Mathematics and Computer Science,
University of Kassel, Wilhelmshöher Allee 73, D–34121 Kassel, Germany
http://www.kde.cs.uni-kassel.de

Abstract. About ten years ago, triadic contexts were presented by Lehmann and Wille as an extension of Formal Concept Analysis. However, they have rarely been used up to now, which may be due to the rather complex structure of the resulting diagrams. In this paper, we go one step back and discuss how traditional line diagrams of standard (dyadic) concept lattices can be used for exploring and navigating triadic data.

Our approach is inspired by the *slice & dice* paradigm of On-Line-Analytical Processing (OLAP). We recall the basic ideas of OLAP, and show how they may be transferred to triadic contexts. For modeling the navigation patterns a user might follow, we use the formalisms of finite state machines. In order to present the benefits of our model, we show how it can be used for navigating the IT Baseline Protection Manual of the German Federal Office for Information Security.

1 Introduction

Concept lattices have proven their high potential for visualizing and exploring datasets in many applications during the last 25 years. This success of Formal Concept Analysis incited researchers to extend it to other types of knowledge representation. Among them are for instances logical extensions, relational data, and power context families. One of these extensions are *triadic contexts*, which were introduced ten years ago by Fritz Lehmann and Rudolf Wille in [14]. They defined a *triadic formal context* as a quadruple $\mathbb{K} := (G, M, B, Y)$ where G, M, and B are sets, and Y is a ternary relation between G, M, and B, i. e., $Y \subseteq G \times M \times B$. The elements of G, M, and B are called *(formal) objects*, *attributes*, and *conditions*, resp, and $(g, m, b) \in Y$ is read "object g has attribute m under condition b. A *triadic concept* of $\mathbb{K}$ is a triple (A_1, A_2, A_3) with $A_1 \subseteq G$, $A_2 \subseteq M$, and $A_3 \subseteq B$ where $A_1 \times A_2 \times A_3 \subseteq Y$ such that none of its three components can be enlarged without violating this condition.

Lehmann and Wille present an extension of the theory of ordered sets and (concept) lattices to the triadic case, and discuss structural properties. This approach initiated research on the theory of *concept trilattices*, which was followed by several researchers (e. g., [1, 2, 3, 4, 5, 6, 8, 10, 11, 15, 16, 17, 18, 20, 21, 22]). Already in the first paper on this topic, Lehmann and Wille elaborated also a visualization of concept trilattices in *triadic diagrams*. But even though there are applications where the natural representation of the data are triadic contexts, the visualization by triadic diagrams never made it into

B. Ganter and R. Godin (Eds.): ICFCA 2005, LNAI 3403, pp. 315–328, 2005.

practice, and there exist only few visualizations of rather small concept trilattices. This is probably due to the complex structure of the diagrams. In this paper, we go one step back and discuss how traditional line diagrams of dyadic concept lattices can be used for exploring and navigating triadic data.

The idea of deriving dyadic contexts from the triadic one is not new. Lehmann and Wille present, for instance, in [14] the derived dyadic context $\mathbb{K}^{(1)} := (G, M \times B, Y^{(1)})$ with $(g,(m,b)) \in Y^{(1)} :\iff (g,m,b) \in Y$ (marked by 'W' below), and its two symmetric variations. In [8], the set B is used to define two modal operators (marked by '$\exists$' and '$\forall$' below). We will use these derivation modes later, but will set them in a common navigation framework.

Our approach for navigating triadic data is inspired by the *slice & dice* paradigm of On-Line-Analytical Processing (OLAP). We present the basic ideas of OLAP in the next section, and show how they may be transferred to triadic contexts. For modeling the navigation patterns a user might follow, we use the formalisms of finite state machine (see Section 3). In order to present the benefits of our model, we show in Section 4 how it can be used for navigating the IT Baseline Protection Manual of the German Federal Office for Information Security. As this model is only a first step to a comprehensive navigation environment for triadic (and possibly other) data, many interesting research questions remain open. They conclude the article.

2 On-Line Analytical Processing and Triadic Contexts

The expression *On-Line Analytical Processing (OLAP)* has been coined by E. F. Codd et al in [7], and stands for the analysis of multi-dimensional data. We will first give a short introduction in the main features of OLAP as far as they are needed in this paper, before informally outlining how we adapt them to triadic contexts.

OLAP relies on the metaphor of a (high-dimensional) cube containing data. One might for instance want to structure sales facts along the dimensions region, product and time. These dimensions span a three-dimensional cube as shown in Fig. 1. The cube is composed of cells, one for each combination of a region, a product, and a day. The cell contains a numerical value indicating how many items of that product have been purchased in the specific region on the given day.

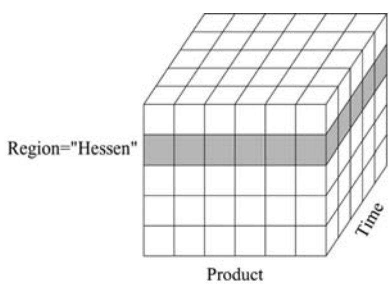

Fig. 1. A data cube