Multidimensional vector coordinates as a method of organisation of large social and economic databases

Chris Leowski

University of Toronto,
140 St. George St., Suite 707,
Toronto, Ontario, M5S 3G6, Canada
E-mail: chris@chass.utoronto.ca

Abstract: Economic and social databases that contain hundreds of millions of data points need to be organised in a way that allows filtering and viewing of data through various sets of dimensions to narrow search criteria and limit the volume of returned data. Vector arrays with individual vectors addressed through multidimensional vector coordinates is one such organisational method, at the core of the largest Canadian national database CANSIM, deployed at Statistics Canada and at the University of Toronto, and widely used for research and teaching in many North American universities.

Keywords: economic and social databases; database organisation; multidimensional vector coordinates; Statistics Canada; CANSIM.

Reference to this paper should be made as follows: Leowski, C. (2010) 'Multidimensional vector coordinates as a method of organisation of large social and economic databases', *Int. J. Information and Communication Technology*, Vol. 2, No. 4, pp.374–385.

Biographical notes: Chris Leowski graduated from the Main School of Planning and Statistics in Warsaw, Poland, where he completed his Doctoral studies in 1976 and worked as an Adjunct Professor of Economics till 1980. In early 1980s, he worked for four years as an Associate Professor, then Professor, of Economics and Statistics at the Graduate Centre for Administrative Systems, Regional Institute of Technology, Mexico, specialising in data modelling and computational data analysis. In 1984, he joined the University of Toronto, where he held posts of Director of the Centre for Computing in the Humanities and Social Sciences (CHASS), Director of IIT and Interim Assistant Dean for Instructional and Information Technologies. During his tenure at CHASS, he created a data centre for social sciences and humanities, including the flagship national Canadian database CANSIM – covering all social and economic aspects of Canadian life.

1 Introduction

Economic and social databases traditionally present information either as a set of time series, where data points are aligned along the time axis, or as a cross-section of data for a point in time or for a time period (e.g., number of hospital beds in each province in December 2006), or a combination of the two in the form of a matrix. With collections of social and economic data reaching hundreds of millions of data cells, the challenge was to create a system where navigation and data retrieval could be done by looking at

selected dimensions of a large body of data. Multidimensional data modelling, already widely used in the commercial sector, became the basis for the new approach. The difference was in scale and in heterogeneity of dimensions, when compared to standard 'star schemas' of a 'single sector' commercial database, e.g., retail data. A typical view of a large multidimensional national database is that of a set of cubes, each cube potentially having not only a different number of dimensions, but also different depths of its dimensions (e.g., various demographic data can be organised by age, sex, marital status, geographical area; while trade data reflect export and import, source and destination, unit of measure, value, quantity, etc.). This paper presents a method for organising a multidimensional database as a set of **«tables of vectors»** (sometimes called **«arrays of vectors»**), with individual vectors addressed through **«multidimensional vector coordinates»**.

2 Rigorous structure through metadata

A well-designed multidimensional database must have a rigorous internal structure that meets many criteria and conditions, the most important ones being:

- All data must be categorised and described – that is, data points must belong to a group, a set, a subset, a topic, etc. There must not be «loose» (uncategorised) data within the database. In other words, the metadata system must be all-encompassing.
- The metadata system must provide a comprehensive «view» of the database across multiple topics, groups, sets, subsets, etc., and across all dimensions within each of those categories.
- The search and retrieval system must allow the user to «drill down» within each group (e.g., from a topic to its sub-topics, then sub-sub-topics, etc.) and within each dimension (e.g., selecting only some members of a dimension), as well as to «roll-up» to higher categories within the same set of dimensions.
- All data must belong to logical storage categories that map into physical storage within the database management system. The mapping may be quite intricate, but it must exist and be unequivocal. Selection of a set of data points for a specific topic, sub-topic and a set of dimensions will map into a set of coordinates and through them to a specific vector of data points within an array. That logical mapping will be then further translated into physical mapping to, for example, oracle tables and to specific SQL statements that retrieve requested data.
- For dynamic databases, update and revision processes must fully reflect the structure of the database. Additions, deletions and revisions are always to a member of the structure, whether it is a new or existing topic, group, subset, or a new or existing dimension or dimension member within a particular grouping category, or new or existing data points, or new or existing descriptive elements within metadata, etc. The structure does not change, but what fills the structure can change constantly. This by no means implies that the structure imposes rigidity, it merely imposes logic and discipline. We can create a new «topic» and allocate various sub-topics, dimensions, members of dimensions, descriptive elements, data points, etc., to that topic – what is important, though, is that the concept of a «topic» forms part of the

structure of the database. In a similar fashion, concepts of a «dimension», a «member of a dimension», a «table», a «vector», a «data point», etc., form part of the structure of the database.

