FIPS PUB 45

FEDERAL INFORMATION
PROCESSING STANDARDS PUBLICATION

1976 September 30

U.S. DEPARTMENT OF COMMERCE / National Bureau of Standards

GUIDE FOR THE DEVELOPMENT, IMPLEMENTATION AND MAINTENANCE OF STANDARDS FOR THE REPRESENTATION OF COMPUTER PROCESSED DATA ELEMENTS

CATEGORY: DATA STANDARDS

APPROVED AS AN AMERICAN NATIONAL STANDARDS INSTITUTE TECHNICAL REPORT BY COMMITTEE X3 COMPUTERS AND INFORMATION PROCESSING.

U.S. DEPARTMENT OF COMMERCE • Elliot L. Richardson, *Secretary*

Edward O. Vetter, *Under Secretary*

Dr. Betsy Ancker-Johnson, *Assistant Secretary* for Science and Technology

NATIONAL BUREAU OF STANDARDS • Ernest Ambler, *Acting Director*

Foreword

The National Bureau of Standards under the provisions of Public Law 89-306 has been given Federal-wide responsibilities for providing technical support and standards for the effective use of computer technology. In this regard, NBS is pleased to publish this technical guide on data standardization which was developed under the auspices of the American National Standards Institute through its Committees X3 and X3L8 sponsored by the Computer and Business Equipment Manufacturers. Additionally, this guide is used for the basis for the development of international standards and guidelines by the International Organization for Standardization. We hope that it will provide a useful technical base for those who are currently involved in or are about to begin standardization activities as part of their information management functions. Within the Federal Government, this guide is provided as a basic technical reference to assist Federal departments and agencies in the development, implementation and maintenance of standards for data elements and representations in accordance with the provisions of Part 6, Title 15, of the Code of Federal Regulations.

Harry S. White, Jr.
Associate Director for ADP Standards
Institute for Computer Sciences and Technology

Abstract

Provides technical and administrative guidelines for the development, use, and maintenance of standards for representing data elements used in computer based systems. Basic concepts and terminology of data standardization are provided in addition to evaluation criteria for assessing various coding alternatives. The guide is used as a basic reference document in the development of Federal and voluntary national and international standards for data. The guide was developed by the X3L81 Standards Task Group of the American National Standards Institute and has been adopted for use by the International Organization for Standardization.

Key words: American National Standards; computers; data; data elements; data processing; information; information processing; International Standards; standards; U.S. Government.

Nat. Bur. Stand. (U.S.), Fed. Info. Process. Stand. Publ. (FIPS PUB) 45, 48 pages (1976) CODEN: FIPPAT

For sale by the National Technical Information Service, U.S. Department of Commerce, Springfield, Virginia 22161

Preface

This GUIDE was prepared by the X3L81 Task Group (Data Standardization Criteria) of the X3 Sectional Committee (Computers and Information Processing) of the American National Standards Institute (ANSI) to assist ANSI groups and others in developing, using and maintaining standard representations of computer processed data elements. This GUIDE contains considerations intended to aid in the design and development of voluntarily adopted uniform practices and standards. The GUIDE is not itself a standard nor is any part of it to be considered mandatory or binding on any individual or organization. A definition of "data element" may be found in Appendix A.

The GUIDE is aimed at both administrative and technical levels of decision-makers. Both groups will require answers at some stage in their involvement with information processing to such questions about coding, codes and forms of data representation as, What are the current standards and where can I find out about them? Who has standardized common data related to my field of interest? How does one engage in data standarization? How can one develop optimum codes and other representations of data? This GUIDE offers some hints and special recommendations along these lines.

It should be pointed out that this report addresses alpha-numeric data only. It does not address, for example, geometric entity data. The material is organized into three main topical areas covering the background and concept of data standardization, codes and coding, and the current organization and activities of data standardization. This GUIDE is intended to be comprehensive while being "modular" in design to permit independent reference to individual sections as required.

None of the material in this document should be considered final. Much of the content is opinion. Some is controversial. Not even all the members of the X3L8 Subcommittee agree completely on all points. Nevertheless, this document does represent the current state of the art according to current authorities on the subject. Since the content is evolutionary, details of the readers' experiences and recommendations for improvement to this work will be appreciated.

Credit must be given to various people who have made the GUIDE possible. The major tasks of writing the primary text and supporting its completion have been borne by Harry S. White, Jr. of the National Bureau of Standards. Thanks are also due to Thornton J. Parker III of the Office of Management and Budget of the Office of the President for the valuable input provided the Task Group in the form of a number of documents and informally expressed views and insights. Additional credit is given to the Bell Telephone Laboratories, Inc., particularly to Arthur J. Wright and Lou Sonntag who furnished a major contribution in providing the Task Group with material contained in their document "Common Language Coding Guide.' Merle G. Rocke of Caterpillar Tractor Co. has provided invaluable assistance to the X3L81 Chairman both by making available substantial portions of his document "Data Codification Principles and Methods" and by volunteering much time, effort, and interest in preparing the GUIDE. Thanks are also forthcoming to the American Institute of Physics for the time made available to the X3L81 Chairman, Arthur R. Blum, which has made possible the completion of the writing and editing chores for the final version of this document.

Contents

Guide for the Development, Implementation and Maintenance of Standards for the Representation of Computer Processed Data Elements

SECTION 1. BACKGROUND

In recent years there has been an enormous expansion in the collection, processing, and exchange of data required for governmental, industrial, commercial, scientific, and computer processed technical information. Such information is essential to the life and operation of modern society.

To serve the vital need for improved communication of information within our society, further technological advances in computers, communications, and allied fields have continued to make possible an increasingly broader integration of data systems and ever greater aggregation and exchange of data among them. These advances have achieved both substantial cost reductions and important improvements throughout the spectrum of data systems and services.

However, the full effect of these advances will not be realized until the data processing and management communities reach a uniform understanding about the common information units and their expression or representation in data systems. This can only be done by developing and and applying appropriate standards.

The need for data standards is not new, but it is ever more pressing. The expansion of data needs within small and intermediate as well as large-scale computer systems—and the prospects of even more sophisticated electronic tools—re-emphasizes the need for data standards. Future applications dictate that action be taken to hasten their development and use. The GUIDE recognizes that standardization must never be undertaken for its own sake, but to promote greater efficiencies and economy, including those cases where the benefits derived are not always self-evident. The GUIDE also recognizes that the community of data users has already grown too large to expect a resolution of all problems. This GUIDE is therefore as a means by which those concerned with development and implementation of data systems can gain an appreciation of the need for more uniform practices and standards and can concentrate on the areas of greatest importance and potential benefit.

SECTION 2. CONCEPT

Data and information are fundamental to human communication.[1] No communication can occur without data or information having been transferred and recognized (or at least conveyed and accepted).

For the purposes of this GUIDE we will call the specific unit of information a data element. In information processing and exchange the data element is used to identify the intended field in a record. The data element thereby forms the fundamental building block out of which all information structures (records, files, and data bases) are made.

The increasing use of sophisticated and rapid methods of handling data has intensified the problem of dealing with meanings. Computerized society is not compatible with ambiguities of language or erroneous numbers. Man can no longer afford to apply ambiguous words or symbols to describe or to fill in the records used in daily life. Woeful is the life of the person—whether a customer, employee, employer, or taxpayer—who tolerates ambiguous meanings or entertains erroneous data values in such computerized records as credit cards, personnel files, purchase orders, tax forms, airline tickets, or utility bills. Considerable effort has been made in recent years

[1] To keep this document as informal as possible such formal distinctions as that between data and information are taken relatively laxly throughout the text. Where a strong contrast is needed, it is assumed that information is the holistic meaning, possibly derived from the assembly, analysis, or synthesis of the data into a previously unknown, unpredicted, and meaningful form. Data by contrast provide the atomic or molecular fragments to be connected.

to bring together the large-scale users of information in government and industry to achieve greater uniformity through clearer understanding (definition) and to facilitate processing of common data through standard data representations.

