

**A Discussion of Data Quality  
for Verification, Validation, and Certification  
(VV&C)  
of Data to be Used in Modeling**

Jeff Rothenberg  
RAND  
1700 Main Street  
Santa Monica, CA 90407



## **PREFACE**

This memorandum discusses data quality issues that have become apparent in performing research conducted for the Defense Modeling and Simulation Office (DMSO) within the Acquisition and Technology Policy Center of RAND's National Defense Research Institute (NDRI), a federally funded research and development center sponsored by the Office of the Secretary of Defense, the Joint Staff, and the defense agencies. However, this paper is *not* an official product of that research effort: it represents the views of the author and not of the DMSO sponsor.

The paper should be of interest to data producers and modeling and simulation data users, as well as all those who rely on such data in performing studies and analyses.



**CONTENTS**

PREFACE ..... iii

FIGURES ..... vii

SUMMARY ..... ix

1. INTRODUCTION ..... 1

2. BACKGROUND AND DEFINITIONS ..... 3

    2.1. Previous definitions of data quality ..... 3

    2.2. Defining some data-related terms ..... 5

3. ASPECTS OF DATA QUALITY ..... 7

    3.1. Data as modeling reality ..... 8

    3.2. Data as residing in databases ..... 10

    3.3. Data quality as suitability for an intended purpose..... 11

    3.4. How can we promote data quality? ..... 12

    3.5. Metadata ..... 13

    3.6. Generating metadata ..... 15

    3.7. The Cost of Metadata ..... 16

    3.8. Explicit VV&C ..... 17

    3.9. Processes that affect data ..... 18

    3.10. Improving processes that affect data ..... 19

    3.11. Data quality is relative to users and their purposes ..... 20

4. PRODUCER VS. CONSUMER (USER) VV&C ..... 21

    4.1. Objective vs. subjective validity ..... 22

    4.2. Phases of V&V and organizational commitment ..... 23

5. A FRAMEWORK FOR IMPROVING DATA QUALITY ..... 24

6. METADATA TO SUPPORT DATA QUALITY ..... 25

    6.1. Database level metadata ..... 28

    6.2. Data-element level metadata (data dictionary) ..... 29

    6.3. Data-value level metadata ..... 30

    6.4. Discussion of metadata categories ..... 30

        6.4.1. Database Level Metadata ..... 31

        6.4.2. Data-element Level Metadata (Data Dictionary) ..... 39

        6.4.3. Data-value Level Metadata ..... 46

    6.5. Tools for creating metadata ..... 52

    6.6. Mitigating storage and transmission requirements for metadata ..... 53

6.7. Allowing the evolution of the metadata structure .....	54
7. THE DATA QUALITY PROFILE .....	54
7.1. Selected metadata categories for a data quality profile .....	54
Database level metadata for quality profile .....	55
Data-element level metadata (data dictionary) for quality profile .....	55
Data-value level metadata for quality profile .....	55
7.2. The data quality profile view .....	55
7.3. Refining and using the quality profile .....	57
8. DATA VV&C .....	57
8.1. Verification techniques .....	58
8.2. Objective validation techniques .....	59
8.3. Subjective validation techniques .....	60
8.4. Certification techniques .....	61
8.5. Prioritizing VV&C .....	61
9. CONTROLLING AND IMPROVING PROCESSES AFFECTING DATA .....	62
9.1. Identify relevant processes throughout the life-cycle of the data .....	63
9.2. Identify “owners” of data and processes .....	63
9.3. Empower/facilitate/support process-control, redesign, and improvement .....	63
9.4. Implement process management .....	64
10. STEPS TOWARD DATA QUALITY .....	66
11. CONCLUSION .....	68
11.1. The cost of (not) improving data quality .....	68
REFERENCES .....	70

## FIGURES

Figure 1: Data as model .....	8
Figure 2: Alternate data “views” of reality .....	9
Figure 3: A framework for improving data quality .....	24



## SUMMARY

Most modeling activities—whether conducted for analytic, predictive, or training purposes—are at least partially data-driven. Their quality therefore depends critically on the quality of their data. Yet it is very difficult to ensure and assess the quality of the data used in modeling: this calls into question the quality of many modeling efforts, while impeding the reuse of data.

This memorandum suggests a strategy for improving data quality by means of two equally important, parallel approaches: (1) performing explicit evaluation of data and (2) establishing organizational control over the processes that generate and modify data. These approaches require: (i) augmenting databases with metadata in order to record information needed to assess the quality of their data, record the results of these assessments, and support process control of processes affecting data; (ii) encouraging producers and consumers (users) of data to implement organizational commitments to perform distinct phases of explicit verification, validation, and certification (VV&C) on their data, using metadata both to direct these activities and to record their results; and (iii) establishing control over the processes that affect data, to improve the quality of data generation, transformation, and transmission, again using metadata both to support this activity and to record its results.

It should be possible to develop automated tools to help capture and maintain metadata whenever generating or modifying data, thereby greatly facilitating this strategy. While the cost of developing and enforcing data quality procedures and tools may be substantial, the cost of *failing* to implement such procedures is likely to be even greater, since it undermines the value of much of the modeling and simulation that is currently performed. Furthermore, procedures and tools can be designed to be tailored to the needs and resources of each user or application, thereby allowing the degree (and therefore cost) of quality assurance to be matched to the perceived risk in each case.

The paper takes a broad, theoretical look at data quality, discusses the metadata required to support data VV&C and data process improvement, and introduces the concept of a “data quality profile” to help users evaluate data for their purposes.



## 1. INTRODUCTION

Although some models may be based on “first principles” that allow them to be relatively independent of data, the vast majority of non-trivial simulation models depend critically on data, whether treated as input or “hard-wired” into the model. While the importance of verifying and validating models has become increasingly recognized, relatively little effort has so far been focused on developing techniques for assessing and improving the quality of their data.<sup>1</sup> Since it is well-understood that the validity of a data-driven model depends on the validity of its data, the burden of verifying and validating a model is often shifted from the model itself to its data. Whether models are used for analytic, predictive, or educational purposes (including training and mission rehearsal), it is crucial that their data be subjected to the same kind of scrutiny that the models themselves undergo. Conscientious modelers may spend considerable effort attempting to do this, but they have so far been hampered by a lack of systematic methods and procedures (not to mention automated tools) for evaluating and ensuring data quality. Such procedures and tools should be designed to be tailored to the needs and resources of each user or application, thereby allowing the degree of effort devoted to quality assurance to be matched to the perceived risk in each case, in order to enhance cost-effectiveness.

Data quality affects the outcome of a model in profound and sometimes devious ways. The well-known adage “garbage in, garbage out” encapsulates this truth but does not do it justice. Nonsensical data may produce nonsensical results, but this describes only the most fortunate case. Garbage data may produce incorrect results that are not obviously garbage: the effects of poor data are often insidiously difficult to detect. Furthermore, as with a leak in a roof (or a bug in a program) the effects of bad data may become apparent only far from the source: these effects may even propagate through several models (for example, when models are used to aggregate results for input to other models), making the source of the problem very difficult to identify. In addition, models often depend discontinuously on their data or are highly sensitive to variations in data values, so that even minor data errors can produce arbitrarily serious problems.

Therefore, if the results of a modeling effort are to be believed, the data used in producing these results must be made at least as credible as the model itself.<sup>2</sup> In many application areas, including much of military modeling and simulation (M&S), achieving such credibility for data requires considerable effort, which is not currently supported in a systematic way. In short, the utility of any modeling application is limited by the quality of its data, no less than by the quality of the model itself—and the quality of much of the data used in many such applications is unknown or undocumented and therefore highly questionable.

The lack of explicit data quality information also impedes the reuse of data in M&S studies that are similar—or even nearly identical—to previous studies. Reuse can greatly reduce the cost of performing a study if it is similar to a prior study, as well as helping to ensure that the results of the

---

<sup>1</sup> The literature tends to focus on the quality of database *design* rather than on the quality of the data in databases [3,4]. Discussions of data quality *per se* tend to focus on specific examples rather than generic methods [1,6,7,8,10]

<sup>2</sup> The term “credible” is used here to mean *justifiably* believable, rather than simply “convincing”.

two studies are meaningfully comparable in cases where that is of interest. However, reusing data requires careful judgment to be applied in tailoring data for each new use. If there is no explicit record of the quality of the data used in an M&S study (or of the rationale for why specific data sources were chosen and how they were evaluated and used), subsequent studies must often start over again without benefit of any previous evaluation or tailoring of data.

The amount of data quality evaluation and assurance effort that is warranted by a particular M&S study will depend on a number of factors, including the importance of the study, how sensitive its results or recommendations are expected to be to its data, and how much risk is associated with its producing incorrect or misleading results. These factors should be balanced against the resources available for performing the study, and the data quality effort should be tailored accordingly. In performing this analysis and deciding how much effort (and funding) to allocate for ensuring data quality, two caveats should be kept in mind. First, data quality “repair” (performed after problems are encountered) is likely to be more expensive and less effective than systematic data quality evaluation and assurance performed in advance; that is, data quality efforts are more likely to be cost-effective if they are planned for and implemented in the early stages of a study. Similarly, it is often the case in practice that the apparent savings achieved by avoiding data quality assurance expenditures are more than offset by the cost of having to redo cases involving flawed data or even having to redo entire studies when data problems are encountered.

The goal of this memorandum is to show how we can realistically evaluate data quality, document these evaluations in ways that are of use to data users (who may not be experts in the subject areas represented by the data they are using), and improve data quality by using these evaluations.

The first step in developing a framework for improving data quality is to understand what we ought to mean by the term. While much has been written about data and database design, it is not always obvious what “data quality” means—or should mean. I therefore first analyze what we mean by data and the use of data in modeling and simulation, in order to derive a deeper understanding of data quality. In light of this understanding, I then consider the processes that create and affect data, in order to understand how data quality can be ensured and maintained.

There are many dimensions of data quality. For example, the “concrete” dimension focuses on data as stored (e.g., in a database or on paper) whereas the “abstract” dimension of data quality focuses on data independent of representation. There is also a “procedural” dimension, focusing on what we do with data, a “life-cycle” dimension, focusing on the different phases of data from creation to use and archiving, and an “organizational” dimension, focusing on how organizations create, store, modify and use data. The discussion here attempts to straddle these various dimensions without limiting itself to any single one of them. In particular, the emphasis here (derived from the perspective of using data to perform M&S studies) is on data in computer-readable databases, though I address some issues surrounding the creation of data (e.g., from measurements of real-world entities or phenomena) and acknowledge issues involving the storage and access of data in non-database (“flat-file”) and even “offline” (e.g., paper) forms, insofar as these have implications for the quality of data used in M&S.

In addition, this inquiry will involve many issues that can be viewed from either an ideal, theoretical perspective or a practical, realistic one. The approach taken here is to explore the theoretical viewpoint first, since this must be the basis for subsequent, pragmatic analysis, rather than attempting to describe the practical reality before having analyzed the theoretical issues. If data quality were a well-defined concept, the more pragmatic approach might be preferable, but I believe that the concept is not sufficiently understood to justify this shortcut. I therefore discuss data quality from a relatively theoretical perspective to make sure the concepts are clear before discussing practical issues. Furthermore, although the focus is on data quality in the context of military M&S, I believe this discussion applies to a much broader range of contexts.

## 2. BACKGROUND AND DEFINITIONS

Discussions of data quality have appeared in a number of recent publications, as have a number of crucial terms relating to data and databases. I therefore first review some existing definitions of “data quality” in other DoD publications and present definitions of a number of data-related terms that are used in this document.

### 2.1 Previous definitions of data quality

Dep Sec Def Memo 13 October 93 defines data quality as “the degree of compliance between an organization’s data environment and its ‘business rules’”. This captures the idea that data quality is relative to the needs of users, but two important things are missing from this definition. First, it assumes that there is always some single organization that is the sole user (or owner) of the data: this may be true for corporate data, but it oversimplifies the situation for M&S data, where different users in different organizations may have different requirements for the same data, arising from their different intended uses. M&S data may not be specific to any single organization, so it may be difficult to ascertain what “business rules” are applicable. More fundamentally, this definition focuses on *verification*, i.e., the degree of consistency of the data with respect to organization-specific criteria; it has little to say about *validation*, i.e., the extent to which the real world is accurately and appropriately described by the given data.<sup>3</sup>

Both DoD 8320.1-M (*Data Administration Procedures* [9]) and the user manual for the Defense Data Dictionary System (DDDS, formerly the Defense Data Repository System, DDRS) quote the *American National Dictionary for Information Systems*<sup>4</sup> in defining data quality as “The correctness, timeliness, accuracy, completeness, relevance, and accessibility that make data appropriate for use”. This is a more useful definition, especially since it includes the idea of appropriateness. Although it leaves the operative words correctness, timeliness, accuracy,

---

<sup>3</sup> See the definitions of “verification” and “validation” below (Section 2.2). Although it is inappropriate to draw too rigid a distinction between these two terms, it is crucial that the concept of comparing data values against the real world be a central aspect of data quality assurance, whether this is considered the sole province of validation (as it normally is) or is allowed to be part of verification as well.

<sup>4</sup> FIPS Publication 11-3, adopted from ANSI X3.172-1990.

completeness, relevance, and accessibility undefined, it provides a good starting point for an exploration of data quality. DoD 8320.1-M (Appendix F) supplies short definitions for the terms timeliness, accuracy, and relevancy, but it does not explain the others nor define overall data quality in terms of these component aspects. This memorandum attempts to provide more concrete definitions for these terms, as well as expanding the concept of appropriateness and integrating the notion of data quality.

Chapter 1 of DoD 8320.1-M discusses data quality as a goal that incorporates data security, defining data quality in terms of availability, accuracy, timeliness, integrity, and need-to-know. Most of the discussion in 8320.1-M that is related to quality concerns data standards and the use of the DDDS; however, Chapter 3 assigns to the DoD Data Administrator (DAd) the job of specifying “quality requirements for Defense data handling facilities” and executing “data quality policies and procedures” while assigning to individual database administrators the job of performing data quality analysis “to detect and prevent data defects before they corrupt databases or end-user applications”. The job of establishing data quality requirements is further elaborated in Chapter 3 as follows:

To ensure data quality, data quality requirements and metrics must be established. Data quality requirements are defined from various authoritative sources during the identification and standardization phases of the data life-cycle. Data quality management is based on the principals (sic) of Total Quality Management as described in the “Total Quality Management Guide” (reference (n)). The Dod DAd, FDAd, and CDAd must ensure that data quality requirement (sic) are identified for all data elements. These requirements are documented in data administration products such as data models, the DDRS [DDDS], and reverse engineering documentation.

By deriving data quality requirements from authoritative sources during the early stages of the data life-cycle, this approach explicitly ignores those aspects of data quality that have to do with the appropriateness of data for a given user’s purposes, which (as discussed in detail below) is a crucial aspect of data quality. DoD 8320.1-M-3 (*Data Quality Assurance Procedures* [10]) spells this out in quite a bit more detail. Unfortunately, operative terms such as accuracy, completeness, consistency, relevancy, timeliness, etc. are defined by reference to 8320.1-M or other standard sources (such as dictionaries), without much elaboration. This builds the entire quality assurance enterprise on a weak foundation, since it remains difficult to be sure that the fundamental terms of the discussion are sufficiently understood.<sup>5</sup> 8320.1-M-3 also discusses the DoD Total Data Quality Management (TDQM) process, but this discussion focuses on administrative issues and is almost completely free of content in terms of what constitutes data quality. Although the Data Quality Engineering (DQE) methodology discussed in Chapter 4 adds some meat to this discussion, it is essentially concerned with verification and says almost nothing about validation. (The DQE software, on which the DQE methodology was based, is discussed in greater detail below, in Section 8.1.)

---

<sup>5</sup> It is hoped that the present discussion avoids being guilty of this same criticism.

The DoD M&S Master Plan (5000.59) adds to the DDDS definition (from DoD 8320.1-M) the following discussion of data quality (taken from *The Digital Geographic Information Exchange Standard*, Edition 1.2, January 1994), which is clearly oriented toward geographic/cartographic data:

Quality statements are required for source, accuracy (positional and attribute), up-to-date-ness/currency, logical consistency, completeness (feature and attribute), clipping indicator, security classification, and releasability.

*Data Management Performance Criteria* (94008AGE-02, October 1994), prepared by the Federal Systems Acquisition Division of The Federal Systems Integration and Management Center (FEDSIM), for the Dept. of Agriculture's Office of Information Resources Management, identifies candidate "performance measures" for evaluating the quality of data, which are to be the first output of its data modernization activity; these include: "cost of data collection, customer usage, ease of understanding, and support for process improvement"; the second output of this activity would be quality data, *per se*, of which candidate examples would include: "accuracy, completeness, customer satisfaction, degree of precision, timeliness".

The TDQM (Total Data Quality Management) group at the MIT Sloan School of Management has published a number of papers on data quality in recent years, including [17] which defines data quality as fitness for use by the consumers of the data. While this is a somewhat circular definition ("data is of high quality if its users consider it to be of high quality") it nevertheless attempts to capture the crucial "appropriateness" aspect of data quality, i.e., that it is relative to the needs of its users. Unfortunately, much of this group's published work makes use of a manufacturing analogy for data, in which data values are considered analogous to manufactured items. This analogy misses one of the most fundamental aspects of data, i.e., that data values *represent* things in the real world, which is utterly foreign to manufactured objects. This line of reasoning therefore tends to ignore—or at least fails to illuminate—many essential aspects of data quality.

Against the somewhat vague background of these existing definitions, the remainder of this paper attempts to elaborate the concepts necessary to understand what is and ought to be meant by data quality.

## 2.2 Defining some data-related terms

Although the word "data" hopefully does not require formal definition,<sup>6</sup> it is nevertheless worth making a few comments about the use of this crucial term. It is particularly important to note that much of the inherent quality of a data value is determined the instant that value comes into being, before it is recorded in a database (or even on paper). The process of generating or creating data typically consists of observing the real world and noting or measuring some aspect of reality, which is to be modeled via appropriate data values.<sup>7</sup> The data creation/generation process therefore has

---

<sup>6</sup> In fact, it is surprisingly difficult to define "data" unambiguously (see [11], Section 2.2 for a discussion of some alternative definitions), with the curious result that many books on data or databases avoid doing so entirely.

important implications for data quality, as discussed in subsequent sections (see especially Section 3.9). On the other hand, an M&S study typically looks for existing data in machine-readable form (i.e., “flat” files or databases), though it may also collect data from paper (published books, handwritten tables, etc.) or even oral sources. Nevertheless, whatever the initial source of data for an M&S study, since the studies that are of concern here are those that are performed using computers, their data must ultimately be represented in machine-readable form. Though some studies may access data recorded in other than database form (e.g., flat files), the trend is likely to be toward the increasing use of databases, since the widespread use of databases is one of the most effective means of encouraging data reuse, reducing error-prone scanning or re-keying of data, and implementing the kinds of data quality improvement methods advocated herein. The primary focus of this paper, therefore, is on data as represented in databases.

Unless otherwise specified, the term “entity” is used here to mean some real-world object or phenomenon that is to be described by data. In its conventional, narrow sense in the data context, an entity is represented by a row in a single relation (table) in a database, but in the larger sense, it may be a highly-structured object, such as an organization, which may be represented only implicitly by “joining” many relations in a database.<sup>8</sup> When it is necessary to emphasize the former (narrower) meaning of this term, it will be qualified by referring to it as “an entity (row) in a relation” whereas the unqualified term “entity” should be understood to mean an arbitrarily complex real-world entity, whether or not it is represented by a single row in a relation. Furthermore, the term “entity” will generally be used in this paper without specifying whether it means an individual, “atomic” entity or a structured, composite entity. Although this distinction may be crucial for some purposes, it can often be ignored in this discussion, since we are concerned with the quality of data items, *whatever* they represent. By avoiding the temptation to make this distinction unless it is absolutely necessary, it is hoped that this discussion will apply equally well to atomic and structured data.

The term “attribute” will be used in its usual sense, to mean some aspect or property of an entity. Strictly speaking, an attribute therefore corresponds to a column in a relation, but when it will not cause confusion, the term will also be used (as is often done) to mean a specific *value* of an attribute (i.e., a cell in a relation, containing the value of the attribute for a specific row or entity). When it is necessary to distinguish between these meanings, the term “attribute value” will be used to denote the value of an attribute for a specific entity.

The terms “data element” and “data item” will generally be used to refer to single, atomic attributes or attribute values, though the possibility of composite data elements is not excluded. As is often done, the term “data item” will be used to mean either the data element itself or a specific value of

---

<sup>7</sup> In some cases, the “real world” being observed may be another model, e.g., when using a model to generate data for some hypothetical system or situation.