The structure of the database is shown here in XML format – this is the metadata view. The overall view of the structure will be followed by description of components and their hierarchies. We then focus on how to filter data through vector dimensions and how to locate subsets of multidimensional data through vector coordinates – the main objective of this exercise.

3 Concepts, components and attributes

Metadata concepts categorise and describe data. They also reflect the logical layout of data and, eventually, map into physical layout. Each major metadata concept may contain a number of hierarchically organised components. Each component may, in turn, have one or more attributes. The encapsulating block is, in XML notation, <DATABASE> </DATABASE>, and it simply denotes the entire database. In real world, the actual name of the database is substituted for the generic DATABASE concept.

The <TOPICS>s concepts consist of only one major component <THEME>, but through the PARENT attribute, it allows for a hierarchy of topics of arbitrary depth. Each individual <THEME>, no matter where it is located within the hierarchy of topics, contains the <TITLE> component with the LANGUAGE attribute. Each <THEME> is identified by its ID attribute, which can be referenced by any other component within the database. The PARENT attribute allows to locate each <THEME> within the hierarchy of topics and sub-topics. Through this simple schema, we are able to fully categorise and describe, in multiple languages, the entire body of data contained in the database, to the extent required for navigation and high level (topical) search functions.

```
<TOPICS>
. . . . . .
<THEME ID="920">
  <TITLE LANG="E">Agriculture</TITLE>
  <TITLE LANG="F">Agriculture</TITLE>
<THEME ID="1000" PARENT="920">
  <TITLE LANG="E">Agricultural products</TITLE>
  <TITLE LANG="F">Produits agricoles</TITLE>
</THEME>
<THEME ID="2024" PARENT="1000">
  <TITLE LANG="E">Crops</TITLE>
  <TITLE LANG="F">Récoltes</TITLE>
</THEME>
<THEME ID="3955">
  <TITLE LANG="E">Arts, culture and recreation</TITLE>
  <TITLE LANG="F">Arts, culture et loisirs</TITLE>
<THEME ID="3238" PARENT="3955">
  <TITLE LANG="E">Recreation</TITLE>
  <TITLE LANG="F">Loisirs</TITLE>
```

```
</THEME>
. . . . . .
</TOPICS>
```

In the example above, Agriculture is a THEME with theme ID = 920. The <TITLE> is a component of each THEME, with the language attribute. That attribute is present in many descriptive components throughout the database, offering a truly multilingual view of the entire structure (metadata) and data. Each THEME is identified by its ID and can have a PARENT attribute. THEME ID = 920 (Agriculture) does not have a parent – it belongs to the first level of the hierarchy of all topics. But the THEME ID = 1000 (Agricultural Products) has its PARENT attribute equal to 920 – Agriculture – which is its parent. Likewise, theme 2024 (Crops) lists theme 1000 (Agricultural Products) as its parent. We can easily envisage a multilayered hierarchy of topics and sub-topics, of arbitrary depth and level of detail. Actual vectors of data points or – at a higher level – collections of such vectors need only to be tagged with a theme id attribute in order to fully categorise data and to allow for a very efficient thematic search.

While the <TOPICS> concept categorises data, the <SOURCELIST> concept refers to a list of all possible sources of data contained in the database. Its main and non-hierarchical component <SOURCE> has an ID attribute, as well as a language attribute. Vectors of data points, and collections of such vectors, can reference multiple sources by listing more than one source ID as their attributes.

```
<SOURCELIST>
. . . . . .
 <SOURCE ID="2334">
  <TITLE LANG="E">Accounting Services Price Index </TITLE>
  <TITLE LANG="F">L'indice de prix des services de comptabilité </TITLE>
 </SOURCE>
 <SOURCE ID="3306">
  <TITLE LANG="E">Adult Correctional Services </TITLE>
  <TITLE LANG="F">Services correctionnels pour adultes </TITLE>
 </SOURCE>
 <SOURCE ID="3312">
  <TITLE LANG="E">Adult Criminal Court Survey</TITLE>
  <TITLE LANG="F">Enquête sur les tribunaux de juridiction criminelle pour
  adultes</TITLE>
 </SOURCE>
. . . . . .
</SOURCELIST>
```