Standardization of the basic units of data requires that variations in the data being interchanged be eliminated or at least minimized wherever possible. It is generally possible at more or less expense to translate identical or very similar data of one system to the format or arrangement of a second system, despite differences between the names, codes, or other representations of the data elements used in the two systems. This is often the case in the trivial instance where two or more data element names refer to the same element, e.g., "Purchaser" or "Name of Customer." But where the same name is assigned to different elements or informational units translation may not be possible (for example, two fields may be called "Status," the first requiring marital condition in a personnel record and the other field querying the condition of a body at a hospital emergency admissions office).

Therefore, the basic unit of information, the data element, has a name which serves to identify it and to distinguish it from other data elements. Typical examples of the names of data elements, which serve to identify the meanings attached to the data fields in records, are "Applicant's Name," "Sex," "Date of Birth," "Place of Birth," "Number of Dependents," and "Social Security Number."

The data element, i.e., the meaning of the data field, can usually be identified by the name of the field. But it remains an open or empty or "unsatisfied" meaning until a specific value is applied to the field. For example, an "Applicant's Name" (data element or data element name) could be "Jones, John Adam" (data item). The meaning of the field is unassigned until the specific value (called data item) is given to it. The data items may be names as in the case above, or in other forms such as abbreviations (of variable length), codes (fixed length), or quantities. His "Sex" would be "Male" which could be coded "M." His "Date of Birth" could be "February 21, 1969" which could be represented "690221." His "Place of Birth" would be "Springfield, Illinois" which could be coded as "1782202." "Number of Dependents" would be "3." Nevertheless, the data standardizer is more concerned with the meaning of a particular field than with the particular names which are applied to it (although uniformity here is very important). For it is the meaning which must be unique and unambiguous and which requires a specific and precise representation.[2]

The gist of the problem of data standardization is that before meaningful data interchange can occur, the sender and receiver concerned must understand the identification and definition of the data elements and data items involved. The codes used in the interchange must be identified and defined. The position or location of the data elements in the record or form must be described.

Mutual understanding and agreement by the parties who interchange data form the basis of data standardization. But success of such standardization depends upon the comprehensiveness of the agreement. The greater the agreement on the national and international levels and the more inclusive the forms of representation, i.e., the names of elements, the codes, the coding methods, and the record forms that are standardized, the more effective will be the efforts in data standardization.

[2] It should be noted that the terms "data element" and "data item" are understood as identical with the COBOL data description terms "data item" and "data value," respectively, although the area of application of data standardization concepts is much wider.

SECTION 3. DATA CHARACTERISTICS

220-174 O - 76 - 2

SECTION 3. DATA CHARACTERISTICS

3.1. Introduction. This section treats the relationship between the data processed in an information system and the entities, events, and properties which the data represent.

Data standardization is essentially concerned with the representation of data elements. It is not things and their attributes which are of primary concern, nor are the data contents, syntactic structures and applications, or machine operations necessarily important in themselves. Although the data standardizer deals with the objects of the everyday world and with many formatting and organizational problems of systems, he sees these two realms from the viewpoint of the representational function of the data.

An essential task of the data standardizer is to obtain agreement on a method of representing data elements (e.g., natural language names, abbreviated names, codes, or even such special use indicators as the name or surrogates of the name of the data field on a record). However, there must be control of the relation between the world and the machine-sensible records and files. The data standardizer seeks to control this relation by working with the formatted data in question, by probing the data characteristics, by testing the suitability of the designation: names, codes, numeric data; perhaps also by structuring and otherwise organizing the designations. He works mostly with data already in records, data expressed in terms of controlled and uncontrolled vocabularies, code, and term sets.

Our task can be summarized in two questions: How can we manage and understand things of our world in terms of data characteristics? How can the data characteristics be defined, represented, and then formatted in data transmissions?

Practical answers to both questions can only be found in the everyday work and long-range accomplishments of the people engaged in data standardization. However, to facilitate their work, the present section on Data Characteristics will provide a tentative reply to the first question, while the remainder of the document will give some hints as to how to answer the second question.

3.2. Viewpoints, Things, and Classes. The world around us is made up of natural and manmade, physical and conceptual, as well as hypothetical or imagined entities. These entities have their own properties and can be related to one another. All of these individual things and notions can be known and designated, and therefore provide potential data to be recorded. For example, the political subdivisions we call countries, or states, or cities, or the physical objects that are arranged and labeled in a warehouse can be considered as data. The characteristics of these data correspond to the characteristics of the original things, or notions, or attributes.

Another example of such a relation might be found in the personnel record of a person where the data element (the meaning of the data field) is "State of Birth"; here the data item or value "Massachusetts," as one unique and ambiguous choice among the other allowable items for States, will satisfy the requirement.

Several processes must take place before a thing or notion can be meaningfully represented in a data processing system, and particularly before data records can be interchanged between systems.

3.2.1. Acceptance. There must be a common acknowledgement of the existence of thing, notion, or characteristics within the given context.

For example, we must agree that Maryland and Virginia exist and are "States of the United States" before they can be coded within the code set for states. Before attempting to communicate about a thing, it is essential to know ***that*** a thing is, to describe ***what*** it is, and to standardize the communication. Consequently, the existence of the object must be accepted by us before we can, for example begin to disagree about whether the boundary line between the two states mentioned is the high or the low water mark of the Potomac River.

3.2.2. Common Viewpoint. People must perceive an object and relate it to existing schemes or to familiar subject fields. However, to share the object with others there must be a mutual agreement about ***what*** it is, so that a common viewpoint concerning it may be established.

It is at this point that the boundary line between the two states becomes important, because uniform specification demands that the things be perceived and described in the same manner. Where the common viewpoint is lacking, specification and standardization may be impossible. For example, a common viewpoint is required to decide whether the Canal Zone, Puerto Rico, Guam, etc., are assigned to either the class "Countries of the World" or the class "States of the United States" or neither.

The standardization process cannot proceed until one has achieved a common viewpoint on whether the object in question is specified as an individual (a thing), or a class. For instance "The United States of America" can be the real individual that belongs to "Countries of the World." From another viewpoint it can be the class which contains "States of the United States" whose members are Alabama, Alaska, etc.

The issue of individual versus class overlaps the problem of uniqueness in the case of common names. For instance, "John Jones" is the name of twelve different individuals in a local telephone directory. Not knowing additional attributes, such as address, the problem of deciding upon uniqueness might be resolved by elimination. This could mean telephoning until the correct individual is located among the other members of the class "John Jones." But even when the individual is identified, true uniqueness is not established until a common viewpoint is determined as to which "John Jones" is meant—John Jones, employee? John Jones, father? Data standardization must resolve all questions concerning class membership. This requirement extends to classes, as in an industrial classification, and to individuals, as in a warehouse inventory.

3.2.3. Terms. A person cannot know and control things or notions unless he can designate them and use the facilities of language to convey his designation to others. Furthermore, the terms or symbols in the designation must also be understood. When we have the name, indicate, identify, describe and quantify (that is, to tell how much, how many, how large, or how long in time), we find ourselves involved in the complex field of semantics.

Data characteristics can be based on physical characteristics. Meaningful data may be derived directly from physical signs. For instance, analog instrument readings convert physical variations into a variety of representations and measurements. There are many other familiar representations of physical states which may be recorded, stored, or displayed in a variety of states and dynamic forms, such as motion pictures, photographs, or drawings.

Data characteristics can also be based on the data derived from such primary readings. One form of machine sensing can be translated into another, e.g., analog readings can be converted to a binary form, or on-line data can be subjected to various forms of interpretation. Such data are meaningful and may be denoted by terms. For example, signs of certain types can be interpreted, named, and quantified by word symbols and numbers. The data characteristics of all such signs exhibit clear cut digitally codable patterns.

Typically, however, data standardization is concerned with conventional symbols, particularly those which express word meanings and discrete quantities. These symbols are commonly applied to data structures, i.e., to the data items and elements, and to the logical records and files organized into data bases and systems.

