Modeling data relationships with named sets

Mark Burgin[1] and H. Paul Zellweger[2]

[1] UCLA, Los Angeles, California, USA
[2] ArborWay Labs, Rochester, Minnesota, USA

Data are representations and containers (carriers) of information while data with their relationships form data structures. That is why modeling data relationships is important for organization and optimization of information processes.

Data have different structures and data processing in general and data mining in particular depends on these structures. The basic data structures include: Boolean values, characters, integers, fixed-precision number values, floating-point number values, arrays, records, lists, streams, sets, multisets, stacks, queues, and graphs to mention just the most important of them. Here in addition to these data structures, we consider named sets and chains of named sets as the fundamental data structures.

So, the question is: Why named sets are really essential and what is so specific about named sets?

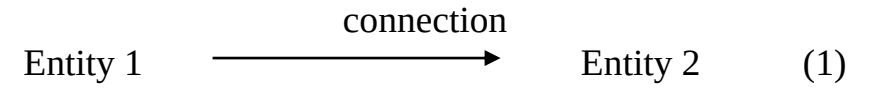
First, it is proved that any mathematical structure is a named set or is built of named sets (Burgin, 2011). For instance, functions, relations, fuzzy sets, multisets, variables, graphs, multigraphs, operators, fiber bundles, and morphisms (arrows) in categories are special cases of named sets. Ordinary sets are also specific named sets, namely, they are singlenamed sets because all elements in a set with the name, say Q, have the common name “an element of the set Q” (Burgin, 2011). In essence, named set is the most fundamental structure in mathematics as all mathematical structures are built of named sets. (Burgin, 2011).

Second, named sets have been extensively utilized in databases and knowledge bases. As relations are a special case of named sets, all relational databases store named sets and work with them (Date, 2004). Named set chains are key structures in temporal databases (Snodgrass and Jensen, 1999; Burgin, 2008). Named sets have been constructively utilized for data visualization and information retrieval in databases (Burgin and Zellweger, 2005; Zellweger, 2017) and database management (Ivanova, 2015).

Third, it is proved that named set (also called fundamental triad) is the most basic structure in nature (Burgin, 2011). As a consequence, named sets become ubiquitous in modeling natural systems.

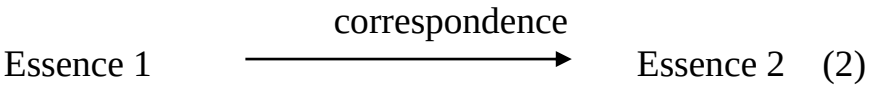
So, what is a named set?

A *basic named set* (also called a *fundamental triad*) has the following graphic representation



(1)

or



(2)

In the fundamental triad (named set) (1) or (2), Entity 1 (Essence 1) is called the *support*, the Entity 2 (Essence 2) is called the *reflector* (also called the *set* or *component of names*) and the connection (correspondence) between Entity 1 (Essence 1) and connection (correspondence) is called the *reflection* (also called the *naming correspondence*) of the fundamental triad (1) (respectively, (2)).

