International Journal of Database Theory and Application
Vol.9, No.10 (2016), pp.47-62
http://dx.doi.org/10.14257/ijdta.2016.9.10.05

Toward a Philosophy of Data for Database Systems Design

Sabah Al-Fedaghi

Computer Engineering Department, Kuwait University
sabah.alfedaghi@ku.edu.kw

Abstract

Data provide the inputs to systems used to understand, explain, manage, regulate, and predict the world in which we live. A basic question in studying data is, What are data? What are and are not data, and how do data become information? Exploring a conception of data is fundamentally a philosophical problem and also an important issue in the area of database design. A firm understanding of the nature of the data being modeled enhances the process of modeling reality, and it helps in establishing a mental map of the computerized domain. This paper proposes a basic definition of data as interpreted things that flow. This definition is used in building structured data (e.g., tuples, tables) that form the foundation of database systems. The notion of things that flow is a concept based on a flow-based modeling language established on machines (extension of the input-process-output model) that create, process, release, transfer, and receive these things that flow. The study uses the proposed basic definition of data to build structured data, hence, applying the definition in constructing a data-based description of particular aspects of database systems.

***Keywords**: What are data? philosophy of data, sense data, database system, conceptual modeling, abstract machine*

1. Introduction

It is difficult to imagine our world without *data* [1] since data are now central to the ways business is conducted and governance enacted. Data provide the inputs to systems used to understand, explain, manage, regulate, and predict the world in which we live [2]. Data are generally understood to be such things as numbers, signals, characters, symbols, images, and bits that constitute the "stuff" for creating information, hence forming a basis for decisions and actions [3]. Data are "the fundamental part of information" [4].

Such an important concept still raises many questions about the nature of data. "Data are never simply just data; how data are conceived and used varies between those who capture, analyze and draw conclusions from them" [5]. Different conceptions of data can produce frustration, error, and miscommunication [6]. Exploring the concept of data is an important issue in the area of database system design. How is it possible to design a database without a firm understanding of the nature of the data being modeled? Such understanding of *what data are* enhances the process of modeling reality, and it helps establish an intuitive mental map of the computerized model.

A basic question in this context is, *What are data?* What qualifies as data, and how does it become information [6]? "Exploring a conception of data is fundamentally a philosophical problem…. this philosophical problem cannot be solved through intuition alone: a methodology is necessary to extract a person's conception of data" [6].

This paper offers a conceptual framework for establishing a philosophical base for data, and it proposes a methodology for constructing a *data-based* model as a starting basis for modeling networks of interconnected concepts at different levels (*e.g.*, information, knowledge). The paper proposes to conceptualize data as *things that flow* with certain characteristics (meaningless, truth-value neutral), and it adopts a new flow-

ISSN: 2005-4270 IJDTA

based diagrammatic language that depicts the movement of things in terms of creation, release, transfer, receipt, and processing. This definition is employed to build *structured data* and also *descriptions* of particular aspects of database systems. The results point to a potential *data* foundation for systems based on *data, information,* and *knowledge*.

The methodology adopted in this research is diagrammatic modeling of the concept of data as things that flow in a *machine*, a term used here to refer to a schema that extends the classical input-process-output model. In information engineering, "diagrams are the best way to communicate a methodology" ([7], referring to [8]). Diagrams are easier to understand and a semi-formalized way of specifying the system under development [7]. Diagrams can represent entities, activities, and causal relationships between these entities and activities [9]. Nevertheless, it seems that the entire approach adopted here could be stated in a mathematical language that formally defines the involved diagram, its types of things that flow, and the utilized structure. Such a project could be developed in the future, after the methodology and diagrams are explored and subjected to public scrutiny.

The paper will progress in the following way:

- Section 2 is a glimpse into the current status of the question, *What are data?* as it summarizes a recent exploration of this question in the literature.
- Section 3 presents a brief description of the diagrammatic modeling language to be used in conceptualizing data differently from the approaches described in Section 2.
- Section 4 provides examples of the diagrammatic modeling language reviewed in Section 3. It also sets these examples in a *philosophical context* in a discussion of *sense-data*; thus, the orientation of the paper includes a philosophical view of the nature of data.
- Section 5 applies the diagrammatic modeling language to some recent definitions of *data*. It also lays the groundwork for Section 6.
- Section 6 defines the concept of data and applies the definition to some relational database structures.
- Section 7 refines the definition of data given in Section 6.

2. Sample Approach to Defining Data

It is useful to consider a recent exploration of the question *What are data?* in order to illustrate the radical nature of our methodology in comparison with current state-of-art conceptualizations of the notion of data. Here I summarize Ballsun-Stanton's [6] Ph.D. dissertation titled, *Asking about Data: Exploring Different Realities of Data via the Social Data Flow* (2012) as a representative work that elaborates on the issue.

Ballsun-Stanton [6] begins with the following:

What is data? That question is the fundamental investigation of this dissertation. I have developed a methodology from social-scientific processes to explore how different people understand the concept of data, rather than to rely on my own philosophical intuitions or thought experiments about the "nature" of data.

Ballsun-Stanton [6] adopts the notion that *data* is a socially constructed term rather than a reflection of some property of the universe; thus, it is subjective relative to the person using the term. Such an approach is taken "because there is no clear consensus on the exact nature of data, much less on the exact nature of data in technical design." Three types of data have been identified:

- Data as communication: To be communicative, data require signs and things to communicate with those signs, *e.g.*, bits or marks on paper. Communicated data can be constructed in two ways: data as a menial-interpersonal communication, and data as a menial-technical communication.
- Data as subjective observations: Data as subjective observations require contextualization and filtering, and everything emits data as sense impressions that

can be captured by us. Data are inherently subjective and produced from observations made by people. Observation allows interpretations of data to be useful information. One example of context in which the nature of data is modulated is the understanding of a sensor's purpose, which is to change the observer's interpretation of the data produced by that sensor.

- Data as measured facts: Objective data require positive effort to generate, along with analysis to uncover the extant patterns of reality, so data must comprise facts, usually numerical and reproducible representations of reality that convey an understanding of measurement quality and units.

Ballsun-Stanton [6] used a Social Data Flow Network (SDFN), combining the idea of the social network with that of the data flow diagram (DFD), as a methodology to help map the "shared and unshared components of a group's social construction of reality as it relates to data flows":

The DFD contributes great ideas to the SDFN. It contributes the idea that data is something that can be modeled. The conception of data embodied by the DFD is that the modeler can translate reality into data-as-bits and that data could be described through text. All actions in the data flow diagram are considered either flows or transformation.