In an area as complex as data standardization, problems of meaning that present communication stumbling blocks arise quite often. An example of a difficulty that could occur when assembling data items is the "language barrier." Different language designations for items of the element "Countries of the World" cannot cross specific language lines without causing confusion or total unintelligibility:

English Language Term	Native Language Term	
France	France	
Germany	Deutschland	
China	Chung Kuo	Depending upon
	Zhongguo	transcription scheme
India	Bhārata	

Influences stemming from general record keeping and data processing contexts, and especially from specialized "closed" systems tend to make the individual and associated terms more rigid, structured, and controlled than would be the case for natural language vocabularies. The clearest

example of structured vocabularies may be found in general classification systems, where all terms in the system are ordered. The ordering of terms can be discovered whether in its full or conventional name form, or in its condensed representation. Terms can be ordered intrinsically or extrinsically.

For example, data terms can have an ***intrinsic*** order given by an ordinal numbering system such as a catalog number used as the unique identifier for stocking and ordering purposes, or a license plate, or a street address number. ***Extrinsic*** order can be given to terms in a classified agreement where subject terms are ordered alphabetically according to their ranking within the scheme.

Quite often, unless code numbers are applied to the terms, there can be no intuitive way of knowing the interrelations of terms just by examination of the terms themselves. Most terms encountered in experience, such as words, names of persons, places and things are ***unordered.***

Ordering interrelates individuals. But regardless of whether the individual (member or members) of a family of terms is taken singularly or is related to the others, it is usually essential to know whether the term itself relates to:

(1) a single ***unique*** thing;

(2) a class or group of things that are accepted as a unity, a composite whole, or a manifold;

(3) many things.

Therefore, by understanding what the term relates to, the data characteristics of the term become uncontrollable. Control of the term makes it possible in turn to control the objects and motions referred to, as well as messages that contain the term. Such control extends to the term is its original mode of presentation, say in its full name form, or in alternate modes or representation, as in coded or abbreviated forms.

The ***criteria*** related to how the data standardizer copes with terms can be crucial. For example, to control the ***language performance*** of the terminology used, attention must be given to the denotative precision or expressiveness (as specified in the term definitions), uniqueness (or seen from the other side, zero ambiguity), compactness and cost in development and implementation of the whole set of terms.

3.2.4. Condensed Representation. Efficiency and economic considerations in data processing require that data elements be represented in a condensed and accurate symbolic form.

The data must be controlled in such a way that the objects and notions are effectively designated and identified, and their meaning is faithfully conveyed throughout the process of representation. The terms or names of the data contents must be abbreviated or coded according to specific rules, but cannot lose any of their precision or uniqueness . . . for human or machine processors . . . in any of their condensed forms. Thus, we expect that the "State of the United States" named "California" may be abbreviated as "CA" and coded "06" with increased efficiency and economy without detriment to the performance criteria mentioned above under "Terms."

Only by working with and through the four basic processes of definitions—acceptance, common viewpoint, terms, and condensed representation—can data description be formalized. For instance, it is possible to standardize a ***code*** for a unique item or a unique class only if:

(1) The uniqueness of the thing or class has been established;

(2) There is acceptance of the specification, description, limits, or properties of the thing or class;

(3) An accepted and unambiguous term is established for the thing or class;

(4) There is acceptance of the code as standing for the term.

3.3. Data and Data Representations

3.3.1. Fundamental Approaches to Data Standardization. The distinction between things, classes of things, and pure classes is so fundamental that it characterizes the individual approaches to data standardization.

In principle and in actual systems design, one of three methods derived from this distinction is often emphasized. Accordingly, the data tend to be treated as:

(1) a unit usually associated with its physical or at least nominal occurrence in data fields of records;
(2) a class based on intrinsic or assigned relations between units which belong to the class; and
(3) classes of information which form part of a defined classification scheme.

The definition of data elements, data items, and activities of data standardization can depend on which approach is adopted:

(1) ***The unit approach***—where the fundamental meaning of the data element is identified with the unit of meaning that occurs in the particular data field of a document or data record. The association is considered so close that the same name is given to the data field, the identifier of the data field, and to the contents as well as the meaning of the contents of the field. The basic unit of meaning, the data element, is duplex. It consists of a general part and a specific component (the data item). For instance, this approach maintains that there is a data element "Date of Birth" that is different from the data element "Beginning of Employment," although both elements will have common data items or values that follow the same formatting specification, e.g., the values of both may appear as "September 8, 1950" or "500908."

Standardization activities are based on the data items, so that uniform representations may be used for the most significant data elements. Consequently, although there are may data elements that apply, for example, to time or to countries, one does not attempt to standardize the data elements. but concentrates instead on the methods of formatting and representing the codes for sets of data items, e.g. (list of geo-political entities).

(2) ***The class approach***—where the data element is considered as a class or category, independent of its appearance or use in any particular record context. The class is considered to be denoted by the intrinsic or assigned relations or attributes of the data items, the members of that class.

Consequently, the class is abstracted from the concrete instances of its occurrence and use (although types of use may be documented and controlled). Considered as the fundamental unit of data, the class itself is standardized: it is treated as a semantic entity and may be analyzed, defined, put into thesauri or dictionaries, and controlled. The class "Date" will, for example, be considered the data element, to be defined and allotted certain allowable names, abbreviations or code structures (perhaps formatted as 720101, or as January 1, 1972). Its uses may be documented at "Date of Purchase" or "Date of Birth," which may or may not share the same name as the data field identifier.

(3) ***Classification approach***—where an entire subject field, perhaps the totality of human knowledge (as in a bibliographic scheme such as the Universal Decimal Classification), a library book location scheme (such as Dewey Decimal Classification), or an industrial classification (e.g., the Standard Industrial Classification), is considered as the main information unit. All particular instances of the component classes or entities then form subdivisions which are hierarchically ranked. Each class or entity may have its own code value, but one which is representative of its relative position within the total scheme. Data can then be either subordinated to a class (under this "subject heading") or comprise the class itself (subject display).

Standardization for this approach requires the specification and structuring of the main scheme and the precoordination of terms for the body of knowledge. It also requires rules for expanding subordinate segments of the scheme, and a method for coding classes and perhaps special attributes of classes (as in facetted schemes).

Common to all three approaches is the basic reliance upon the ***value*** of the unit of meaning (found primarily in a data processing context such as a data field), the value which we have called the data item or data variable above. The data item is always the expression of what is selected as the unit of meaning (or that which is considered fundamental) which is listed as one of the items in a code or other representational structure.

For the sake of simplicity and to preserve the elementary open relation which binds entities and attributes to a data field only by a non-committal linguistic tie, we will adhere to the unit approach through these Guidelines.

3.3.2. Data Elements. The data element is the meaning of the data field, and the data record which contains this field will be only as accurate as the data which it contains.

To ensure optimum accuracy, data handling systems are carefully designed to preserve the precision of the data characteristics throughout all operations. Trained specialists are employed to give particular attention to the design and organization of forms, reports, files, and data formats. Words and symbols used in the procedures and systems descriptions are carefully chosen so that effective communications are facilitated. In most instances, forms are so designed and partitioned that each element on the form can be completely described in detail. Instructions for completing or filling out the form are devised to permit the recorder to provide the needed information both accurately and without ambiguity. The individual units of information which are found in these boxes or fields on forms, records, and files have open meanings which require certain data for the meanings to be satisfied. These units and meanings are the ***data elements***.

A data element is a unit of meaning made up of two parts, a general component which designates the information required (something previously unknown and meaningful to the recipient), and a ***specific*** part which supplies the data required, i.e., that which when recorded indicates a particular fact, condition, qualification, or measurement. The specific part stated as a value or the representation of a term is called the ***data item***. Data items can therefore be expressed as names, abbreviations (including name truncation), codes, or numeric values. For example, the specific values associated with the general component "Color of Dress" can be expressed as a name "Blue," abbreviation "BL," or code "12." Alternatively, one could also use an instrument to measure the color temperature and express the result quantitatively, such "5500 ° K" (Kelvin).

3.3.2.1. Complex (Composite) Data Elements (Data Chains). The meaning of a data field is usually simple, i.e., the data element or the data element name connotes a singular object or notion, such as "Color of Dress" connotes "Blue" or "State of Birth" connotes "California." The basic meaning requires only one thing or notion to satisfy its unique intent.