<sup>8</sup> This generalization of the term “entity” is motivated by the desire to keep this discussion independent of any specific database paradigm: whereas in most cases, the word “entity” will have the meaning it has in the relational database or entity-attribute paradigms, it is also allowed to mean a structured object, e.g., in an object-oriented database.

it; when this would cause confusion, the term “data value” is used to mean a specific instance of a data item (i.e., a value in an individual cell in a relation).

Finally, the following definitions of verification, validation, and certification are used here:

**Data Verification:** The assessment of data with the intent to ensure that they satisfy specified constraints and relationships, conform to specified data standards, and are transformed and formatted properly for their intended use. Data user verification performs this assessment using specifications derived from the intended use of data in a particular model for a particular purpose. Data producer verification performs this assessment using standards and specifications derived from the producer’s mission statement or the requirements of a specific data user or community.

**Data Validation:** The assessment of data for their intended use by evaluating the methods by which data values have been derived and comparing those values against independently-acquired values that are either known or best-estimates. Data user validation performs this assessment with the intent to ensure that data are appropriate for use in a particular model for a particular purpose. Data producer validation performs this assessment with the intent to ensure that data satisfy stated validity criteria and assumptions, derived from the producer’s mission statement or the requirements of a specific data user or community.

**Data Certification:** The determination that data have been verified and validated. Data user certification is the determination by the application sponsor or designated agent that data have been verified and validated as appropriate for the specific M&S usage. Data producer certification is the determination by the data producer that data have been verified and validated against documented standards or criteria derived from the producer’s mission statement or the requirements of a specific data user or community.

Note that the above definition of “validation” includes the validation of methods used to produce data as well as checking individual data values. This is discussed in further detail below (in Sections 3.10 and 9.3).

### 3. ASPECTS OF DATA QUALITY

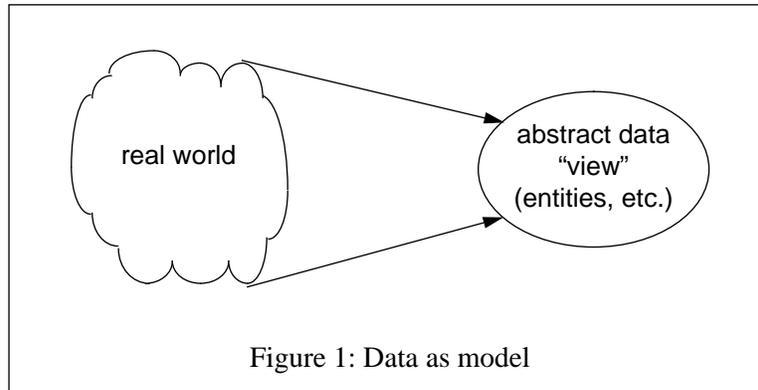
My approach to understanding data quality relies on the notion of data as modeling reality.<sup>9</sup> I therefore motivate and discuss this view before discussing data quality *per se*.

---

<sup>9</sup>For relevant discussions of modeling, see [12] and [13].

### 3.1 Data as modeling reality

For our purposes, data can best be thought of as a kind of model, i.e., the result of attempting to describe the real world (Figure 1). Any such description of reality is always an abstraction, always



partial, always just one of many possible “views” of reality. (We are mainly concerned here with “abstract” data, i.e., *conceptual* entities, attributes, values, relationships, etc. This discussion will remain largely independent of the specific representations chosen for these abstractions.<sup>10</sup>)

Any given real-world entity, process, or phenomenon can be modeled by many different data views, depending on one’s purpose (Figure 2). For example, the speed of a ship might be represented by one of several symbolic values (e.g., *slow*, *medium*, or *fast*), by a single numerical value (e.g., *20*), by a table of numerical values representing speeds under different sea and load conditions, by a set of parameters to an arbitrarily complex function of such conditions, etc. In each such case, a single aspect of reality (the speed of the real ship) is represented by a data value. In order to emphasize this modeling aspect of data, a collection of data values might be referred to as “a data-founded model” (i.e., a model *consisting* of data).<sup>11</sup>

The choice of what kind of data to use to model reality is made prior to generating the data, just as the choice of how to build any kind of model is made before beginning to build the model. The appropriateness of this choice (for example, which of the above ways to represent the speed of a ship) is a crucial aspect of the quality of the resulting data, regardless of what data *values* are

---

<sup>10</sup> Representing data concretely (e.g., in a database) involves mapping abstract data items into specific data modeling constructs, such as the entities, attributes, values, and relationships of some database formalism. This can be done in many ways: each such mapping can in turn be thought of as a model of the abstract data it represents. This level of representation introduces its own quality issues (having to do with datatypes, encodings, field lengths, etc.), but the present discussion focuses on higher-level issues of data quality, i.e., how well abstract data values represent the real world for a desired purpose.

<sup>11</sup> This awkward term is chosen reluctantly, in light of the fact that the more tempting term “data model” is reserved for a different concept, namely some particular, formal way of organizing a collection of data. Similarly, the term “data-based model” is too easily misunderstood as “database model” (which would presumably be a model of a database), while alternatives such as “data-oriented model” are too weak.

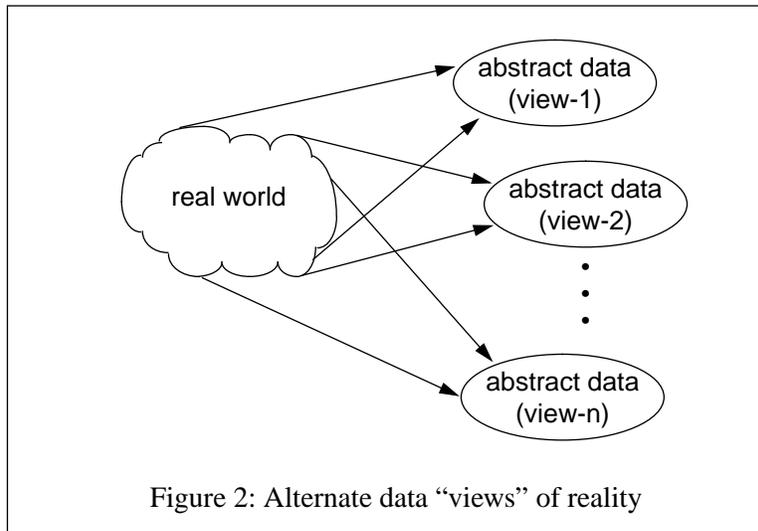


Figure 2: Alternate data “views” of reality

ultimately recorded. Note that this modeling phase includes formal “data modeling” as well as any prior conceptual modeling, in which abstract choices are made about how to view (i.e., model) the world.

Once a conceptual model has been chosen, data values representing the real world according to this conceptual model are either obtained from pre-existing sources or are generated by observing and measuring either the real world or (in some cases) another model that is being used as a surrogate for the real world. This generation process plays a crucial role in providing high-quality data values, since it is the source of those values whose quality will ultimately be evaluated. The generation process may include its own mechanisms for calibrating, verifying, testing, and validating its measurement equipment or procedures. Yet whatever this process entails, its eventual outcome is a set of data values that purport to model the real world in some way that is appropriate for some range of purposes. These values implicitly form a data-founded model; they may be transmitted from their point of generation to some other location, and they may be transformed into something other than their “native” form in the process, but sooner or later they must either be recorded (whether on paper or in some machine-readable form, such as a database) or “consumed” by some process (e.g., a model) that relies on them.<sup>12</sup> In most cases of interest, however, these values will be recorded and stored somewhere, whether to allow them to be reused in the future or to provide an audit trail for the process that is using them. It is therefore useful to consider the data values that constitute a data-founded model as being “born” only at the moment when they are stored, while recognizing that the birth process itself is of crucial importance in providing data quality.

<sup>12</sup> Data values may be used by a model or simulation without ever being recorded anywhere, for example, when they are produced from instrumentation of live phenomena.

### 3.2 Data as residing in databases

Although a data-founded model can be considered in the abstract, it is convenient for many purposes to equate such a model with a concrete database containing its data values. This provides a static view of data as residing in a database, though it does not capture the dynamic aspects of data generation or of the conceptual design of a data-founded model prior to the generation and storage of its constituent data values. These dynamic aspects must be addressed by focusing on the processes that generate and modify data (as discussed below in Section 3.9). Nevertheless, databases provide a useful focus for many data quality issues.

In many cases, M&S studies access their data from myriad, fragmented sources, some of which are not databases at all. These sources can include paper records, “flat files” (i.e., data tables represented as unstructured text), the direct output of software processes such as models and simulations<sup>13</sup> or of hardware sensors, data values that have been generated or invented by the simulation modelers or by domain experts to fill gaps in existing data, etc. Legacy systems may be especially likely to utilize sources of these kinds, possibly combined with data from actual databases. While the data quality improvement enterprise must ideally address all of these different kinds of data sources, there are a number of reasons to view the problem as one of improving the quality of the data in databases. First, even when a study collects data from disparate sources, it typically integrates them into a database of some sort prior to use: whereas the form of this database may be idiosyncratic to the model, it is nevertheless a database in some sense, and subsequent repetitions of the given study—or related studies—may be able to benefit by reusing its data. It is therefore often useful to think of the data used by a study as residing in a database, whatever form the original source data may be in.

Moreover, the trend in new M&S systems is generally toward the increasing use of formal databases as sources of data for studies; much of the current effort in the data area is focused on the collection and integration of existing data sources into common “federated” databases that use standard data elements and standard structures. Whether such databases remain relational or evolve toward object-oriented forms, they represent the wave of the future: M&S systems are increasingly likely to find their data in formal databases.<sup>14</sup>

Finally, while it is meaningful to speak of the quality of an abstract data value that is not represented in any database, it is less useful to speak in the abstract of recording evaluations of this quality or attempting to utilize such evaluations. One of the purposes of this paper is to propose

---

<sup>13</sup> Although data may be generated by software processes in some cases, the present discussion does *not* consider the output of arbitrary programs to be “data” unless this output is used by some other process *as* its input data. It is difficult to draw a sharp distinction here, since a model may take part of its input from some other model or some other process that is generating the data the model needs. However, it does not seem useful to consider all information passed between two software processes to be “data” regardless of the role that this information serves for the receiving process. In the limit, a broad definition of this kind would treat as “data” all parameters passed between subroutines (and by extension perhaps, all data structures in programs). This seems well beyond the scope of the concept of “data” for our present purpose. In particular, accepting this definition would expand the scope of “data quality improvement” to include most aspects of software design and implementation, which does not appear to be useful. Note, however, that this distinction may not be drawn in the same way by the DIS (Distributed Interactive Simulation) community.

mechanisms for recording and utilizing evaluations of this kind, and these must be represented in some concrete form. Whereas there may be no logical necessity to use databases to record this information, no reasonable alternative has the requisite property of making the information widely available and easily accessible within the M&S community. This argument implies that “quality metadata” (defined in Section 3.5 below) should reside in a database. Though it does not necessarily follow from this that the data described by this metadata must reside in the same database as the metadata itself (or for that matter, in any database) it is logically advantageous for data to reside in the same place as the associated quality metadata.

For all of these reasons, it is not often necessary to distinguish between a data-founded model and its representation as data in a database. The term “database” will therefore be used as a surrogate for the term “data-founded model” in the remainder of this paper. This is done as a matter of linguistic convenience, despite the facts that a given model or simulation may get its data from a number of distinct data-founded models, that a given data-founded model can be implemented by many alternative database designs, and that the process by which the values in a database are generated is at least as important an aspect of data quality as the properties of the values themselves or the properties of the database in which they reside.

### **3.3 Data quality as suitability for an intended purpose**

Recognizing data as a kind of model leads to the crucial recognition that it is meaningless to speak of having perfect, complete, and correct data, just as it is meaningless to speak of a perfect model. The quality of any model is relative to its purpose. It is impossible to define the quality of a model without first defining its intended relationship to the reality that it models, where this relationship must be derived from the purpose for which the model is intended.<sup>15</sup> Similarly, the quality of data (as modeling the real world) is relative to the purpose (or purposes) of the data. This is not to say that there are no objective aspects of data quality (such as accuracy and consistency), but even these must always be interpreted in terms of the purpose of the data. For data intended as input to a simulation model, the purpose of the simulation model itself (or more precisely, the intended use of that model) becomes a key factor in understanding the purpose of the data. Note that it may be difficult to define the purpose of generic data (such as geographic data) which may be used by many different models for many different purposes.<sup>16</sup> Nevertheless, it is crucial to recognize that the quality of *any* data is always at least partially relative to the intended purpose or range of purposes of the data.

---

<sup>14</sup> In this regard, two related DMSO-sponsored efforts have addressed the problem of legacy systems: the Joint Data Base Elements (JDBE) Project at the Fort Huachuca Electronic Proving Ground (EPG) has developed a reverse engineering methodology for determining subject area data requirements from pre-existing (“legacy”) databases, and the M&S Information Management (MSIM) Project has developed a reverse engineering technique for determining data requirements from legacy models and simulations.

<sup>15</sup> See [12] for a detailed discussion of this.

<sup>16</sup> It is an open issue how best to characterize such generic data, as well as how many categories it may be useful to distinguish along the continuum between generic and specific data. This paper simply refers to the endpoints of the continuum (generic vs. specific), while recognizing that this distinction may be ambiguous or too coarse.

Viewing data as a kind of model suggests that “*Data quality*” is a measure of the suitability of data for its intended purpose (or range of purposes). Since purposes are properties of people, the above might be rephrased as: “*Data quality*” is a measure of the suitability of data for its users’ intended purpose(s). This defines a source of requirements for M&S data quality: a particular user engaged in a particular modeling project using a particular simulation model (or collection of such models) for a particular purpose levies a specific quality requirement on the data to be used in that project.

This highlights what should be an obvious point—but is easy to miss: A prerequisite for any effort to improve data quality should be to identify the potential users (customers) of that data and understand the purposes and intended uses they have in mind. This is an inescapable first step in defining validity criteria for data, since such criteria are necessarily relative. In the case of generic databases, it may be tempting to skip this crucial step, but skipping it is perilous. Although this step may be unnecessary in some cases, e.g., for a database of logarithms or trigonometric values whose uses are so well understood that their quality (beyond their precision and resolution<sup>17</sup>) is truly independent of their use, it should never be taken for granted.

### **3.4 How can we promote data quality?**

Having established the above conceptual framework, how can we promote and improve data quality in the interest of improving the quality of modeling and simulation? I believe this can best be done by using a two-pronged strategy to improve data quality by means of two equally important, parallel approaches: (1) performing explicit evaluation of data and (2) establishing organizational control over the processes that generate and modify data. These approaches require: (i) augmenting databases with metadata in order to record information needed to assess the quality of their data, record the results of such assessments, and support process control of processes affecting data; (ii) encouraging producers and consumers (users) of data to implement organizational commitments to perform distinct phases of explicit verification, validation, and certification (VV&C) on their data, using metadata both to direct these activities and to record their results; and (iii) establishing control over the processes that affect data, to improve the quality of data generation, transformation, and transmission, again using metadata both to support this activity and to record its results. It should be possible to develop automated tools to help capture and maintain metadata whenever generating or modifying data, thereby greatly facilitating this strategy.

There are several points worth noting about this strategy. First, although much of the focus of this paper is on explicit data VV&C,<sup>18</sup> this is not an end in itself but rather a way of improving both the quality of the data we use in M&S and our understanding of and confidence in that quality. In particular, VV&C must be used to improve certain key processes (notably those that generate data in the first place) in order to improve the quality of future data rather than just “repairing” existing

---

<sup>17</sup> The term “precision” is used here in its literal sense, i.e., the number of significant digits in a numerical value, whereas the term “resolution” is used to mean the level of detail represented by data.

<sup>18</sup> For further discussion of VV&C, see [16].

data. Although relatively little formal VV&C may have been performed in the past, the same effect has often been achieved by repeating modeling studies in their entirety or by performing numerous additional modeling runs to address problems as they have occurred. Part of the intent of performing explicit VV&C in advance is to reduce (if not eliminate) the need for such repair efforts. Nevertheless, sometimes the best way to evaluate data quality is to use data as input to a model and run that model to see if it produces expected results. Similarly, multiple model runs may be used to perform sensitivity analysis as a means of data quality evaluation. In such cases, VV&C may legitimately involve running models, yet the intent is still that this should ideally be done *prior* to performing a particular M&S study.

In all cases, an M&S study should involve the preparation of a VV&C plan, typically in coordination with a VV&A plan for any simulation models it expects to use. A VV&C plan should include an analysis of the potential risk of using poor data and an assessment of what constitutes an acceptable level of risk for the study at hand. Based on this assessment and the available resources, the plan should provide a rationale for how VV&C is to be performed, tailoring the strategy presented in this paper to the needs of the user, the study coordinator (or sponsor), etc. One crucial aspect of this tailoring process is the determination of how quantitative and objective the evaluation of data quality can be for the study at hand (and within the resources available). While quantitative, objective criteria for quality are generally preferable, they are not always possible or affordable; qualitative, subjective criteria<sup>19</sup> such as comparison against a range of “base case” scenarios, expert opinions (e.g., from a “Council of Colonels”), or even “gut feel” may have to be accepted in some cases, recognizing that the user must generally bear the cost of much of the VV&C to be done in the course of a study.

While the above strategy includes a number of interrelated ideas, many of them involve metadata. A discussion of metadata therefore provides a useful starting point for our exploration of data quality. However, the reader should bear in mind that the need for metadata is derivative: metadata issues should not be allowed to overshadow the primary issues of how to improve data quality—metadata is merely a convenient and concrete way to organize and structure our investigation into data quality.

### **3.5 Metadata**

For our purposes, the most useful way to define metadata is simply as data about data. Any description or amplification of data can be thought of as metadata. (This definition has the desirable side-effect of defining metadata as data, which allows metadata to be about metadata as well; this is elaborated below.) When it is necessary to clarify this distinction, we can refer to “ordinary” versus “meta” data. In some cases, this distinction may be moot. For example, a given database might contain an integer value for a time datum in one field and the symbolic value “msec” in a related field (specifying that the time datum is to be interpreted as milliseconds). One model might read the symbolic value of the units field in order to interpret the time value, in which case the units

---

<sup>19</sup> For additional discussion of the distinction between objective/subjective and quantitative/qualitative criteria, see Section 6, page 27.

field would serve the role of data for this model, whereas a different model might ignore the units field (for example, implicitly interpreting the time value as being in milliseconds), in which case the units field would serve the role of metadata for that model.

The purpose of metadata is generally to supply context for data, helping users interpret and use data appropriately. There are many possible kinds of metadata, corresponding to many possible needs, such as understanding the meaning of a database, documenting its origin, controlling or improving access to it, making it more easily shared across different disciplines, or evaluating and recording its quality. Although the present discussion is concerned with only a single kind of metadata (that which is required to support data quality), we must recognize that a database may include many other kinds of metadata as well. In addition, although metadata (as well as data) can conceptually be abstract, we are concerned here with concrete representations of both data and metadata in online form.<sup>20</sup> To fulfill its purpose, a concrete collection of quality metadata must logically adhere to the data whose quality it describes: this implies that quality metadata should—at least from a logical perspective—reside in the same database as the associated underlying data. This can be implemented in various ways (including “joins” across separate databases or “views” within a single database) to allow separating data from metadata for certain purposes. For example, a user evaluating a database for possible use may need the quality metadata for that database but not the data values themselves, whereas a user employing a database that has already been evaluated and chosen for a particular use may no longer need the quality metadata associated with that database. Nevertheless, quality metadata should be logically (if not physically) coupled with data in all cases, to allow users to access quality information about their data whenever necessary.

It is necessary to introduce metadata to help measure and improve the quality of data because most data collections (and even many formal databases) maintain insufficient information about the intended use of their data to allow assessing their quality. First, metadata can directly support data quality improvement by providing the information needed to perform constraint-satisfaction checks (type checks, consistency checks, sanity checks, etc.); much of this kind of metadata should be provided by a good data dictionary (though it may not be in all cases). Yet the importance of metadata goes far beyond this. Since data should be viewed as modeling reality, it is vital to include sufficient metadata to help a potential user understand the assumptions and limitations implicit in a collection of data, in order to evaluate it for the user’s intended purpose. Metadata can inform this evaluation process by telling the evaluator the intended range of purposes and the conditions of derivation of the data. In addition, the results of such evaluations can be recorded in metadata kept with the data, so that future users can benefit from past evaluations. In particular, “certifying” data requires that trusted parties must use metadata to record the fact that V&V has indeed been performed on the data: metadata must describe the V&V that has been done. Finally, metadata can supply the context needed to control processes that affect data: for example, metadata can record historical values of the metrics used to maintain process control over such processes. Metadata should be seen as a means to an end, enabling the intelligent interpretation and evaluation of data and the processes that affect data. It is therefore useful to examine the kinds of quality-related

---

<sup>20</sup> One of the implicit but crucial advantages of online data and metadata is that they should never have to be re-entered or “re-keyed” for use by an online system; this alone eliminates a significant source of error.

metadata that should be recorded, as a way of identifying the kinds of activities that should be performed to ensure data quality. This is done in detail in later sections.