It can be noted that the approach in [6] is one of surrender, simply hovering over *data* since "there is no clear consensus on the exact nature of data." The approach in this paper is that this significant concept, which forms the foundation of such newly claimed paradigms as data science, big data, and data revolution, is worth further exploration.

3. Machines of Things that Flow

For the sake of a self-contained paper, this section briefly reviews the diagrammatic model that forms the foundation of the paper's development. The model has been adapted to several applications [10–14].

The so-called flow thing model (FM) is a diagrammatic language that uses *flow things* to represent a range of things, for example, information, signals, concepts, pieces of data, and so on. Flow things are the "stuff" that is being *created*, *released*, *transferred*, *processed*, and *received* in the abstract flow machine shown in Figure 1. Hereafter, flow things are referred to as *things*, and flow machines as *machines*.

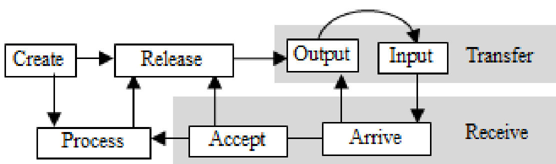


Figure 1. Flow Machine

FM language depicts processes formed by up to six stages (states) occurring along the stream of flow (Figure 1). The stages in Figure 1 can be described as follows:

Arrive: Data reach a new machine; *e.g.*, a new measurement (say, temperature) *arrives* at an alarm machine.

Accepted: Data are permitted to enter a machine; *e.g.*, timing data *arrive*; however, the alarm machine will reject any data outside an *accepted* range. If arriving things are always accepted, *Arrive* and *Accept* can be combined as a **Received** stage.

Processed (changed): Data are manipulated to change their form without creating new things, *e.g.*, timing data are converted from standard time to universal time.

Released: Data are marked as ready to be transferred outside the machine, *e.g.* sent emails are stacked, waiting for reconnection with the communication channel.

Transferred: Data exit (*e.g.*, through signal leave ports) and are transported somewhere from/to outside the machine.

Created: New data are born (created/appear) in a machine, *e.g*., a thermometer produces a new temperature value.

FM also uses the notions of *spheres and subspheres* to include machines and their flow connections that form the "region" of flow circulation. A sphere is where the flow "takes place." It may contain several machines besides *itself*, *e.g*., an employee machine has the *person-herself* machine, the *name-thing* machine, the *salary-thing* machine, *etc*. Flow indicates *change* in a stage or machine that situates the thing that flows. It is the transport of things within and across machines.

FM also utilizes the notion of *triggering*. Triggering is the activation of a flow, denoted in FM diagrams by a *dashed arrow*. The resultant FM diagram is a network of interlinked flows that together provide a complete representation of the system.

As will be suggested, data are *a type of flow thing* related by flows and structuring to assemble structured data. These flow things can be given conceptual labels that place interpretations on the data, thus transforming them as information.

Note that a machine (as well as the description of a machine) could be a thing that flows in another machine, as exemplified in the famous halting problem.

4. Examples: Sense-Data

This section has two purposes. First, it provides examples of FM, the diagrammatic language reviewed in the previous section. Second, it provides these examples in the philosophical context of *sense-data*; thus, it serves the aim of the paper of exploring the issue of *What are data?* Specifically, the material in this section focuses on works by the British philosopher Bertrand Russell as discussed by Savage [15].

Note that the material does not include philosophical speculation on the issue of *sense-data*; rather, the paper is a demonstration of the representational capability of FM to express certain characteristics of sense-data. Representation of some of Russell's philosophical concepts may be inaccurate because the author's expertise is computer science, not philosophy, but this is beside the point in this section as the involved diagram could be easily modified to reflect the correct concepts.

4.1. Source of Sense-Data

Traditionally, sense-data are the ultimate data in a standard foundationalist account of empirical knowledge: the completely certain, immediate, precise data of experience from which all other empirical truths are inferred. [15]

Discussing Russell's *My Philosophical Development* (1959), Savage [15] notes that Russell at first:

- **"**thought that *sensation* is a fundamentally relational occurrence in which a subject is 'aware' of an object,"
- "had used the concept *'awareness' or 'acquaintance'* to express the relation of subject to object," and employed the term "*sense-datum*" to denote the object of sensory acquaintance. [italics added]

Later, in *The Analysis of Mind* (1921), Russell explicitly abandoned "sense-data." As Savage explains, Russell had previously defined "sense-data" in *The Problems of Philosophy* (1912) in a famous passage:

Let us give the name of "sense-data" to the things that are immediately known in sensation: such things as colors, sounds, smells, hardnesses, roughnesses, and so on. We shall give the name "sensation" to the experience of being immediately aware of these things. . . . If we are to know anything about *the table*, it must be by means of the sense-data—brown color, oblong shape, smoothness, *etc*.—which we associate with the table. [15] [italics added]

In FM, the *table example* is drawn as shown in Figure 2. In the figure, the "properties" of table shape (circle 1), color (2), and smoothness (3) are (natural) *flow things* that flow through the table. These properties trigger (4–6) the creation of *sense-data* that are released and transferred to the sensation in a person's mind (7–9). The machine of the table itself (as a physical entity) (10 – includes only Create) is not enclosed by a box for simplicity. The convention of not enclosing in a box is followed also for Person's mind (11) and properties of the table.

