December 6, 1995

A MODEL FOR MATERIALS DATABASES

Gio Wiederhold
Laie Caindec
John McCarthy
Ken Reifsnider

ABSTRACT
We present a model for materials data, using terminology and concepts from the database community, to define the complexity and needs which materials data impose on database technology. Many terms are defined so that the model can be used by materials scientists. The intent of the model is to build a bridge between these two disciplines, since current technology has not adequately served the community which needs computerized access to materials data.

1. INTRODUCTION

The need to make information about materials engineering rapidly available to researchers and designers of advanced materials, has grown exceptionally fast during the last decade. Unfortunately, because of the extraordinarily broad domain of materials data, materials engineers are finding it difficult to acquire reliable, up-to-date test data on newly developed composites. This yet inadequately addressed need has brought about the invention o new software and hardware technology designed specifically to fit the requirements of advanced materials engineers. CIM (Computer Integrated Manuafacturing) and CAD/CAM (Computer Aided Design and Maufacturing) systems have generated a great deal of interest in research-oriented environments, and are now becoming more commercially practical.

To anticipate some of the upcoming requirements of materials scientists, Stanford University, with the NBS and the NFS, recently hosted a Materials Databases Workshop. The central objective of the workshop was to develop a greater sense of the issues involved with integrating computer technology with materials data. Another was to combine the expertise of the three fields we believe are the most closely related to future development of materials database management. Broadly, these three fields were Materials Science, Engineering, and Computer Science.

Beyond these three major divisions, the workshop was divided into a set of study groups: the representation of materials information, the computerr-user interfaces, choosing and utilizing appropriate technology, and the role of expert systems. The attendees distributed themselves fairly evenly throughout these groups, encouraging a true sharing of perspectives.

†Gio Wiederhold is an Associate Professor of Research at Stanford University in Medicine and Computer Science. Laie Caindec is a recent graduate of Stanford University in Computer Science, now at Oracle Corporation. John McCarthy is a Researcher and Consultant at the Lawrence Berkeley Laboratory, in Berkeley, CA. Ken Reifsnider is the Reynolds Metals Professor at the Virginia Technological Institute in Blacksburg, VA. Please refer any questions to the first author.

Based on the proceedings of this workshop, a prototypical model for a materials database system was proposed. This paper focuses on the modeling aspect of the proposed system.

1.1 Existing Problems

We foresee many new problems which may directly result in implementing a comprehensive materials database model. However, even the existing systems have problems which require more research. Several factors which currently inhibit optimal utility of a materials database are identified below:

1. Access to existing data resources is inconvenient; most information is still on paper.
2. Searching for materials by constraints on properties, and secondary variables, etc. is difficult.
3. Descriptions of materials properties, especially as a function of geometry, environment, treatment, and use are inconsistent and incompatible.
4. Data formats for describing advanced materials are unsuitable.
5. Interfaces provided by existing database management systems to to support CAD (Computer Aided Design) and CIM (Computer Integrated Manufacturing) systems are generally poor. These systems require interfaces that permit flexible and effective transfer of data into programs, rather than friendly query language access. Efficiency is a concern as well since data is often voluminous.
6. Quality assurance for individual data sets, databases, and entire systems is inadequate.

In this paper, we focus on the proper description of materials databases, since resolution of this area is fundamental to making progress with any of the factors identified above.

The new systems should optimize use of recent software and hardware technology to deliver materials data. In addition to utilizing state-of-the-art technology, it may be critical to integrate materials database management with expert systems. Unfortunately, the majority of expert systems are currently created to only meet specific needs. Expert systems models which would be most useful to materials information systems should be capable of being generalized for various materials engineering applications. The develpment of interfaces to support database access for expert systems is an active research area [Kerschberg 87].

1.2 Model Approach

We first present a discussion on the computerization of advanced materials, then we define a database model prototype which could support future integration of materials databases. The intent of this model is to provide an abstraction which can link users and contributors of materials data. The commonality is grounded in database science and technology.

To incorporate the large number of descriptive parameters inherent to materials data, the model accommodates various representations of data which might be included in the domain of materials data. These representations must accommodate hierarchies of properties and materials, descriptive variables, independent parameters, and *textual notes*.

The model must make a tradeoff in dealing with the specifics of materials science requirements and the capability of current database technology. The general model presented here then does not deal well with special features or properties unique to small groups of materials, and the consideration of new, specialized and composite materials. Some such problems are addressed in Section 4.1. At the same time, the requirements we pose are too difficult for direct translation into existing and general database management systems. A layer of software may be able provide the mediation needed between materials systems and database systems, albeit at a loss in performance [AFIS 89].

In the conclusion we identify research issues to be addressed if materials databases are to become an active component of automated engineering support systems. A glossary provides access to the considerable body of terminology encountered at this juncture of three disciplines.

2. COMPUTERIZED DESCRIPTION OF MATERIALS

The dominating characteristic where materials data differ in structure from data encountered in traditiional database is the complexity of their description. Whereas the common model of a database record has a simple key identifier, say, the `employee_number`, and associated with that identifier a long string of dependent attributes, as `name, age, salary, job_title, department, ...`, a piece of materials data needs many attributes to describe it. There is the `type` of the material, its `processing`, its `shape`, its `temperature`, etc. The dependent data are correspondingly few, principally its `strength` and `stiffness`, and secondary parameters as its `specific_weight` and `ductility`.

A further feature of materials is the need to characterize the properties and response of these materials under *extreme conditions*. Such conditions frequently involve combinations of severe environments, such as severe cyclic or dynamic mechanical loading, extreme high temperature or low temperature, and severely aggressive chemical environments. The arrays of properties and performance mentioned above must be determined as a function of various combinations of these influences that are of interest to specific applications. It is very difficult to find a general representation of these highly variable circumstances and countably infinite combinations of constituent properties and their configurations.

The challenge is further heightened by the necessity to characterize changes in properties and performance as a function of the history of application of the influences commonly applied to these materials. For example, the most important information for materials to be used at very high temperatures or other severe conditions is their rate of degradation under a specific set of conditions, since the permissible utility of those materials is limited by those rates. Moreover, the engineering design of structures has reached the point at which safety and reliability are critical design objectives and criteria for certification in many instances. These service-related characteristics depend on the manner in which properties and performance change during service. The quantification and description of these changes is essential to the complete description of materials.