On the other hand, some data elements are complex. Their total meaning requires a chain of secondary meanings and, as a result, a composite group of data items to be entered into the data field to fulfill their primary meaning.

For example, the data element named "Mailing Address" may require data items which express the notions of name, street number, street name, apartment number, room number, building number, organization, city, state, county, and ZIP Code. Similarly, the specific representations associated with the data element named "Birth Date" convey the notions of year of birth, month of birth, and day of month of birth.

3.3.2.2. Data Elements Used for Matrices (Variable Name Data Element). Related to the complex data element, although more highly structured by extrinsic ordering (see 3.2.3.), are the data elements used in matrices or tables or arrays of data elements (see especially 4.4.1.1.). The name of the matrix may be considered a complex data element which refers to or intends the subordinate data elements that form the headings of the rows and columns.

The subordinate elements are organized in arrays that are peculiar to the type of matrix at hand, for example:

Educational Level of ADP Management and Supervisory Personnel

Educational Level (V_1)	Management Category (V_2)		
	Data Processing	Systems Analysis	Programming
College Degree	49.1%	50.7%	34.8%
Some College	38.2%	38.6%	47.6%
High School Graduate	11.5%	9.6%	16.3%
None of the Above	.4%	.1%	.2%
No Response	.8%	1.0%	1.1%

In this case, the name of the data element and its specific value can be identified in the following way:

> The percentage of ADP management and supervisory personnel with educational level of (V_1) in management category of (V_2).

All the subordinate data elements in the matrix are jointly identifiable by that single complex name, and when values are supplied in the columns for the variables V_1 and V_2, each specific value of the matrix can be explicitly identified. For example:

> The percentage of ADP management and supervisory personnel with an educational level of (V_1 = college degree) in management category of (V_2 = system analysis) is 50.7 percent.

3.3.2.3. Primary Data Elements and Attribute Data Elements. The data elements within a data system collectively make up larger units of data called ***records***. Within the records (whether they be forms, reports, or logical computer records), we may find that at least one data element, which we will call the primary data element, stands out, and has a certain primacy and logical privilege over the others.

A primary data element is the element which serves as a unique meaning "key" to distinguish a particular entity from others.

The element is therefore used as an identifier for the entity or entities and is qualified by the other data elements in the record. In many such cases, the primary data element is at the same time a record key or provides a sort key in machine sensible records. For example, in a personnel record which contains information concerning a particular individual within the organization, the following data elements may be used: Social Security Account Number, Name, Date of Birth, Personnel Grade, Salary, Job Title, Organization Assignment, and Home Mailing Address.

If the organization is small, the name of the individual usually provides the unique identifier for him and serves as the primary data element which is qualified by the other data elements. In a larger organization where several persons may have identical names, the Social Security Number plus the name may furnish the primary data element or key to identify the individual. If necessary, two or more data elements may be used collectively to provide uniqueness, and each may be regarded as primary. The remaining data elements in the record then simply qualify or further describe the entity (whether it be a person, place, thing, or notion) which has been identified by the primary data element(s). Qualifying data elements are called ***attribute data elements***.

In the example above, Date of Birth, Personnel Grade, Salary, Job Title, Organization Assignment, and Home Mailing Address are attribute data elements.

Attribute data elements can be chained together or "nested." For, in some cases, attribute data elements may have qualities which also are identified in the record. In these instances an attribute data element may be qualified by another attribute data element. In a personnel record like the one mentioned above, we could find attribute data elements named "Spouse's Name" and Spouse's Birth Date"; the data element "Spouse's Birth Date" is an attribute data element of the attribute data element named "Spouse's Name."

Depending upon the structure of a particular record or file, what might be a primary data element in one record may be an attribute data element in another record. Likewise, an attribute data element in a given record could in another record be a primary data element.

3.3.3. Data Representations Other Than Codes. It was mentioned earlier that both the general and specific parts of the meaning of data, although especially the latter, can be represented in such various forms as names, abbreviations, codes, and quantitative (numeric) expressions. Blue as a specific value associated with the data element named "Color of Dress" can be represented as a name "Blue," as an abbreviation "BL," as a code "12," or can be measured and expressed quantitatively as "5500 K." Similarly, the general portion of the data element can be represented as a name, "Color of Dress," as an abbreviation "CLR-OF-DRESS," or as a code "COD." Each of these forms of representation, ***names***, ***abbreviations***, and general ***quantitative expressions*** have characteristics of their own which need and are given further explanation in this Section. Data codes are treated as fixed length representations and are discussed independently in greater detail in Section 4.

3.3.3.1. Names. Natural Language terms are the most common designators of data structures. As it was pointed out in Section 3.2.3.—Terms, the terms used for logical data structures, i.e., data items and elements, records, files, forms, and whole data bases, are basically built up of meanings. These meanings are indicated by a variety of representational expressions, such as names, abbreviations, special symbols, and codes. But names are generally the most suitable universal and familiar forms for representing the meanings of the data elements and, where non-quantitative, their data items. The principal function of names is for the identification of objects, qualities, quantities, and notions, for the purpose of aiding human recognition and manipulation of the things and ideas encountered in experience.

But it must always be remembered that any specialized use of natural language, such as for identifying the meaning of a data field or its content, is governed by the same laws and constraints as any other use of natural language. And natural language is notorious for its imprecision in conveying meanings uniquely and without ambiguity. For example, quite often the only clue to the exact meaning of a name is provided by the format, context, or overall situation within which the natural language name appears. For instance, the data element named "Grade" is used in the following records:

School Personnel Record:

Name	Grade
John Smith	4

Employment Personnel Record:

Name	Grade
John Smith	GS-12

College Transcript Record:

Name	Course	Grade
John Smith	Biology 251	B

However, the meaning of "Grade" is different in each case and requires a distinctive full name which in some way reflects the context within which the element is used. "Grade" implies "School or Class Grade' in the first example, "Civil Service or Personnel Grade" in the second, and in the last "Course Grade." In the interchange of data among various data systems or even among the components of the same system, it is necessary that the context in which the names are used be known and specified (explicitly or by default) before communications can be accomplished unambiguously.

The proper context can be established and defined in several different ways: (1) The data elements can be related to the larger context in which they appear, i.e., to the particular records or reports in which they are used or to the primary data elements which they qualify; (2) The data element name can be expanded or otherwise modified to include essential words which establish the proper context; or (3) The definition or explanation of the data element name can mention in which context the data element is used. System descriptions and documentation often employ combinations of these techniques to describe data elements.

In effect, a data element may have more than one name by which it is identified. The same is true for the names of specific data items associated with a data element. The names may be the actual names used internally on the reports and forms in a particular system, or even the field labels or name tags by which the same element is identified. Alternatively, they may be more universally understandable names, perhaps full explicit names with appropriate descriptions, which should be used to communicate the data element outside the particular system. Both the local (or internal) names and the interchange name (the explicit form which indicates the full context) must be identified in any complete data system description.

Both natural language and the terms which name and describe data characteristics reflect the real and conceptual world which contains all of the things and notions of human experience. But it would be a mistake to assume that natural language is or necessarily should be identical with data terms. The language of data terms is in one sense not natural. It is called into being in order to identify only those objects and ideas which find their way into records and other data structures. However, it also borrows heavily from natural language for designators or descriptors to name, identify, classify and otherwise describe much of its contents. As such, it is a subject of natural language. But the data terms also draw upon a variety of formalized representations such as highly structured code sets for rigorously unambivalent connotation of their meanings.

However, even if the contents are different, the processes used in the language of data terms coincide completely with those of natural language when selecting and assigning names for data structures. Therefore, the naming of such data structures as data items and elements involves all the linguistic description and prescription that would apply to naming anything else. Names are called nouns in grammar. Hence, the grammatical conventions that apply to nouns and related word formations also apply to the data item, element, record designator or descriptors.

