Chapter 15

KStore:
A Dynamic Meta-Knowledge Repository for Intelligent BI

Jane Campbell Mazzagatti
As It Is Inc., USA

ABSTRACT

KStore is a computer data structure based on the Phaneron of C. S. Peirce (Peirce, 1931-1958). This structure, called a Knowledge Store, KStore or simply K, is currently being developed as a storage engine to support BI data queries and analysis (Mazzagatti, 2007). The first Ks being constructed handle nominal data and record sequences of field/record data variables and their relationships. These rudimentary Ks are dynamic allowing real-time data processing, ad hoc queries and data compression, to facilitate data mining. This paper describes a next step in the development of the K structure, to record into the K structure, meta data associated with the field/record data, in particular the column or dimension names and a source indicator.

INTRODUCTION

Phaneroscopy is the description of the Phaneron; and by the Phaneron I mean the collective total of all that is in any way or in any sense present to the mind, quite regardless of whether it corresponds to any real thing or not (Peirce, 1931-1958).

The Phaneron is Peirce’s cognitive model. As Peirce describes the Phaneron, it contains anything that the mind requires to think, that would be experiential data and relational information, as well as processes or strategies to manipulate and/or retrieve information. All of these elements of the Phaneron would exist within the Phaneron and in the same triadic form that Peirce describes.

A rudimentary triadic structure, K, based on the Peirce phaneron, that creates a triadic recording of basic data and relational information, has been described and implemented (Mazzagatti, 2006). The resulting K structure is truly unique and has many attributes relevant to BI. The K structure compresses the data being recorded by reusing K nodes that represent data elements.

DOI: 10.4018/978-1-60960-595-7.ch015
The locking of the K structure is at the K node level and the K structure does not contain any calculated values such as probabilities, so that the K structure remains dynamic allowing recording, building and querying to occur simultaneously.

The relationships between the data elements at all levels are recorded during the building process. All possible indexes are incorporated into the K structure, so no other data structures such as cubes are required to query for results within a context. This allows ad hoc querying at any time.

This K structure with a set of K transversal routines forms the engine for a prototype data analysis system that is very efficient for organizing and querying large field/record data sets (Mazzagatti, 2007).

Development of this K structure continues with the focus on incorporating meta data information into the K structure containing the field/record data. Some of the meta data will become an integral part of the K structure and some of the meta data will be referenced and processed by routines external to the K structure.

**BASIC K STRUCTURE**

Given the dataset in Figure 1, a diagram of the resulting K data structure would look like Figure 2. All possible data elements or particles, in this case the alpha-numeric character set, are represented by the K nodes at the lowest level of the K structure.

The middle level of the K structure contains the sequences of K nodes referencing the K nodes of the lower level (characters) and representing the sequences of those lower level K nodes that are the field variables. The end K nodes of middle level will be used to create top level.

Finally the top level of the K structure contains the sequences of K nodes referencing the end K nodes of the middle level (the field variables) and representing the sequences of those middle level K nodes that are the records.

**Recording Event Sequences**

As the Peirce materials suggest each level of the K structure is constructed in the same way using the basic triad (Mazzagatti, 2006).

The basic triad shown in Figure 3, shows the recording of two sequential events by a K node at a level above the two events. This recording is accomplished by establishing a set of pointers that link a pair of K nodes, called a bi-directional pointer. Each K node in the pair contains a pointer that points to the other K node.

The bi-directional pointers allow the K structure to be traversed in any direction, even changing direction if required. For example, the solid line that connects the BOT K node with the next K node is created when a pointer in the BOT K node points to the second K node and a pointer in the second K node points to the BOT K node.