2.1 Determination of Properties and Performance Data

Beyond the complexity and sophistication of descriptive representations suggested above, the test methods to determine these properties and characteristics must be available. Professional societies such as ASTM, have defined standard test methods, but full information may not be vailable for all materials. The selection of the most relevant test data for a given application is probably beyond today's automation. *Classical* properties such as stiffness and strength suffer less from this malady than do less classical performance characteristic definitions for phenomena such as creep rupture, static and dynamic fatigue, and damage accumulation.

Another important feature related to the determination of properties and performance is the importance of statistical variations of these characteristics. Processing plays a very great role in the final character of materials. Consequently, variations in properties can be significant from *batch to batch*.

The model which follows provides a framework for organizing the large quantity of data we will find in computerized materials systems.

3. A Computational Model for Material Databases

In order to share an understanding of the requirements specific to materials information databases, we propose a high-level model. This model is an abstraction of information discussed at the workshop, as well as concepts drawn from related areas such as Engineering Information Systems and statistical databases. This model specifies the structure of materials information. Means to achieve efficient access are a subsequent topic, where the users' demands for information can be quantified vis-a-vis the model.

3.1 Significant Aspects of Materials Databases

While the components of Materials Database Systems (MDBS) can be mapped to components of other types of databases, some differences stand out and make it difficult to implement effective MDBS on current commercial, off-the-shelf systems (COTS) or general Database Management Systems (DBMS). We recapitulate the issues and relate now materials database (MDB) terms to DBMS terms by placing them in (parentheses).

- The number of identifying (key) attributes is much greater than those seen in current commercial applications. An MDBS often requires as many as 20 keys where a commercial DBMS normally requires only 1, 2, or 3.
- Much of the actual data does not exist at the base levels, but only at higher levels of abstraction.
- Textual notes and descriptors are important for understanding applicability of the data.
- Interpolation functions may be attached to the domains.
- Extrapolation functions may be attached to the domains.
- The number of dependent attributes per cell (tuple) is small, but the specific attributes vary from material to material.

Broadly speaking, one consults a computerized materials database or system of databases for one of three purposes:

1. to retrieve the properties data for a particular material
2. to find a short list of candidate materials with a specified set of properties
3. to sort particular materials into groups serving some functional purpose or application area.

For all these purposes a designation or identification scheme is required which will define and differentiate materials with significantly different properties.

The designation or identification of materials poses some particular problems in a computerized information system of broad scope and applicability. These problems derive partly from the fact that the computer's search and retrieval programs mindlessly look only for matches with characters strings and partly from the fact that the user of the information and his application cannot always be anticipated. To dig more deeply into the matter, we should examine some dictionary definitions. The critical definitions of designation, identification, and characterization are discussed in Westbrook [West 88], which is reported in the ASTM International Symposium of 1988.

3.2 Identification Parameters

One of the most outstanding features of materials data is the large number of descriptive parameters which are necessary to precisely identify observed properties. For example, the yield strength of a traditional material, `Steel`, includes not only a hierarchy of material identification parameters, I, which can be expressed by a special symbol ($\longrightarrow\!\!*$):

I1 `Steel` $\longrightarrow\!\!*$ `alloy steel` $\longrightarrow\!\!*$ `Ni-Cr-Mo steel` $\longrightarrow\!\!*$ `AISI 4340 = 145 ksi`

but also a number of independent parameters:

I2 `test-sample-configuration`
e.g., `bar`
I3 `heat-treatment`
i.e., `Q + T`
I4 `test-temperature`
I5 `Source`
i.e., of test information

Each of these independent parameters or descriptive variables, Ij, denotes a set of dependent descriptives which make up the components of a conceptual vector, or tensor, in an all encompassing *Data Matrix*, D. (The *Data Matrix* will be explained in fuller detail below.) For example, the identifying parameter, `Source` (as opposed to *identification*), of the sample material, would be represented as the following conceptual tensor, in the materials matrix:

I5 `Source`
`= US Steel, Heat 2317, Oct 1951`

For this independent identifying parameter, `Source`, the actual data describing the sample material are `US Steel, Heat 2317, Oct 1951`, which collectively make up the tensor component $I5$ of this material, in the Data Matrix D. Dependent on each identifying parameter then, are a set of actual data describing the material.

$D \rightarrow$ {strength, hardness, machinability, etc.}

Due to the large number of independent descriptive variables, say m, and the non-trivial size of the sets of dependent variables, the overall size (N_D) of the conceptual data matrix (D) grows quite rapidly. The following equation describes the multiplicative growth of the data matrix.

$$N_D = \prod_{j=1}^{m} d_{Ij}$$

where d is the size of each of the domains of the independent variables $j, j = 1, \ldots, m$. Clearly, the search domain of an average query could be quite large.

For complex materials, each of the m independent variables may themselves be a multidimensional tensor. Assigning a standard tensor permits the establishment of a canonical ordering within that domain. However, the domain will be large, further exacerbating the problems associated with the large matrix size, N_D.

Where domains are real numbers, as opposed to integers, domain sizes appear infinite. The size of the resulting table of data must be reduced by selecting an appropriate representation granularity, say `Melting_Temperature = 100K, 200K, 300K, ..., 1000K`. Some domains are effectively *open-ended*, such as `test-sample-configuration`, and may be defined by a special reference symbol, $\succ\!\!-\!\!-$, pointing to an auxiliary table, A_I, which defines all valid configurations more precisely.

An important category of descriptive information, vital to the interpretation of the values in the matrix D are *textual notes.* To the extent that they determine the values in D, they are also best modeled as a class of independent variables, It. For search purposes, it is best to enter textual notes as specific attributes, if possible. An example might be:

$It1$ `warning`
`= strength adversely affected by surface scratches`

$It2$ `application`
`= underwater construction`

Newer and specialized materials especially, for which our experience and understanding has not reached a point of convenient and complete classification, may have many such descriptive attributes.