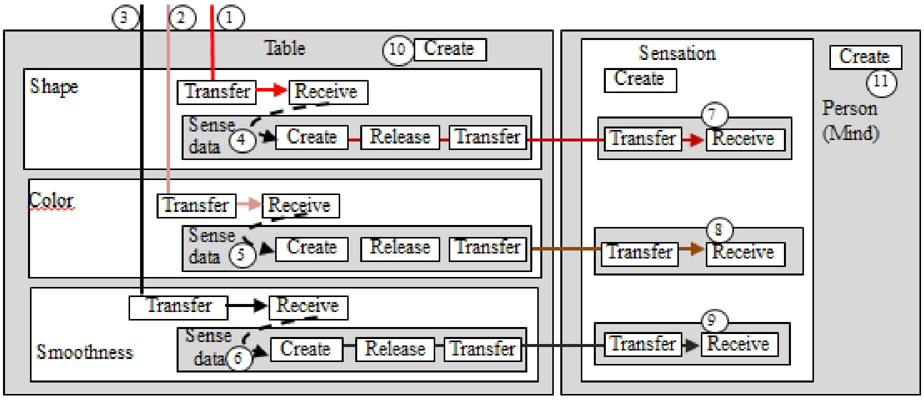


Figure 2. Sense-Data of a Table

In the figure, the Table machine contains three submachines of shape, color, and smoothness. For example, a color is a *flow thing* since it can be created, released, transferred, received (*e.g.*, color flows from the environment to a chameleon, or from sunsets to human eyes), and processed (diluted). Figure 2 assumes that the properties are different from the sense-data, *e.g.*, the shape machine has two submachines: (i) the "shape property" itself (has no enclosing box by convention), and (ii) the sense-data submachine. However, in a different philosophical conception, it could be assumed that the properties flow to the table and are received by sensation as *data*, as shown in Figure 3.

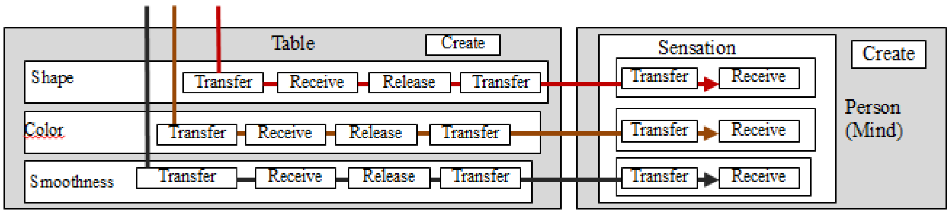


Figure 3. Another Conception of Sense-Data of a Table

It is clear now that the FM representation is a mere language that provides models for different interpretations of a domain (*e.g.*, reality). If Russell were alive, a non-philosopher such as a computer scientist could ask him to clarify the relationships between sense-data, properties of things, and the "source" of sense-data. It might be that philosophers have explored such issues; however, the diagrammatic representation immediately uncovers these differences. For computer scientists it is similar to the process of drawing a flowchart, which requires understanding to make decisions on variables, control flow, selections, and repeat structures. Of interest in Figures 2 and 3 is *figuring out* the position of *sense-data* in the scenario of sensing some natural object such as a table. This may lead to an understanding of the question, *What are data?*

4.2. Simple and Complex Data

According to Savage [15], Russell's list of examples of sense-data—*e.g.*, colors, smoothness—seems to suggest that sense-data are particular instances of simple properties. However, later, two examples are given: *That patch of red is round*, and *This is to the right of that*.

These are contrasted with a judgment that "simply asserts the existence of the sense-datum, without in any way analyzing it": that expressed by "There is such-and-such a patch of red," or "There is that." [15]

Figure 4 shows the FM representation of the judgment expressed by *That patch of red is round*. Note that in the context of representation, *Create* refers to the existence of the patch. The statement can be interpreted as awareness of the presence of the patch. In this case, we must consider the mentality of the speaker (person's mind) in the FM representation, as shown in Figure 4. In the figure, the red patch (1) is in a certain location (2), and its sense-data flow to the speaker, where data are processed (3 and 4) to trigger (5 and 6) awareness (7) of "that" patch. The thick vertical bar (8) indicates that awareness appears as the result of receiving both patch data and location data (The thick bar can be represented as a machine, but for simplicity this is avoided here).

Accordingly, the diagrammatic representation seems to raise awareness of such loaded terms as "that" in this example. Note that *location data* are flow things.

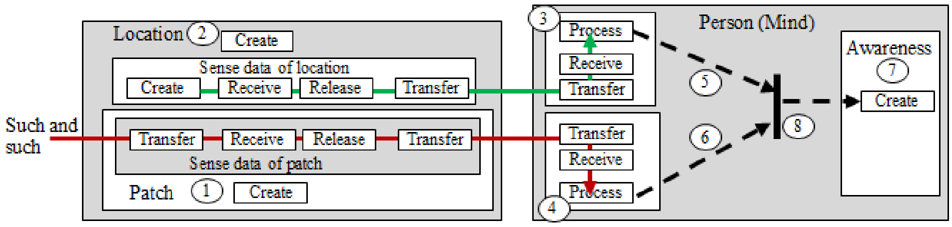


Figure 4. Flow of Awareness of that Patch of Red is Round

4.3. Judgment and Complex Data

For Russell in the context of discussion, *judgment* is a multiple relation of a subject or act to the several objects concerned in the judgment [15, Pre-abandonment Doctrine of Sense-Data]:

When we judge (say) "this is red," what occurs is a relation of three terms, the mind, and "this," and red. On the other hand, when we perceive "the redness of this," there is a relation of two terms, namely the mind and the complex object "the redness of this." (*Principia Mathematica*, vol. 1, as reported in [15])

Figure 5 shows the FM representation of *This is red*. The figure expresses the process of a *person pointing at and asserting (judging) that the thing is red*. Note that the act of pointing is a flow thing. The result is a map like portrait that embeds a basin of dynamic behavior involving the person's mind and a pointing act upon the *thing* with the red attribute. The concept *This is red* depicted as an FM diagram shows the scenario of steps invoked in the domain where *This is red* is spoken.

Figure 6 is a diagrammatic depiction of *The redness of this*. The pointing is directed toward the redness; since the redness is part of the thing, then it is directed at the thing.

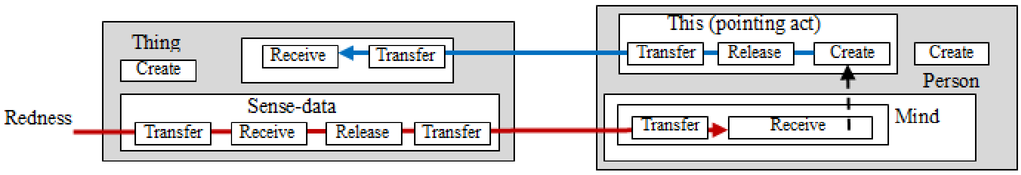


Figure 5. FM Representation of This is Red

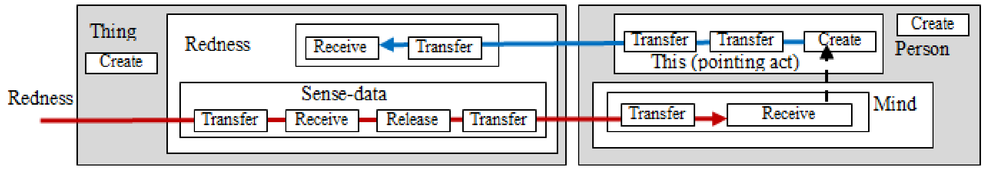


Figure 6. FM Representation of the Redness of this Thing

4.4. Perceiving Complex Objects

According to Savage [15], flowing illustrates the relational type involved in complex objects, as follows:

The complex object "a-in-the-relation-R-to-b" may be capable of being perceived; when perceived, it is perceived as one object. Attention may show that it is complex; we then judge that "a and b stand in the relation R." An example of a complex object of this type is the fact described by the sentence "This is to the left of that."