The assignment of names for data structures must be based upon a variety of considerations. These include grammatical specifications. These grammatical specifications for nouns as parts of speech, and the various criteria of language performance mentioned at the end of Section 3.2.3: (1) denotative precision or expressiveness of the noun or noun embedded syntactic formations; (2) uniqueness of reference, that is, does the data name or noun refer to: (a) a simple unique thing, (b) a class or group of things accepted as a unit, a composite whole, or a manifold, or many things; (3) compactness of expression; and (4) cost in development, implementation, and maintenance of the whole vocabulary or set of names. For additional guidance in developing acceptable names, refer to ISO document R704-1968, "Naming Principles."

Grammar

The subject of the grammar of nouns and related formations is far too extensive for exhaustive treatment here. But a brief review of its cogency in the naming of data structures is apt and, it is hoped, will serve to stimulate further inquiry on the part of the reader.

The number of things and notions that people see, even from a common viewpoint, is greater than the number of nouns that people use to designate them. One result of this was shown above in the use of the noun "Grade." A noun can have more than one meaning: each such noun may be a name for two or more things. A second result of the lack of available nouns can be that some names are not single words. A name that is not a single word is not strictly speaking a noun, but rather a syntactic formation where the noun is embedded as a nucleus in a name "cluster." The noun is the essential element in the cluster. For example, the ***noun nucleus*** in the data element "State of Birth" is the noun "State," where the other words are ***modifiers*** of the nuclear or principal noun.

Therefore a noun cluster is a grammatical construction which contains a noun as its nucleus, preceded and/or followed by modifiers of the nuclear noun. Nouns may appear singly or as one of several words in such nuclei, and are characterized by their singular or plural forms (usually ending in -s or -es, although some nouns have irregular plurals).

Nouns can be modified by various modifiers or adjectival units that consist of a single word, or one or more groups of words. A variety of modifiers occur, such as ***determiners*** (of uniqueness or possession), as the articles ***the***, ***a***, ***your***; ***numerals***, such as "***First*** Position Held" or "Choice ***One***"; ***adjectives*** such as "***Principal*** Function"; noun adjuncts, as "***Data*** Name"; or ***phrases*** as in "***State of Birth,***" or "Status ***at Time of Resignation.***"

A noun can be the name of two or more things in two ways. First, there is the way discussed above for the element "Grade," and then there is another, still more far-reaching manner: Two things can have the same name because people recognize that both things are in some sense ***the same.***

If different things are not the same, each is unique, and if it has to be named, generally deserves a ***proper noun*** to identify it. On the other hand, if we recognize sameness, and find that the same name can be applied to two more things, we are dealing with a class. The name applied to the class or to each member of this group is a ***class*** or ***common noun.*** The things or notions that are named by class nouns can be counted: if the class is void, the number of members is zero; if it is a singleton, the number is one. If there are more members than one, a variety of grammatical number

220-174 O - 76 - 3

words can generally be attached to the member nouns, including ordinal numbers, indefinite articles, etc. Nouns can also denote a multiplicity of members in the case of mass or collective nouns such as the word "Carbon" when describing the composition of diamond, coal and graphite. The same word can be a class noun in the item "Carbon" for data element "Office Supplies." The mass noun never requires an article.

The composite noun is a syntactic formation which includes the nuclear noun cluster. For example, a data processing operation might be called "Personnel Data Throughput" or "A Sort by Name"; The file may be "Salesforce by Major City."

The attributive elements include single words (***Personnel Data***) and prepositional phrases (***by Major City***). The formations which cluster about the nuclear or principal noun can become quite complex. Various grammatical forms can adhere to these clusters, e.g., "***Current Awareness Alerting*** Service," where there is a composite adjectival modifier which contains a ***verbal noun*** (alerting), all of which are attributive to the nuclear noun "Service." ***Possessive nouns*** can be considered under this heading, as in the element "***Vendor's*** Name."

The composition of name parts presupposes a conciseness and compactness of expression. Precision, clarity, and familiarity of the words used in names cannot be compromised by the need for compression, and some degree of optimization may be required.

The nouns used for naming entities at various levels in the data structure are not absolute, and can often be used at other levels. For example, "Virginia" may be the name of the data item for the data element "State of Residence." It becomes an important noun modifier in the data element name "Population of the State of Virginia." The question of level assumes great importance in the hierarchical ranking of names, as appears in a classification system. The effort needed to organize the name structures according to the levels required by class distinctions then becomes a significant cost parameter.

The section on names would not be complete without mentioning a few other cost factors involved in the overall process of naming data structures. Development costs as well as operating costs can apply to:

data collection—entity search, naming, definition, and preparation for encoding;

name control— the compilation and implementation of vocabularies in the form of dictionary entries, lists, thesauri, classification schemes;

maintenance— updating procedures, organizational assignments of term and coding control where the centralization versus local file trade-offs are vital; providing access to file contents possibly through publication, display terminals, etc.

Name Definition

A central issue in data standardization is the meaning of the data terms rather than the word forms and word syntax. As a result, the definition of names is of major importance. Improper definition can seriously impede data interchange.

The cost of definition can be very high. But if definition is not performed from the most general yet most common point of view, data interchange may still not be possible. Certain data systems which have developed highly standardized defined vocabularies in unique controlled environments may not be able to converse with systems in different environments. Although term definition may be present, a universal viewpoint related to the names and their definitions may be lacking. For example, both systems in two hypothetical different environments may use the same code set and format for "Date," perhaps expressed as "730325." Yet "Shipping Date" from a military point of embarcation will not have the same sense as "Shipping Date" for the local delivery of a small commercial parcel. A contractor who deals with both environments may find that there cannot be a universal definition which accommodates both meanings. Two definitions may be required.

3.3.3.2. Abbreviations. An abbreviation is a shortened form of a word, term, or phrase. Abbreviations improve the communication process by presenting information to be read by humans quickly, accurately, and with ease. The abbreviation saves space and time, and it provides a convenient, compact way of reducing long and complicated words or phrases that may often be repeated.

The names of data structures, particularly the data elements and data items, frequently lend themselves well to abbreviations. Nevertheless, there is no widespread standard method of abbreviation. Among the styles and forms of abbreviations, there are two tendencies toward commonality. First, individual disciplines and organizations produce lists of abbreviations that become authoritative for their industry or special field of interest, such as the list used in the publications of the American Chemical Society. The second movement is to establish rules and algorithms for the generation of uniform abbreviations. An example of this technique may be found in the American National Standard for the Abbreviation of Titles of Periodicals (ANSI Z39.5-1969).

The difference between abbreviation and other forms of coding is not self-evident. Abbreviations are generally developed for human handling, since codes are more suited for such machine applications as on computers, card and paper punches, and similar keyboarding devices, as well as on communication machines. Nevertheless, many of the same basic criteria are applicable to both these forms of representation and to the methods of deriving them.[3]

(1) Each word in the name should be compressed to require as little keyboarding time and storage space as possible.

(2) There should be no loss of discrimination and uniqueness between the original name and the compressed representation.

(3) The compressed forms should be at least as readily recognizable, learned and recalled by humans, and as easily transmitted ***without error*** as the original names.

(4) To retain optimum discrimination the compressed form should be mnemonically similar to the original name.

(5) Whenever possible, the abbreviated form should be capable of being systematically transformed back into the original name when desired.

(6) Whenever possible the abbreviated words should sort in the same alphabetical order as the original name.

These requirements are basic in the sense that at least two must be used in any efficient abbreviation scheme or code structure, but are ideal in the sense that all can rarely be applied at the same time.

At the risk of arbitrariness, the abbreviation may be generalized from its common appearance in text, and defined as a mnemonic code with a variable length. When existing or constructed abbreviations have a minimal number of characters, will alphabetize in a desired sequence, and are easily manipulated as well as mnemonic, then the abbreviation set ***is*** identical to the code.