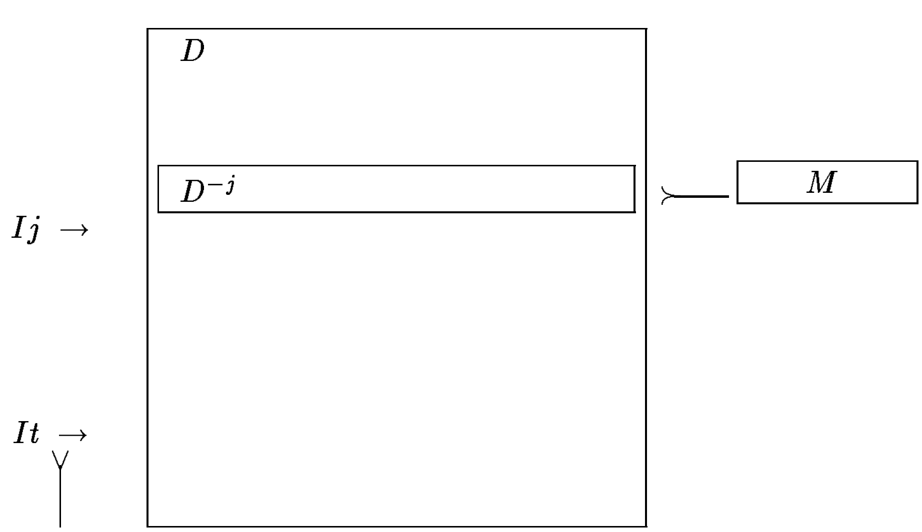
3.3 Components of the Model

In addition to the data matrix D we need to define its components, and a number of ancilliary structures. We will first list then, and sketch their relationship graphically, and then elaborate on them.

- D The base matrix of data values.
- V Set of values within a data cell of D.
- Ij The independent variables identifying the data values.
- d Domain values for the identifying data values.
- D^{-j} Data values at higher levels of abstraction, omitting one or more identifiers j.
- F Interpolation/Extrapolation functions to permit computation of domain values.
- M Modifiers for data values.
- It Textual identifiers.
- R References cited by textual identifiers.

3.4 The Graphic Model

We model then the components of a materials database as follows:



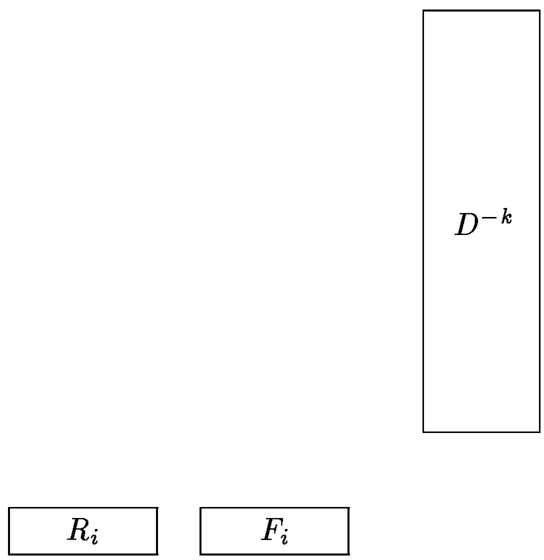


Figure 2. Graphical Overview of Model

We can now summarize the objects in the materials database environment which an MDB has to consider. The relative importance of these objects can differ greatly; some may not exist at all in some materials databases. The criteria for the exclusion may be intentional or accidental. If an existing database system approach is imposed on the materials database then it is likely that some components will be missing or under-emphasized.

3.5 The Data Matrix, D

We now describe the model elements of the data matrix itself. While the concept of the data matrix is intuitively simple, it is also naive. For any realistic number of independent variables I, the value of N, the matrix size, is infeasible and large. The following two approaches, *Interpolation* and *Abstraction*, reduce the size of N, but also introduce complexity.

The Data Matrix can contain identifiers and at least four *types* of variables, V, or data.

1. `identifiers`
   e.g., `4043 steel from Nippon steel, heat no. 4213`
2. `descriptive characteristics`
   e.g., `suitable for underwater construction`
3. `independent variables`
   e.g., `temperature, environment, form`
4. `dependent variables`
   e.g., `ultimate tensile strength`

In this example, `identifiers` determine `descriptive characteristics`, and both the `identifiers` and the `independent variables` determine the `dependent variables`.

3.5.1 Sparseness

As indicated, this matrix provides a conceptual basis for information structuring, but does not provide a path for data storage management. First of all, we must note that such a matrix will be very *sparse*. Only small subvectors in some specific dimensions will be known.

We can expect that even in conventional materials, the density will be a few percent, and in advanced materials much less than one percent.

In the following paragraphs, we consider how to deal with the problem of retrieving information in such a sparse environment.

3.5.2 Interpolation

Interpolation is needed because our data matrices are so sparse. A relational model database is apt to return a `null` for any combinations combinations of identifying key parameters, for which no explicit data exist.

If the designer needs such data, the materials expert can take nearest neighbor values and produce a credible estimate by interpolation. Automation of this process means that the database needs a rule of the form `''If value is null, try to interpolate"`.

Interpolation means that there should be values above and below the searched for data in all dimensions; otherwise, we are forced to *extrapolate*.

Associating an interpolation function with an independent identification parameter, Ij, can greatly reduce the number of entries in the domain, d_{Ij}. Critical points may be explicitly represented, and more than one function may be needed for distinct ranges, as determined by phase changes and the like.

Using the same function for extrapolation is also possible, but carries a much greater risk. Outside of the measured data points may be discontinuities not captured by the interpolation function. If an interpolation has many parameters in order to fit a data interval precisely, then the risk of extrapolation is especially high. Confidence in extrapolation should be directly proportional to the number and *naturalness* of the parameters used to describe the function.

Under naturalness we mean a function which mimics the physical behavior of the materials system, rather than a function, as a general polynomial, which is adequate by numeric measures but which requires ever increasing terms to span wider ranges of values. An example of a natural function is a series incorporating the term $1/T$ for thermodynamic paparameters as enthalpy. Use of a Langrange or power series, with terms of the form aT^i leads to forced fits with very poor extrapolation behavior.

Extrapolation is quite a different matter than interpolation in that the bounds of uncertainty or probability error can usually be precisely stated for interpolation, but are large and frequently indeterminate for extrapolation. Furthermore, the appropriate interpolation or extrapolation function and its adequacy (both in the mathematical and *naturalness* senses) varies from property to property and perhaps also for the same property, from material group to material group.