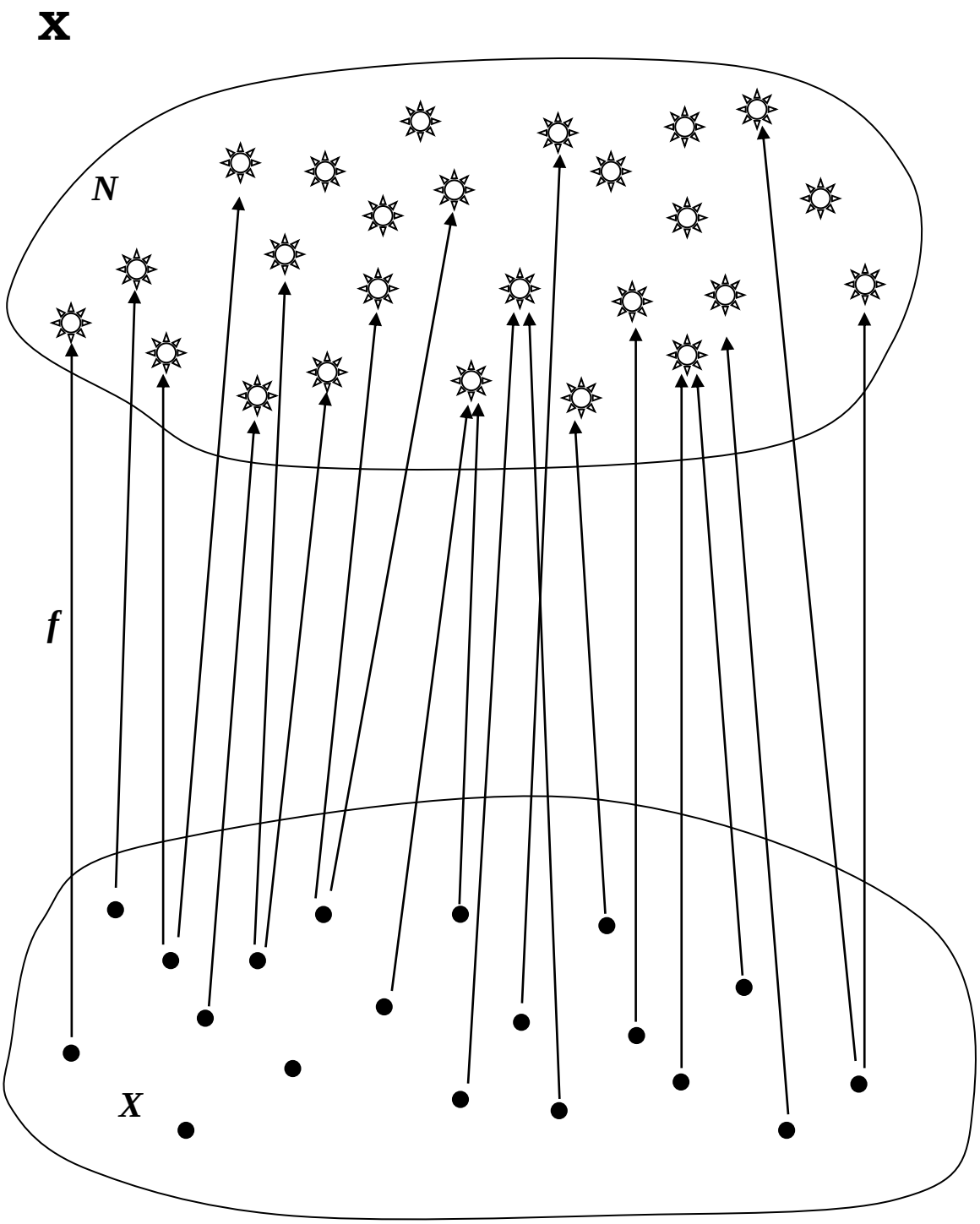
In the symbolic form, a *named set* (also called a *fundamental triad*) **X** is a triad (*X*, *f*, *I*) where *X* is the *support* of **X** and is denoted by S(**X**), *I* is the *component of names* (also called *set of names* or *reflector*) of **X** and is denoted by N(**X**), and *f* is the *naming correspondence* (also called *reflection*) of the named set **X** and is denoted by n(**X**). The most popular type of named sets is a named set **X** = (*X*, *f*, *I*) in which *X* and *I* are sets and *f* consists of connections between their elements. When these connections are set theoretical, i.e., each connection is represented by a pair (*x*, *a*) where *x* is an element from *X* and *a* is its name from *I*, we have a *set theoretical named set*, which is binary relation.

A *bidirectional named set* has the same structure but in it, the correspondence (e.g., a binary relation) *f* goes in two directions.