Since metadata can describe metadata as well as “ordinary” data, a potential confusion arises. Just as it may be important to have metadata specifying the source, time of entry, certainty, etc. of an ordinary data item, it may be important to have this kind of metadata for some metadata items themselves. In particular, it is vital to have metadata that can help a user evaluate the quality of the quality metadata in a database. For example, it may be important to know the source of the metadata that assigns a low certainty to a given data item. Similarly, a metadata item may have missing or inapplicable values, or it may have an unknown source or time of entry, which would be indicated by metadata about that metadata item. It may even be useful to verify, validate, and certify the metadata in a database prior to performing VV&C on its data; in practice, however, this is unlikely to be warranted, except possibly for databases that are both critically important and static enough to be described by static metadata.

Thinking of metadata as hierarchical allows us to say that metadata that describes metadata can still be called “metadata” (though it can also be called “meta-metadata” if this distinction is necessary). It may not make sense to have all possible kinds of meta-metadata for every metadata item (any more than it may for every ordinary data item<sup>21</sup>), but the potential for such meta-metadata must be there for when it is needed. Metadata may even apply recursively: for example, every data item has a source (which is metadata), but every metadata item specifying the source of a data item also has a source. Fortunately, the source information for most metadata in a given database will tend to be the same (or will fall into one of a small number of groups of metadata items, each sharing the same source); therefore, it will rarely be necessary to specify source metadata for more than one level of metadata above the ordinary data level. At worst, it will normally be sufficient to supply source metadata for an ordinary data item plus source meta-metadata for this source metadata, without recursing to additional levels. Because this recursion rarely exceeds one level, we need not normally bother to distinguish between the term “metadata” (which applies to ordinary data) and the term “meta-metadata” (which applies to metadata): we can simply refer to the upper level in all cases as “metadata” and the lower level as a “data” (which may be ordinary data or metadata). Similarly, the term “meta-attribute” will be used here to denote a particular metadata item, whether that metadata item describes data or metadata: for example, the time of entry of any data or metadata item is a meta-attribute of that item. Much of the discussion of metadata describing ordinary data items will therefore also apply to metadata describing metadata (i.e., “meta-metadata”), and much of the discussion of ordinary data items will apply to metadata items as well.

### 3.6 Generating metadata

Although in principle, metadata can be added to a database at any time, in practice, it may be difficult to construct meaningful metadata retroactively. For example, the source or date of a data item may be difficult to determine or reconstruct if it is not recorded when the data item is first

---

<sup>21</sup> For example, a data item that represents a static quantity, such as a numerical constant or conversion factor, may not warrant having a timestamp.

generated. Similarly, reconstructing the rationale for the design of a “legacy” (i.e., pre-existing) database or for choices of its data element definitions, domains, etc. may be rather difficult. This has serious implications for legacy databases, many of which may defy the generation and addition of meaningful metadata at this late stage in their life-cycle. In general, the greater the difficulty encountered in attempting to create metadata for any database the greater the indication that the database is not well documented or easily understood and may therefore be of dubious value.

For this reason, it is generally preferable to generate metadata for a data collection when that collection itself is generated or first assembled into what we refer to as a “database” (for convenience, even when it is not literally the case). Different databases come from a wide range of sources (as discussed above in Section 3.2), but in all cases the original context and documentation of the data form the basis for the metadata describing that database. If the data source for a new database is some other, pre-existing database, then “generating” metadata for the new database may be a simple matter of transcribing or translating existing metadata or other documentation. If a new database contains “new” data (e.g., resulting from some new measurement process or generated by running some model), then metadata for the new database must be generated along with the data: this will typically be straightforward, since the required metadata will consist essentially of a description of the process, assumptions, and conditions surrounding the generation of the new data. Furthermore, in many cases, straightforward extensions to database tools should allow automatic or semi-automatic capture of metadata as a database is generated (for example, supplying the date at which new data items are entered).

### 3.7 The Cost of Metadata

Enumerating desirable metadata categories quickly makes it apparent that metadata can potentially require more storage than the ordinary data in a database. This is a sobering prospect, which threatens the entire data quality enterprise. If the required metadata will swamp a database with additional information, how can anyone generate, maintain, and use such metadata? Fortunately, it is not often necessary to provide unique values for every metadata item for every data (or metadata) item. Many metadata items can be allowed to “default” to their nominal or expected values. There is a natural inheritance structure for much metadata, wherein a metadata item for a specific data (or metadata) item at the data value or data-element level will often default to a corresponding metadata value from the database level. For example, the time of last update for a data item might default to the time of last update for the database as a whole (if the entire database has been revised). Similarly, meta-attributes for a specific instance data value will often take their values from the corresponding meta-attributes of the data-element definition for that data item. Finally, some metadata values will tend to be the same across most or all attributes of a given entity while others will tend to be the same for a given attribute across most or all entities. Taken together, these factors suggest that there may be relatively few *distinct* metadata values in any given database, compared to the *potential* number of such values (which, in the worst case, would be the number of meta-attributes times the number of data values, including any metadata values which themselves require meta-attributes). This implies that the storage burden for metadata may not be as severe as it appears. In addition, the cost of creating metadata may be greatly reduced by the design and widespread dissemination of tools that perform automated or semi-automated

generation and capture of metadata at the time of data generation or transformation (as discussed further in Section 6.5).

### 3.8 Explicit VV&C

One of the main motivations for data quality metadata is to support meaningful data VV&C. Metadata can help provide the necessary context to interpret and evaluate data appropriately, as well as providing a place to record the results of VV&C activity. The VV&C itself may consist of many different kinds of testing and evaluation, ranging from simple consistency and sanity checks (e.g., checking for non-negative counts for physical inventory items, known geographic identifier codes, aggregate attributes having values equal to the sum of the values of their components, etc.) to high-level judgments as to the suitability of various classes of data for their intended or expected application. In particular, it is useful to distinguish between two kinds of data VV&C, which may be performed at different times by different parties: these are referred to here as “producer” vs. “consumer” (or “user”) VV&C and are discussed in more detail below.<sup>22</sup>

This paper adopts the conventional distinction between verification and validation, as elaborated above (Section 2.2, page 7) for data. Verification has to do with checking that a data item is of the proper form, is of the required type and within a specified range, is consistent with other data in a database, satisfies specified constraints, or in general conforms to “self-contained” specifications that do not require reference to the real world (that is, to anything other than the data and the specifications themselves).<sup>23</sup> Validation involves checking that a data item correctly represents that aspect of the real world which it is intended to model: this goes beyond checking that it is of the right *form*, requiring that it also be “correct” (where “correctness” mean modeling reality in the desired way). This distinction may not always be a rigid one, and it can be counter-productive to argue about the precise boundary between these two activities. Nevertheless, the distinction is useful and is made throughout this paper. A simple rule of thumb is that verification can (at least in principle) be performed without reference to anything outside the database itself, whereas validation cannot.<sup>24</sup>

In addition, validation consists of more than simply comparing data values against other known values (or the real world): whenever a data transformation is employed, the transformation process itself should be validated to ensure that the transformed data will be valid—in addition to any explicit validation checks that are applied to transformed data values. This essentially involves performing V&V (or VV&A) on the transformation process: this is just one of a number of ways

---

<sup>22</sup> Note: “producers” in this context include all intermediate suppliers, providers, and managers of data, as well as originating sources.

<sup>23</sup> Verification includes what is called “data editing” in the traditional data processing world.

<sup>24</sup> Validation *may* also refer to criteria within the database, such as the intended accuracy of the data in the database, but it will *always* (in general) also refer to some reality outside the database, whether this is the real world *per se*, some other database that is serving as “ground truth” for this one, or the opinion of a subject-matter expert.

in which VV&C of data and VV&A of models (or processes) are intertwined, as discussed further in the next two sections.

Validity has two distinctly different aspects, which we can refer to as “objective validity” and “appropriateness” as discussed below. The first of these corresponds to what most people intuitively mean by validity: objective validation ideally requires comparison of data with the real world. In some cases this is impractical (as it may be in the case of *any* model) and must be replaced by “expert judgment” as a validity criterion; but the choice of this expedient should always be carefully noted and regarded with skepticism. On the other hand, evaluating the “appropriateness” of using data for some purpose typically *requires* judgment—by the user, who must be considered an “expert” on the subject of the intended use of the data.

Certification consists of an official, authoritative recording of the fact that data V&V has been performed and has led to some specific evaluation of the suitability of the data in a database for some use or range of uses. Certification of data as being appropriate for a wide range of uses would in theory be more valuable than certification for a specific use; but the relevance and credibility of any certification with respect to any particular use will generally be proportionate to the specificity of the certification for that use. Therefore, overly general certification should always be suspect.

Note that it is not the intent of this paper to discuss specific VV&C activities in detail, despite their centrality to ensuring data quality. The current discussion is rather intended to establish a framework within which a rich set of VV&C activities can (and must) be elaborated in future discussions. The paper merely discusses the theoretical issues surrounding the improvement of data quality and proposes the concept of a data quality profile along with the definition of a complete framework of metadata required to support data VV&C.

### **3.9 Processes that affect data**

While it is useful for many purposes to consider data as residing in databases (as argued above in Section 3.2), it is equally important to consider the processes that create or change data. These “data-affecting processes” include those that generate data in the first place, those that modify, edit, aggregate, or derive data, and those that transform, transmit and propagate data for use in other databases or as input to models or simulations. (Processes that propagate data may include the use of simulation models whose output produces input data for other purposes; in some cases, processes may be intended to transmit data “transparently” without changing anything, but if the potential for change is present, it must be recognized.)

As discussed in Section 3.1 above, the process of generating or creating data typically consists of observing the real world and noting or measuring some aspect of reality that is to be modeled via appropriate data values. In some cases, this may involve collecting or using existing data, whether from databases, “flat files” of unstructured data, text files, paper or oral sources; or it may involve using the direct output of software processes (including models and simulations) or of hardware sensors, or using data values that have been generated or invented by modelers or domain experts to fill gaps in existing data. Since the M&S studies that are of concern here are those that are

performed using computers, their data must ultimately be represented in machine readable form: this need not be a formal database in all cases, but we can speak of it as a database for convenience.<sup>25</sup> It is therefore appropriate to consider “data generation” as consisting of the process of acquiring or otherwise creating data (from whatever source and by whatever means) to produce a machine-readable form (i.e., database) that can be accessed by M&S users.

Data generation involves numerous measurement, calibration, and representation issues that are beyond the scope of the present discussion. However, since data generation forms the ultimate foundation of data quality, these issues are central to any attempt to improve the processes that affect data, as discussed below.

Of course, the initial form used to represent data may not be the eventual form that a particular M&S user accesses: data may be transformed many times between initial generation and eventual use. However, we can view data as residing in a database at each stage after its generation; from the point of view of its result, each data-affecting process after generation simply transforms data from one database form to another. The next section discusses ways of improving these processes themselves.

### **3.10 Improving processes that affect data**

In addition to performing VV&C *per se*, it is important to improve the generation and manipulation of data. Whereas the purpose of V&V is to *evaluate* the quality of existing data, the intent here is to *improve* that quality. Ideally, there should be a close relationship between these activities: On the one hand, V&V should examine the processes that affect data, as well as evaluating the data *per se*, while on the other hand, these processes should aid V&V to the extent that they can guarantee that their results are valid. In particular, the methods used to produce and modify data should themselves be validated wherever possible; as noted above, this amounts to performing VV&A on all data transformation processes.

Except for the decay of media or inadvertent changes introduced by what are intended to be transparent processes (such as the occurrence of undetected errors when copying files), data can change *only* when data-affecting processes are performed. Therefore, improving the quality of these processes should improve and ensure the quality of the data they produce and manipulate. As has been pointed out in the literature (Redman 1992) there is an interplay between data V&V and controlling the processes that affect data. If a database is relatively static, then performing V&V on it may be the preferred strategy, since it must be done only infrequently to establish the quality of the data. On the other hand, if a database is fairly dynamic (i.e., frequently affected by various processes), then data V&V would have to be performed repeatedly to ensure data quality; in such cases, it may be more cost-effective to try to control and improve the processes that change the database, in an effort to *maintain* its quality, rather than continually having to reestablish it. This

---

<sup>25</sup> As suggested above, capturing data in machine-readable form as early as possible is likely to be the most effective means of encouraging data reuse, reducing error-prone scanning or re-keying of data, and implementing the kinds of data quality improvement methods advocated in this paper.

trade-off applies reasonably well to simple, corporate databases or to any database whose uses are well defined and static. However, when generating new data, these two approaches may coalesce into one, since data V&V would be performed as part of the process of generating the data in the first place. Furthermore, the situation is often more complicated than this simple dichotomy suggests.<sup>26</sup>

In simple cases, if all relevant data-affecting processes (including, conspicuously, those that generate the data to begin with) could be proven “correct” and appropriate, it might be unnecessary to verify or validate the data ever again. Unfortunately, the real world is not this simple. For one thing, these processes may in general be harder to validate than the data they produce, in which case validating the processes would afford no leverage over validating the data (keeping in mind that when generating new data, the two approaches may become intertwined). In addition, it is rarely possible to guarantee that data will not be changed by some unknown (and uncontrolled) process; therefore, validating recognized data-affecting processes is insufficient to guarantee the quality of the data. Moreover, since the validity of data is relative to a user’s intended purpose, it is meaningless to think of validating data once and for all—by *any* technique—except in cases where users’ purposes (to the extent that they affect what is meant by the validity of the data) are known in advance and never change. In particular, it is unlikely that data-affecting processes can ensure that data will be appropriate for a given user’s intended purpose if this purpose was unknown when these processes were created. In such cases, controlling the processes that affect data may provide some general quality control, but it cannot obviate the need for explicit data V&V, which must take into account the specific intended purpose and use of the data.

### **3.11 Data quality is relative to users and their purposes**

The above discussion repeatedly refers to users of data and the fact that the purposes and uses they have in mind determine the relevant criteria for the validity of data. It is therefore worth focusing briefly on users and uses. In our context, a given use of data involves the employment of one or more models or simulations to perform some analysis, predict some outcome, support training, etc. It may not always be possible to identify specific users of data, especially when examining generic databases, but it should be possible to find *representative* users even in these cases.

The range of intended uses that is attributed to the expected users of a database should be a key factor in designing the database, creating and modifying the processes that populate or maintain it, and identifying potential V&V techniques to ensure its quality. Furthermore, these expected users and uses and their impact on the design of a database and its data-affecting processes should be described in the metadata associated with the database—even a short discussion of these issues, consisting of a paragraph or two, would add considerable value to many databases. This can provide early-warning flags for the inappropriate use of a database, i.e., when a proposed usage conflicts with (or goes far beyond) the intended uses for which the database was designed.

---

<sup>26</sup> Redman gives an excellent discussion of many of these issues, albeit in a context that is at once somewhat abstract and oriented toward the commercial world, in which databases are typically generated and owned by the organizations that use them, and in which they are often used for more straightforward purposes than as input to complex models.

Although it is possible that a given database may be appropriate for use by user populations (or for purposes) that differ from those for which it was intended, this should always be cause for skepticism.

Even different user populations with similar purposes may implicitly make different assumptions about their data. For example, logistics modelers may require data about equipment and vehicles (such as “footprint” sizes, areal capacities, and various tonnage measures for shipping) that may be irrelevant to combat modelers. Similarly, data-affecting processes such as aggregation may be defined differently by different users: for example, a resource shared by several units may be counted once for each unit or once in total, depending on whether the user is considering concurrent or sequential employment (with or without loss and replacement). Careful evaluation of the assumptions embodied in a database is therefore vital. Only by making such assumptions explicit (in metadata) can they be examined and evaluated as necessary.

#### **4. PRODUCER VS. CONSUMER (USER) VV&C**

As mentioned above, the data in some databases (e.g., geographic information, numerical tables, etc.) may be quite generic. It may be tempting to believe that generic databases can be verified, validated, and certified once and for all when they are created, without regard to their specific use, since they contain relatively “objective” data. Yet as pointed out above, *any* data can embody assumptions that are inappropriate for certain uses. The appropriateness of a database for some specific use *must* therefore be evaluated by the user,<sup>27</sup> regardless of where the database falls on the generic-specific continuum. Does this imply that the entire burden of data VV&C must therefore fall on the user? What can the producer<sup>28</sup> of a database do to help ensure its quality?

Since verification is concerned with self-consistency and conformance to explicit specifications, it is possible for a producer to *verify* a database in a number of ways that are independent of any specific use. Such verification can be performed in accordance with specifications represented by the data model and any other relevant metadata, which describe data format, type, relationships, constraints, etc. Although it is possible that a specific user may levy additional verification requirements on the data (e.g., more restrictive value domains, conformance to expected distribution or auto-correlation results, constraints on relationships among various data items, etc.), the producer of a database can at least verify that it meets its stated specifications for internal consistency, format, precision, etc. This implies that verification can often be performed in large part by the producer of a database, though the consumer (user) may always perform additional verification if desired.

Recalling that data should be viewed as modeling reality, the next question is: Can a database be *validated* by its producer, independent of the purpose of any specific user? Unlike verification,

---

<sup>27</sup> This need not be done *literally* by the user: it may be performed by any suitable representative of the user who is involved in a specific use of the database.

<sup>28</sup> Again, for our purposes, “producers” include intermediate suppliers and managers as well as originators of data.

validation asks whether a database corresponds to the real world in some way. Here again, metadata associated with a database may enable its producer to perform *some* validation in the absence of knowing what use any specific user may intend for the database. For example, the metadata for a geographic database may say that certain features are shown with a specified spatial resolution and accuracy; this can be validated by checking the data for these features against reality, regardless of whether the features themselves or their specified resolution (level of detail) or accuracy (closeness to reality) are *appropriate* for use as input to any specific model invoked by any specific user for any specific purpose.

#### 4.1 Objective vs. subjective validity

It is useful to distinguish between the *objective validity* of a database (with respect to the reality it is intended to model, according to its specifications) and the *appropriateness* of using that database for some specific purpose, i.e., its *subjective validity*. A producer may (in some cases) be able to use metadata to provide objective modeling criteria for validating data against the real world (though in other cases, the best the producer may be able to do is warrant data as being presumably appropriate for some expected range of uses). On the other hand, a user (consumer) must provide criteria for whether a database is appropriate for its intended use. These latter (user appropriateness) criteria are entirely relative to each user's intended use of the data and cannot in general be represented in a database (even as metadata) in advance, since they come into existence only when some user evaluates that database for some intended use.<sup>29</sup>

Of course, the distinction between data producer and user may not always be a rigid one: particular organizations or individuals may play both roles in a given case and may play different roles in different cases. The reason for making this distinction is not to impose an inflexible formalism on the processes of creating and using data but rather to illuminate their salient aspects.

If a user evaluates a database for some use and decides to use it, then the evaluation criteria and results should be recorded in the metadata for the database. Yet if a user evaluates a database for some use and *rejects* it as inappropriate (according to some criteria), it may be unrealistic to expect that user to document these criteria and the reasons for this rejection in the database. In particular, a user may evaluate a number of databases, some quite casually; it is probably unreasonable to ask a user to add metadata to every database that is rejected, simply to document the reasons for rejecting it. Nevertheless, if a serious evaluation is performed, it would be very helpful to document the criteria and results of this evaluation, even if the database is rejected; otherwise, the data quality metadata will be heavily skewed in the direction of positive evaluations. Furthermore, once an evaluation has been performed for some purpose, it would be useful to record its results in the database to avoid or simplify re-evaluating it for this same purpose in the future, whatever the

---

<sup>29</sup> To the extent that appropriateness criteria for a given purpose can be stated and evaluated consistently by different parties—whether data producers or different data users—it might be argued that these criteria are not necessarily strictly subjective (although they *are* necessarily relative to the use at hand). It might therefore be more precise to call them “inter-subjective” to emphasize the fact that others besides the ultimate user may be able to recognize and agree on what these criteria should be for a given use; for simplicity, however, they are referred to as “subjective” here.

conclusion of the earlier evaluation. Even though every individual use of a database is potentially different, there *may* be occasions when a database is reused for the same purpose (e.g., where a given study or type of study is repeated under very similar conditions); particularly when such reuse is likely, users must be encouraged to record both their evaluation criteria and the results of their evaluations, even for databases that are rejected.

## 4.2 Phases of V&V and organizational commitment

The above arguments imply that data verification and validation should be split into two “phases” that can be performed by different parties at different times. In many cases it is likely that much of the required verification of a database can be performed by its producer. However, only *objective validation* can be performed by a database producer; the *appropriateness* of a database for a user’s intended purpose (i.e., its subjective validity) must be evaluated by the user (who may also perform additional objective validation).<sup>30</sup> In both phases, whenever V&V is performed on a database, some authority may “certify” the fact that it has been performed and interpret the results of this V&V to evaluate the suitability of the database for some purpose or range of purposes, recording this certification in the database’s metadata.