Any use of interpolation or extrapolation should be reported to the user of a materials database. Use of extrapolated values might be discouraged.

3.5.3 Abstraction or Data Reduction

By abstraction, we mean the reduced data matrix, D^{-i}, created when one or more independent attributes are not recorded for a sample material. *Abstraction* may be an awkward term for the major technique in materials data reduction, because the absent information may not even exist in the matrix, or may exist sparsely. For example, we may not always have data on differences in `heat-treatment` or `test-sample-configuration`. Furthermore, if we talk about identification of a `Steel` we will not always have information about the specific `batch-no`.

The problem with this type of data reduction, and commonly associated with the concepts of abstraction, is the loss of information for those variables which depend on all of the `identifiers` of a given material. Note that abstraction is an example of *vertical* data reduction (see Figure 1), and may be contrasted with *horizontal* data reduction provided by interpolation.

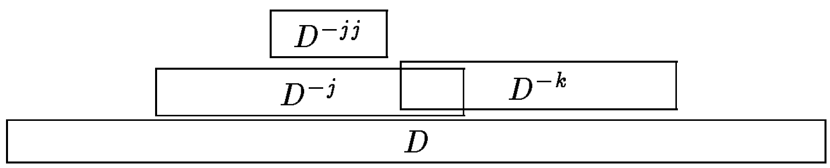


Figure 1. Vertical Data Reduction

In practice, there would be many intermediate levels of data attributed to a given class of materials. At the base level, D, all information for a sample material would be available. From the example in the discussion of *Identification Parameters*, we have the following material identification hierarchy:

$D =$

`Steel` $\longrightarrow$ `Alloy Steel` $\longrightarrow$ `Ni-Cr-Mo Steel`

As a vector, this data conceptually represent D, for the batch it specifically describes. By removing the composite information attribute, `Ni-Cr-Mo`, the remaining data for this class of materials are effectively *reduced* or abstracted. The resulting tensor can be represented as D^{-k}, where k uniquely identifies the parameter, `composite-information`. The hierarchy of information given in this tensor can further be reduced by removing each independent attribute, until finally, the only datum left is D^{-jj}, representing `Steel`.

Some variables, say density, do not depend on all of the identifiers, and are hence properly attached at higher levels of abstraction.

3.6 Storing Sparse Data

While we model the data matrix as a convential high-dimensional cube, it is clear that it will not be stored in that manner. We cannot now be definite on the best representation. For computational purposes, a sparse matrix can be stored by accociating with each value index lists of the predecessor and successor values in each of the m dimensions [Knuth vol.1].

For persistent storage we might prefer a relational or a tabular form where each known value is preceded by a list of all the associated descriptors in a canonical order. All the m descriptors must be listed in an associated *schema*, so that the canonical position can be ascertained. The schema is not shown in the graphical model.

A data record in relational form will look as follows:

$$\boxed{\mathrm{I1, I2}, \ldots, \mathrm{I}j, \ldots, \mathrm{I}m \;:\!\rangle\; \mathrm{d1}, \ldots, \mathrm{d}\ell, \ldots, \mathrm{d}k.}$$

We use the symbol $:\!\rangle$ to separate identifies from data values.

To describe both identifying and data attributes we will $m + k$ schema entries. Details of such organizations can be found in [Wiederhold 83]. Each schema entry includes

1 The name of the attribute.
2 Text further describing the attribute.

3 A code for the type of attribute, as integer, category (discrete, code, text), etc.
4 a domain specification for the attribute, as
   4.1 The range for integers.
   4.2 The category values for categories.
5 The interpolation function appropriate for the domain, to be used if the value is missing but in range.
6 The interpolation bounds for such functions, i.e., gaps that cannot be covered with any confidence.
7 The extrapolation function appropriate for the domain, to be used if the value is missing, and out of range. This function may be identical to the interpolation function in item 5, or may be simpler (more linear), to avoid excessive errors.
8 The extrapolation gaps for such functions, i.e., limits on the range that might be covered, beyond which a value of out-of-range must be returned.
9 A standard base value.

As indicated above, the values returned by the interpolation function include codes where the limits are exceeded. Provision for such codes is now included in the IEEE floating-point standard [IEEE fltpt]. Other components of the value returned are a positive and a negative uncertainty bound.

Where the data to be entered is incompletely specified, that is not all descriptive parameters have been provided, their records can be entered into the table with the standard-base-value. Now values can still be obtained in range searches for data. That those data were incomplete given must be entered at least into the associated textual notes, so that a report can explain this type of uncertainty in the result.

3.7 Locating Information

Now the searches for information in this sparse environment can be performed using a form of range search parameters; if no exact match is obtained, the predecessor and successor values are retrieved, and the interpolation function appropriate for that domain is invoked to compute an estimate and its uncertainty. We discuss those functions further below.

If a range parameter is not given, the standard-base-value is substituted. The value of any substituted parameters are reported with the result, so that the user will be aware of the assumptions made.

Query languages available for relational databases do not deal adequately with sparse information. Some work in the management of large statistical tables may be relevant [Ozosoyoglu 89], but this work again is designed to interact with people. Our materail systems will have to interface with processing programs serving design environments.

3.8 Use of Interpolation

While we can easily imagine a reasonable interpolation function when used with a graph of values, it is quite difficult for someone without much material experience to map abstracted data back down to a lower level.

For a system to work in this model, we must require that there will be an interpolation function for every domain. Today, sometimes interpolation functions are provided within the handbooks listing materials data, but rarely for each possible parameter. The engineer then makes up interpolation function as needed — unless the use has the capability to perform specific tests for the case being considered. A specific test reduces the undertainty associated with an interpolation function.

The results of an interpolation function should not only be a value, but also the uncertainty — perhaps in the form of a standard error — associated with the value. Furthermore some special values can be returned:

oor No value can be returned
 – given when the parameter is *out-of-range* and no reliable extrapolation exists.

nil No value can be computed
 – given when no function exists.

n.a. A value of n.a. may be returned for ranges where it is clear that the material should not be used.

Interpolation may not be feasible for some discrete categories. The solution is then to request information for only a specific value of this parameter. A `HELP` function must be available to provide advice on what range of values is usable.