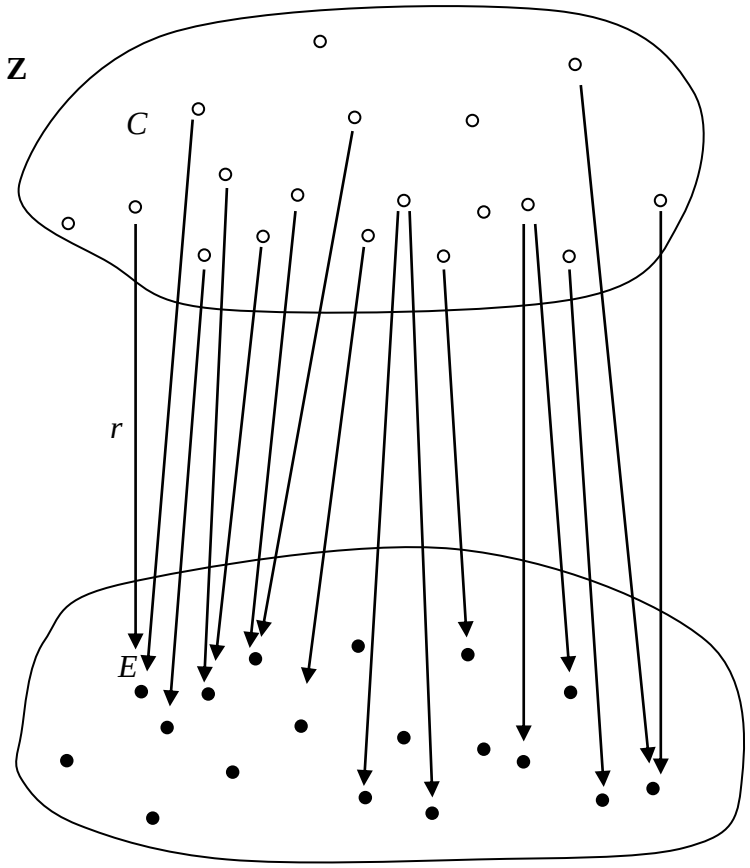
Two model examples of a named set:

X is a group of people, N is a set of their names and f is the connection between people and their names.

X is a collection of Internet resources, N is a set of their Internet names and f is the connection between resources and their names.



A set-theoretical named set $\mathbf{X} = (X, f, N)$



A set-theoretical named set $\mathbf{Z} = (C, r, E)$

An important peculiarity of utilization of named sets in databases is that algorithms in general and software systems, in particular, for operation with data are also specific named sets and systems of named sets. Namely, they are algorithmic named sets and their systems, i.e., such named sets in which the relation *f* is an algorithm or a program.

Manipulation with data demands various operations and in the case of using named sets for data representation, a variety of operations, such as mappings of different kinds, union, intersection, difference, renaming, naming, interpreting, and reinterpreting, is provided by the theory of named sets (Burgin, 2011). These operations have various applications in database management, data modeling, data organization, data mining, data transformation, and information search.

Applications of named sets in computer science, computer and network technology and programming

1. Naming systems and in particular, intentional naming systems

INS (Intentional Naming System) is a new naming system intended for naming and discovering a variety of resources in future networks of devices and services.

2. Naming schemas and in particular, intentional naming schemas
3. The Domain Name System (DNS)
4. Layered Naming Architecture for the Internet
5. Location-independent naming
6. Labeled database management
7. Naming and binding of objects
8. Named graphs
9. Named parameters
10. Content routers for high-speed name-based operations
11. Name-oriented networking or named data networking (NDN)

Burgin, M. (2008) Structural Organization of Temporal Databases. In: *Proceedings of the 17th International Conference on Software Engineering and Data Engineering* (SEDE-2008), ISCA, Los Angeles, California, pp. 68-73

Burgin, M. *Theory of Named Sets*, Mathematics Research Developments, Nova Science, New York, 2011

Burgin, M. and Zellweger, P. (2005) A Unified Approach to Data Representation. In: *Proceedings of the* 2005 *International Conference on Foundations of Computer Science*, CSREA Press, Las Vegas, pp. 3-9

Date, C. J. *An Introduction to Database Systems*, Addison Wesley, Boston/San Francisco/New York, 2004

Ivanova, K. (2015) Storing and processing accounting information based on collect/report paradigm, International Journal *Information Content and Processing*, **2**(2), 103-114

Snodgrass, R.T. and Jensen, C.S. *Developing Time-Oriented Database Applications in SQL*, Morgan Kaufmann, 1999

Zellweger, H.P. (2017) A Decision Tree Interface Based on Predicate Calculus, 21st *International Conference on Information Visualisation* (*IV* 2017), pp. 188-193