Thus defined, several techniques for deriving abbreviations are commonly used:

a. ***contraction***—the shortening of a word, syllable, or word group by systematic omission of an internal letter or letters. For example, "abbrvtn" for "abbreviation."

b. ***truncation***—the shortening of words by the omission of letters at either end. For example, right end truncation retains the proper number of characters at the left end and deletes all the remainder up to the end of the word, e.g., "Wash" for "Washington." Left truncation drops letters from the left end, e.g., in the list:

"h, A.R.	for	"Smith, A.R.
h, Dick		Smith, Dick
h, Paul		Smith, Paul
m. Thomas"		Smith, Thomas"

c. the formation of ***acronyms***—forming words from the initial letter or letters of each of the successive parts or major parts of a compound name. For instance, RADAR for Radio Detection and Ranging.

[3] Cf. Charles P. Bourne, Methods of Information Handling, p. 46 (John Wiley & Sons, N. Y., 1963).

In the absence of any universal authority for abbreviations one can recommend the use of the abbreviation list and individual entries in Webster's New Unabridged International Dictionary. However, care must be exercised. Only unique and unambiguous abbreviated word forms should be assigned to the data terms in question. The elimination of letters within words or phrases tends to produce undistinguishable character clusters, e.g., "DA" may be an abbreviation for "Day," "District Attorney" "Department of the Army." "No" may be an abbreviation for the chemical "Nobelium," for the direction "North," or for the word "Number."

In addition to the criteria basic to both abbreviations and other codes listed above, the following suggestions may be useful in the development of uniform abbreviations for data terms:

Abbreviate significant words in the name, allotting a consistent maximum number of words to be used and word types (e.g., articles, conjunctions, prepositions) to be dropped.

Words with a small number of characters (say, four or five) when used alone should generally not be abbreviated.

For mnemonic purposes the first letter of a name word should be present in the abbreviation.

Initial capitals or all capitals should be used.

Consistency is of major importance: either use periods at the end of all abbreviated words or omit final periods (preferred, since people often inadvertently omit them).

The same abbreviation is used for singular and plural forms of the same words.

Given a choice of deletion, consonants are more important than vowels in the abbreviation, initial letters than final.

If a conventional abbreviation already exists, it is preferrable to a newly developed one, provided that it conforms to the other criteria mentioned above.

The abbreviation should be as universally understandable and recognizable to human beings as possible and not merely provide a jargon "shorthand" version of the name, for example, one should avoid giving the initial letters of a data term such as "INOC" for "Identification Number of Consignee."

In a compound name, the order of abbreviated words should follow the same sequence as the original name.

Abbreviations must be developed with consideration given to existing software constraints. For example, in a COBOL environment it is essential that non-connected words should not begin or end with a hyphen, must have at least one alphabetic character, and names used as tags must be restricted to a word-length no greater than 30 characters.

3.3.3.3. Quantitative Data. Quantitative data provide a numeric answer to such questions as "How much?" "How many?" "How large?" "How long in time?" or "How frequently?" The numerals in quantitative data represent numbers which express the limits of quantities and magnitudes. The meaning of the quantity or magnitude is a data element and is connoted by the name of the element, e.g., "Length of Runway." This meaning is satisfied by furnishing the appropriate numeral, which is the proper data item for the specific element, e.g., to satisfy "Length of Runway" one could specify "800 feet." Numerals often include a wide range of expressions, such as whole numbers, ratios, exponents, fractions, and constants.

The degree of preciseness needed within a given system determines the form of the particular quantitative representation. For example, the unit cost of certain supply items may be expressed in dollars, cents, and mills as $1.035, the sales price of these items is expressed in dollars and cents as $1.04. The inventory of a business may be expressed in dollars as $82,520 or in thousands of dollars as $82.5. On the extreme end of the economic scale are the expressions of the gross national product or national debt which are expressed in billions of dollars. Similarly, the precision of irrational numbers (numbers which cannot be exactly expressed as a ratio of two integers) will depend upon the specificity required. Pi, which is used as the symbol to denote the ratio between the diameter and circumference of a circle, may be expressed as 3.14, 3.1416, or 3.14159265 . . . depending upon the requirements of the system.

The same value can also be expressed numerically in several different ways. For example, 3½ hours can be expressed as 3.5 hours, 3 hours 30 minutes, or 210 minutes. Likewise, 100 meters can be expressed as 0.1 kilometers or 10,000 centimeters or 100,000 millimeters.

However, upon closer examination of quantitative data it is found that all have certain fundamental characteristics which can be described. These are:

(1) All quantitative data expressions have some form of numeric expression. The most common form used in human-to-human communications is that of the decimal (base 10) system. However, numbers represented in computers are generally converted to binary form (base 2) or binary coded decimal form. Some computers have the capability of representing two or more binary coded numerals in a single computer word. This type of expression is commonly called packed numeric representation.

(2) All quantitative expressions have an expressed or implied radix point (called decimal point in decimal representations). Generally, when a quantity is expressed without a radix point, it is interpreted to be an integer (a whole number).

Some computers have a floating point capability. This capability allows a wide range of magnitudes to be represented to a given precision by means of a limited number of digits. For example, in a decimal system which uses only three digits to represent significant digits, the number 134,000,000 ($= 1.34 \times 10^8$) would appear as 1.34, 8 (where 1.34 are the significant digits and 8 is the exponent of the base 10). Likewise, 0.0134 ($= 1.34 \times 10^{-2}$) would appear as 1.34, −2, and 1.34 ($= 1.34 \times 10^0$) would appear as 1.34, 0.

(3) Normally, quantitative expressions have an expressed or implied sign (+ or −). Usually, unsigned quantities are considered to be positive. When the quantity is negative, the sign is usually expressed (explicit).

(4) Quantitative data representations, which indicate measurement, usually require an expressed or implied unit of measurement (e.g., dollars, meters, degrees, percent, etc.) Some measurements, however, do not have a unit of measure expression (e.g., dress, hat, and shoe sizes.)

Quantitative data reflect the degree of preciseness, approximation, range, or tolerance either as part of the representation or in the definition of the data element (e.g., a ship's position may be defined to be accurate within plus (+) or minus (−) one nautical mile, or the result of a computation may be expressed as being accurate within certain maximum (+) or minimum (−) limits).

Quantitative representations are frequently rounded in systems applications. Rounding is a systematic way of shortening an expression (e.g., Pi expressed as 3.14159265 . . . when rounded to four decimal places would be represented as 3.1416).

Another method of shortening is that of truncating a representation (this applies to indicative as well as quantitative expressions). Truncating simply is the act of dropping a certain number of characters (or digits) from an expression (e.g., Pi expressed as 3.14159265+ when truncated to four decimal places would appear as 3.1415). Both rounding and truncating degrade the preciseness of expression.

In the interchange of information among or between systems, it is essential that these fundamental characteristics of quantitative data be thoroughly described and understood by both the sender and receiver.

3.4. Summary (Section 3). The following is a summarization of Section 3, DATA CHARACTERISTICS. Presented is a short statement that attempts to summarize the major concepts presented in Section 3 above.

3. Data Characteristics

3.1. Introduction. Data standardization is concerned with:

analysis and control of the relation between the data processed within an information system and certain entities, events, and properties in the world of human experience;
representation of these things and notions by names, codes, and numeric description.

3.2 Viewpoints, Things, and Classes. Characteristics of data correspond with appropriate degrees of precision to the original things, notions, or attributes.

3.2.1. **Acceptance.** Common acknowledgement of the existence of things, notions, or characteristics is essential to begin data collection.

3.2.2. **Common Viewpoint.** To achieve standardization the objects concerned must be perceived from a common viewpoint, related to familiar subject knowledge, specified and subjected to mutual agreement concerning what it is. Definition must be made according to the uniqueness, individuality, and class membership of the objects of the data.

3.2.3. **Terms.** Objects to be standardized, that are seen from a common viewpoint, must be named, described, and quantified. Typically data standardization is concerned with conventional symbols, particularly with terms which express word meanings and discrete quantities. Data terms relate to data structures, i.e., data items, elements, and logical records and files that are organized into data bases and systems.

Terms may be ordered intrinsically or extrinsically, or be unordered. To be standardized they must relate to (1) a single unique thing, or (2) a class or group of things accepted as a unity, a composite whole, or a manifold, or (3) many things.

To control the language performance of the terminology used, attention must be given to the denotative precision (accuracy), or expressiveness (definition), uniqueness, compactness, and to the cost of developing and implementing the set of terms.