The interpolation function need not be restricted to conventional polynomial interpolation. For instance, for the domain `heat treatment`, if no data are fund suitable for estimating the effect, the minimum value —, i.e, the value for `no-heat-treatment` can be returned for the strength of the material. The uncertainty component then has only a positive range.

The establishment of a comprehensive set of interpolation functions will require time and effort. Experience can be gained with simple materials before attempting to cover complex materials. The proposed `HELP` function will be necessary for a long time to assist the user to browse through the multi-dimensional space represented by the data matrix.

Without interpolation automated use is severely compromised. Programs to aid design must be able to access materials data without the need for human intervention. Although a single intervention may be a problem in the design process, repeated interventions, in order to achieve an optimal material selection are certainly a hindrance. Summarization is disabled as well.

A problem not yet addressed is how to provide for interaction of multiple interpolations. While some common cases should not require special consideration, there are obviously cases where the interaction is such that the uncertainty is greatly increased, or where the range of validity becomes less than for any function individually. For summarized results the use of multiple interpolations should remain feasible.

3.9 Data Modifiers

The set of data itself D is rarely absolute. There will be variation in the material, variation in the measurements, questions of trustworthiness of the measurements, as well as outright errors. We cannot deal well with errors, except by setting error bounds, which will cause data to be rejected upon input.

3.9.1 Measurement Quality

Most entries in D are obtained by measurements. Associated with specific sets of observations we might find degrees of tolerance and uncertainty:

$M1$ `tolerance is` $\pm$ `10`
$M2$ `standard error in testing is 0.3`

3.9.2 Trustworthiness

There is of course also the user's interpretation and belief in the validity of the results. While a specific user may place a belief assessment into the set of modifiers, a more general model would associate the belief with the `Source`, or rather with a crossproduct of independent attributes such as `Source` $\times$ `Material`. Auxiliary tables, keyed to these attributes, might represent this level of the users' interpretation.

There may be multiple, intersecting abstractions D^{-j}, D^{-k} over the same, probably non-existent base data matrix D, depending on the descriptors Ij, Ik which have been omitted.

Distinct abstractions may also be characterized by objectives: for many electrical applications the effects of `micro-structure` may be immaterial, for space applications `environmental variables` will be fixed.

By utilizing two or more intersecting abstractions together, expectations of materials performance may be obtained for which results might be valid for the design phase, but subsequent specific testing may become a higher imperative which would otherwise be the case.

3.10 Incompleteness

The conceptual m-dimensional matrix, D, has sides of cardinalities d_{Ii}. Unfortunately, especially when we consider the actual data subsets, as defined via abstraction, the overall domains may be incompletely covered. For example, the `temperature` range is likely to be incomplete for many of the `test-sample-configurations`.

3.11 Textual References

We introduced above a distinguished, irregulat class of descriptors, textual notes, As a domain the collection of those notes may be modeled as a vector, It, a special class of tensor. This vector establishes an open-ended list with the text of these notes. Whenever comments are identical for multiple samples, it is desirable to reference the same note, so that eventual patterns may be established. The elements of the materials matrix can then refer to a subset of textual notes, with any entry being `present`, `undetermined`, or `known-to-be-absent`. This vector is open-ended. Distinct vectors may be appropriate for different classes of materials. The combination of these vectors form a unique matrix for each class of materials.

The matrices for similar materials may be similar in size, use similar domains, and share the auxiliary reference tables of textual notes.

Textual references are not constrained to be literal strings. They may also be indirect, namely references to the literature. Of course, if the reference is to a large object as a reference to a textbook, only a small part of the referenced information will be relevant. For an actual implementation, direct and indirect information may be combined: the abstract itself will be directly available; for more detail, one may indirectly access further data via the author's name and publication information.

3.12 Model relationships

We can identify the following relationships connecting the components.

- ID: $I1, \ldots, Ii \rightarrow D$ The data values are determined by the set of their identifiers.
- DV: $D_{1,\ldots i} \longrightarrow\!\!\ast\; V$ Each data cell is composed of a (small) set of data values, where $1 \ldots i$ could either be different properties or different measurements of the same property.
- IIt $I \longrightarrow\!\!\ni It$ Some data descriptors are textual.
- ItT $It = (0,1)t, t \in T$ There is a set of textual descriptors from which zero or more may be selected.
- tR $t \longrightarrow\!\!\ni tr, tr \succ\!\!\longrightarrow R$ Some textual descriptors reference documentation.
- DM $D \longrightarrow\!\!\ni M$ Data elements may have modifiers.

$D - D$ $D* \longrightarrow D_{-j}$ Data may be abstracted.
dF $d \longrightarrow\!\!\ni F$ Domains may have interpolation functions for elements $V \in D$

3.13 Computerized Description of Advanced Materials

Advanced materials present a special challenge to the computerization of the properties and performance specifications of advanced materials. Such materials are *advanced* because of highly specific characteristics which make them uniquely suited to special engineering applications such as high speed vehicles, high performance engines, and other applications requiring materials that can sustain severe mechanical, chemical, and thermal conditions. The most dramatic development in the field of advanced materials is the emergence of a class of materials called *composites*. Composites are mechanical mixtures of the distinctive constituents, phases of different materials, which can be combined in order to achieve a synergistic blend of properties which cannot be obtained by use of each of the constituents alone.

The *art* and *science* of composite materials has grown dramatically over the last twenty years at a rate which suggests that they may dominate the field of *advanced materials* as early as the year 2000. These materials present a very special challenge to efforts to computerize their description, since they are designed for specific applications and are, therefore, difficult to classify since they are as various as the engineering circumstances they serve. The following summary presents a brief discussion of this situation in general and of the special challenge associated with composites in particular.

Examples of Advanced Materials

Examples of advanced materials include high temperature materials and other advanced alloys, composites, hybrid materials, and coatings. General features, particularly important to establishing proper representations of such materials, include the dependence of properties and performance upon micro-structural configurations as well as property definitions of the constituent materials. Other features include a dependence on fabrication and synthesis techniques, a lack of standardized test methods and characterization schemes for the determination of properties and performance, and an uncommon need to represent the combined response of such materials to multiple excitations and circumstances, such as mechanical, thermal, and chemical loadings.