Figure 7 expresses the complex relational object *This is to the left of that*.

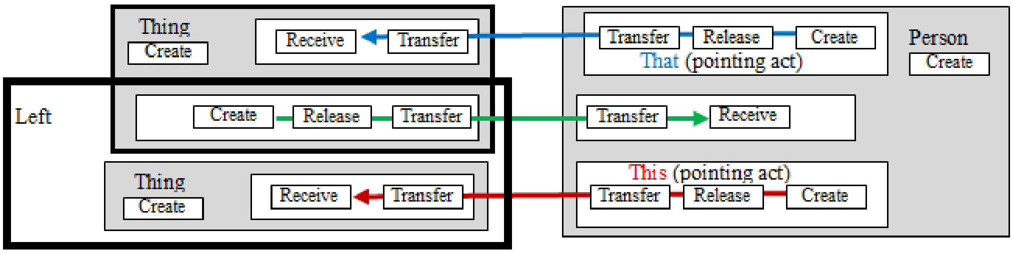


Figure 7. FM Representation of this is to the Left of that

4.5. Post-abandonment Doctrine

In *My Philosophical Development* (1959), Russell abandoned the distinction between the sensation and the sense-datum:

Accordingly, the sensation that we have when we see a patch of color simply is that patch of color, an actual constituent of the physical world, and part of what physics is concerned with. A patch of color certainly is not knowledge, and therefore we cannot say that pure sensation is cognitive. Through its psychological effects, it is the cause of cognitions, partly by being itself a sign of things that are correlated with it, as *e.g.*, sensations of sight and touch are correlated, and partly by giving rise to images and memories after the sensation is faded. [15]

According to Savage [15], Russell could be using the word "physical" here to refer to physical, physiological, or neurological processes:

The sensation, which is now held to be indistinguishable from the original sense-datum, is said to be the cause of the images and memories that are the psychological or cognitive ("mental") components of the causal process of perception… sensing … is replaced by the having of a perceptual image, and the sense-datum is replaced by the neural excitation that causes the image. [15]

Figure 8 shows the FM representation of sense-data as the cause of the images and memories. Sense-data flow to a person (circle 1 in the figure) to be processed physically (2), physiologically (3), and neurologically (4). Note how the FM representation forces the modeler to explicitly distinguish different flows (shown in different colors in the online version of the paper). Since it is not clear how these processes are performed, we assume that the physical, physiological, and neurological processes are executed in this sequence independently. The figure can be modified if the processes are performed in parallel or cyclic fashion. At the end, the neurological process triggers the creation of images (5).

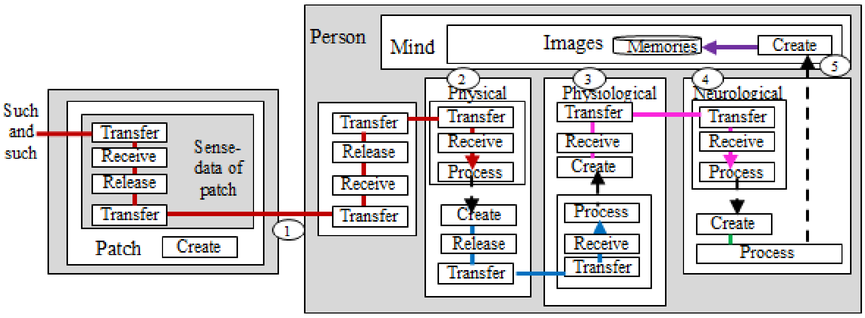


Figure 8. Sense-Data as the Cause of Images and Memories

With this demonstration of FM modeling, it is beneficial to clarify the notion of a machine. Figure 1 implies that the *passive* flow thing is an *object* that is being created, processed, received, released, and transferred. Furthermore, it implies that there is a subject, an agent or activator that is creating, processing, receiving, releasing, and transferring the thing. Furthermore, the *thing* may exist in relation to another thing.

Consider Figure 9, which expresses a scenario of a patient consulting with a physician who creates data about his or her health. In the figure, data are *created* by the physician *about* the patient's health. The machine is *patient's health*. Data are not a machine; rather they are mere flow things. However, for the sake of simplicity, the health box can be deleted. This simplification has been applied in many figures. For example, Figure 10 elaborates on details omitted from Figure 9.

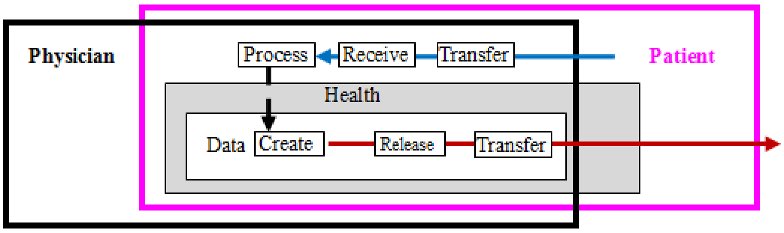


Figure 9. Machine vs. flow thing

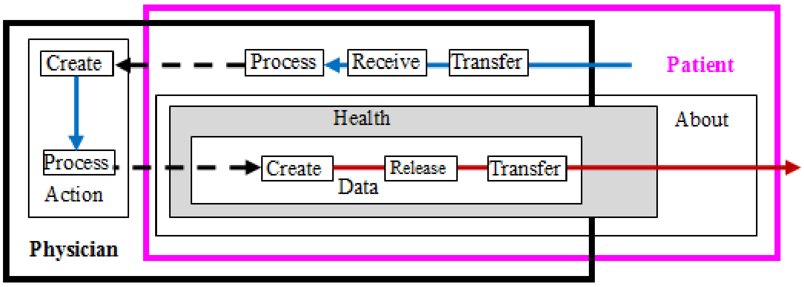


Figure 10. Detailing the Creation of Data

5. What are Data? Current Definition from the FM Perspective

This section considers some recent definitions of *data* and projects them onto the FM conceptualization of *things that flow*. The aim here is to find in these definitions the notions of creation, processing, release, transfer, and receiving, the five components of things that flow. An attempt is made in this section to link the concept of data with the phenomenological project in philosophy.