To summarize, it is useful to distinguish between “producer” VV&C and “consumer” VV&C, where the former is done by the producer of a database, using metadata to specify criteria for V&V that can be performed independently of any specific use of the database, while the latter is done by the user of a database in order to evaluate its appropriateness for some intended purpose.

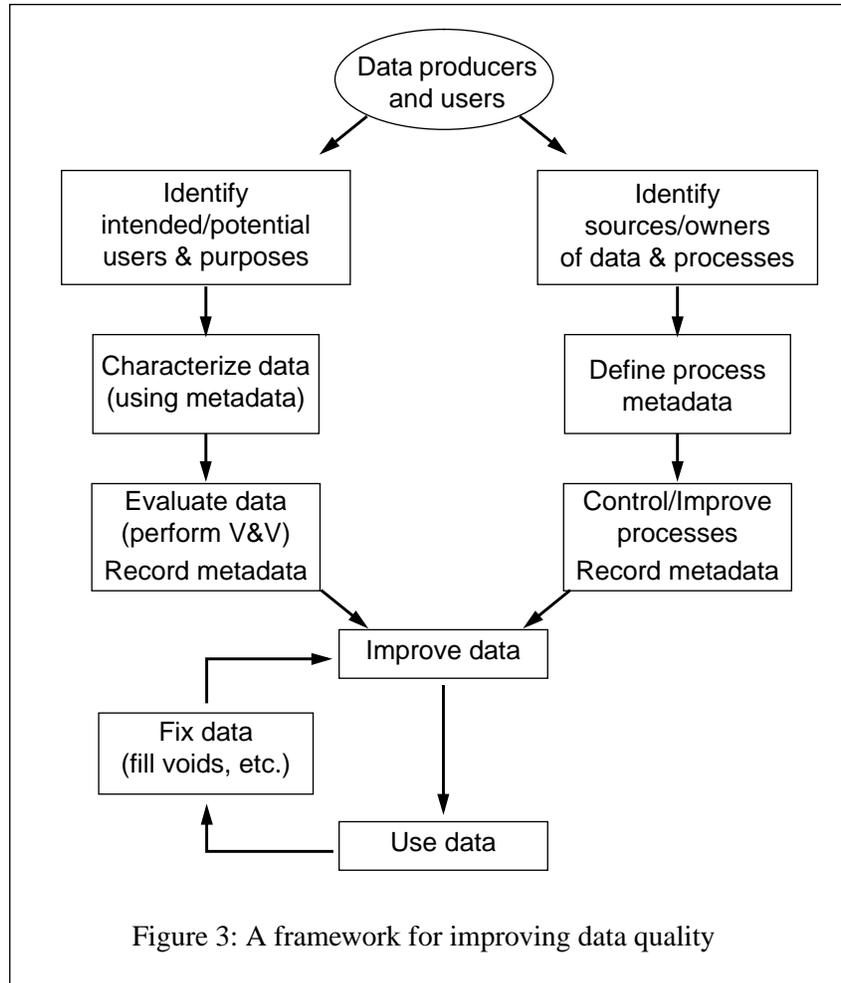
It is crucial that both producer and user organizations commit to performing the VV&C required by this approach. Without such commitments—along with the appropriate allocation of effort and funds and the establishment of suitable incentives within each organization—the enterprise described herein cannot succeed. Well-meant intentions can only be realized if “implementation” issues are given serious attention, especially in situations where existing organizational incentives are likely to thwart efforts to improve data quality.

---

<sup>30</sup> In this regard, it may also be useful to consider models and simulations as “users” of data. One might hope that a model’s purpose would be better defined than that of its human user, but this is rarely the case. In fact, models suffer from the same problem as data: their purposes are rarely made explicit. This makes it difficult to subdivide the problem of validating data with respect to a human model-user’s purpose into the obvious subproblems of (1) validating the model with respect to its user’s purpose, and (2) validating the data with respect to the model’s purpose. Since this attractive strategy is unavailable, data validation must be performed by viewing the model and its user as a single composite entity and asking whether the use of the data by this entity is valid. Unfortunately, this greatly complicates the problem, since it requires not only understanding the human user’s purpose but also understanding the applicability (and therefore the workings) of the model that the human employs in a particular case. Unfortunately, this validation process must be repeated for every combination of new user, purpose, model or simulation, and database.

## 5. A FRAMEWORK FOR IMPROVING DATA QUALITY

An overall framework for improving data quality should include parallel efforts to perform VV&C and to improve the processes that affect data. This is illustrated in Figure 3. This figure is *not*



intended as a flowchart of the activity to be undertaken by an individual data producer or user in evaluating data: rather, it attempts to show the parallel processes that should be followed by a community of producers and users in order to improve the quality of their data. In particular, the top left activity (“Identify intended/potential users & purposes”) should not be interpreted as something that the designer of a new database might undertake but rather as an action to be undertaken by an entire data community in order to define appropriate metadata requirements for use in VV&C. Similarly, the top right activity (“Identify sources/owners of data & processes”) is not intended here as something that an individual user might undertake prior to performing an M&S project but rather as an action to be undertaken by an entire data community in order to identify which organizations perform data-affecting processes on a given database.

The key to both recording the specifications needed to perform V&V and recording a full history of VV&C activities on a database is the definition and maintenance of appropriate metadata. Similarly, it is necessary to define and maintain appropriate metadata to support and record the results of process control.

The actual VV&C that must be performed on data may be divided into producer and user VV&C as suggested above. Every opportunity to verify and validate data should be considered, recognizing that the cost of this activity must be balanced against its perceived benefits (i.e., its potential to improve data quality and the contribution that this can make to the overall modeling activities in which the data may participate). Producers and modifiers of databases must be given the resources and mandate to perform VV&C along with their other responsibilities. Users must be encouraged, motivated, and ultimately funded to perform user VV&C as an integral part of their using—or even considering the use of—a database. Guidelines and tools for aiding the V&V process should be developed, supported, publicized, and made widely available.<sup>31</sup> Database management systems should be modified to integrate these guidelines and tools, so that users can perform V&V as a natural concomitant of creating or using a database.

The data-affecting processes that generate, modify, transform, and propagate data should be examined, controlled, and improved whenever possible, in order to improve the quality of the resulting data. One major challenge to data quality is to propagate and transform metadata as necessary when deriving new databases from existing ones. This is an important aspect of controlling the processes that affect data (including metadata): to the extent that this can be automated or aided by automated tools, it will greatly reduce the cost (and so improve the viability) of the data quality approach presented here. However, automation of this kind must be employed with caution, since improperly transforming metadata may corrupt it, and corrupted metadata may be worse than no metadata at all.

The following two sections discuss the metadata required to improve data quality. Later sections expand on the above discussion of data VV&C and how to control and improve the processes that affect data.

## **6. METADATA TO SUPPORT DATA QUALITY**

There are many different kinds of metadata that can be useful for improving data quality. As discussed above, the motivation for adding metadata to a database is that the information represented by the data *per se* is simply insufficient for many uses. A data value that consists of the number “3” for (say) the number of wheels on an airplane omits far more than it contains. For example, it does not tell us what is meant by a wheel (is it, for instance, equivalent to a tire, or are wheels in one-to-one correspondence with tires?), it does not tell us whether this is a normal or exceptional value, it does not tell us how certain or reliable the value is or who entered it or when it was entered, it does not tell us whether the value has been checked by anyone, and if so by whom and when, etc. A missing or null value (or “data void”), whether represented by a zero, a blank, or

---

<sup>31</sup> See Section 8 below for further discussion of V&V and tools to support it.

some other “null” indicator, typically tells us even less: it does not tell us whether there *should* be a value present, whether the value is missing because it is inapplicable, because it is unknown, whether it is a known exception or error, etc. In short, a database without metadata is often of limited value and most likely of indeterminate quality.

The purpose of this section is to describe the full range of metadata categories that would enable the development of a data quality profile to support data quality measurement and improvement. Whereas a “profile” suggests something that contains the *results* of evaluating the quality of the data in a database, this is possible only if the database also contains all of the metadata needed to *perform* such evaluations in the first place. Therefore, in order to develop the metadata requirements for a data quality profile, it is necessary to discuss a much broader range of quality metadata, of which only a subset will constitute the profile *per se*. Though practical considerations may limit our ability to generate all of the kinds of metadata discussed here for particular databases, it is important to identify these categories if only to map out the full space of quality-related metadata. Only after this space has been articulated will it be possible to make informed decisions about which categories are of the greatest value in specific cases.

I refer to a specific, populated instance of a database as an “instance database” or “dataset”. It is also useful to introduce the term “use-instance” to denote a specific occurrence of using an instance of a database (i.e., a specific dataset) for some purpose; this provides a way of distinguishing between a potential *kind* of use of a database (which I call simply a “use”) and an actual instance of using a database. Different users (or even a single user) may employ a given database for similar purposes (i.e., the same *kind* of use) on different occasions, where each such employment represents a distinct use-instance, uniquely characterized by a specific user, dataset, version, purpose, and time of use of the database.<sup>32</sup>

It is helpful to discuss metadata at three distinct levels: the database level, the data-element (or data dictionary) level, and the data value (or instance data) level. All of the metadata at the top level describes the database as a whole, whereas metadata at the lower two levels can be thought of as being logically replicated for each data item in the database. In addition, some of the metadata at each level (including the top) can be thought of as being replicated for each specific *use* of the database or even for each specific *use-instance* of the database: much of the evaluation of a database (or an instance of a database, i.e., a dataset) is meaningful only in the context of some specific intended use or of some specific use-instance. For example, a given data value may be appropriate (e.g., accurate enough or at the right level of resolution) for training purposes but not for analysis, resulting in different evaluations for these different uses. Alternatively, a given value may be appropriate (for the same kinds of reasons) for one training exercise but not for another (or for one analysis study but not for another), resulting in different evaluations for these different use-instances.

---

<sup>32</sup> To make certain that this distinction is clear, consider the following analogy: The “uses” (i.e., kinds of use) of a car include shopping, commuting, racing, etc., whereas individual “use-instances” would include going to a grocery store on a specific occasion, driving to work on a specific day, and racing in a particular race.

At the top (database) level—and to some extent at the middle (data-element) level as well—there are two distinct *kinds* of metadata, namely that which describes the abstract database (in the absence of any instance data) and that which describes an instance of the database (populated with instance data). At the lowest (data-value) level, the first of these kinds of metadata makes no sense: since the data-element level already serves as an abstraction of the data-value level, all abstract metadata pertaining to the data values should appear at the data-element level, leaving no abstract metadata at the data-value level.

The metadata requirements for each of these levels are first presented in outline form (subsections 6.1, 6.2, 6.3), where each item corresponds to a category of metadata. This is followed by a more detailed discussion of each category (in subsection 6.4). This discussion does not attempt to introduce a data model for the required metadata, since that would be premature: it is necessary to obtain consensus on the need for metadata and the appropriate categories before attempting to define specific metadata elements, as required for a data model. I present only those categories of metadata that are relevant to data quality. All of the categories of metadata discussed here are necessary either for evaluating or recording data quality; those used to *report* data quality results (and provide the minimum context necessary to understand those results) constitute the data quality profile *per se* and are marked with “\*” to show where the quality profile fits in the universe of quality metadata. The distinction between the data quality profile (discussed in Section 7) and this larger universe of quality metadata is not a rigid one: the quality profile is simply a *view* (in the database sense) into this universe, which can be extended as required.

Note that many of the quality metadata items described here are qualitative in nature (e.g., textual). It should not be surprising that the evaluation of quality is necessarily somewhat qualitative. This must not be confused with the objectivity of quality assessments: an assessment may be objective despite its being qualitative, and making an assessment quantitative does not prevent its being subjective. Nevertheless, to the extent that quality assessments can be made objective and (secondarily) quantitative, they are likely to be of greater value.

The final subsections of this section discuss a number of issues concerning metadata, including the need for tools to facilitate the generation of metadata and keep metadata and data synchronized with each other (subsection 6.5), ways of mitigating the storage and transmission requirements for metadata (subsection 6.6), and the need for mechanisms to allow the evolution of the metadata structure described herein (subsection 6.7).

## 6.1 Database level metadata

- \* • Overview of DB
  - Description & meaning of DB
  - Intended use/range of purposes and constraints of DB
  - Requirements for access and use
  - Description of and rationale for structure/design of DB
  - Global relationships to other DBs
  - Update-cycle information for the DB
- \* • Source information for the DB
  - Source & source credibility
  - Classification, accessibility, reproducibility information
  - Release authority for DB
- Characterization
  - \* – Intended resolution (level of detail) and rationale
  - Intended quality (accuracy, completeness, currency, etc.)
- Cross data-element information
  - Constraints
  - Distribution measurement information
- \* • Measured quality (overall *and* for each use)
  - Overall accuracy, consistency, completeness, currency, etc.
  - Clarity, flexibility, robustness of the DB design
  - Appropriateness for intended use
- Process control information
  - \* – Descriptions of (& references to) processes used to derive data (*and* metadata) in this DB
  - Rationales for choosing these processes
  - Agents responsible for developing, maintaining, & performing these processes
- Status/History/Configuration Management information
  - Overall current status of DB
  - Version history, times/sources of data/metadata modifications and  $\Delta$ s
  - \* – Usage (who has used the DB? for what? with what models? with what VV&C?)
- \* • VV&C audit trail

(Note: items marked with an asterisk also serve as part of the data quality profile *per se*.)

## 6.2 Data-element level metadata (data dictionary)

- Meaning of this data element & its meta-values & metadata
  - What this data element represents (and what it *doesn't*)
  - Meanings of nulls (unknown value, applicability of attribute, special values, etc.)
  - Meanings of exceptions, uncertainty metadata, etc.
- Source & update-cycle information for this data element
  - Allowing multiple sources with multiple, irregular update-cycles
- \* – Expected “degradation mode”
- \* – Classification, accessibility, reproducibility information, & release authority
- \* • Derivation/transformation information
  - Aggregation or other derivation information
  - Transformation process information
  - Process control data
- Constraints, relationships to other data/DBs
  - Including entity/attribute completeness, etc.
- Domain/datatype & units-of-measure
  - \* – Rationale for these & their portability, flexibility, etc.
  - Usage-specific restrictions of this element’s domain, including rationales
- \* • Resolution, precision, intended/expected accuracy
  - Including rationales, representation-dependence & portability
- \* • Appropriateness of this data element for intended use
  - Meaning, derivation, constraints, domain, resolution, intended accuracy, etc.
- History of changes
  - Audit trail of evolution of domain/type/units choices
  - Times/sources of data element modifications and  $\Delta$ s
- \* • VV&C audit trail
  - Concerning the appropriateness of this data element, its domain, type, units, etc.

### 6.3 Data-value level metadata

- \* • Quality (overall *and* for each use)
  - Accuracy, certainty (validation results)
  - Consistency (verification results)
  - Currency (expiration dates, “degradation modes”, etc.)
  - Appropriateness for intended use
  - Sources and quality of metadata
- Annotation
  - For caveats, special values or cases, etc.
- Source information
  - Source, derivation, time of generation/entry, etc.
- Next-source information
  - Describing when updates are expected (from where) & what they may offer
- \* • Derivation/transformation information
  - Aggregation or other derivation information
  - Transformation process information
  - Process control data
- Transformation audit trail
  - \* – How this value has been transformed
  - \* – Information on in-progress transformation transactions
    - Times/sources of data element modifications and  $\Delta$ s
- \* • VV&C audit trail
  - For VV&C that has been done on this value
  - Including “scope” of validation *and* of certification

### 6.4 Discussion of metadata categories

This section discusses each category of metadata in more detail, with one subsection describing each of the three levels of metadata (database, data-element, and data-value). A number of similar metadata categories exist at each of the three levels; when the meaning or use of these categories differs at each level, these differences are discussed below. In some cases, however, the meaning of a given category is essentially the same at each level: in these cases the category is discussed when first introduced, and later occurrences are simply described as having the obvious, analogous meaning. It should be kept in mind that metadata at the lower two levels is (at least potentially) replicated for each data item in the database, whereas metadata at the top level describes the database as a whole and is therefore not replicated within a given dataset.<sup>33</sup>

Some metadata items are *descriptive* whereas others are *evaluative*. A descriptive metadata item supplies an objective attribute of a data item (e.g., its source, datatype, meaning, objective accuracy, etc.), whereas an evaluative metadata item provides an evaluation of how appropriate a

data item is for some purpose. A descriptive metadata value may change—if the objective attribute of the data item it describes changes—but it will not change simply because the data item is being used for different purposes. Conversely, an evaluative metadata value may happen to be the same even when the data item it describes is used for different purposes, but it will generally be different for each such case, since the evaluation of the appropriateness of the objective attributes of a data item for different purposes will generally be different. “Evaluative” metadata is therefore necessarily “use-specific” metadata describing the appropriateness of a data item for a specified purpose.

#### 6.4.1 Database Level Metadata

This level of metadata describes the database as a whole. It consists mostly of textual documentation describing the intent, source, and various characteristics of the database and its design. It provides vital context for evaluating the quality of the database in terms of its source and its intended range of uses. While it may be unrealistic to expect metadata to provide *all* necessary context for evaluating the quality of a database, this should be thought of as an ideal to be approximated. It may always be necessary for users to seek subject-matter experts to help them evaluate the appropriateness of using specific data for a specific purpose, and it is certainly not my intent to suggest that metadata should be thought of as the *only* channel for communicating contextual information about a database, excluding the possibility of users contacting data providers directly to learn about their databases. But it should ideally not be necessary for users to be able to find others who are expert *in the use of a specific database*. That is, a database should ideally contain enough context to allow a subject-matter expert to evaluate its appropriateness for some purpose without additionally requiring that expert (or the user) to be expert in the idiosyncrasies of the database itself. In practice, it may be difficult to eliminate the need for such idiosyncratic knowledge entirely, but this should be the ultimate goal.

As mentioned above, metadata at this level may fulfill either of two roles, namely describing the database in the abstract or describing some particular (populated) instance of the database (i.e., a specific dataset, sometimes called an “instance dataset”). Where this distinction is not stated below, it is implied that whenever a specific instance dataset differs from the database in the abstract or warrants further description, metadata for this instance should be provided. Note, however, that the notion of an instance of a database (i.e., a dataset) need not imply a synchronized, static collection of data values corresponding to a specific “version” or “release” of a database. On the one hand, a given database design might represent scientific data collected from a particular kind of experiment: each time this experiment is run, it generates a new dataset, consisting of a static, populated version of this database, in which all values are synchronized. On the other hand, some databases are far more dynamic than this: their values may be updated on a continuous basis, with or without synchronization, producing no discrete “versions”. Overall or aggregate metadata for

---

<sup>33</sup> The top-level metadata describing a database *may* have to be replicated across different *instances* of the database (i.e., datasets), because different datasets may have different, specialized database-level attributes. This implies that somewhat different versions of the database-level metadata for a given database may exist in different instance datasets; however, a given instance database will never contain more than a single set of database-level metadata.

such dynamic databases may be relatively meaningless; metadata for any individual data value in such a database may be quite distinct from the corresponding metadata for other data values in the database (i.e., for other entities, or rows) and may not “inherit” from metadata at the database or data-element (column) levels: for example, the time of entry would be different for data values entered at different times. With this caveat, I nevertheless use the term “instance database” or “dataset” to include dynamic databases of this sort, since a given database *design* may be instantiated as a number of different “instance databases” each of which may be static or dynamic.

Although datasets are often obtained from data sources or suppliers, users may themselves supply or modify datasets under certain circumstances. Users often supply missing data, correct apparent errors, or even supply intentionally artificial data for a variety of legitimate (and questionable) purposes. For example, when running training exercises or games, it is often deemed necessary to modify data to keep an exercise on track, produce a pedagogically desirable result, or incorporate the knowledge or opinions of experts engaged in the exercise. Similarly, analytic studies often require the generation and maintenance of multiple base cases and variant “excursions” from these base cases, for analyzing alternative scenarios, performing sensitivity analysis, etc. In all such cases, users become data suppliers, and they must take responsibility for updating appropriate metadata to record the fact that they have made such changes; this should include metadata indicating their own assessment of the quality of their changes and providing the information needed for others to evaluate this quality independently. If a modified database and its metadata are to be subsequently made available to the original data supplier or other potential users, procedures must be developed for communicating in this “upstream” direction. Similar considerations apply to reporting data or metadata errors detected by database users.

- \* • Overview of DB

- Description & meaning of DB
- Intended use/range of purposes and constraints of DB

This should provide an overall, textual description of the database, including a discussion of its intended range of appropriate uses and any constraints on its intended use (e.g., “not to be used for navigation”). Particularly for constraints, it is helpful to provide some rationale or explanation, to ensure that the constraint is understood and not ignored inadvisably. As the database evolves, this description should be updated as necessary: any mismatch between the documented intended meaning of the database and a user’s intended use of it should be cause for concern (rather than being ignored on the basis that the database description may be out of date). This description should include a discussion of each specific dataset (populated, instance database) for which this database design is used; as discussed above, each instance of a database may be static or dynamic, and this aspect of each instance database should be documented as part of its description. These are probably the most important items of quality-related metadata that can be attached to a database, so the accuracy and completeness of this information are vital; however, in many cases this documentation need not be extensive and may range from a paragraph or two of text to several pages.