The Challenge of Describing Advanced Materials

Advanced materials are frequently *advanced* not only because of the properties of the micro-constituents, but also because of their arrangement. The global properties of composites, for example, are generally anisotropic such that their response depends on the direction of loading relative to certain special directions defined by micro-constituent arrangements. The immediate consequence of this situation from the standpoint of representation is that a single property such as stiffness or strength becomes a tensor array of properties which may have as many as 36 entries to provide a complete description of the material characteristics. Moreover, a large class of advanced materials consists of laminated plies of a given composite material system wherein the individual plies differ in orientation and combine to form a laminate which has yet another set of specific properties and performance characteristics. These lamination arrangements and other similar configurational variations are sometimes said to be *mini-structures* to contrast them in dimensional scale with the *micro-structure* normally associated with constituents in a composite material system or alloy. In total, this range of configurational variation is countably infinite and is the common objective of optimization efforts to achieve the most desirable properties and performance of advanced materials. From the standpoint of database representations, this countably infinite set of tensor arrays is a major expansion of the *simple set* of material properties and characteristics that is commonly associated with *traditional materials* that provides a unique and permanent definition of the engineering utility of those materials.

For advanced materials it is even harder to characterize the properties and response under extreme conditions. Beyond the greater complexity and sophistication of descriptive representations suggested above, efforts to establish databases for advanced materials are further frustrated by the fact that test methods to determine these properties and characteristics are still under current development, and consensus methods for many of the properties and performance determinations have not been established. Largely because of the complexity of the situation, test methods are commonly developed for circumstances of particular interest to various applications, and variations in those test methods are frequently made as variations in the applications suggest. Although notable efforts are being made to establish test methods, especially by professional societies such as ASTM, these efforts seriously lag the development of advanced materials. Additional identifiers should include specifics of the methods of determination of their properties. Consequently, the data matrices become yet larger than those mentioned above and will even more from absent entries.

Another important feature related to the determination of properties and performance is the importance of statistical variations of these characteristics in advanced materials. It was mentioned earlier that processing plays a very great role in the final character of such materials. In general, not only is processing difficult to control, and difficult to repeat, but it presents a sundry of difficulties in and of itself. Consequently, variations in properties are surprisingly large from *batch to batch,* or even within a given batch, or in some cases within a specific specimen or component. It is impossible to properly discuss the properties and performance of these materials without taking a rigorous quantitative approach to these statistical variations. Indeed, the design of advanced structures with these materials and the subsequent certification for service is frequently based on allowable levels of applied loading determined from a statistical analysis of an allowable likelihood or rate of failure determined from these statistical variations in material properties and performance.

These problems present further challenges to the management of materials data. Their solution may have to await the resolution of the simpler materials database issues, but must be kept in mind when planning for advanced applications.

3.14 Implementation of Materials Databases

Although in principle the storage

3.15 Active Interaction with Design Programs

When we have design materials which can be structured to satisfy specific properties the concept of a passive database, which simply returns stored values, retreats even further. Now the database system, or rather an expert system interacting with the database, must receive specifications of desired properties ($d\ell$). Computational functions can now return descriptor values (Ij). The relational model actually can treat data attributes independently of whether they are identifiers or dependent data, but the concept of reversible computations over these elements is beyond current models or implementations.

4. CONCLUSIONS

The intent of this discussion and the model is to bring materials scientists and computer scientists closer together, so they can better focus on the hard issues of dealing with materials data. The model concepts obviously stem from the computer science domain. While the model itself may seem complex, we believe it to be comprehensive. Comprehensiveness can help avoid the trap of computer scientists to oversimplify the world of materials data, which would eventually result in inadequate solutions for materials scientists.

4.1 Issues in Computerizing Descriptions of Advanced Materials

Attempts to establish computerized descriptions of advanced materials require a number of decisions in addition to the requirements and constraints mentioned above. Since descriptions may be peculiar to applications which drive the development of advanced materials, information regarding the manner in which data was obtained and the manner in which materials were prepared is essential. Efforts to achieve standardization are critical, but non-standard situations must be expected and properly handled. Careful consideration of the likely user of descriptions must be given for decisions regarding the need to evaluate data and the methods of formatting the data for presentation. Because of the volume and variability of material data which must be handled, search schemes are critical. It would appear that expert systems have a natural application to this situation. Representation of dynamic data is essential.

The complexity and sophistication of computerized descriptions of advanced materials may exact somewhat peculiar requirements on such databases. Since these materials are so strongly driven by applications, it may be that databases will be driven by groups of applications as well. Some of the databases that have currently been established are tailored to groups of users who have peculiar applications or classes of applications which are best served by the structure of the database involved. Since the cost of these databases is substantial, and cost recovery is a persistent and potentially disabling problem associated with these databases, it may be that databases tailored to a specific market which benefits sufficiently to merit sustaining the cost of maintenance of that system is a viable development route. COMPOSINET, an on-line information database constructed for the use of the Industrial Affiliates of the Center for Composite Materials and Structures at the Virginia Technological Institute, is an example of such a dedicated database developed for a specific purpose. However, in order to achieve generality, gateway arrangements are necessary to combine specific databases developed for specific purposes.

4.2 Directions for Future Research

While it is important to study how current technology may be developed further to accommodate the needs of a materials database system, it is also essential to plan for new technology needed in this area. An important result of this workshop was a list of several projects for future research. Out of the list emerged three general classifications:

(a) New Technology
(b) Adaptation of Existing Technology
(c) Adoption of Existing Technology

There is a great range of research needed in this area. Standardized general descriptions of advanced materials must be achieved before computerized descriptions can be achieved.

Of course, this can proceed no more quickly than fundamental research on the properties and performance of these materials. Following is a brief list of requirements involved in the task of creating a more successful materials database system:

1. Issues in compatibility and integration of data and users should be addressed.
2. Standards for testing, generating, storing, and using data should be developed.
3. Model systems and exploratory descriptive techniques should be developed on a trial basis.
4. Support of existing systems for exploratory development should be considered.
5. Advanced concepts such as the use of fuzzy set theory in the representation of indefinite data sets should be investigated.