Logical organisation of data is achieved through the concept of <TABLE>, which contains a number of table components and attributes. We might think of a <TABLE> as a body of data organised along certain logical criteria – any set of features that make it convenient and useful to group them together. Physically, a TABLE is a collection of VECTORs of data points. A more important feature, though, is that a <TABLE> contains data viewable through a fixed set of dimensions. Those dimensions are different for different tables, but within a given table, all data are viewed through the same set of dimensions Thus, if a table is a collection of demographic data, it may be based on such dimensions like: age, sex, geography, with each having a different «depth» (the sex

dimension will have only two members: male and female, while geography and age dimensions can have many more). A <TABLE> can reference multiple sources of data and it can also provide information on more than one <THEME> within the <TOPICS> concept. The main purpose of the latter is navigation and search functions, while the main purpose of <TABLE> is grouping of data points.

<TABLE>s have components and attributes through which they reference SOURCES and TOPICS, provide FOOTNOTES that explain more arcane aspects of information, describe UOM – units of measure, CLASS – classification system, KEYWORDs useful in user searches, FRQ – frequency of data points, ACC – access level to data (e.g., public, restricted, suppressed) and many more. The primary components, though critical to understanding a multidimensional database and how data are referenced and extracted through «vector coordinates», are <DIMENSION>s and <VECTOR>s.

<DIM> – a dimension – is a table component that offers a specific «view on data». We intuitively understand the time dimension (the classic organisation of data points into a time series), the geographical dimension – view on data by country, state, province, municipal area, etc., the age dimension, the two-member sex dimension, etc., but many data can be viewed from yet different angles. To understand how a dimension is built, let us examine more closely the «Geography» dimension:

```
<DIM ID="GEOGRAPHY" TYPE="0" FTNREF="">
  <TITLE LANG="E">Geography</TITLE>
  <TITLE LANG="F">Géographie</TITLE>

  <MEM ID="1" PARENT="0" LEFT="0" TERM="0" CLSREF="0" CLSCOD=""
  UOMREF="0" FTNREF="">
  <TITLE LANG="E">Canada</TITLE>
  <TITLE LANG="F">Canada</TITLE>
  </MEM>

  <MEM ID="2" PARENT="1" LEFT="0" TERM="0" CLSREF="0" CLSCOD=""
  UOMREF="0" FTNREF="">
  <TITLE LANG="E">Newfoundland and Labrador</TITLE>
  <TITLE LANG="F">Terre-Neuve-et-Labrador</TITLE>
  </MEM>

  <MEM ID="3" PARENT="1" LEFT="2" TERM="0" CLSREF="0" CLSCOD=""
  UOMREF="0" FTNREF="">
  <TITLE LANG="E">Prince Edward Island</TITLE>
  <TITLE LANG="F">Île-du-Prince-Édouard</TITLE>
  </MEM>

  <MEM ID="4" PARENT="1" LEFT="3" TERM="0" CLSREF="0" CLSCOD=""
  UOMREF="0" FTNREF="">
  <TITLE LANG="E">Nova Scotia</TITLE>
  <TITLE LANG="F">Nouvelle-Écosse</TITLE>
  </MEM>
. . . . . .
</DIM>
```

Dimension has a title, with multiple language attributes, and members whose 'location' within a dimension is fixed. Thus, in our example above, member <MEM ID = 1> – Canada – has PARENT = 0 (that is, no parent), and LEFT = 0, that is no siblings. The next member of the same dimension, with ID = 2 – Newfoundland and Labrador – has PARENT = 1 (that is, Canada), and no siblings to its left. The following member, with ID = 3 – Prince Edward Island – has the same parent (Canada), and a sibling to its left (LEFT = 2), which is Newfoundland and Labrador. This dimension consists of Canada (the parent) and its provinces listed East to West. Data viewed through this particular dimension will always have data points listed for the entire country and then by province from East to West. When retrieving data, users may select all or only specific members of the dimension. The PARENT attribute allows to 'roll-up' data (e.g., all provinces add up to the entire country), while the LEFT attribute shows 'horizontal' layout of members within each particular dimension. Let us look at one more – less obvious, although self-explanatory – dimension that has only three members:

```
<DIM ID="CLASSOFWORKER" TYPE="2" FTNREF="">
  <TITLE LANG="E">Class of worker</TITLE>
  <TITLE LANG="F">Catégorie de travailleurs</TITLE>

  <MEM ID="1" PARENT="0" LEFT="0" TERM="0" CLSREF="0" UOMREF="0"
  FTNREF="">
  <TITLE LANG="E">Total employed, all classes of workers</TITLE>
  <TITLE LANG="F">Emploi total, toutes les catégories de travailleurs</TITLE>
  </MEM>

  <MEM ID="2" PARENT="1" LEFT="0" TERM="0" CLSREF="0" UOMREF="0"
  FTNREF="">
  <TITLE LANG="E">Employees</TITLE>
  <TITLE LANG="F">Employés</TITLE>
  </MEM>

  <MEM ID="3" PARENT="1" LEFT="2" TERM="0" CLSREF="0" UOMREF="0"
  FTNREF="8">
  <TITLE LANG="E">Self-employed</TITLE>
  <TITLE LANG="F">Travailleurs indépendants</TITLE>
  </MEM>
</DIM>
```