– Requirements for access and use

This should provide general information that a user needs to know in order to determine whether this database is likely to be accessible and usable, in order to make a quick determination as to its suitability for a given purpose. Access information should identify the owning agency and point of contact (POC) providing phone and FAX numbers, e-mail and postal addresses, etc., as well as identifying the classification level, whether the database is compartmented or restricted to government or contractor users, what restrictions apply to its access and use (such as the need for any special permission to use the database), and any copyright or foreign distribution requirements or constraints that apply to it. Information about how to use the database should include an overview of any general or specific user requirements, such as whether any special software or hardware are required to use the data, whether pre- or post-processing is required, etc.

– Description of and rationale for structure/design of DB

Every database should include a description of its design and structure and a discussion of their rationale, relating them to the intended purpose and use of the database. The description should include such overall aspects as the language and format of the database. The rationale, among other things, serves as a consistency check against the discussion of intended use: both the intended use and the rationale for the design of the database should be updated whenever the database is extended or changed in significant ways. (Wherever the term “rationale” is used in this paper, it is intended to mean a discussion of why a given choice was made, including what options, if any, were considered, why they were rejected, and why the chosen option was considered the best one.) Furthermore, because such choices may become outdated as conditions and assumptions change, it is important that additional metadata be associated with each argument, decision, assumption, or line of reasoning in a rationale, specifying when it was deemed relevant (i.e., currency metadata) by whom (i.e., source metadata), and with what certainty (certainty metadata). In some cases, a rationale will consist of a collection of choices, all sharing a uniform time of entry, source, and certainty; but it is important to allow adding or retracting arguments in support of previous decisions (in the manner of truth maintenance or argumentation support systems), if initial conditions become invalid, thereby undermining initial assumptions.

– Global relationships to other DBs

This should provide an explicit discussion of the overall relationship of this database to any others. Additional metadata (described below) specifies the actual derivation process by which data in this database may have been generated using data from other databases, so the discussion here should be at a conceptual level. That is, it should explain any semantic and/or historical relationships between this database and any others, making clear whether the relationship is expected (or required) to continue to hold true. For example, a database may be derived from an older database which it supplants (in which case continued correlation between the databases is not expected), or it may be intended to represent a dynamic summary or “view” of data in another, dynamic database (in which case the two databases may be required to remain “in sync”). This discussion should

include any intended aggregation or other level-of-detail relationships between this database and any others. This metadata may be of particular relevance in evaluating the appropriateness of using a database for some specific purpose.

- Update-cycle information for the DB

Many databases are revised on a regular (or irregular) basis. Different levels of update may be performed by different sources or agents, and some revisions may affect only certain subsets of the data in the database. The discussion here should explain how often, how regularly, and how extensively the database is expected to be updated. This has some overlap with “currency” metadata discussed below, but the emphasis here is on giving an overview of when, how, and by whom the database is revised or reissued, rather than on how current it may be at any given time; this should also give a potential user an overview of when (and how extensively) the database can be expected to be revised next.

- \* • Source information for the DB

- Source & source credibility

Source information is of particular interest to the Intelligence community, but all users should be concerned with the source of any database they intend to use—and particularly with the credibility of that source. Credibility may be difficult to quantify, but even a qualitative, textual discussion of the credibility of the source of the data in a database can be very helpful. The credibility of the source of a database is an important aspect of its quality. Note that the source and credibility of this source and credibility metadata are also important: this is one category of metadata that is potentially recursive—who has certified a given source as being credible, and is that certifier credible?

The source of a database need not denote a single agent or organization. Some databases may exist in different versions (i.e., be “poly-instantiated”), resulting in multiple sources, and different data items in a database may have different sources. In addition, there is often a distinction between the *immediate* source of a database and its *ultimate* source. For example, the immediate source of a database that is derived from other databases (via aggregation, selection, or other transformations) would be the organization that performed this derivation, but the ultimate source of the original data that was transformed may be a different organization. Even if no significant transformation is performed, intermediate data suppliers may collect and distribute data from various sources, and this process may be repeated at several levels. Furthermore, the “ultimate” source of a database may be construed as the originator (generator) of the data or as the organization that takes ultimate responsibility for maintaining the database, where these often differ. A metadata supplier must therefore decide whether the “source” for a database should be the immediate source of the data or its ultimate source (in either of the two senses above) or even the entire chain of intermediate sources between the ultimate source and the immediate source. Arguments can be found for any of these alternatives, and the correct choice may depend on the particular database in question. In some cases, the immediate source of a database may be incidental, whereas in some cases the

ultimate source may be unknown; in general, the immediate source is likely to be known with a high degree of confidence, whereas the ultimate source may be ambiguous or conjectural. In some cases, recording an appropriate point of contact in the source organization may be more useful than attempting to capture a complex sequence of sources; note however that such information (or any information about sources) may become outdated as organizational roles, telephone numbers, and locations change. Attempting to record chains of intermediate sources may be especially likely to result in outdated information, since this amounts to encoding in the metadata for a database the organizational relationships among stakeholders for that database, which may well be more volatile than the database itself.

- Classification, accessibility, reproducibility information

Detailed information as to formal accessibility should be included here. These characteristics of the database may bear only indirectly on its quality, but they are relevant if only as an indication of whether the database can be (or is likely to have been) exposed to public review.

- Release authority for DB

This is a formal requirement for classified databases, but it is useful information for any database, even if the “release authority” is informal. Knowing who has responsibility and authority for releasing data may help the user assess its quality and understand (or find out) anything about the database that is missing from its metadata.

- Characterization

- \* – Intended resolution (level of detail) and rationale

This should describe the intended overall level of resolution of the data in the database—including the rationale for choosing this level, in terms of the stated purpose of the database and its design, source, and relationship to other databases. To the extent that the database *cannot* be characterized as having a single, uniform level of resolution, this lack of consistency should be made explicit and justified in terms of the intended uses of the database.

- Intended quality (accuracy, completeness, currency, etc.)

This should be a statement of intention, against which actual accuracy, completeness, currency, etc. can be measured. (The meanings of these terms are discussed below.) From the point of view of the database designers and maintainers, the quality of the database should match (or exceed) its intended quality. From the point of view of the users, the intended quality may be of less importance than the measured quality, but the intended quality nevertheless indicates a likely upper bound on the level of quality of the database, since effort (and funding) are unlikely to be expended to achieve a higher level of quality than intended.

- Cross data-element information
  - Constraints
  - Distribution measurement information

This describes consistency constraints across data elements and statistical checks to be applied to distributions of values across different data elements in the database. (Metadata for such checks applied to distributions of values of *single* data elements should be specified at the data-element level.)

- \* • Measured quality (overall *and* for each use)

This is an overall assessment of the quality of the database. It should summarize the measured quality of the individual data items in the database, as well as the results of any overall evaluations that may have been performed (such as statistical tests of distributions of data, consistency checks among related data items, spot checking for accuracy of data values, etc.). It is possible to assess the quality of a database *either* with respect to some specific use or range of uses of the database *or* in general. For example, if the database has been used for some number of specific purposes (e.g., as input to various modeling studies), each such use-instance will be documented in the usage metadata discussed below; a separate quality assessment for each use-instance should be recorded here and linked to its corresponding usage history. The quality metadata for a database will therefore in general consist of a number of replicated sets of metadata, each set corresponding to one historical use-instance of the database. In addition to this usage-linked quality information, there should generally also be an overall quality assessment of the database, which should include a summary of the quality assessments of all of its use-instances.

- Overall accuracy, consistency, completeness, currency, etc.

The term “accuracy” is used to mean a measure of how well a datum matches some real-world entity or phenomenon that it is intended to represent; this forms the basis for its *objective validity* (as discussed above). In some cases it may be possible to provide numerical bounds on the measured accuracy of all data items in a database; failing this, a qualitative assessment of the overall accuracy of the database should be given (possibly broken down into subsets of the data, which may have different accuracy). When accuracy cannot be measured objectively at all, evaluation by a subject matter expert may have to be used instead. Whereas accuracy involves a comparison of data values with external entities, “consistency” is a result of *verification* efforts, which measure how well different data items within a database agree with each other or with specified criteria (such as falling within expected ranges, belonging to acceptable enumeration sets, or other constraints). Such criteria and constraints may be different for different instances of a database (i.e., different datasets) and for different use-instances of a given dataset: for example, a given usage may impose additional constraints or restrictions on the allowable values for a data element. At the database level, consistency assessments include aggregate evaluations derived from measurements of specific data items *and* overall measures of consistency across data items

(standard database terminology refers to such specified consistency criteria as “business rules”). As with accuracy, consistency may vary for different subsets of data in the database.

The “completeness” of a database is an assessment of how much of its intended content is present; evaluating completeness is another aspect of verification. Metadata at this level should provide an overview of how much null (or “void”) data is present in the database, in order to support alternative “voids management” strategies. Either data producers or users can identify missing data and take action to supply missing values, e.g., by seeking to find or generate correct values, interpolating, or “faking” data in some way. Database-level metadata should record the fact that such activities have been performed, whereas lower-level metadata should supply more detailed information about what values have been added to the database.

The “currency” of a database is an assessment of how up to date it is (which again may be different for different subsets). I exclude “relevance” from this “overall” list, since that is by definition relative to a user’s intended use of the database.

This database-level metadata must also specify any desired consistency measures *across* the above metadata measures for different instances or versions of the database. For example, if the measured accuracy of two different versions of the same database differs by more than some specified variance limit, this might alert a user (or the maintainer) of the database to problems with the data or with the validation process used to measure accuracy.

The actual measures of accuracy, consistency, completeness, etc. may result from multiple V&V processes, so they must be multi-valued. Beyond this, they are left unspecified at this point.

– Clarity, flexibility, robustness of the DB design

At the database level, it is appropriate to evaluate the design of the database itself with respect to its stated purpose and intended use. This involves assessing the clarity and appropriateness of the design, as well as its flexibility and robustness across likely extensions of its intended purpose. Although it is possible for a rigid, inextensible design to satisfy a specific intended use, it is generally preferable for a database design to allow for growth and modification of its original purpose. This information should include a discussion of (and rationale for) the normal-form of the data, and any relevant data modeling information, including the design’s relationship to the DoD Data Model and any use of extended form (e.g., for pointers, algorithms, etc.). Note that meta-attributes such as clarity, flexibility, and robustness are inherently qualitative (though not necessarily subjective). Although this information may not be of interest to a user who anticipates using a database only once, it may be relevant to a user who is contemplating the use of a database over an extended period of time or for an open-ended range of uses; it may also be of interest to a maintainer or manager of database who is trying to assess its value in terms of its potential applicability and longevity.

- Appropriateness for intended use

This should provide a high-level evaluation of how appropriate the database is for its stated purpose and intended range of uses. Since this assessment may be quite different for different uses, it should be stated separately for each specific use, and it should be explicitly linked to the usage metadata described below; only if the intended use of the database is quite narrow will it be meaningful to provide a single, overall evaluation of its appropriateness.

- Process control information
  - \* – Descriptions of (& references to) processes used to derive data (*and* metadata) in this DB
    - Rationales for choosing these processes
    - Agents responsible for developing, maintaining, & performing these processes

This should provide a high-level discussion of the processes that are used to derive, generate, collect, and transform the data (and metadata) in this database. In addition to documenting these processes *per se*, it is important to document the rationale for choosing each such process, so that these can be understood and evaluated against the stated purpose of the database (as well as against a given intended use). Documenting the parties responsible for choosing and developing these processes facilitates evaluating these choices, both by making apparent the implicit goals and needs of the agents involved and by allowing direct contact with those agents to clarify the rationale for any choice that may be inadequately documented. (It is important to include agents and processes that apply to *metadata* in this category as well as those that apply to data *per se*.) This information is of particular relevance in evaluating the appropriateness of using a database for some specific purpose.

- Status/History/Configuration Management information
  - Overall current status of DB

This should be a concise, high-level statement of the condition of the database, indicating whether it is in transition, how stable it is, and what expected future changes will affect it. For databases that are updated and reissued on a periodic (or ad hoc) basis, this should include “configuration management” information that explains how versions are maintained and by whom, as well as references to descriptions of any standard methodology or software used for version control. Such information will be database-specific, but it may be useful to attempt to develop or adopt a standard scheme or formalism that could describe the status of the majority of databases of interest.

- Version history, times/sources of data/metadata modifications and  $\Delta$ s

Explicit version history should be maintained, showing which agents revised the database at which times and what kinds of changes they made. Changes to structure, content, or meaning of both data and metadata should be described at a conceptual level, so that a user can evaluate the semantic impact of any revisions. Note that the impact of a revision is not necessarily highly correlated with

the number of data items changed: for example, restructuring or redefining the meaning of data items may not involve any changes to data values at all.<sup>34</sup> At the database level, the crucial need is to capture a qualitative sense of the extent and import of a revision in semantic terms.

Although the primary purpose of this metadata is to record official changes to a database by the agency or organization that owns and has responsibility for maintaining it, this can also be used to record any changes made by a user organization, i.e., when customizing a database for a specific use or when correcting errors found in a copy of the database (whether or not this is associated with a specific usage of the database). Whenever a user finds errors in a database, established procedures should allow notifying the owner of the database, but the user's copy of the database should in any case be annotated to show what modifications were made, by whom, when, and why; in addition, errors should be analyzed to determine whether additional V&V may be warranted to catch similar errors in the future. Whenever relevant, all change information (whether it is the result of customization or of correcting errors) should be linked to usage metadata, to allow reconstructing the motivation for making such changes and the conditions under which they were made.

\* – Usage (who has used the DB? for what? with what models? with what VV&C?)

This should provide as complete a usage history of the database as possible, including a point of contact for each instance of use (use-instance) and a high-level description of what the database was used for, how it was used, what VV&C was performed specifically for this use, and how appropriate the database was ultimately found to be for this use. The VV&C information stored here overlaps with the VV&C audit trail (below) to which it should be linked.

\* • VV&C audit trail

A high-level audit trail of VV&C should be maintained for the database as a whole, describing the history of quality assessment efforts applied to this database, and allowing non-binary, conditional and qualitative certification results to be recorded. This information should be linked to the usage history described above as well as to VV&C audit trail information at the data-element and data-value levels, both to avoid redundancy and to provide three distinct—but consistent—levels of VV&C metadata, ranging from overall assessments of the quality of the database as a whole to specific results for individual data items.

#### **6.4.2 Data-element Level Metadata (Data Dictionary)**

This level of metadata describes data elements and their *possible* values—not their *actual* (instance) values. This provides the semantics for interpreting data, as well as the rationale for

---

<sup>34</sup>The metadata for a data field named “size” might be changed to reflect the fact that the size field had in fact always represented volume but had previously (and erroneously) been described as representing area. This changes the *interpretation* of every value of this data field without changing any of the values themselves.

representing real-world entities or phenomena in terms of particular data items. It characterizes the sources of specific data items, explains how they are derived, provides consistency criteria and other constraints, and makes explicit any relationships between data items in the database and other data items either within the same database or in other databases.

In general, the purpose of this level (corresponding to a data dictionary) is to provide an abstract description of each item that appears at the data-value level. Nevertheless, as mentioned above, metadata at this level may still serve either of two purposes, namely describing a data element in the abstract or describing some particular instance of a data element (i.e., in a specific dataset). For example, a given data element (viewed abstractly) might be capable of naming any country, whereas its use in a specific instance of the database may be restricted to naming countries in a particular geographical area. It could be argued that “instance” metadata of this kind should be stored at the data-value level; however, it seems to make more sense to keep this kind of instance metadata at the data-element level, at least when it corresponds to related abstract metadata at this level (such as in the above example, where it describes the domain of a data item). This points out that while descriptive metadata at the data-element level (e.g., domain information) *usually* remains the same for a given data element across all uses, it *can* change if a particular use specializes or modifies the domain of a data element, its entity/attribute completeness, or some other objective attribute. On the other hand, evaluative metadata—which describes the appropriateness of a data item for a specified purpose—here as elsewhere, is always use-specific.

In addition, since each data element definition corresponds to an “attribute” or “column” of a relation, each data-element level metadata value supplies a logical “default” or “inheritable” value for the corresponding metadata for each instance (data-value level metadata) of this attribute within a given instance of the database. For example, if the metadata for a data element specifies a source for a given attribute in a given instance database, this source metadata can be thought of as being inherited by every value of this attribute in this instance database.

For standard data elements (SDEs), some data element metadata may come from the DDDS (Defense Data Dictionary System, as described in [9]) or from some other standard data dictionary. Further, many of the metadata items described here for a given data element will derive (or “inherit”) their values from row (entity) metadata values in an instance database. For example, if the data (and metadata) items for a given entity (represented by a given row) in an instance database come from a single source or were entered at the same time, then this source or timestamp metadata would be inherited by all data and metadata attributes (columns) of this entity (row).

- Meaning of this data element & its meta-values & metadata
  - What this data element represents (and what it *doesn't*)
  - Meanings of nulls (unknown value, applicability of attribute, special values, etc.)
  - Meanings of exceptions, uncertainty metadata, etc.

This describes the semantics of a data element, explaining what it is intended to represent (and what it is not intended to represent, if this is at all ambiguous, which it often is). This should include a discussion of the meanings of any null or other exceptional values for this element (different

kinds of null values may represent such things as whether the value of a given instance of this element is unknown, unknowable, of unknown knowability, of unknown applicability, known to be inapplicable, etc., whereas other exceptional values may imply that the value of a given instance is a known exception, an error, etc.). While the specific null and exceptional values allowed for this data element should be described under its domain (discussed below), the discussion here concentrates on the *meanings* of the various kinds of nulls and exceptions represented (rather than on the domain values used to represent them).

- Source & update-cycle information for this data element
  - Allowing multiple sources with multiple, irregular update-cycles

This refines the source and update-cycle metadata for the database as a whole: it focuses on the source and revision of a particular data element, which may be different for different data elements within the database. Different levels of revision may occur, corresponding to more or less complete revisions by more or less authoritative sources or agents. For example, a database may be “cleaned up” (e.g., by having consistency checks applied to it) more often than it is regenerated from its source; regeneration may result in more up-to-date values, but it may introduce new errors or inconsistencies as well. This metadata must allow for multiple sources (authoritative & other) having multiple, irregular update-cycles.

As with the corresponding database-level metadata, the distinction between this and “currency” metadata is that this gives a potential user an overview of how often this data element is likely to be revised and, in particular, of when (and how extensively) it can be expected to be revised next.

- \* – Expected “degradation mode”

Metadata for each data element should include information as to the “mode” in which values of that data element are expected to degrade over time: some values become continuously less accurate or less meaningful as they age, whereas others remain entirely valid until they “expire” (i.e., when some event changes the reality which they represent). This “degradation mode” metadata should be recorded here at the data-element level if it applies equally to all instance values for the given data element in a given instance dataset (and for a given use-instance), but it should be recorded at the data-value level if it is specific to a particular instance value of this data item.

- \* – Classification, accessibility, reproducibility information, & release authority

This is analogous to the equivalent metadata at the database level but applied to the data element, if the data element differs from the database as a whole in any of these attributes.

- \* • Derivation/transformation information
  - Aggregation or other derivation information
  - Transformation process information
  - Process control data

This specifies whether and how values for this data element are derived from other data, and it describes any transformations that are applied in generating this data element. The discussion here should describe any aggregation or other derivation method used to generate this data element, and it should refer to any other data values used in this derivation. The derivation process itself should be described in full mathematical detail, and each source datum used in the derivation should be identified by its meaning, its source (e.g., another database), its name, expected range of values, etc. If a data value is computed as the result of running some program (including, but not limited to, some model or simulation that may use other data as input), then the version and source of that program *and* of its input data must be documented here. Similarly, any transformations that are normally applied to this data element should be described in full detail, and any relevant process control information for this data element (such as historical values or statistical distribution information about process control parameters) should be recorded here as well. Any derivation or transformation process that is too complex to be fully described in simple mathematical form should be documented by a complete process description: this may be supplied in place, either in text or as a formal process model (using a standard process modeling methodology, such as IDEF0), or it may refer to an external description of the process in some appropriate form or publication. This metadata is of particular relevance in understanding and evaluating the appropriateness and quality of a data element. Note that this information may be different for different instances of a given data item in different datasets.