The need for research for the the computerization of advanced materials is great, but the need for success in this area is even greater. The economic success and safety of our country greatly depends upon our ability to develop, describe, and use advanced materials most efficiently. This is an area in which the complexity and sophistication associated with materials data, prohibits a trial and error approach. We have reached an era in which we must be guided in our technical endeavors by logic and rational anticipation; we must know what we should do before we do it. This cannot be done without a rigorous and vigorous approach to the management of our knowledge and experiences.

Acknowledgements
The authors acknowledge the assistance of thew National Science Foundation and the National Bureau of Standards for funding this workshop. We especially extend our gratitude to Howard Moraff from NSF, and John Rumble from NBS (now NIST).

We would also like to acknowledge all other workshop participants by name, we hope we have represented their views correctly: Attila Bruckner, Douglas Brutlag, Peter Burke, Jim Davidson, Ralph Harrison, David Hartzband, Sharif Heger, Donald Hillman, Larry Kerschberg, Mohammad Ketabchi, Billy V. Koen, Christopher LeMaistre, John McCarthy, Howard Moraff, Mervin Muller, Frank Olken, Bo Parker, Gerald Reisz, John Rumble, F.J. Smith, Stanley Su, Y.T. Thathachari, Miles Townsend, Ellis Verink, Volker Weiss, and Jack Westbrook.

5. GLOSSARY

abstraction Abstraction is the process of providing information at a higher level (e.g., ignoring some of the variables that can affect data values). Abstraction may involve a computational **summarization** or just a selection of certain parts of the original information.

access The opportunity to enter a computerized data system to read, write, search or edit records.

advanced materials Materials which are engineered to have specifications which exceed those of commonly available materials.

aggregation Aggregation is an operation where data are summarized to a higher level, and a **key attribute** controlling the aggregation is deleted. For instance, a computation to compute the `MEAN density` of a set of `poly-vinyls` is an aggregation. Aggregation functions such as `COUNT`, `SUM`, `MAX`, `MIN`, and `Standard Deviation` may be part of a summarization.

architecture The physical and logical structure of a computer system. Includes methods to handle input/output, storage, memory, and the instruction set.

association The term indicating a relationship among entities or a connection among database relations.

attribute The attribute of a data value is its name, and such a name is commonly associated with a column in a table of records. In a database system the attributes for each file of records are listed in the **schema**.

characteristics This word is used to describe properties, independent variables, etc. pertaining to a material. For example, characteristics might include `color`, `ultimate tensile strength`, `form`, etc..

composite Mechanical mixtures of distinctive constituent phases of different materials which are combined in order to achieve a synergistic set of characteristics which cannot be obtained by use of any of the constituent materials alone.

computerized Converted to a form (digitized) which can be stored, read, and retrieved by a computer.

connection A formalization of relationships to describe structural semantics and constraints among relations.

constraint A rule which limits the range of data values.

data A set of technical information represented by tables, graphs, models, numbers, or symbols. Note that data differ from information.

data abstraction Providing the user with an abstract view of the data so that details of how the data is stored and maintained is hidden. Data abstraction must often be weighed against being able to retrieve information efficiently.

data capture Collecting data and representing data in such a way that the data in the database *captures* the nature of the object being represented.

data evaluation The examination and validation of data by experts to assure that data conform with phenomenological or theoretical models.

data independence A characteristic of most DBMS that the data in the computer system is independent of the ways in which users view that data.

data item The smallest unit of data that has meaning in describing information; the smallest unit of named data.

data model A data model is a collection of conceptual tools for describing data, data relationships, data semantics, and data constraints.

data reliability see **reliability**

data security see **security**

data semantics The meaning of data, primarily established through the relationship of a data element to other elements and descriptive information.

data structure How data are organized and placed into memory. The data structure in a database should reflect **data semantics**.

database A **database** is a collection of related data. The data storage for a database is accomplished by the use of one or more files. A comprehensive database should contain all the information to manage some enterprise, say, a business, a scientific study, or a governmental department. Less comprehensive data collections that support some part of an enterprise are also commonly called databases. A data collection managed by a single-application program is best not considered to be a database.

database model A type of model (relational, network, etc.) used to represent a data model.

database management system (DBMS) The collection of software required for using a database that handles the storage allocation, retrieval and update of records by presenting multiple views of the data to the users and programmers. A database-management system controls redundancy of records and provides security and data independent of a database. Database-management systems allow the user to sort through large volumes of files and display relationships that would be too time-consuming and sometimes virtually impossible to generate in the manual system.

dependent Data values that depend on other, key values for their meaning.

engineering information system (EIS) A computer system which combines databases and data processing programs to aid in the design, building, and making of engineered systems, often for electronics.

expert system A computer-based system that emulates the performance of a personal expert or consultant. A computer system design based on logic or artificial intelligence.

extrapolation The estimation of a materials data values beyond the range of recorded measurements .

fields **Fields** contain the basic values which comprise a record. The content of a field is provided to users or their programs on request, say, `14Oct41`, when `John Jones'` birthdate is wanted. Computational processes can manipulate such values. The values are associated with a **type** and a **domain**.

front end A configuration of hardware and software that provides the preliminary interaction with the user including sign-on protocols, tutorial material, database directories, as well as the base system tasks such as line control, message handling, and transmission error control.

graphical database A data file that includes pictorial capabilities for representation of information in the file.

heterogeneous system A system composed of a diversity of computers, for example, IBM PCs, VAXs, SUNs, etc.

incompleteness Having only sparse information, i.e., only measurements for some of the combinations of key parameters.

information Knowledge acquired or derived; knowledge that can be collected, stored or filed, retrieved, and used to create other knowledge. To be useful, information must be pertinent, communicable, comprehensible, dependable, and convenient.

information hiding Withholding information, such as processing rules, from the user to keep users from employing such information and bypassing formal interfaces.

information management system (IMS) An organized body of knowledge, provided with appropriate inter-linkages between its subparts, a variety of access mechanisms and user assistance.

integrity The avoidance of accidental loss of consistency in data.

interface The interface is the common boundary between two physical devices or two distinct communications systems, two software programs, the computer system and the user, etc. The interface may be mechanical, such as the RS232 standard defining the physical plug connection between devices; or it may be a software program, for example, as the interface allowing messages to be passed between systems operating with different protocols or speeds.

interpolation The estimation of a materials data values beyond the range of recorded measurements .