4 Vectors and their 'multidimensional coordinates'

The <VECTOR> concept is the second, after <DIMENSION>, critical element in understanding how data are organised and retrieved in a multidimensional database. Vectors are components within a <TABLE> and thus share dimensions with that table, as well as certain attributes (for example, data frequency). We might think of vectors as collections (e.g., time series) of data points. Most vectors in economic and social databases are, in fact, time series, but there also exist vectors of data points with no time

dimension present – that is why we differentiate between vectors and time series, the latter being just one, though a very common one, representation of vectors. A TABLE in the CANSIM database is thus an ARRAY OF VECTORS, all of which share the same set of dimensions and other attributes. Shared dimensions and attributes are listed at the table level, while each vector may also have its own attributes, different from those in other vectors within the same table (e.g., start date and end date of its data points). A vector attribute called multidimensional coordinates determines what data points are included in the vector. These coordinates list members of each of the dimensions that the data cells that are present in the vector refer to. The example below shows three vectors that belong to a specific table that covers Labor Force Survey data. The table has four dimensions: Geography, Urban Rural, Class of Worker, and Industry. The first dimension has 11 members (Canada and provinces), the second dimension has eight members, the third one has three members, and the last one, listing various industry groups, has 18 members. Multidimensional coordinates = «1.1.1.1», which is one of the attributes of the first vector, mean: take data points for the first member of the first dimension, filter them through the first member of the second dimension, then through the first member of the third dimension, and finally through the first member of the fourth and last dimension of this table. What we end up with is a vector of data points that refer to total employed in all industries (member 1 of the 4th dimension), all classes of workers (member 1 of the 3rd dimension), covering total urban and rural areas (member 1 of the 2nd dimension, in Canada) (member 1 of the 1st dimension). Likewise, the second vector below (ID = 29756299) covers data for Newfoundland and Labrador (member 2 of the 1st dimension), Urban Fringe (member 4 of the 2nd dimension, Employees only) (member 2 of the 3rd dimension), in Other Services (member 18 of the 4th dimension).

<VEC ID="29755636" **COORD="1.1.1.1"** DEC="1" SCAL="3" AGG="0"
STRTDATE="20010101" ENDDATE="20060601" TERM="0" FRZ="0"> </VEC>

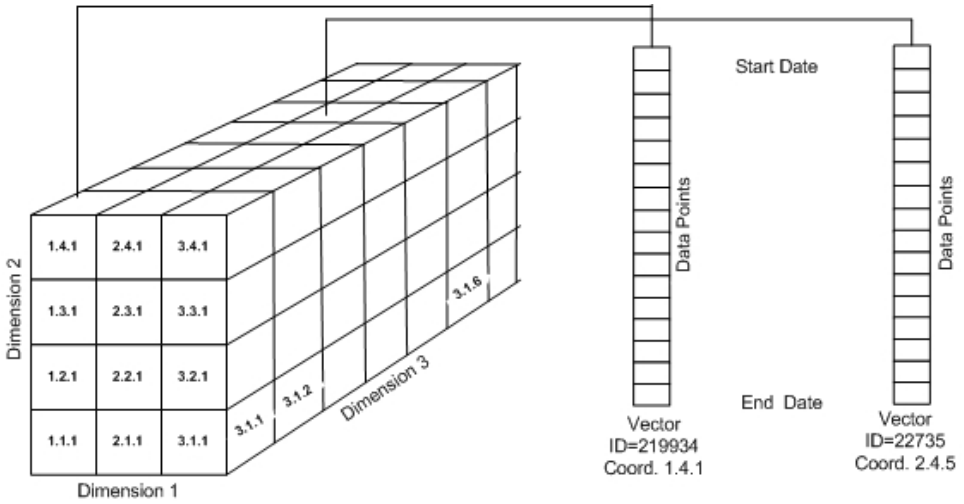
<VEC ID="29756299" **COORD="2.4.2.18"** DEC="1" SCAL="3" AGG="0"
STRTDATE="20010101" ENDDATE="20060601" TERM="0" FRZ="0"> </VEC>