- Constraints, relationships to other data/DBs

This metadata describes any constraints that apply to this data element, beyond those implied by its domain and datatype. In particular, this is the place to document any desired consistency constraints involving this data item, including what are often referred to as “business rules” for the database; such constraints might involve relationships between this data item and others in this database or in other databases. For example, if a data item represents an aggregation of other data items (whether in this database or some other database), this constraint should be stated here, giving a complete specification of how “aggregation” is to be performed in this case. Metadata at this level may also describe consistency or statistical checks to be applied to distributions of values of single data elements (i.e., distributions of values in columns). Constraints and relationships should be described in text, along with their rationales; they should also be stated formally, in terms of mathematical relationships or references to procedures or programs used to enforce them. Ideally, constraints and relationships should be represented in a declarative, machine-interpretable formalism, to facilitate automated consistency checking. In particular, if automated or semi-automated tools are used to perform consistency checking, then the consistency constraints and relationships used by those tools should be derived directly from these metadata descriptions, in order to avoid the configuration management problem of having to maintain consistency between

the documentation of those constraints (in metadata) and their implementation (in the rules used by the tools).

- Including entity/attribute completeness, etc.

Each data element should be characterized in terms of its “completeness” in both of two senses. A data element representing some real-world entity is “entity complete” if every real-world instance of this entity is represented in the database (that is, if the database includes a row containing an instance of this data element for every instance of the entity in the real world). This says that the database models a “closed world” (with respect to this data element) in which every entity is known; if a data element is *not* entity complete, then the database models an “open world” (with respect to this data element). Independently of this, a data element is “attribute complete” if every instance of the data element (which in general represents some attribute of some entity) has a value. This says that there is a value for this attribute (column) for every row in the database. If a data element is *required* to be attribute complete then it is called “obligatory” (as opposed to being optional). These two kinds of completeness information should both be present for every data element. The first of these (entity completeness) cannot be verified by reference to the database alone—it must be validated with respect to the real world—whereas the second (attribute completeness) can be verified simply by checking that each such attribute in the database has a non-null value. Entity completeness is therefore a validity constraint, which appears in the database in the form of a metadata assertion about the relationship of the database to the real world, whereas attribute completeness is a computable feature of a given dataset; that is, a metadata assertion that a database should be attribute complete can be verified by checking for non-null values of that attribute in a given dataset, without reference to the real world. Each of these types of completeness therefore requires two metadata items: one to specify whether the database is *intended* to be complete in the given sense and the other showing the result of checking (by validation or verification, respectively) whether the database is *in fact* complete in the given sense.

- Domain/datatype & units-of-measure

Units of measure should be stated explicitly for any data element for which this may be ambiguous. In addition, I distinguish between the “domain” of a data element (which is the possibly infinite set of allowed values for the data item) and its “datatype” (which is the type of value allowed). For example, a data element representing 3-character country codes has the type “text” (limited to 3 characters), whereas its domain is the set of legal country codes, rather than all possible 3-character strings. The more informative (i.e., restrictive) this metadata can be made, the more useful it is for automated data checking; it may therefore be helpful to introduce a third concept in addition to the “base type” of an element and its domain. The base type (e.g., text, integer, real) specifies the underlying value set from which a data element’s values are drawn, which is generally too unrestrictive to be of much help in verifying data. Similarly, if the domain of a data element is interpreted as the universe of values from which the data item’s value is drawn (e.g., all possible country codes), this may still be too unrestrictive, since, for example, a particular instance of a database may be restricted to codes representing countries in a specific geographical area.

Each domain should be defined (and *named*) so as to make its intended meaning clear: ideally, data elements whose values are drawn from a single (semantically defined) set should use the same domain, whereas data elements whose values are drawn from a semantically different set should use different domains. The criterion for whether two sets are semantically different is not simply whether they contain the same values: a given set of values (such as “1, 3, 5”) may be used for many semantically different purposes, e.g., as arbitrary identifiers (such as part numbers), odd numbers, numbers differing by 2, etc. The notions of “name-equivalence” and “subtypes” from programming language theory are useful in this regard. Name-equivalence treats two potentially-distinct collections of values as being the same only if both the names of these collections *and* the values they contain are the same: that is, two collections containing the same values are considered different if and only if they have different names. A domain definition should consist of a descriptive name intended to convey the semantics of its values (e.g., OddIntegers, PartNumbers, EuropeanCountryCodes, etc.); this name should be used everywhere that the same semantics are intended (so that data elements using “1, 3, or 5” to represent selected odd integers would use the domain OddIntegers, whereas other data elements using “1, 3, or 5” as part numbers would use the domain PartNumbers). Similarly, the notion of “subtypes” (or subdomains) should allow defining domains that are subsets of other domains, such as EuropeanCountryCodes as a subtype of CountryCodes. Unfortunately, the need for overlapping, semantically distinct sets can get quite complex, resulting in a proliferation of subdomains. Rather than registering all such variant subdomains as standards, they may be represented as use-instance metadata at this (data-element) level.

This metadata may be different for different instance databases, e.g., when a data item has a more restricted domain in one dataset than in another. Variations like these are represented explicitly by the “usage-specific restrictions” metadata subcategory below.

\* – Rationale for these & their portability, flexibility, etc.

The rationale for the choice of units, datatype and domain for a data element should be discussed in sufficient detail to allow a potential user of the database to evaluate these choices, particularly with respect to their portability and flexibility (i.e., for unanticipated uses of the database). This metadata is of particular relevance in evaluating the appropriateness of using a database for some specific purpose. Note that meta-attributes such as portability and flexibility are inherently qualitative (though not necessarily subjective).

– Usage-specific restrictions of this element’s domain, including rationales

It is often useful to restrict the domain of a data element for a specific use, without redefining the data element. For example, if a data element represents country codes, a specific use might focus on a particular geographic area, for which only a subset of all country codes would be legal. Rather than restricting the definition of the domain for this data element (which would then make the database inappropriate for use over less restricted geographic areas), this metadata category provides a separate place to record such usage-specific restrictions, as well as their rationales. In

addition, it may be useful to keep historical usage data here, recording when specific restrictions are used for specific purposes.

- \* • Resolution, precision, intended/expected accuracy
  - Including rationales, representation-dependence & portability

This should describe the resolution (level of detail) and precision (number of significant digits in numerical values) of this data element, as well as its intended and expected accuracy. The discussions of resolution and intended accuracy here refine those of the database as a whole. In addition, this level of metadata should discuss any representation issues (such as precision limits imposed by field-length or encoding), and should make clear any potential portability problems introduced by both abstract and representation-dependent choices made here.

- \* • Appropriateness of this data element for intended use
  - Meaning, derivation, constraints, domain, resolution, intended accuracy, etc.

This provides an evaluation (as a result of user V&V) of how appropriate this data element definition is for some specific intended use. In many cases this will default to the higher-level evaluation of how appropriate the database as a whole is for this use, but it may conversely be useful to evaluate data-element level choices in order to arrive at the overall database-level evaluation of appropriateness.

- History of changes
  - Audit trail of evolution of domain/type/units choices
  - Times/sources of data element modifications and  $\Delta$ s

At this level, a history should be kept of any changes to the definition of this data element, i.e., its type, domain, units, or meaning. The times and sources of any such modifications should be stored in this metadata, and the changes themselves should be recorded in sufficient detail to allow a user to evaluate the meaning and impact of these changes.

Just as at the database level, although the primary purpose of this metadata is to record official changes to a database by the agency or organization that owns and has responsibility for maintaining it, this can also be used to record any changes made by a user organization, i.e., when customizing a database for a specific use or when correcting errors found in a copy of the database (whether or not this is associated with a specific usage of the database). Customization for a specific use at this level should normally be represented by “Usage-specific restrictions” metadata (discussed above), but actual errors and any generic customization of data element definitions (e.g., changes that would be shared by all users in a given organization) should be represented here. Although it may be relatively rare for users to find errors in data element definitions, whenever a user does find such errors, established procedures should allow notifying the owner of the database, and the user’s copy of the database should be annotated to show what modifications were made, by

whom, when, and why; in addition, errors should be analyzed to determine whether additional V&V may be warranted to catch similar errors in the future. Whenever relevant, all change information (whether it is the result of customization or of correcting errors) should be linked to database-level usage metadata and data-element level usage-specific restriction metadata, to allow reconstructing the motivation for making such changes and the conditions under which they were made.

- \* • VV&C audit trail
  - Concerning the appropriateness of this data element, its domain, type, units, etc.

At this level, VV&C audit information concerns any evaluation that has been performed on the data element definition itself, i.e., its type, domain, units, and meaning. This information should be linked to the usage history described above (both at the database level and the data-element level) as well as to VV&C audit trail information at the database and data-value levels.

### 6.4.3 Data-value Level Metadata

This level of metadata describes actual instance values of data, including any annotations or comments about exceptional values, missing data, etc. A given instance database is by definition a single version of the database as a whole, but it may contain individual data values derived from different sources or different generation cycles. Metadata at this level documents a specific version of a specific dataset. The emphasis here is therefore on the quality and specific characteristics of the data values that are present (or missing) in this dataset. In recognition of the fact that data values do not always adhere to the expected characteristics of their corresponding data elements, it is important to allow metadata at this level to replicate and specialize the corresponding data-element metadata for a given data value (for example, specifying some additional restriction on the domain of a given attribute for a given entity), as well as allowing ubiquitous free annotation at this level.

Metadata for a given data value may in some cases be inherited from the corresponding metadata for the entity (row) or attribute (column) of this data value. In practice, this should greatly reduce the amount of metadata required to characterize the quality of a dataset.

- \* • Quality (overall *and* for each use)

This metadata category provides *measured* quality of individual data values, i.e., the degree to which they satisfy the accuracy requirements, constraints and other relationship specified by their corresponding data element definitions and the database as a whole. In addition, this must include measurements of the *appropriateness* of these data values for their intended use. Although it is desirable to make these metadata items as concrete and quantitative as possible, quality assessment may inherently involve qualitative judgements and evaluations; therefore, it is important to allow textual annotation of all metadata items in this category to make sure that any necessary context or explanation can be recorded.

A given instance data value may be used on a number of occasions for a number of different purposes, or “use-instances” as discussed above. Just as it is possible to assess the quality of a *database* either with respect to some specific use or in general, so it is also possible to assess the quality of a data *value* in either of these ways. For example, a data value may have a specific, measured, objective accuracy, which would be the same for all use-instances, but the *appropriateness* of this level of accuracy may be different for each use: that is, the quality of this same objective accuracy may be “good” for some uses and “poor” for others. Therefore, a separate quality assessment of each data value for each use-instance should be allowed here, linked to its corresponding usage history at the database and data-element levels. The quality metadata at the data-value level will therefore in general consist of a number of replicated sets of metadata, each set corresponding to one historical use-instance of a given instance dataset. In addition to this usage-linked quality information, there should be an overall quality assessment of the values in this instance database, independent of (and/or summarizing) the quality assessments for some or all use-instances of the dataset; it may be feasible to derive this overall assessment automatically from individual use-instance assessments.

– Accuracy, certainty (validation results)

This should provide one or more measures of the accuracy of this data item, i.e., its correspondence to the real-world entity or phenomenon that it is meant to represent, as evaluated by some objective validation effort. This should also include some measure of the certainty of these accuracy measures. This is the “ground truth” on which all higher-level quality evaluations of the objective validity of an instance database must ultimately rest. There may be multiple measurements of the accuracy of a given data item, corresponding to different validation techniques or distinct validation efforts; each such measurement should be explicitly linked to the particular validation process (whether automated or manual) that produced it. Accuracy should always be thought of as metadata that refers to something outside the database itself, whereas certainty should be thought of as metadata about the accuracy metadata; certainty is therefore a measure of the quality of the accuracy assessment itself. This metadata provides merely the “bottom line” result of any accuracy evaluations; the evaluation procedures themselves are documented by other metadata.

– Consistency (verification results)

This measures the consistency of this data item with any constraints or other relationships specified for it at the data-element or database levels, as evaluated by some verification effort. There may be multiple measurements of the consistency of a given data item, corresponding to different verification techniques or distinct verification efforts; each such measurement should be explicitly linked to the particular verification process (whether automated or manual) that produced it. As with accuracy metadata, this provides merely the “bottom line” result of any consistency evaluations; the evaluation procedures themselves are documented by other metadata.

- Currency (expiration dates, “degradation modes”, etc.)

This provides information about the currency of instance data. There are two aspects of currency: information about when a data value was created and information about how long it can be expected to remain valid. In general, it is not possible to tell how current a data value is simply by knowing how long ago it was created; some old values may remain valid indefinitely, whereas some recent values may become obsolete very quickly. Currency metadata should therefore include timestamps for when a data item was originally generated (if known), when it was first added to this database, and when it was last updated or validated.<sup>35</sup> But it must *also* include information about when a data item can be expected to become obsolete (i.e., an “expiration date” or “last valid date”) or when it is expected to be superseded by newer data. Finally, currency metadata for a data item should include information as to the “mode” in which the data item is expected to degrade over time: some values become continuously less accurate or less meaningful as they age, whereas others remain entirely valid until they “expire” (i.e., when some event changes the reality which they represent). This “degradation mode” metadata should default to the value recorded for it at the data-element level if it applies equally to all instance values for the given data element in this instance dataset (and for this use-instance), but it should be recorded here at the data-value level if it is specific to a particular instance value of this data item (for example, instance values from different sources might degrade differently). That is, degradation mode metadata may attach to an attribute (column) or to an individual cell (item), depending on whether individual instance values degrade differently.

- Appropriateness for intended use

This provides an evaluation (as a result specifically of *user* V&V) of how appropriate this data item is for some specific intended use of this dataset. In some cases, this may default to the corresponding evaluation at the data-element level for this data item, which evaluates whether this data item is appropriately *defined* for a given use-instance. However, the appropriateness of the measured accuracy, certainty, consistency, currency, etc. of an instance value may become apparent only as the result of user V&V applied to an instance dataset: the result of this evaluation should be recorded at the data-element level if it applies equally to all instance values for the given data element in this instance dataset (and for this use-instance), but it should be recorded here at the data-value level if it is specific to a particular instance value of this data item. For example, if user V&V finds the measured objective accuracy of the values of a given data element in a given dataset to be generally appropriate for a given use-instance, this should be recorded at the data-element level; but if this objective accuracy differs significantly across individual data values for this data element (i.e., for different entities in the dataset), or if the objective accuracy is constant but the *appropriateness* of this accuracy is different for different entities, then these distinctions should be

---

<sup>35</sup> Even if the time of generation or entry for a data item is not known precisely, it will often be possible to specify earliest or latest bounds on these unknown times; for example, the earliest time at which any metadata values are entered for a data item should provide a latest bound on when the data item itself was entered (assuming the metadata is not entered prior to entering the data item itself), just as the date-and-time of entry provides a latest bound on the date-and-time of generation for any item.

recorded at the data-value level. Note also that the appropriateness of a value may be affected by the *certainty* of its accuracy or of some other metadata.

– Sources and quality of metadata

This describes the sources and quality of metadata values themselves: it is vital information, without which a user may have little reason to trust the quality metadata in a database. This is analogous to (and should have the same form as) the source and quality metadata described above for ordinary data.

Recalling the recursive aspect of metadata, any metadata item is itself a data item, which should be described by metadata at the data-element level and which has an instance value at the data-value level of the database. As discussed above (in Section 3.5) this can be somewhat confusing, since every metadata value, whether it is at the database, data-element, or data-value level, is itself an instance value, i.e., an instance of an abstract metadata element, as described at the (meta) data-element level (that is, the level at which metadata data-elements are defined). Ordinary data values have a similar duality, in that they are instances of abstract data elements that are described (by metadata) at the data-element level, but instance values of ordinary data items are found only at the data-value level, whereas instance values of metadata items are found at all three levels.

As a simple example, consider a metadata item that consists of a timestamp (T1) specifying (for example) when some data value (D) was last validated. Since T1 is an attribute of the instance value D (not of the data element description of D) T1 should be considered data-value level metadata. Since the timestamp is a (meta) data item in its own right, it must be described by additional metadata at the (meta) data-element level, specifying the datatype, units of measurement, precision, etc. of T1. Now suppose it is also important to know when the timestamp T1 was itself entered into the database (this could be useful for verifying the currency of the database). This would require a new timestamp (T2) that serves as metadata for the first timestamp (T1). T2 would be described by its own metadata at the (meta) data-element level, though much of its description at that level would be identical to that of T1. (T2 and its data-element description metadata at the (meta) data-element level can be referred to as meta-metadata, but for most purposes, it is less confusing to refer to meta-metadata simply as metadata.)

Fortunately, this recursion is easily terminated, since it is rarely meaningful to have metadata describing metadata beyond one extra level (as in this case)<sup>36</sup>. Nevertheless, it *is* often important to

---

<sup>36</sup> Furthermore, it should not be necessary to provide data-element level meta-metadata within the database, since the structure of the metadata itself and the rationale for this structure (which is what would be described by such meta-metadata) would be the same for all databases that conform to the quality criteria discussed herein. That is, this memorandum itself serves as the rationale and context for providing the quality metadata items it proposes: any database whose metadata structure conforms to this logical model can simply reference this memorandum (or subsequent documents or specifications derived from it) as the source for the meta-metadata context describing that metadata structure. Ideally, a single such quality metadata structure (with possible variants and options) should be adopted for the vast majority of all M&S databases, so that this single reference should serve as a common source of data-element level meta-metadata for all databases adhering to this common quality metadata structure.

have this second level in order to evaluate the metadata in a database. It is possible that not *all* metadata values will require second level meta-metadata of this kind, but all crucial, quality-related metadata should be described by meta-metadata that describes the source and quality of the metadata: knowing the source and quality of a quality assessment makes it much more meaningful.

- Annotation
  - For caveats, special values or cases, etc.

One of the shortcomings of many databases is their lack of provision for adding comments. While this capability should ideally be available everywhere in the database (on all data and metadata items at all levels), it is especially crucial for quality-related metadata. It is suggested above that all metadata items in this category allow textual annotation, but it may also be useful to have special metadata dedicated to any overall comments or qualifications pertaining to the quality assessment of the instance data in a database.

- Source information
  - Source, derivation, time of generation/entry, etc.

This refines the source and generation metadata at the database and data-element levels: it focuses on the source and revision of a particular data value, linking this to corresponding information (stored at the data-element level) about the kind of revision that produced this value (i.e., whether this value was produced by a more or less complete revision performed by a more or less authoritative source or agent). As is true for the corresponding database and data-element level metadata, there is some overlap between this metadata and the currency metadata for this data value, the emphasis here being on the source and derivation of the data value.

- Next-source information
  - Describing when updates are expected (from where) & what they may offer

This also overlaps with update-cycle metadata at the database and data-element levels, as well as with the currency metadata for this data value. The focus here is on the source and time of the *next* expected update for this data value; this gives the user a context within which to evaluate the currency of the data value and facilitates making informed decisions about whether and when to wait for the next revision of a database. This information should ideally specify whether the next update is expected to provide a newly-generated value for the given item or simply a possible revision of the current value, based on error-checking, user-feedback, etc. (This distinction has to do with whether or not a new data-collection/measurement effort is anticipated for this item: if it is not, then the next revision should be interpreted as a refinement or correction of the current value rather than a new value representing a new observation of the real world.)

- \* • Derivation/transformation information
  - Aggregation or other derivation information
  - Transformation process information
  - Process control data

This refines metadata at the data-element level: it describes the derivation and transformation processes that produced this data value, which may not be identical to the intended processes for this data element. In addition, it supplies any specific statistical information or history that describe the processes applied to this data value, which again may be distinct from the corresponding information at the data-element level.

- Transformation audit trail
  - \* – How this value has been transformed
  - \* – Information on in-progress transformation transactions
    - Times/sources of data element modifications and  $\Delta$ s

This overlaps derivation/transformation metadata to some extent but focuses instead on the history of actual transformations that have been applied and the resulting changes that have been made to this data value, including information about any ongoing transformation processes that may be expected to affect this value. A history of previous values and the changes that have occurred can be a great help in performing sanity checks and in evaluating the stability and evolution of a database. Changes should be described semantically, in terms of their significance and impact, which is often not apparent from raw values themselves; reasons for changes (e.g., detection of previous errors, redefinition of the basis for computing a value, etc.) should be recorded here whenever possible.

Just as at the database and data-element levels, although the primary purpose of this information is to record official changes to a database by the agency or organization that owns and has responsibility for maintaining it, it can also be used to record any changes made by a user organization, i.e., when customizing a database for a specific use or when correcting errors found in a copy of the database (whether or not this is associated with a specific usage of the database). Whenever a user finds errors in a database, established procedures should allow notifying the owner of the database, but the user's copy of the database should in any case be annotated to show what modifications were made, by whom, when, and why; in addition, errors should be analyzed to determine whether additional V&V may be warranted to catch similar errors in the future. Whenever relevant, all change information (whether it is the result of customization or of correcting errors) should be linked to database-level usage metadata and data-element level usage-specific restriction metadata, to allow reconstructing the motivation for making such changes and the conditions under which they were made.