Etymologically, the word *data* "is derived from the Latin *dare*, meaning 'to give'" [16][17], indicating *what is given by phenomena*, as measured and recorded in various ways. Accordingly, a phenomenon encompasses:

- Object (Husserl's transcendental reduction)
- Being (Heidegger's existential reduction)
- "Given-ness" or event (see [18]–[20])

The phenomenon "shows itself first insofar as it is given, before possibly being qualified as a being or as an object" [19]. Butchart [20] explains, "In the correlation between appearing and that which appears, appearing is not considered as a *datum for the conscious* subject but as 'the givenness of what appears'" (quoting Marion's *Reduction and Givenness* [18], p. 32). In this case, a datum can be defined (by the author of this paper) as what flows from "the givenness of what appears," or *the taken-ness of the conscious subject*. Additionally, Butchart [20] states that "givenness precedes both intuition and intention [both can be called *Mind* in our rough computer science terminology] because the sense they make is only for and through an appearance"; hence, we can conclude that data are the *taken-ness* that follow the givenness and precede intuition.

Givenness of a phenomenon can be represented in FM as release and transfer (appearance) and taken-ness as transfer and receive, as shown in Figure 11. For simplicity, we will draw the concept of data created directly by the phenomenon.

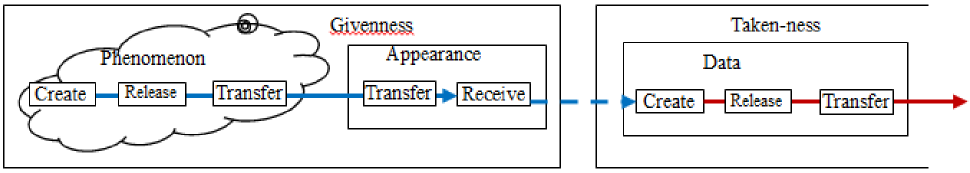


Figure 11. Data as what follows Appearance

The importance of data lies in their being *the* gateway to study any phenomenon; after all, "science is not about data; it is about phenomena" [21]. In the following, we relate words used in defining data to FM stages: Create, Process, Receive, Release, and Transfer.

- On one hand, from the meaning "to give" and from the FM view, *data* can be regarded as things that are *released and transferred* (given) by phenomena. On the other hand, according to Costantino [22], one could argue that *data* are often being *taken* rather than being given. Additionally, data are the material *extracted* through observations, computations, experiments, and record keeping ([23], as reported in [16]).
- From such viewpoints as well as use of FM, one can view data as *things that are transferred and received* ("taken") through observation, recording, *etc.*

Additionally, According to Costantino [22], "In general, data [are] being *processed* through algorithms."

- In FM language, data are things that are *processed* by algorithms.

Additionally, Costantino [22] states that "Every time someone interacts in and with a system little bits of data are being *created*."

- In FM language, *data* are things that can be *created*.

These descriptions of data can be put together to say, data are things that are created, processed, released, transferred, and received. That is, data are *a type of flow thing*.

Furthermore, additional descriptions of data have been given that can help in recognizing this *type* of flow thing. According to Costantino [22], "It can be arguable that data in its raw form has a neutral, almost apolitical character, but the way it is being tracked and used, is mostly not neutral, nor a political anymore." Kitchin [16] states that data may be understood as being pre-analytical: "There is no false data. Compared to a fact, data cannot be proven wrong. False data is always still data."

- In FM language, data are flow things that by themselves have no logical content, *i.e.*, they are neither true nor false.
- In FM language, data are flow things that by themselves have no meaning.

According to Costantino [22], "Through data every one of us has a twin made out of the data ... being collected on you. This information is being sold, processed and used in a way that transforms it into knowledge." A "data twin thing" can be generalized as data of phenomena, illustrating what is meant by "meaningless data" mentioned previously.

- In FM language, a *physical thing* ("every one of us") has a *data thing* that forms its "twin," made of data that are created, released, transferred, received, and processed, as illustrated in Figure 12.

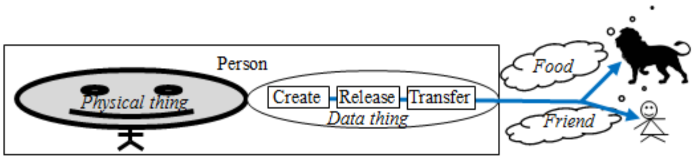


Figure 12. Data are Flowthings

In this context of defining data, the U.S. National Science Board (NSB) [24] defines data as,

Any information that can be stored in digital form, including text, numbers, images, video or movies, audio, software, algorithms, equations, animations, models, simulations, *etc.* Such data may be generated by various means including observation, computation, or experiment…. Data are inherently collective and come in sets—the collation of many individual data.

Text, numbers, images, video and movies, audio, software, algorithms, equations, animations, models, simulations, *etc.* are things that may be "generated" (created), but also processed, released, transferred, and received.

The definition can be oriented to FM-based language by restating it as, data are *things* such as interpreted text, numbers, images, video or movies, audio, *etc.*; however, "software, algorithms, equations, animations, models, and simulations" are *machines*. For example , an algorithm receives data, processes, then outputs results in the classical input-process-output model.

Additionally, the NSB definition raises the important point that machines themselves can be flow things through other machines as, for example, in the famous halting problem of Turing machines. However, such a thing is not data because it can be "activated" as a machine. Previously we defined data as a type of flow thing. Now, we see *a thing that is not data*. As Costantino [22] states, "It can be arguable that data in its raw form has a neutral, almost apolitical character." *Data have no machine characteristics, they do not create, process, receive, release, or transfer other things*.

A *non-data* thing can have machine components and be defined by them. Suppose a *Person* is described in terms of his or her physical body and health condition, as shown in Figure 13. Suppose that the person feels sick and takes a taxi to the hospital. As shown in the figure, in the taxi only the person's body is *of interest to be modeled*, but, implicitly, health condition flows with the body, then reappears at the hospital as an object of processing. Note that the data in the figure do not comprise a machine, as discussed previously.