knowledge base The component of a knowledge-based system that contains the systems knowledge. It is generally regarded as the most important component.

knowledge-based system (KBS) A computer program containing knowledge about objects, events, situations, and courses of action that emulates the reasoning processes of human experts in a particular domain. The components of a knowledge-based system are the knowledge base, inference engine, and user interface. It may also contain a Truth Maintenance System separate from the inference engine. Types of knowledge-based systems include rule-based systems and model-based systems.

knowledge manipulation Processing of the rules or semantics that describe the data, rather than the data itself.

knowledge representation The techniques used to store information in a knowledge base. Knowledge representation techniques include logic (also called formal methods), semantic networks, production rules, frames, and scripts. It is also common to see systems classified as storing Attribute-Value pairs, Object-Attribute-Value triplets, or Frames.

materials identification The data necessary to specify a material. For alloys, ceramics, and polymers, this includes chemical properties, such as, molecular weight, micro-structure, and heat treatment.

materials performance Mechanical, corrosion, and physical properties of a material that help determine the service life of a product.

materials properties The results of analyses of groups of individual, experimental, property data leading to single values or relationships indicative of the performance of a material. The basis ma be typical or nominal values based upon averages of individual test results, minimum values based upon some statistical analysis, or representative of curves deduced by materials experts.

materials designation cross-reference A table of synonyms for specification of a material. It can contain trade names, common names, and chemistry.

materials profile A listing of desired properties for a particular material with quantitative values specified whenever possible in terms of allowed range, minimum, maximum, etc.

metadata A detailed description of data in a database that provides systematic information for users, applications programs, and database management software. Data about data. Metadata can itself be manipulated and searched.

model An abstraction of a probable domain, to permit formal and simplified treatment of salient issues.

model-based system A type of knowledge-based system, usually intended for diagnostic purposes, that is based on a model of the structure and behavior of the device it is designed to understand.

object A collection of data to describe a single entity or event. The description of a single materials instance, including data for a set of temperature conditions may be organized into an object type. An object may also include procedures for retrieving, manipulating, or updating data within that object. Object data are typically not **normalized**.

object class or type Prototype for an object.

object-oriented paradigm Computing methods based on the concept of objects.

object-oriented programming Programming methods based on the use of items called objects that communicate with each other via messages. In terms of Knowledge-Based Systems, the main difference of Object-Oriented Programming from Frames is the communication between objects by the sending and receiving of messages. It is important to note that Object-Oriented Programming exists as a programming methodology separate from Knowledge-Based Systems. Object-Oriented Programming provides an ideal framework for the concept of concurrent, asynchronous operations.

performance The amount of time for a program, or a part of a program, to complete.

persistence Essential attribute of a database for which information is retained independently of program termination.

property data The calculated results of individual experimental property tests; a description of materials behavior under a stated condition or process.

prototype A model created for and used to simulate a small-scaled version of the larger, actual thing. Prototypes are largely used in research to test the functionality of an idea.

quality assurance A planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established technical requirements.

query A statement requesting the retrieval of information from a database.

query language A language especially designed to aid the user in accessing a database.

records A **record** is a collection of related **fields** containing elemental data items. Fields maybe related to each other because they describe some specific instance, say, a person, an object, or an event. A record of an employee file will contain such information as name, the social security number, date of birth, tax information, current job assignment, and rate of pay for a particular employee.

relational database A database in which the conceptual files are all relations. Related data items of single-valued, simple attributes constitute tuples.

reliability The ability of an item to perform a required function under stated conditions for a stated period of time. With respect to data, reliability indicates the dependability of the information in terms of accuracy and precision.

SQL A standard for a relational database language.

security The protection of computer hardware and software from accidental or malicious access, use, modification, destruction, or disclosure. Security also pertains to personnel data, communications, and the physical protection of computer installations.

semantics Semantics is defined to be the meaning of data. The semantics of individual data elements are expressed by linking them into objects, and the semantics of objects are expressed by having them participate in relationships with other objects. The linkages and relationships are described by meta-data.

structural information Formal description of relationships among data which can affect the structure of the database.

thesaurus The vocabulary and associated terms to be used in indexing to ensure retrieval. Tight vocabulary control with continual updating is essential to retrieval of pertinent information.

uncertainty The concept in a knowledge-based system of a piece of information or a conclusion having a likelihood somewhere between True and False. Some systems have built in mechanisms for handling uncertainly. The definitions used and the mechanisms employed differ considerably between systems.

user interface The virtual boundary between the user and the computer system. Its main components are the command language, formats for data representation, input and output devices, sequencing and timing representation, etc.

view A view is a subset of a database appropriate for a particular user or application. It is obtained by **selecting**, **projecting**, **joining**, and **aggregating** data from base relations in the database.

6. REFERENCES

1. [IEEE fltpt]
2. [Grat86] W. Grattidge, J.H. Westbrook, J. McCarthy, C.J.M. Northrup, and J.R. Rumble: "Materials Information for Science and Technology (MIST): Project Overview"; NBS Special Publication 726, 1986, Department of Commerce.
3. [Nort86] C.J.M. Northrup, J.L. McCarthy, J.H. Westbrook, and W. Grattidge: "Materials Information for Science and Technology (MIST): A Prototype Materials Properties Data Network System"; *Materials Properties Data-Applications and Access*, *ASME*, PV&P and Computer Engineering Division, 1986, p.167.
4. [Ozsoyopglu89] G. Ozsoyopglu, V. Matos, and Z.M. Ozsoyopglu: "Query-processing Techniques in the Summary-by-Example Database Query Language"; ACM TODS, to appear 1989.
5. [West88] Jack Westbrook: "Designation, Identification and Characterization of Metals and Alloys"; *ASTM International Symposium 1988*.
6. [Weis88] Volker Weiss: "Towards a Computerized Materials Characterization Expert System"; Syracuse University, Syracuse, NY 13210.
7. [Wiederhold 83] Gio Wiederhold: *Database Design*; McGraw-Hill, 1983.
8. [Wiederhold 86] Gio Wiederhold: "Objects, Views, and Databases"; IEEE Computer, 1986.
9. [Wiederhold 89] Gio Wiederhold: "The Architecture of Future Information Systems"; Stanford, CSD, 1989.