In both cases, we will see data points from 01.01.2001 to 01.06.2006 as indicated by start date and end date attributes of each vector (frequency of data is an attribute at the table level). In other words, we have a four-dimensional, entity denoted by four coordinates for each vector and translating into four specific dimensions, from which «hangs» a vector of data points, which itself displays data along the fifth dimension (time).

A table of vectors sharing a set of dimensions can be illustrated as a multidimensional cube (Figure 1, for obvious reasons, restricts the cube to only three dimensions) and the entire database is a collection of such cubes – a multidimensional meta-space in which float multidimensional spheres comprised of arrays of vectors. Spheres may be of different shapes, reflecting their different number of dimensions and some of those spheres, though not many, will be cubes, where only three dimensions are present, as in Figure 1. In those simple three-dimensional entities, each table (cube) is combined of smaller ‘cubicles’, the number of which is the product of all its dimensions. Coordinates point at the combination of dimensions that each vector of data refers to, and the actual

vector of data cells «hangs» from its 'cubicle', with the number of data cells defined by its attributes.

Figure 1 Multidimensional (3 × 4 × 7) cube representing a single table (array of vectors)



A complete high level XML view of the structure only – without any contents – of the CANSIM database, but with its components and attributes, is provided in Appendix 1. For ease of reference, the main concepts and components are printed in black, while attributes are grey. In reality, as deployed at Statistics Canada and at the University of Toronto, the database comprises approximately 3,000 tables (arrays of vectors) of varying number of dimensions and over 41 million individual vectors. They reference various sources of information, listed in the SOURCELIST section, and refer to various THEMES from the TOPICS section.

The actual data points that fill each vector can be represented either as records of a flat CSV file or as an XML structure. Appendixes 2 and 3 offer both views of data cell records. Each data cell belongs to a table, then to an individual vector within that table that has a value and a number of attributes, including time reference if the vector is a time series. Once individual vectors are selected for retrieval – based on the user's navigation through topics, sub-topics, tables, dimensions and members of each dimension – data cell records can be easily retrieved through the index on the vector id column of such records.

Acknowledgements

The paper is submitted to the International Conference on Data Management (ICDM 2008) at IMT Ghaziabad, India – February 2008.

Appendix 1

High level XML view of the structure of the CANSIM database

```
<CANSIM>

<THEMES>
<THEME ID="n" PARENT="a">
<TITLE LANG="E" English_text</TITLE>
</THEME>
. . . . . .
<THEME ID="n+m" PARENT="b">
<TITLE LANG="E" English_text</TITLE>
</THEME>
</THEMES>

<SOURCELIST>
<SOURCE ID="n1">
<TITLE LANG="E" English description f the source</TITLE>
</SOURCE>
. . . . . .
<SOURCE ID="n1+m1">
<TITLE LANG="E" English description of the source</TITLE>
</SOURCE>
</SOURCELIST>

<TABLE ID="nnnnnnn" FRQ="ff" TERM="0|1" ACC="0|n"
CONTENT="YES|NO" LONGITUDINAL="" SOURCE="nnnnnn"
THEME="nn1 nn2 nn3" FTNREF="n"
<TITLE LANG="E" English title of the table</TITLE>
<FOOTNOTE ID="n"
<TEXT LANG="E" English text of the footnote</TEXT>
</FOOTNOTE>
. . . . . .
<FOOTNOTE ID="n+m"
<TEXT LANG="E" English text of the footnote</TEXT>
</FOOTNOTE>
<UOM ID="n" DEFAULT="m">
<TEXT LANG="E" Unit of Measure</TEXT>
</UOM>
<CLASS ID="n">
<TEXT LANG="E" Classification</TEXT>
</CLASS>
<KEYWORD ID="1">
<TEXT LANG="E" Keyword 1</TEXT>

</KEYWORD>
. . . . . .
<KEYWORD ID="n">
```