- \* • VV&C audit trail
  - For VV&C that has been done on this value
  - Including “scope” of validation *and* of certification

At this level, VV&C audit information concerns all evaluation that has been performed on this data value. This information should be linked to the usage history and VV&C audit trail information described above at the database and data-element levels. VV&C information should include a discussion of the *scope* of whatever validation has been performed, i.e., how extensive and intensive a validation effort has been undertaken; similar information may be helpful for verification as well, but it is more crucial for validation, since verification can be characterized more easily and unambiguously by reference to verification criteria, constraints, and specifications within the database itself, whereas validation necessarily involves evaluating data values with respect to the real world. To put this another way, it is relatively easy to verify the verification process, since this should not require access to anything outside the database itself; but verifying the validation process necessarily requires access to external information, which may often be expensive or difficult to obtain. For this same reason, it is important to characterize the certification process, to provide credibility for whatever V&V has been performed on the database; in particular, it is important to specify the *scope* of the certification process, i.e., how carefully and exhaustively the certifying agent or agency examined the V&V process and its results before certifying that the claimed V&V was indeed performed and that its results warranted the certified conclusion that the database was found appropriate for a given purpose or range of purposes.

## 6.5 Tools for creating metadata

As suggested above, the cost of creating metadata may be greatly reduced by the design and widespread dissemination of tools that perform automated or semi-automated generation and capture of metadata at the time of data generation, editing, or transformation. The logical metadata structure presented above allows extensive use of “inheritance” of metadata for related data items and/or data items that are generated, edited, transformed, or entered at the same time. For example, in many cases, all data items entered in a given session will share most if not all of their metadata, except for any special cases, which should (hopefully) be apparent to the person performing the data entry. Data entry tools should be able to set such inheritance linkages by default: that is, data items entered by a given person in a given organization at a given time should all be linked to a common metadata description of this single data entry session. Such tools may even be able to help identify special cases as lying outside of specified bounds or differing in some other pre-defined way from the bulk of the data.

In addition, data entry tools may be able to collect useful statistics on the usage patterns of the metadata items themselves. This would help database designers analyze and evolve the metadata structures of their databases (see Section 6.7 below).

Finally, the use of automated tools for generating and maintaining metadata is probably the best way to ensure that metadata and associated data remain “in sync” with each other, i.e., that metadata will be updated automatically as needed to correctly describe changing data.

Tools of this kind would need to be linked to the metadata structure of the database for which data is to be entered. This should be done dynamically, by having the tools access a data model (data dictionary) for the metadata, so that a single tool could be used to enter or modify data for any variant of the metadata structure described here. This would allow these tools to be used across a wide range of database designs, as well as allowing for the inevitable evolution of the metadata structure of each database. Design criteria and abstract specifications for such tools should be developed and published, in order to encourage database management system software developers and database managers to develop appropriate tools.

## 6.6 Mitigating storage and transmission requirements for metadata

The amount of metadata described above may appear overwhelming, but there are two factors that should help reduce the burden of storage and transmission of metadata to acceptable levels. First, there is considerable inherent redundancy in the metadata for a given database, as represented by the inheritance or defaulting of metadata values (e.g., for data items that share a common source or time of entry). This implies that many metadata items for many data items in a database will share common values. Implementations of this metadata structure should capitalize on this redundancy by using pointers rather than replicating values for shared metadata, wherever possible.<sup>37</sup> This would have two advantages: It would be a more accurate representation of the semantic relationships among metadata values in the database, showing when metadata items for certain data values inherit their values from the metadata of other data values; and it would drastically reduce the storage requirements for metadata, since many—or even most—of the metadata values in a given database are likely to be shared in this way.

The second mitigating factor is that it may be unnecessary to carry all of the metadata for a given database along with the database itself for all purposes. Although the collection of metadata for a database is logically an integral, inseparable part of the database, giving the database its context and meaning, not all of this metadata may be required for all purposes. For example, much of the metadata described above is designed to *enable* the evaluation of the quality of the data in a database: most of this is not needed by a user organization that is simply examining evaluations performed by previous producers or users (though this organization would still need to be able to add metadata giving its own evaluation of the database for its own purpose). Similarly, once a database has been evaluated and chosen for a particular use, most of the quality metadata for that database can subsequently be ignored by the user—unless using the data raises questions whose answers require access to the metadata. This suggests that the metadata for a database may be profitably divided into several segments, not all of which may be needed for any given purpose, thereby reducing transmission and storage costs. However, even metadata segments that are deemed optional for a given purpose must remain easily accessible on demand, in case questions

---

<sup>37</sup> Note, however, that these shared metadata relationships must be represented so as to be easily broken when distinct metadata values become necessary for data items that previously shared their metadata. For example, if a collection of data items are all entered at a given time, they may initially share a single metadata item specifying their time of entry; but if one of these data items is subsequently edited, its time of entry must suddenly take on a new, unique value, rather than pointing to the shared metadata for the initial collection.

arise that require access to them. A structure logically similar to that which is currently evolving for the World Wide Web would satisfy this requirement: hypertext links to metadata would make it easy to access remote metadata segments without having to copy them along with a database for all purposes.

## **6.7 Allowing the evolution of the metadata structure**

The quality metadata structure described herein must be recognized as an initial design. The lack of existing databases exhibiting such metadata and the attendant lack of experience in performing systematic data quality evaluation and improvement make it unrealistic to expect this to be the ultimate design for an appropriate quality metadata structure. The intent of this memorandum is to identify and describe those categories of metadata that appear most likely to help improve the quality of M&S data, but this initial design must be refined on the basis of experience. It is therefore important to establish mechanisms for generating feedback on the value and use of the metadata proposed above, so that this feedback can be analyzed from time to time and used to modify the metadata structures of existing and new databases, in whatever ways seem appropriate. One way to generate such feedback would be to perform periodic analysis of usage patterns of the metadata for a database. Unused or universally-defaulted metadata items may be candidates for elimination, whereas the heavy use of explanatory annotation or apparent work-arounds may indicate the need for additional or different metadata. As suggested above, data entry tools may be able to collect useful statistics on such metadata usage without adding to the data entry task.

## **7. THE DATA QUALITY PROFILE**

A data quality profile for a database should be thought of as a *view* (in the database sense) of selected metadata items from the above categories for that database, combined and presented in ways that facilitate assessing the quality of the database. Quality metadata should (a) allow potential users to evaluate the relevance and appropriateness of a database for some intended use; (b) allow data suppliers to evaluate the results of their data generation and transformation processes; and (c) help database administrators, maintainers, and owners evaluate the overall quality, utility, and value of a database for a given purpose or range of purposes. Of these three potential uses, the primary focus of the quality profile is (a), whereas (b) and (c) may require access to the larger universe of quality metadata discussed above. The following sections extract and reorganize selected metadata from this universe into a “data quality profile” view and discuss it from the perspective of the primary consumer of the quality profile—the database user.

### **7.1 Selected metadata categories for a data quality profile**

The quality profile consists of metadata from the following categories (extracted from the more complete list above). When all subcategories of a category belong in the profile, the category as a whole is shown without its subcategories, whereas when only some subcategories of a category belong in the profile, they are listed under the category:

### **Database level metadata for quality profile**

- Overview of DB
- Source information for the DB
- Characterization
  - Intended resolution (level of detail) and rationale
- Measured quality (overall *and* for each use)
- Process control information
  - Descriptions of (& references to) processes used to derive data (& metadata) in this DB
- Status/History/Configuration Management information
  - Usage (who has used the DB? for what? with what models? with what VV&C?)
- VV&C audit trail

### **Data-element level metadata (data dictionary) for quality profile**

- Source & update-cycle information for this data element
  - Expected “degradation mode”
  - Classification, accessibility, reproducibility information, and release authority
- Derivation/transformation information
- Domain/datatype & units-of-measure
  - Rationale for these & their portability, flexibility, etc.
- Resolution, precision, intended/expected accuracy
- Appropriateness of this data element for intended use
- VV&C audit trail

### **Data-value level metadata for quality profile**

- Quality (overall *and* for each use)
- Derivation/transformation information
- Transformation audit trail
  - How this value has been transformed
  - Information on in-progress transformation transactions
- VV&C audit trail

## **7.2 The data quality profile view**

Combining related categories from the different levels produces the following condensed view of the quality profile. For each new, combined category, applicable levels are shown in brackets, using the abbreviations DB (database level), DE (data-element level), and DV (data-value level). Note that all of the *results* of V&V are represented in the category “Measured Quality” whereas the rest

of the profile consists essentially of contextual information that is needed to help interpret the measured quality of the database.

- Overview of DB [DB, DE]
  - Description & meaning of DB [DB]
  - Global relationships to other DBs [DB]
  - Source, credibility, classification, accessibility, reproducibility & release authority [DB, DE]
  - Update-cycle & expected “degradation mode” information [DB, DE]
  - Intended resolution (level of detail) and rationale [DB, DE]
  - Rationale for domain/datatypes & units-of-measure & their portability, flexibility, etc. [DE]
- Measured Quality (overall *and* for each use) [DB, DV]
  - Overall accuracy, consistency, completeness, currency, etc. [DB]
  - Clarity, flexibility, robustness of the DB design [DB]
  - Accuracy, certainty [DV]
  - Consistency [DV]
  - Currency (expiration dates, “degradation modes”, etc.) [DE, DV]
  - Appropriateness for intended use [DB, DE, DV]
  - Sources and quality of metadata [DV]
- Status/History/Configuration Management information [DB]
  - Usage (who has used the DB? for what? with what models? with what VV&C?) [DB]
- Derivation/transformation information [DB, DE, DV]
  - Descriptions of (& references to) processes used to derive data (& metadata) in this DB [DB]
  - Aggregation or other derivation information [DE, DV]
  - Transformation process information [DE, DV]
  - Process control data [DB, DE, DV]
- Transformation audit trail [DV]
  - How this value has been transformed [DV]
  - Information on in-progress transformation transactions [DV]
- VV&C audit trail [DB, DE, DV]
  - For DB as a whole [DB]
  - Concerning the appropriateness of this data element, its domain, type, units, etc. [DE]
  - For VV&C that has been done on this value [DV]
  - Including “scope” of validation *and* of certification [DV]

### **7.3 Refining and using the quality profile**

The above view is intended to help a potential user (or other interested party) grasp the quality of a database or a specific dataset. It provides context to help a user understand the relevance, suitability, and appropriateness of a database for some intended use, while providing specific, quantitative evaluations of the objective quality of the data, as well as a comprehensive history of how the database has been used, how it has been derived and transformed, what its current state is, and who has evaluated it for what purposes and by what means. The larger collection of metadata defined in previous sections provides additional context, including the information needed to perform such evaluations in the first place. The quality profile itself is intended to be the minimum collection of information that conveys the quality of a database. The final specification of a data quality profile will be possible only after one or more pilot studies have been performed using the strawman profile outlined here; such studies should also help us understand how to use a data quality profile in the pursuit of improved data quality.

The most striking aspect of the profile is that it does not provide a simple “bottom line” quality measure. Evaluating the quality of a database for a specific intended use is not a trivial matter, and the complexity of the quality profile highlights this fact. However, the information required to build and maintain a quality profile for a database is neither obscure nor especially difficult to obtain (at least for new databases—legacy databases present additional problems<sup>38</sup>). If nothing else, the definition of a data quality profile and the broader range of quality metadata discussed above provide a set of criteria for improving the quality of the data we rely on when performing modeling studies for a wide range of critical purposes. Even if pragmatic considerations limit our ability to produce quality profiles that meet these criteria, it is important to define them as a first step toward improving data quality.

In addition to the fact that a data quality profile cannot provide a simple, scalar measure of the quality of a database, it is equally important to keep in mind that it is only one component of a data quality evaluation and improvement strategy. The generation and maintenance of the remainder of the quality related metadata for a database is equally vital, as are the other aspects of our overall strategy for data quality improvement: having producers and consumers (users) perform explicit VV&C, using metadata both to direct these activities and to record their results, and controlling the processes that affect data, to improve the quality of data generation, transformation, and transmission. The remaining sections return to these other aspects of data quality improvement.

## **8. DATA VV&C**

The first step in performing data VV&C is to develop criteria against which the data will be evaluated. It is meaningless to say that data will be verified or validated without specifying the criteria for these assessments. Since producer and user VV&C differ in certain key respects, the

---

<sup>38</sup> The requirement to generate metadata describing legacy databases should not be seen as making it more difficult to use such a database but rather as making explicit how difficult it already is to use it appropriately and safely for any purpose—or with any model or simulator—other than that for which it was originally intended.

criteria they develop will tend to be rather different. Moreover, different producers may have quite different criteria, as may different users.

Techniques for data VV&C should include generic methods applicable to most databases and specific methods evolved for particular kinds of data (and metadata), particular kinds of databases, or particular databases, as well as potentially different methods that are appropriate for data producers, maintainers, suppliers, and users. Ideally, these various methods should be collected and packaged as tools or tool sets for data VV&C. Such tools should ideally also provide ways of capturing anomalies, corrections, and annotations (whenever these occur) and channeling them back to the database provider or maintainer. Further experience with one or more pilot studies will be needed before producing an initial specification for such a tool set; however, it is possible to outline the kinds of VV&C techniques we might expect to find useful.

In addition, as mentioned above, it is crucial to secure commitments to perform VV&C from both producer and user organizations. Such commitments—as well as the attendant allocation of effort and funds and the establishment of suitable incentives within each organization—are essential to the success of data VV&C. The effectiveness of this “implementation” strategy will ultimately determine the outcome of the data quality improvement enterprise.

## **8.1 Verification techniques**

Data verification techniques essentially involve checking that a data value meets some specification, such as being of a required datatype, in a specified domain or range, or satisfying some kind of constraint or consistency check. Many database management systems include facilities for performing some checks of this kind (i.e., maintaining “referential integrity”), and there has been considerable work in recent years in building expert systems that can perform even more sophisticated checks. Case-based reasoning techniques may also be applicable here, since different consistency constraints and criteria may apply to a given database under different conditions (or for different datasets), as indicated by values in the database. Traditional statistical techniques can also be applied to this problem, for example to check that a value is within an allowed range of variation for a collection of other values (or of past values for the same data item). Often even trivial statistical techniques, such as looking for values that appear only once (or with very low frequency) in a given data field can identify anomalous data. Similarly, auto-correlation techniques applied to successive versions of a database can frequently detect errors that have been introduced by revision. Additional semantics can be supplied to produce more powerful constraints, for example specifying required or expected relationships among data fields or aggregates of values.

The use of metadata to support verification should make it easier for producers and users to incorporate additional specifications and constraints of these kinds, to be used by automated or semi-automated processes that can verify that data values meet these constraints. In addition, data visualization techniques can be employed to help both data producers and users see patterns in their data, find outliers, and generally use their own subject area expertise to verify that data values appear reasonable.

The Data Quality Engineering (DQE) tool (initially developed by the Marine Corps but extensively reengineered by USCENTCOM) is an example of a tool that performs semi-automated data verification. It has been designed to allow a data center to run a set of rule-based checks against each new version of a database to look for anomalies and report these back to the data supplier. In principle, it could also be run by a user, perhaps with a slightly different (expanded) set of rules. The rules are expressed in terms of SQL queries against the database, making it easy to check for things like missing values, values out of expected ranges, etc. In principle, these could be extended to allow sophisticated statistical checks on distributions of values for a given attribute (field) across all entities or to verify expected relationships between different fields of different entities. Note, however, that all such tests are verification checks: they do not directly support validation, i.e., checking whether a data value is accurate or correct, except in terms of whether it matches its expected type, domain, range, etc. Extending these rules to perform validation would require them to refer to external data values or other representations of reality outside the database itself. In addition, the mechanism for reporting errors back to the data supplier assumes that the database will eventually be corrected; but errors fed back to a supplier are generally not corrected until the next release of the database, when new errors may well be introduced with new data. From the user's perspective, this may not be enough: whoever performs these verification checks on the data should be able to annotate incorrect data items and optionally provide their own values for such items, to allow using the data as is, since any database must always be expected to have errors in it whenever it is used. Nevertheless, despite these limitations, DQE is an excellent—and so far, apparently unique—example of a tool that begins to address the need for semi-automated data quality evaluation.

## 8.2 Objective validation techniques

Unlike verification, which can be performed on the data in a database without regard to anything outside of the database itself (assuming that the database includes metadata supplying specifications for constraints, consistency checks, and so-called “business rules”), validation necessarily requires reference to the real world, outside the database. It is therefore more difficult to see what kinds of tools can help the validation process. Data validation, as discussed above, involves both objective and subjective facets. The objective facet of validation asks whether a data item correctly represents that aspect of the real world which it is intended to model (according to some implicit or explicit specification of how closely and in what way it is intended to model that aspect of reality). As discussed above (in Section 4.1) this kind of validation may be performed by a data user, but it can often be performed by the producer of the data, if the criteria for objective validation are explicit. If a map claims a stated positional accuracy for features of a certain kind, it should be possible for a data producer (or anyone else) to validate that claim by checking the map data against the real world.

As with validation of any scientific measurement or statement of supposed fact, absolute validation is not well defined.<sup>39</sup> For example, “validating” a measurement by repeating the same measurement process does not add as much confidence to the validity of the original measurement as does re-deriving the value by independent means. Performing a ground survey to confirm the accuracy of a geographical position that was originally derived from a satellite survey might provide more

confidence in the accuracy of that position than would reevaluating or even repeating the satellite survey: independent measurement techniques guard against systematic errors in the measurement process itself. Similarly, a value that was originally computed from theoretical “first principles” would be better validated by attempting to measure it in an experimental setting than by recomputing the theoretical value, even by means of a different computational process. As a final point on this subject, it is worth reiterating that “expert judgement” (or “subject area expertise”) should be considered a validation technique of last resort, and—especially if it is the only source of validation for a data value—it should always be regarded with suspicion and skepticism. Considerations such as these must become a routine part of data validation if the process is to have any real substance.

In addition (as argued above in Section 3.8), validation consists of more than simply comparing data values against other known values (or the real world). Data transformations themselves should be validated to ensure that they transform data in valid ways. This requires performing VV&A on all such data transformation processes.

### **8.3 Subjective validation techniques**

The subjective facet of validation involves evaluating the appropriateness of data for a given purpose (as discussed above in Section 4.1). This is most naturally performed by a user, though in some cases a data producer may be able to anticipate the purpose of some or even most users of a given database. Evaluating the appropriateness of data for a given purpose may actually involve additional objective measurement. For example, a user may need to determine whether a given data value is representative of real world values in the user’s domain or context: the answer to this question may be quite different from the result of objective validation performed in a less restricted domain or context. Yet even in the absence of any such reference to objective, real world values, the user must always determine whether a database contains values that are appropriate for the use at hand. In the context of performing some kind of study using a model or simulation, this may require a detailed understanding of the data requirements and/or modeling techniques employed by the model, as well as a thorough comprehension of the semantics of the study itself. There may not in principle be much that can be said about this process in the abstract: it may be necessary to rely on the user to apply whatever criteria are needed to determine whether a given database is appropriate for a given use. Nevertheless, the metadata associated with a database should provide sufficient context to allow an arbitrary user to evaluate the appropriateness of that database for an arbitrary purpose, and it should provide ways of documenting the subjective validation processes performed by various users, attaching these evaluations to the database for use by subsequent users, as well as by the maintainers of the database, who may want to improve its applicability.

In particular, it is necessary to provide metadata to represent user validation for each specific use of a database. In cases where a “use” corresponds to a “study” which may in principle be repeated

---

<sup>39</sup> Validation is generally understood as the failure of falsification. Most scientists subscribe to the view that an assertion is considered valid only to the extent that (a) it is in principle susceptible to falsification, (b) it has repeatedly exposed itself to falsification attempts, and (c) it has withstood all such attempts (so far).

sometime in the future (whether by the same users or by others), it is important to record sufficient information about the study itself and its director or other relevant parties to allow future users to contact these parties for elaboration of their experiences.

#### **8.4 Certification techniques**

It is possible to define certification as simply recording the fact that data V&V has been performed, but to be useful, certification should be more than just a V&V audit trail (which is already included in the data quality profile metadata). A more valuable use of certification is as an authoritative testimony to the effect that the data in a given database or dataset has been evaluated and found to be of appropriate quality for some more-or-less specific purpose or according to some stated criteria. At one extreme, this might be fairly independent of any specific purpose, i.e., stating that a database has been verified and validated in objective terms, according to criteria stated in the quality metadata for the database (qualified, if necessary, by additional specifications in the certification itself), with the caveat that this does not necessarily imply the suitability of the database for any particular purpose. For a generic database whose likely uses can be predicted with reasonable accuracy, certification might warrant the database as being appropriate for this predicted use (or any of a predicted range of uses), with the caveat that this does not necessarily imply the suitability of the database for any *other* purpose. Finally, at the other extreme, certification might warrant a database (or specific dataset) as being appropriate for a specific purpose, where this purpose is described in sufficient detail to allow future users to evaluate its relevance to their own needs.