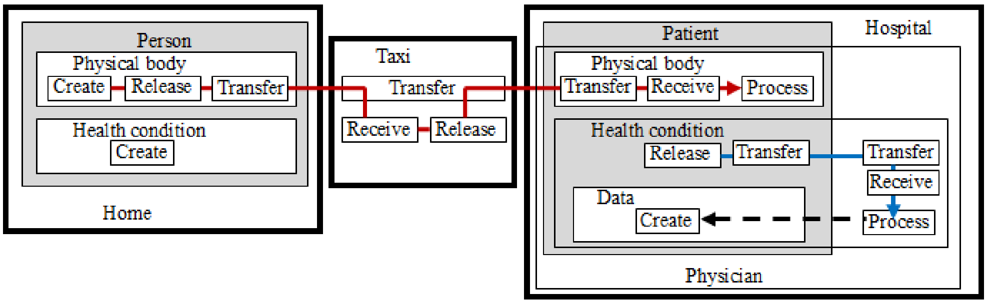


Figure 13. Person's Physical Body as a Flow Thing

Another interesting point in the NSB definition is that "Data are inherently collective and come in sets—the collation of many individual data." Note that in FM, a set is a machine, as will be made clear in the next section when we show how to construct a machine of structured data (tuples) from individual data sets.

6. Application to Some Relational Database Structures

Introductory books on the topic of database systems do not provide a definition of data [25]. A representation commonly associated with object-oriented programming, with labels corresponding to attribute names of objects, is as follows:

A typical computer science text will identify "label:value" or "label:(tupleVal1, tupleVal2, ..., tupleValN)" (or some other isomorphic structure) as a conceptual representation for a *datum*. The latter is heavily utilized within the Relational Model... While label:value and its variants are representations for data, *they are not, technically, definitions for data*. A particular label:value pair may simply be a value, possessing *no extrinsic meaning and reflecting neither fact nor information about a world*. As values, these representations are commonly used with records and tagged unions. [25] [Italics added]

Accordingly, our next interest is in the relationship between value1 and value2 (in our terminology, Data 1 and Data 2) in label_r:(label1:value1, label2:value2). This would be a first step in constructing Relations (tables) from two "basic pieces of data from two different sets," where a more general structure (more than two pieces of data) can be constructed in similar fashion. In database terminology, the next step in the FM methodology is to construct the relation $R(a_1, a_2)$ from the attributes a_1 and a_1. Because this paper focuses on the mere notion of *data* and not on a database that embeds semantic notions (*e.g.*, name of relation, labels of attributes), we will conceptualize a relation as a table constructed from tuples and data items that have indices indicating the columns where data belong.

Note that after defining *data* as a special type of flow thing, we move to extend the definition to structured data and sets of data, as shown in Figure 14, with a data-oriented simple database model of normalized relations.

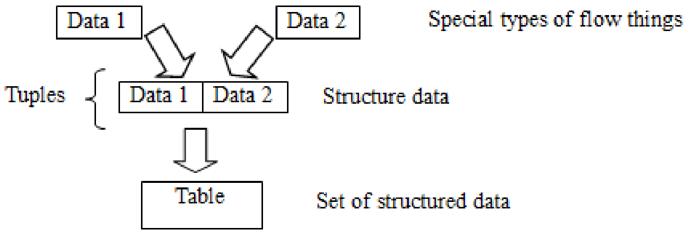


Figure 14. Moving to Structured Data as Flow Things

Consider a *basic* data item. Figure 15(a) shows that data are captured by some measurement of a certain phenomenon. Figure 15(b) applies such a description to Heat and the data are converted to numeric form; however, for simplicity the figure can be reduced to Figure 15(c). In Figure 15(d), processing the data triggers the creation of information by assigning the meaning *temperature* to the data. Figure 15(f) is a simplification of Figure 15(d).

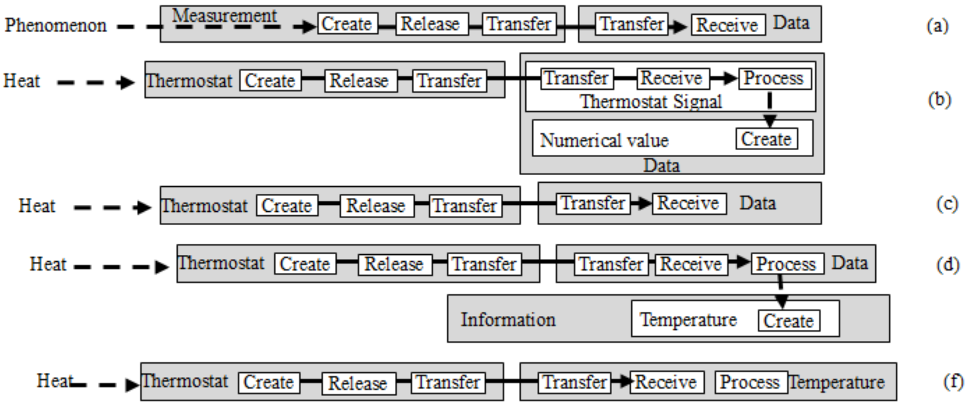


Figure 15. Basic Description of Data Captured from a Phenomenon

Accordingly, data are defined as *uninterpreted flow things*, with interpretation referring to such notions as meaning and truth-values. Note that data in this case are not necessarily raw data (*e.g.*, from a natural source). One agent's information might be another agent's data, because information is *transferred* in the form of data, and *arriving* data become interpreted, thus triggering the *creation* of information. If a sending agent sends the information "salary 224" and another agent interrupts its flow in the channel of communication, the interrupting agent will find *data* (string of bits) but will receive *information* only if it has an interpretation key. This conception is the key idea in Shannon's communication model: *information → data sent by source → data in channel → data received by destination → information*, simplified as *information → channel → information*. Accordingly, data as *flow things* can be produced by *de-interpreting* the *flow-things' information*, and information can be produced by interpreting data. In this case, the flows of data form the underlying communication system in the entire FM diagram. In order to communicate information you have to *de-interpret* it as data.

Data can be structured as shown in Figure 16, where a two-tuple is constructed from two data items. The tuple involves the construction and the resultant tuple itself. Note the similarity of this "structuring" to declarations in object-oriented language.