High level XML view of the structure of the CANSIM database (continued)

```
<TEXT LANG=”E” Keyword n</TEXT>
</KEYWORD>

<DIMENSION ID=”Dimension 1” TYPE=”0|1|2” FTNREF=””
<TITLE LANG=”E” English description of dimension 1</TITLE>
<MEMBER ID=”1” PARENT=”0” LEFT=”0” TERM=”0” CLSREF=”0”
UOMREF=”n” FTNREF=””
<TITLE LANG=”E” Dim member 1 </TITLE>
</MEMBER>
. . . . . .
<MEMBER ID=”n” PARENT=”m” LEFT=”i” TERM=”0” CLSREF=”0”
UOMREF=”n” FTNREF=”n”
<TITLE LANG=”E” Dim member n </TITLE>
</MEMBER>
</DIMENSION>
. . . . . .
<DIMENSION ID=”Dimension n” TYPE=”0|1|2” FTNREF=””
<TITLE LANG=”E” English description of dimension n</TITLE>
<MEMBER ID=”1” PARENT=”0” LEFT=”0” TERM=”0” CLSREF=”0”
UOMREF=”n” FTNREF=””
<TITLE LANG=”E” Dim member 1 </TITLE>
</MEMBER>
. . . . . .
<MEMBER ID=”n” PARENT=”m” LEFT=”i” TERM=”0” CLSREF=”0”
UOMREF=”n” FTNREF=”n”
<TITLE LANG=”E” Dim member n </TITLE>
</MEMBER>
</DIMENSION>
<VECTOR ID="first vec id" COORD="1.1.1.1" DEC="n" SCAL="n" AGG="0"
STRTDATE="start date" ENDDATE="end date" TERM="0" FRZ="0">
</VECTOR>
. . . . . .
<VECTOR ID="last vec id" COORD="x.y.z.w" DEC="n" SCAL="n" AGG="0"
STRTDATE="start date" ENDDATE="end date" TERM="0" FRZ="0">
</VECTOR>
</TABLE>
. . . . . .
<TABLE ID=”last table id” FRQ=”ff” TERM=”0|1” ACC=”0|n”
CONTENT=”YES|NO” LONGITUDINAL=”” SOURCE=”nnnnnn”
THEME=”nn1 nn2 nn3” FTNREF=”n”
<TITLE LANG=”E” English_text</TITLE>
… [same structure as for the previous table]
</TABLE>

</TABLES>

</CANSIM>
```

Appendix 2

XML view of data cells

<RECORD TYPE=”C”
ARRAY=”nnnnnn”
VECTOR=”mmmmmmm”
REFDATE=”YYYYMMDD”
REFDATE2=”YYYYMMDD”
SECURE=”public|secure|unreleased”
STATUS=”normal|naval|toosmall“
VALUE=”data_point_value”
SYMBOL=”none|p|f|r”
RELEASETIME=”YYYYMMDDHHMMSS”
</RECORD>

Explanations

- The TYPE of a cell record is always ‘C’.
- Data point belongs to a VECTOR within a CANSIM ARRAY (TABLE).
- Each data point references a point in time (REFDATE in YYYYMMDD format).
- REFDATE2 is used if and only if the data point references a time range, otherwise it is null.
- SECURE attribute denotes whether the data point in question is public, secure or not yet released.
- STATUS attribute can be either normal, or ‘not available’ or ‘too small to be expressed’.
- VALUE contains the value of the data point.
- SYMBOL indicates the symbol to use when displaying the data point (p = preliminary, f = forecast, r = revised).

Appendix 3

CSV view of data cells

The example below shows six data cell records in CSV format, two each for vectors V1552, V1553 and V1554 from the table (vector array) 303-0019. The actual values for each data point are in bold face and they refer to October and November of 2006 (data frequency is an attribute at the table level, not included at the cell level, but most likely, these data points refer to monthly values. All data points in this example have been released on 29 January 2007 at 8:30 AM.

"C","3030019","V1552","20061001","","public","normal","**192475**","none","200701290
83000"

"C","3030019","V1552","20061101","","public","normal","**187873**","none","200701290
83000"

"C","3030019","V1553","20061001","","public","normal","**7050**","none","20070129083
000"

"C","3030019","V1553","20061101","","public","normal","**5723**","none","20070129083
000"

"C","3030019","V1554","20061001","","public","normal","**633**","none","200701290830
00"

"C","3030019","V1554","20061101","","public","normal","**444**","none","200701290830
00"