In general, the form of a certification should be that of a logical argument in which the intended use is described first and is used to motivate the choice of V&V techniques that were applied to the data; the scope and results of this V&V are then described, analyzed, and summarized, leading to the conclusion that the database (or dataset) is (or is not) appropriate for the intended use. This conclusion should be qualified as necessary, and negative conclusions should be recorded as well as positive ones, as discussed above (see Section 4). Any necessary context or criteria for the conclusions reached should be included as part of the certification metadata. In addition, the credentials of the certifying agent or agency should be documented as well, along with a timestamp and any other contextual information that might help the user evaluate the relevance, currency, and credibility of the certification itself.

#### **8.5 Prioritizing VV&C**

Since the cost of VV&C may be significant, it is clearly important to establish priorities for when and how extensively to perform it. Ideally, this should be decided in a top-down manner, starting with the cost of various levels of VV&C and the benefits and risks that would be expected to result from “good” versus “bad” choices for the real-world decisions that are to be made based on the results of a given modeling study using a given database. Sensitivity analysis could then (again, ideally) be used to decide on appropriate levels of VV&C to be applied to various values or subsets of values in the given database, in terms of the impact that correct versus incorrect data values

might have on the real-world outcome. Unfortunately, performing sensitivity analysis of this kind is quite impractical in most realistic modeling efforts<sup>40</sup>, so expert judgement must generally be substituted for formal analysis. This means that there is currently little choice but to “eyeball” the potential impacts and risks associated with incorrect data values. The determination of how much VV&C is appropriate in a given case must therefore be based on little more than guesswork. Nevertheless, given limited resources, it is worth trying to perform *some* analysis, however informal, to attempt to apply the available resources for VV&C where they are likely to be most useful. Furthermore, as discussed above (in Section 3.4), it may be useful to analyze the cost of VV&C even though formal cost-benefit analysis may be infeasible. In particular, if techniques can be developed for performing VV&C at reasonable cost, it should be possible to justify a certain level of VV&C even without a rigorous analysis of cost-benefit tradeoffs.

## 9. CONTROLLING AND IMPROVING PROCESSES AFFECTING DATA

The second aspect of the approach recommended here seeks to improve the quality of data by improving the processes that generate and affect it. Data-affecting processes include all those that create or potentially change data, i.e., those that generate, modify, edit, aggregate, or derive data, and those that transform, transmit and propagate data for use in other databases or as input to models (where processes that propagate data may involve the use of models whose output is used as input data for other purposes). As discussed above (in Section 3.10), whereas validating data-affecting processes might in theory reduce the need to perform explicit VV&C, these processes may be harder to validate than the data they produce, and it is questionable whether data-affecting processes can ensure that data will be appropriate for a given user’s intended purpose if this purpose is unknown when these processes are created. Therefore, improving data-affecting processes cannot obviate the need for explicit VV&C (especially user VV&C); nevertheless, it is an important way to improve the quality of data in any database and to ensure that the benefits of VV&C are not lost when transforming or transmitting data from one database to another.

As is the case for explicit VV&C, process improvement also requires organizational commitment to be effective. The following sections outline an “implementation” strategy that attempts to address these issues as well.

Improving data-affecting processes can be thought of as performing VV&C on these processes themselves. For a given database of interest, this requires the following steps:

- Identify relevant processes throughout the life-cycle of the data
- Identify “owners” of data & processes
- Empower/facilitate/support process-control & improvement

---

<sup>40</sup> For a possible way around this problem, see [14]; for other discussions of sensitivity analysis see [2] and [17].

### **9.1 Identify relevant processes throughout the life-cycle of the data**

It is assumed that databases of interest will be chosen on the basis of their recognized importance or use in specific modeling studies of interest. The generic types of data-affecting processes are mentioned above, but the specific processes that affect a given database, dataset, or data value must be identified on an individual basis. It may be useful to develop a process model for a data-producing organization or for users of a given database, describing what they do with the database in question. A process model for a given organization may involve the use of many different databases, which may undergo shared or unique processing. This modeling process itself may produce tangible data quality improvement, since it may reveal redundant or conflicting processing of data that can be eliminated or combined to reduce inconsistencies in the data. Alternatively, a dataflow model may be produced for a given database, showing where its data values come from and how they are processed. Whatever modeling technique is employed, it must cover the entire life-cycle of the data, from generation or initial derivation, through any and all modification, revision, transformation, and propagation. Each data-affecting process should be analyzed to determine its likely impact on data quality, its potential for improvement, an estimate of the cost of improving it (if possible), and an estimate of the potential benefit of improving it, in terms of its effect on various uses of interest.

### **9.2 Identify “owners” of data and processes**

It is equally important to identify “owners” of both data and the processes that are applied to data. Process modeling or dataflow modeling may be helpful in revealing *de facto* ownership, but it is necessary to understand that this may not be synonymous with recognized authority. In one sense, true ownership of data (or of a process) cannot be arbitrarily “assigned” to an organization: it must be actively “accepted” by that organization, in order to be meaningful. Furthermore, an owner must not only *take* responsibility but must also *have* the necessary capability to administer the data or process appropriately and must be *given* the recognition, authority (including release authority for classified information), and resources to do so. Cases of multiple, conflicting, or overlapping ownership must be identified and resolved before proceeding beyond this stage.

### **9.3 Empower/facilitate/support process-control, redesign, and improvement**

Having identified appropriate databases, data-affecting processes, and owners, it is next necessary to empower those owners (or other agents, if appropriate) to design and implement process control and improvement techniques. This involves obtaining a consensus among the appropriate stakeholders (owners, users, funding providers) to agree that this is worth doing and then proceeding on a case-by-case basis. Here, as elsewhere, organizational factors must be managed appropriately to ensure that there is a true commitment to performing substantive process improvement.

Each data-affecting process should be analyzed to determine whether its improvement is warranted and is likely to be cost-effective; in particular, the methods used to generate data should be

examined to see if they warrant being subjected to formal validation. Each process that is chosen for improvement should then be analyzed to see what kind of process-control techniques can be applied to it and/or in what ways it can be redesigned to be improved. This process itself (redesigning and/or controlling a data-affecting process) should be subjected to VV&C to ensure that it is done appropriately and will in fact lead to improved data.

#### **9.4 Implement process management**

From a management point of view, the process of controlling and improving data-affecting processes involves a number of steps. The following is adapted from the corporate world (see Redman 1992) but should apply reasonably well to the DoD environment. It seeks to use standard process-control techniques to improve the quality of each data-affecting process. For each process identified by the procedure described above:

- Establish a process owner and management team
- Describe the process qualitatively
- Establish a measurement system
- Establish process control over the process
- Identify and select improvement opportunities
- Make and sustain improvements

Each of these steps will be discussed in turn:

- Establish a process owner and management team

The appropriate owner of a given data-affecting process may be obvious, or it may have to be decided by some appropriate authority, especially in cases where multiple agents or agencies have investments in the process. Once an owner is established, a “process management team” for the process should be established, consisting of members of the owner organization and possibly others from relevant organizations. This team must embody the functional expertise needed to analyze and modify the process as necessary: its members should understand the data and the specific processing required by the data-affecting process under consideration.

- Describe the process qualitatively

The process management team for a process should analyze the process and identify both the suppliers and the consumers (customers) of the data involved in the process. Ideally, this can be done by considering each data-affecting process to be a separable part of the overall processing that is applied to the given data from its generation to its ultimate use. By considering each such process in isolation, the team should be able to limit its analysis to the *immediate* suppliers and consumers of the data from the perspective of this process: for example, if a process is intended to transmit

data without modification, the needs of its “customers” can be defined straightforwardly as requiring that the data supplied to the process by its “suppliers” be received without change. The sequence of activities and processing involved in the process should be analyzed qualitatively but in detail. The needs of the data consumers (i.e., the “customer requirements”) should be clearly identified and mapped into the roles of everyone involved in the process, so that their activities can be derived directly from the customers’ needs. This analysis may suggest ways of re-engineering the data-affecting process under consideration, in which case this may be done before proceeding with subsequent steps.

- Establish a measurement system

In order to apply process-control techniques to the data-affecting process in question, it is necessary to find a way to measure the performance of the process. Common process-control wisdom suggests that the best strategy is to pick a small number of the most relevant aspects of the process to measure, rather than attempting to measure everything. In the case of data-affecting processes, these measures may reflect attributes of the process itself (e.g., the time taken to process a dataset or the number of queries of the input dataset required to produce the output dataset) or they may reflect attributes of the resulting data (e.g., its conformance to expected statistical measures or similar “verification” constraints). This measurement system is a crucial aspect of process-control, since it serves as an indicator of the health of the data-affecting process; it should therefore be allowed to evolve continuously in an effort to improve the control of the process in question. Both the choice of which attributes of the data-affecting process to measure and the techniques for measuring them should be subjected to continuous re-analysis. Since these measurements are not of intrinsic interest (being used merely as proxies for the health of the data-affecting process itself) it should be possible to change the attributes being measured or the ways they are measured without negatively affecting the measurement process: any measurement that better reflects the health of the data-affecting process in question is preferable to an inferior measurement. The measurement system can and should therefore be improved continuously.

- Establish process control over the process

The measurement system should be used to implement standard process-control techniques, essentially establishing a statistical measure of whether the data-affecting process is performing as expected. Standard 3-sigma variation or some other criterion can be used, as appropriate, to determine when the process is performing within acceptable limits. Whenever the established criterion is violated, the data-affecting process is identified as potentially in need of improvement (though a particular violation may, upon further analysis, turn out to be simply an unusual but legitimate fluctuation in the performance of the process).

- Identify and select improvement opportunities

At this point, if not before, cost-benefit analysis should be used to determine which data-affecting processes have the greatest potential impact on the quality of the decisions that are to be made

based on studies using the data in question. The implicit prioritization resulting from this analysis must be combined with the results of process-control monitoring to determine which data-affecting processes offer the best potential return for improvement. This final prioritization should ideally be done by a high-level team consisting of representatives of the process management teams that have analyzed each of the data-affecting processes under consideration. The result should be a set of focused improvement projects (or tasks) with quantitative goals for improving specific data-affecting processes.

- Make and sustain improvements

Once the above projects or tasks have been chosen, individual “improvement teams” should be formed for each such project, to attempt to improve the data-affecting processes identified as most worth improving. It is vital that the necessary authority, resources, and direction be supplied to these teams by some appropriate agent or agency, and that their progress be supported and monitored by this agency. If the goals of each team have been correctly formulated in quantitative terms, it should be possible to measure their progress toward improving the quality of the processes that affect the data in question.

## **10. STEPS TOWARD DATA QUALITY**

The approach described above is hypothetical and must be refined through experience before it can be recommended for widespread use. It would therefore be best to try this approach out in one or more “pilot” projects to evaluate and improve the quality metadata framework in general, the data quality profile in particular, and the data VV&C and process control strategies outlined above.

To this end, one or more candidate databases should be identified for pilot data quality improvement projects. An ideal candidate database would be one for which it is relatively easy to identify clear-cut “customers” (users), purposes, owners, and sources of the data involved and for which specific producer and user organizations are willing and able to work together toward data quality improvement. In addition, the generation and manipulation processes for this data should not be too complex or involve too many different participating organizations, and the update-cycle of the data should be relatively short, to allow seeing the effect of applying process control to its data-affecting processes for two or more cycles, within the duration of the pilot project. The database should also be amenable to clear-cut, identifiable V&V techniques, and it should be ripe for quality improvement (i.e., there should be some reason to think that its quality might be improved). Furthermore, the database should be representative of other databases of interest, to improve the chances that the results of the pilot project will generalize to other cases. Finally, the quality of the chosen data should have an identifiable impact on the quality of some modeling or simulation effort that is of recognized importance.

Having chosen a suitable database for a pilot study, the procedures discussed above should be applied in an attempt to improve its quality. Customers for this data should be identified, and their potential purposes (i.e., uses for the data) should be enumerated. Owners and sources of the data

in the database should be identified, as well as any other stakeholders who may have an interest in the data or play a role in its production, transformation, or consumption.

Once the database owner and customers have been identified, all of the processes that generate and manipulate the data in this database should be identified, along with the owners of these processes and any relevant relationships among them. This may be done by some combination of dataflow analysis (performed from the perspective of the database itself) and process modeling (performed from the perspective of the owner or customers of the database). This analysis may suggest ways of re-engineering the generation or propagation of this data, which may short-circuit the pilot project to some extent, skipping some of its subsequent steps; this would presumably be a salutary effect of the pilot project, and it should be recognized as a positive outcome of the data quality improvement process, despite the fact that it may make it impossible to assess the full potential of the rest of that process. Whether or not such re-engineering is performed at this stage, one or more specific data-affecting processes should be identified as candidates for improvement.

The chosen database should be examined in terms of the above analysis to identify applicable V&V techniques that can be used to evaluate the quality of the data. Using the strawman metadata model presented above, a specific metadata model should be defined for the database and should be populated with metadata to permit the application of the identified V&V techniques. A specific VV&C plan should be formulated at this stage, to schedule initial V&V on the database. At the same time, an overall process management plan should be formulated for the database, identifying data-affecting processes that should be examined and controlled, while addressing organizational issues such as authorizing, empowering, and supporting responsible parties in this pursuit.

With the metadata structure, VV&C plan, and process management plan in place, the actual evaluation and improvement of the database can proceed. Initial producer and user V&V should be performed on the database to produce a “baseline” evaluation of its quality. The results of this (and subsequent) V&V should then be used to improve the quality of the data in the database, correcting any errors or anomalies found, while recording the results of V&V in metadata and certifying the database as appropriate for its intended use. In parallel with this, process control and improvement should be applied to those data-affecting processes identified as good candidates for improvement. The effects of process control and improvement may not be apparent until data has gone through the affected processes several times, so the final evaluation of the pilot project should be performed only after a “steady-state” result has been achieved.

A final evaluation of the pilot project should be conducted by repeating the initial producer and user V&V and comparing the results with those of the original baseline. In addition, qualitative evaluation of the approach should be performed by all parties involved in the data improvement process. Costs and benefits of the pilot project should be estimated, measured, and evaluated throughout the life of the project, and an attempt should be made to generalize these results to see how they would scale up for use with other databases. Ideally, after the initial pilot project has performed its VV&C on the database, a different user group should be asked to try to use the results of this VV&C to evaluate the database for a different purpose; the experience gained from this subsequent effort should help in evaluating the utility of the VV&C process for future users.

## 11. CONCLUSION

The motivation for seeking to improve data quality is derived from the desire to improve the results of using models and simulations in a broad range of studies, investigations, training, and decisionmaking activities. The quality of the results obtained in these studies has direct impact on the outcome of activities such as procurement, organizational redesign and downscaling, system acquisition and integration, operational planning, operations, the performance of systems, etc. This impact may be in terms of the financial cost or efficiency, timeliness, risk, or ultimate effectiveness of the implemented decisions. Improving these results requires improving the decisions that produce them, which in turn requires improving the answers that are produced by the models used to help make these decisions.

Most uses of models for studies of this kind are largely data-driven. The quality of most modeling activities therefore depends critically on the quality of their data. Although there are many other aspects of modeling that are equally deserving of quality improvement, the well-known adage “garbage in, garbage out” suggests that the databases used to drive modeling studies should be subjected to far more stringent quality control than has heretofore been the case. The current lack of explicit data quality measures and control procedures makes it difficult to ensure or assess the quality of most existing databases—particularly large, complex databases whose users may be far removed from their producers—and this calls into question the quality of many modeling efforts.

I have suggested attacking this problem by means of two interrelated and parallel strategies: (1) performing explicit evaluation of data and (2) establishing organizational control over the processes that generate and modify data. These approaches require: (i) augmenting databases with metadata in order to record information needed to assess the quality of their data, record the results of such assessments, and support process control of processes affecting data; (ii) encouraging producers and consumers (users) of data to implement organizational commitments to perform distinct phases of explicit verification, validation, and certification (VV&C) on their data, using metadata both to direct these activities and to record their results; and (iii) establishing control over the processes that affect data, to improve the quality of data generation, transformation, and transmission, again using metadata both to support this activity and to record its results. I believe that automated tools can be developed to help capture and maintain metadata whenever generating or modifying data, thereby greatly facilitating this strategy; specifications for such tools will be developed in the future, as experience is gained in applying this strategy. This memorandum represents part of an ongoing effort to develop guidelines for metadata, VV&C techniques, and process control procedures to improve the quality of the data used in modeling. A proposed set of such guidelines is presented in [16].

### 11.1 The cost of (not) improving data quality

Adding and maintaining metadata to databases, performing and recording the results of explicit VV&C, and placing critical data-affecting processes under process control are all potentially expensive undertakings. Measuring and monitoring the quality of databases and their associated processes while maintaining and propagating the resulting metadata so that users can evaluate the

quality and appropriateness of databases for specific purposes will require a significant investment. Yet without increasing our attention to data quality, how can we rely on the results of even the simplest database queries, let alone the results of studies that use data in complex and often obscure ways?

Although the cost (in terms of both money, effort, or other organizational resources) of developing and enforcing data quality evaluation and improvement techniques will be substantial, I believe that the cost of *failing* to implement such techniques is even greater, since it undermines the value of much of the modeling and simulation that is currently performed (both within DoD and elsewhere) for analytic, predictive, and training purposes. The potential cost of improper decisions based on inappropriate results from such modeling efforts is likely to far outweigh the cost of implementing the techniques presented here to improve data quality.

## REFERENCES

- [1] Belair, Robert R. (ed.), 1985. *Criminal justice information policy: data quality of criminal history records*, U.S. Dept. of Justice, Bureau of Justice Statistics, Washington, D.C.
- [2] Blanning, R. W., 1987. "Sensitivity Analysis in Logic-based Models," *Decision Support Systems*, Vol. 3, pp. 343-349.
- [3] Bruce, T.A., 1992. *Designing Quality Databases with IDEFIX Information Models*. Dorset House, NY (ISBN 0-932633-18-8).
- [4] Inmon, W.H., 1992. *Data Architecture: The Information Paradigm*. QED Information Sciences, Wellesley, MA (ISBN 0-89435-358-6).
- [5] Kent, W., 1978. *Data and Reality*, North-Holland.
- [6] Liepins, Gunar E., and V. R. R. Uppuluri (eds.), 1990. *Data quality control: theory and pragmatics*, Marcel Dekker, Inc., New York.
- [7] Magnusson, David, and Lars R. Bergman (eds.), 1990. *Data quality in longitudinal research*, Cambridge University Press, New York.
- [8] Naus, Joseph I., 1975. *Data quality control and editing*, Marcel Dekker, Inc., New York.
- [9] OASD/C3I, 1994. *DoD 8320.1-M: Data Administration Procedures*. DTIC, Alexandria, VA.
- [10] OASD/C3I, 1994. *DoD 8320.1-M-3: Data Quality Assurance Procedures (Draft: February 1994)*. DTIC, Alexandria, VA.
- [11] Redman, Thomas C., 1992. *Data Quality Management and Technology*. Bantam Books (ISBN 0-553-09149-2).
- [12] Rothenberg, J., 1989. "The Nature of Modeling." In *Artificial Intelligence, Simulation, and Modeling*, L. Widman, K. Loparo, and N. Nielsen eds. John Wiley & Sons, 75-92. (Reprinted as N-3027-DARPA, The RAND Corporation, November 1989.)
- [13] Rothenberg, J., 1990. *Prototyping as Modeling: What is Being Modeled?*, The RAND Corporation, N-3191-DARPA.
- [14] Rothenberg, J., N. Z. Shapiro, and C. Hefley, 1990. *A 'Propagative' Approach to Sensitivity Analysis*, The RAND Corporation, N-3192-DARPA.
- [15] Rothenberg, J., and I. Kameny, 1994. "Data Verification, Validation, and Certification to Improve the Quality of Data Used in Modeling", *Proceedings of the 1994 Summer Computer Simulation Conference (SCSC'94)*, (La Jolla, CA, July 18-20, 1994), pp. 639-44, Society for Computer Simulation (SCS) (ISBN 1-56555-029-3).
- [16] Rothenberg, J., W. Stanley, G. Hanna, and M. Ralston, 1997. *Data Verification, Validation, and Certification (VV&C) Guidelines for Modeling and Simulation*, The RAND Corporation, PM-710-DMSO, September, 1997.

[17] Strong, D. M., Y.W. Lee, and R. Y. Wang, 1997. "10 Potholes in the Road to Information Quality", *Computer*, Vol. 30, Number 8 (August), pp. 38-46.

[18] Suri, R., 1989. "Perturbation Analysis: The State of the Art and Research Issues Explained via the GI/G/1 Queue", *Proceedings IEEE*, Vol. 77, No. 1 (January).

[19] Wilson, Thomas F. (ed.), 1986. *Data quality policies and procedures: Proceedings of a BJS/SEARCH conference*, U.S. Dept. of Justice, Bureau of Justice Statistics, Washington, D.C.