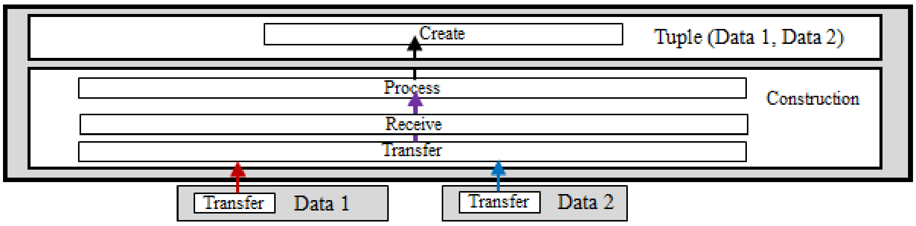


Figure 16. Structured Data

Figure 17 shows the construction of a relation (table) by the insertion of tuples into the table. Figure 18 is a representation of the selection operation.

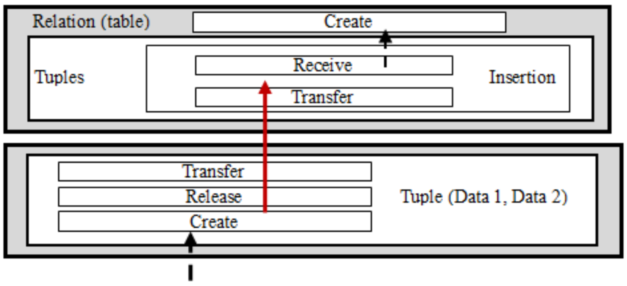


Figure 17. Table Constructed by Repeatedly Creating Tuples

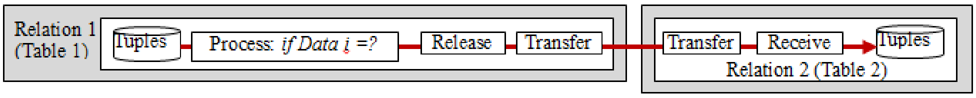


Figure 18. Selection

In FM, a table is an infrastructure of machines. Figure 19 illustrates this concept for a table with three tuples and two columns. It includes three types of machines:

- Machines of individual items: 1, 1 through 3, 2 (indexed as in a matrix), as shown in the figure
- Tuple machines that receive things from machines along the rows of tuples
- Column machines that receive things from machines under the columns

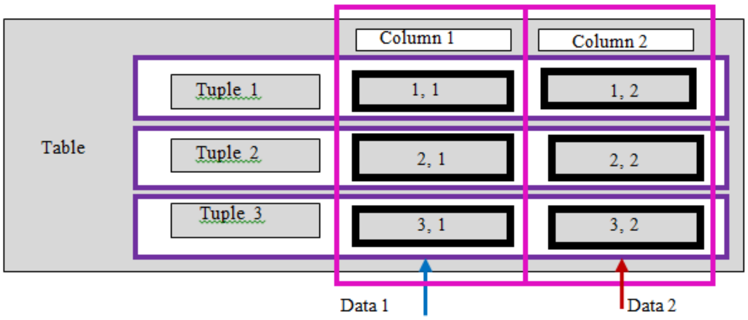


Figure 19. Illustration of Machines in a 3 by 2 Table

Figure 20 shows the structured data form of a table with two columns. Because of the data-oriented focus of study, an index 1 or 2 is associated with these data items instead of the previously mentioned "labels" (*e.g.*, temperature, time, *etc.*). The figure is a data-based definition of a two-tuple table (circle 1) that can be generalized. Limiting the number of columns to two, Figure 20 shows the general structure of a table (circle 1) for multiple tuples. The two machines (i, 1) and (i, 2) (circles 2 and 3) of tuple i (4 – purple color in the online version) receive Data 1 and Data 2 directly. The tuple machine receives these values and creates the tuple itself by the submachine Construction (5). The submachine indicated by circle 6 denotes a mechanism to facilitate the indexing of different tuples in the table. We have not added machines that produce the whole table from the set of tuples because they were illustrated previously.

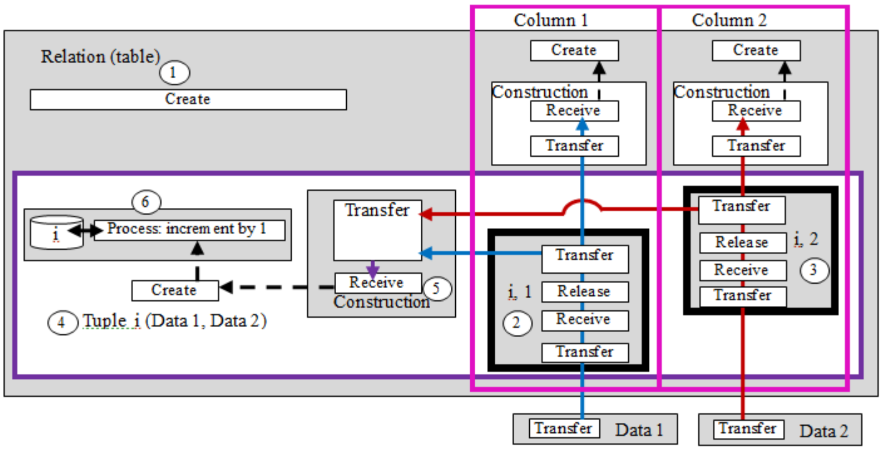


Figure 20. Sample Definition of a Table with Two Columns

The conceptual picture of the table contains machines that handle different data within it: individual values, tuples, columns, and the table, in terms of processing, creating, releasing, transferring, and receiving data. The whole structure can be generalized to more than two columns. Figure 21 shows the flow of data to form the projection of column 2.

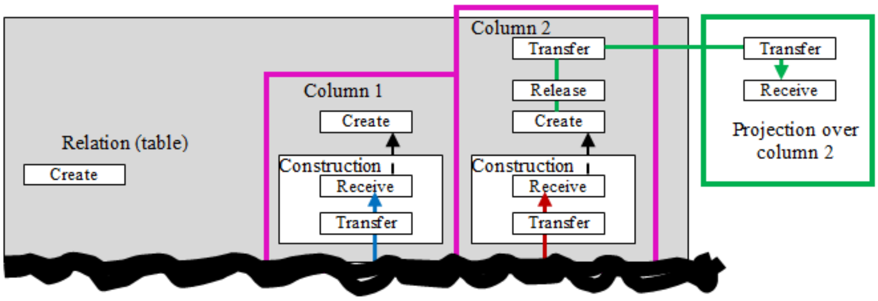


Figure 21. Projection over Column 2

8. Conclusion

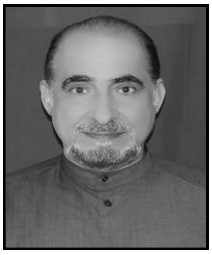
This paper has introduced a philosophical basis of data by grounding the definition of data on the notion of uninterpreted flow things. This conceptualization is used to build structured data (*e.g.*, tuples, tables) that form the foundation of a database system. The development of such a concept is based on a modeling language built on abstract machines that create, process, release, transfer, and receive things. The results of definition of data is used to build structured data, hence, applying it in constructing a description of some aspects of database systems.