3.2.4. **Condensed Representation.** A condensed and accurate symbolic form is needed to represent data terms. Objects and notions are effectively designated and identified and their meaning is effectively conveyed by abbreviating and coding their names. Four minimum coding requirements are given.

3.3. Data and Data Representations

3.3.1. **Fundamental Approaches to Data Standardization.** There are three methods of approaching data standardization based on the distinction between particulars, classes of individuals, and pure classes: (1) the unit approach; (2) the class approach; (3) the classification approach.

3.3.2. **Data Elements.** The data element is the meaning of a data field, which may also be found to be represented in records, forms, reports, and other formatted data in files. It is composed of two parts, a ***general*** component and a ***specific*** part (the value or data item).

3.3.2.1. **Complex Data Elements.** A complex data element entails a chain of secondary meanings and, therefore, requires representation by a composite group of data items, as "Mailing Address" requires name, street number, street name. . . . city, state . . . etc.

3.3.2.2. **Data Elements used for Matrices.** The name of the matrix is used as a complex data element which refers to or intends surbordinate data elements that form the headings of the rows and columns.

3.3.2.3. **Primary Data Elements and Attribute Data Elements.** The data element used as an identifier for the given entry or entities and which is qualified by the other elements in the record is the ***primary*** data element. The element or elements which qualify it are ***attribute*** data elements.

3.3.3. **Data Representations Other Than Codes.** The general and specific parts of the meaning of data can be identified and represented by names, abbreviations, and quantitative expressions.

3.3.3.1. **Names.** Names are the most universal and familiar forms for representing the meaning of data elements and items. Specifically, they provide natural language identification for the data counterparts of objects, qualities, and notions within reports, forms, and record data fields. Language ambiguities may be reduced by proper use of grammar and possibly eliminated by reference to context or with definition. Differentiation is needed where the same data structure has more than one name, just as when one name applies to more than one data structure. This problem may be resolved by definition, although there are situations where more than one definition is required.

3.3.3.2. Abbreviations (Variable length representations). The abbreviation is a shortened form of a word, composite term, or phrase considered as a variable length mnemonic code. Basic criteria are given for word compression. Several techniques are described for the derivation of abbreviations, particularly by contraction, truncation, and the formation of acronyms. A number of further suggestions for the derivation, formatting, and style of abbreviations are offered.

3.3.3.3. Quantitative Data. Quantitative data provide a numeric answer to such questions about quantities and magnitudes as "How many?" "How large?" "How long in time?" or "How frequently?" The degree of precision and various quantitative expressions are treated as significant data characteristics.

SECTION 4. BASIC CODING METHODS

4. Basic Coding Methods

4.1. Introduction. This section provides a description of basic coding methods, including advantages and disadvantages of each method. It is intended to assist data standardization task groups in selecting the most appropriate code structure for each particular application.

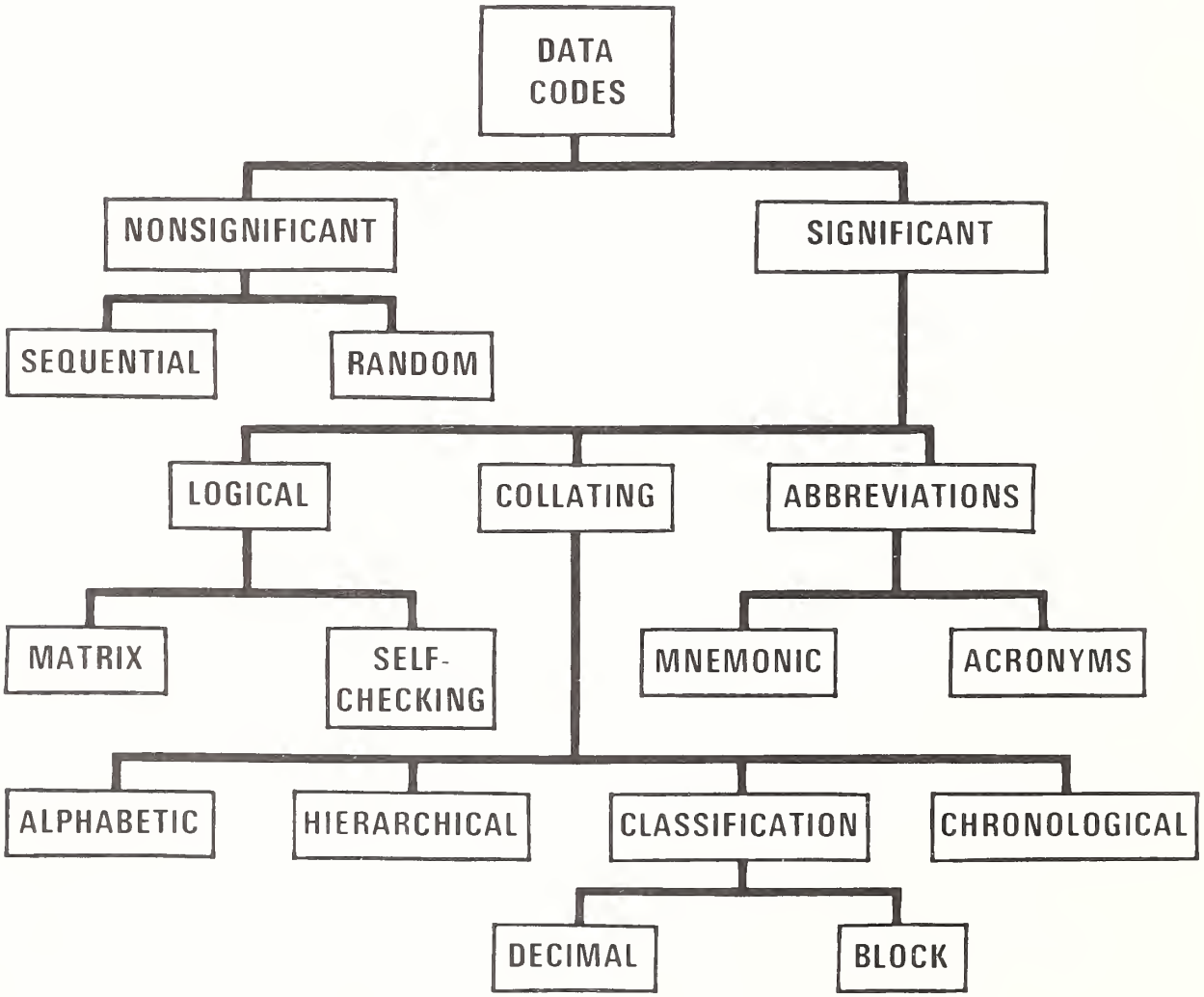
A code is an ordered, shortened, fixed-length data representation. Codes are designed to provide unique identification of the data to be coded. To accomplish this, there must be only one place where an identified word or phrase can be entered in the code structure and, conversely, there must be a place in the code for everything identified. It is imperative that this "mutually exclusive" feature is built into any code structure.

The choice of code structures is fairly extensive. The following information, however, should help lead toward selection of the best method.

Section 4.2. is a chart outline of the coding methods discussed in Section 4. This set of code structures is not entirely comprehensive, but does include all the significant types. Further, these are "pure" codes—and many data codes are actually combinations of these basic types.

For additional information on coding methods as well as indepth reports on psychological studies, etc., from which much of the content of this GUIDE was taken, refer to Appendix C, "BIBLIOGRAPHY."

4.2. Forms of Data Codes



4.3. Nonsignificant Codes. Individual values of nonsignificant codes are meaningless without some defined relationship to another entity set or sets and are assigned only to provide unique identification to the entities coded. The sequence number and the random number are the two most commonly used nonsignificant codes.

4.3.1. Sequential Code

Sequential (Serial or Tag) Number. The simplest to use and apply, the sequential method of coding is merely the arbitrary assignment of consecutive numbers (beginning with, say, "101") to a list of items as they occur, just as employee numbers might be assigned to employees as they are hired. The code value has no signi ficance in itself but does uniquely identify the entity.