The approach is motivated by the claim that there is “no clear consensus on the exact nature of data, much less on the exact nature of data in technical design” [6]. The premise in this paper is that this significant concept, which forms the foundation of such newly claimed paradigms as data science, big data, and data revolution, is worth further exploration. Though the paper does not claim to offer a complete solution to this problem, the proposed conceptualization of data seems promising for establishing a philosophical basis of data and introducing a methodology for constructing a data-based model.

References

[1] S. D. Ben, “Data”, http://www.med317.co.uk/archives/tag/kitchin, **(2015)**.
[2] R. Kitchin and T. P. Lauriault, “Towards Critical Data Studies: Charting and Unpacking Data Assemblages and Their Work”, paper presented at The Programmable City, Working Paper 2, http://www.maynoothuniversity.ie/progcity/, **(2014)**.
[3] R. McLean, “Data”, http://www.med317.co.uk/archives/719, **(2015)**.
[4] K. Unsworth, “Information Policy: Global Issues and Opportunities for Engagement”, Bulletin of the Association for Information Science and Technology, www.asis.org/Bulletin/Jun-14/JunJul14_Unsworth.html, vol. 40, no. 5, **(2014)**, pp. 46-49.
[5] R. Little, “Data: ‘I know your name!’ (The Goonies, 1985)”, http://www.med317.co.uk/archives/736, **(2015)**.
[6] B. B. Stanton, “Asking About Data: Exploring Different Realities of Data via the Social Data Flow Network”, Ph.D. Thesis, Methodology, The University of New South Wales, **(2012)**.
[7] W. Brits, “The Relationship between Organizational and National Culture and the Use and Effectiveness of Systems Development Methodologies”, MS Dissertation, **(2011)**.
[8] D. E. Avison and G. Fitzgerald, “Information Systems Development: Methodologies, Techniques & Tools”, McGraw-Hill, **(2006)**.
[9] C. E. Stinson, “Cognitive Mechanisms and Computational Models: Explanation in Cognitive Neuroscience”, Ph. D. Thesis, School of Arts and Sciences, University of Pittsburgh, **(2013)**.
[10] S. Al-Fedaghi and M. Almutairy, “Applying Thing-Oriented Modeling and Patterns”, International Journal of Software Engineering and Its Applications, vol. 10, no. 4, **(2016)**, pp. 143-160.
[11] S. Al-Fedaghi, “Drafting Blueprints for Car System Requirements”, International Journal of Hybrid Information Technology, vol. 7, no. 2, **(2014)**, pp. 57-70.
[12] S. Al-Fedaghi, “System for a Passenger-Friendly Airport: An Alternative Approach to High-Level Requirements Specification”, International Journal of Control and Automation, vol. 7, no. 2, **(2014)**.
[13] S. Al-Fedaghi, “Flow-based Enterprise Process Modeling”, International Journal of Database Theory and Application, vol. 6, no. 3, **(2013)**, pp. 59-70.
[14] S. Al-Fedaghi and Fajer Al-Kanderi, “Integrating Security Concerns into Software Development”, International Journal of Security and Its Applications, vol. 7, no. 3, **(2013)**.
[15] C. Wade Savage, “Sense Data in Russell’s Theories of Knowledge”, in C. Wade Savage and C. Anthony Anderson, eds, Re-Reading Russell: Bertrand Russell’s Metaphysics and Epistemology: Minnesota Studies in the Philosophy of Science, vol. 12. University of Minnesota Press, **(1989)**.
[16] R. Kitchin, “The Data Revolution: Big Data, Open Data, Data Infrastructures & Their Consequences”, SAGE Publications Ltd, ISBN 978-1-4462-8747-7, **(2013)**.
[17] D. Rosenberg, “Data before the Fact”, paper presented at 126th Annual Meeting of the American Historical Association, **(2012)**.
[18] J.-L. Marion, “Reduction and Givenness: Investigations of Husserl, Heidegger, and Phenomenology”, T. A. Carlson (trans.). Northeastern University Press, Evanston, IL, **(1998)**.
[19] J.-L. Marion, “Being Given: Toward a Phenomenology of Givenness”, J. L. Koskey (trans.). Stanford University Press, Stanford, CA, **(2002)**.
[20] G. C. Butchart, “An Excess of Signification: Or, What Is an Event?”, Semiotica, vol. 187, no. 1/4, **(2011)**, pp. 291–307.
[21] J. Bogen and J. Woodward, “Saving the Phenomena”, Philosophical Review, vol. 97, **(1988)**, pp. 303–352.
[22] M. Costantino, “Data”, http://www.med317.co.uk/archives/tag/kitchin, **(2015)**.
[23] C. L. Borgman, “Scholarship in the Digital Age”, MIT Press, Cambridge, MA, **(2007)**.
[24] U.S. National Science Board, “Long-lived Digital Data Collections: Enabling Research and Education in the 21st Century”, Report of the National Science Board. National Science Foundation, Arlington, VA, **(2005)**.
[25] Cunningham & Cunningham, Inc., “What Is Data”, online. http://c2.com/cgi/wiki?WhatIsData, **(2016)**.

Author



Sabah Al-Fedaghi, holds an MS and a PhD in computer science from the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, Illinois, and a BS in Engineering Science from Arizona State University, Tempe. He has published two books and more than 250 papers in journals and conferences on software engineering, database systems, information systems, computer/ information privacy, security and assurance, information warfare, and conceptual modeling. He is an associate professor in the Computer Engineering Department, Kuwait University. He previously worked as a programmer at the Kuwait Oil Company, where he also headed the Electrical and Computer Engineering Department (1991–1994) and the Computer Engineering Department (2000–2007).