This method makes no provision for classifying groups of like items according to specific characteristics and cannot be used where such requirements exist. It is practical only for coding entity sets where the only requirement is a short, convenient, easily applied representation.

The advantage of the sequence code is its ability to code an unlimited number of items by using the fewest possible code digits. As new items occur they are simply assigned the next-higher unused number in sequence.

This number is frequently used to give a unique reference number to entities (e.g., countries) which are composed of several elements identifiable in their own right (e.g., states, cities). With proper controls it is extremely useful in many applications and usually exists as a part of other more specialized coding schemes.

4.3.2. Random Code. The term random number is frequently applied erroneously to the sequential code just described. The difference between a sequential and a random code is the number list from which the code values are assigned. The random code is drawn from a number list which is not in any detectable order or sequence. There are computer programs available to produce these random number lists. Each additional item to be coded is given the next number in the random list. This method forces the coder to look up the next number on the list because there is no logical way to predict what the next number will be when the last used number is known.

In a sequential list, if 200 were the last number assigned, the next one will be 201. The next number on a random list might be 163.

This forced look-up is supposed to reduce errors in coding, but in actual use it tends to introduce problems of control. Properly controlled sequential lists have proved less error-prone than random lists.

4.4. Significant Codes. Codes are designed to provide unique identification of the words or phrases being coded. In other words, in a coded set of entities, no two entities should be assigned the same code. If in addition to providing unique identification of entities a code is so designed to furnish additional meaning, this type of code is called a significant code. The additional meaning supplied by the significant code can yield logical significance, collating significance, or mnemonic significance.

4.4.1. Logical Codes. Individual values of logical codes are derived in conjunction with a consistent, well defined, logical rule or procedure (algorithm). Two examples are the matrix code and self-checking code.

4.4.1.1. Matrix Code. This code is based on x-y coordinate locations or longitude-latitude coordinates. It is useful in coding two component relationships. Code values can be formed by assigning the "XY" coordinate numbers or by assigning sequence numbers. (The squares in the example are numbered both ways for illustration.) A code value is merely read from the appropriate square in the table when assigning code values to an entity. When decoding, the code value is located in the matrix and appropriate XY attributes are obtained. For example:

X \ Y	1 = Round		2 = Square		3 = Rect.		4 = Oval		5 = Irreg.	
1 = Round	11	(01)	12	(05)	13	(09)	14	(13)	15	(17)
2 = Square	21	(02)	22	(06)	23	(10)	24	(14)	25	(18)
3 = Hex.	31	(03)	32	(07)	33	(11)	34	(15)	35	(19)
4 = Oct.	41	(04)	42	(08)	43	(12)	44	(16)	45	(20)

(Note: Numbers in parentheses are merely the matrix location sequence numbers; the other numbers are the resulting code values.)

4.4.1.2. Self-Checking Codes. It is possible to append to a code an additional character which serves the purpose of checking the consistency or validity of the code when it is recorded and transferred from one point to another. This character, which is commonly called a ***check character,*** is derived by using some mathematical technique (algorithm) involving the characters in the base code. The check character feature when utilized provides the capability of detecting most clerical or recording errors. These errors are categorized in four types, i.e., transposition errors (1234 recorded as 1243), double transposition errors (1234 recorded as 1432), trar scription errors (1234 recorded as 1235), and random errors (1234 recorded 2243) which are multiple combinations of transposition and transition errors.

Several different techniques are employed to generate the check character. Each method has its advantages and disadvantages based upon the complexity or capability of the equipment involved in the data system and the degree of reliability essential to the particular application. For purposes of demonstrating the technique, one typical system which is prevalently used in credit card applications is described below:

Given the base code 457843, the check character is derived in the following way.

Each position of a character in the base code is given a weight (the amount by which it is multiplied to derive a product). In this example, the least significant position (rightmost position) is given a weight of 2, the next one, and so forth (alternating positions 2, 1, 2, 1, 2, 1. . . .) until all positions are assigned weights.

4	5	7	8	4	3	(base number)
1	2	1	2	1	2	(weight)
4	10	7	16	4	6	

Each character in the base number is multiplied by its weight producing the above products.

The individual digits of these products are then added to produce a sum of the digits:

$$4 + 1 + 0 + 7 + 1 + 6 + 4 + 6 = 29 \text{(sum)}$$

The sum is then divided by 10 which produces a quotient of 2 and a remainder of 9. (10 is referred to as the modulus, i.e., the number which is used to divide the sum of the digits to arrive at a remainder) :

$$29 \div 10 = 2 \text{ plus } 9 \text{ remainder.}$$

The remainder is then subtracted from the modulus (10 in this case) to produce the check character

$$10 - 9 = 1 \text{(check character)}$$

Thus the base number plus the check character would be

$$4578431$$

In application, the full number including the check character is recorded. The check character is then used in the following way to determine the validity or consistency of the recorded number.

Weights are assigned to the positions as before except the check character is given a weight of 1 and other positions are alternately assigned weights of 2, 1, 2. . . .)

4	5	7	8	4	3	1	(base number plus check character)
1	2	1	2	1	2	1	(weights)
4	10	7	16	4	6	1	(products)

Products are generated as before.

Digits are added as before.

$$4+1+0+7+1+6+4+6+1=30$$

This sum is then divided by the modulus (10), producing a quotient of 3 and a remainder of 0.

$$30\div 10=3 \text{ plus } 0 \text{ remainder}$$

Now examine the remainder. If it is zero, then the number checks. If other than zero, an error has been detected.

This particular self-checking system will detect 100 percent of all transcription errors, 97.8 percent of single transposition errors, and 90 percent random errors. It will not detect double transposition errors. For additional methods, refer to the text on error detecting and error correcting codes in Appendix C.

4.4.2. Collating Codes. Collating codes are by far the most directly useful and the most frequently used. The collating code structure is designed so that when sorted by the code number, the items represented by the codes are placed in a predetermined sequence. This sequence is frequently the sequence of the output required from the computer for optimum use by people.

4.4.2.1. Alphabetic Codes. For maximum effectiveness, alphabetic coding requires placement of all items in alphabetic sequence, then assignment of a code of ever-increasing value. Future sorts on the code put the items in the original alphabetic sequence. For example:

01—Apples
02—Bananas
03—Cherries
04—Dates

Normally, space is left between each item for future expansion. This code has some very strong points in its favor:

- Ease of sorting into desirable output format.
- Ease of maintenance.
- Accessibility to the code list without initial encoding.

Unfortunately, this code has some disadvantages that can result in problems that are extremely expensive to correct. This is especially true in large, scattered data systems where high rates of corrections or additions are necessary to maintain the list.

These disadvantages include:

- The necessity of coding the entire item list at one time to get reasonable spacing for new entries.
- Crowding that requires renumbering to maintain sequence of new entries.
- Relatively short life.
- The necessity of central control of number issues.

This code does, however, have a very useful place. Proper system design can utilize its good points and eliminate many of its shortcomings for certain applications.

4.4.2.2. Hierarchical Codes. The hierarchical code is a collating code which ranks entities or attributes by relative levels. It is very useful for many diverse applications. In its simplest expression, the hierarchical code arranges items in a predetermined sequence. The sequence may be increasing weight, length, diameter, or other single attribute of the items.

As code requirements become more complex, pure hierarchical coding is seldom sufficient for large systems. New ways to create hierarchies have been developed using the basic technique in combinations with other codes. Hierarchical codes are still of great value in specialized applications or supplementary to a larger code system for indicating increasing values, organization structures, or levels of data summary control.

4.4.2.3. Chronological Codes. As the name implies, a chronological code is assigned in the order of events so that each code has a higher value than the last code assigned. This is essentially the same approach as nonsignificant sequential. The difference is the attachment of time significance to the code number assignment.

4.4.2.4. Classification Codes. Classification is best described as the establishment of categories of entities, types, and attributes in a way that brings like or similar items together according to predetermined relationships. A classification is by nature an ordered systematic structure.

The design of a classificatory structure must satisfy two basic requirements: (1) comprehensiveness and (2) mutual exclusiveness of its categories. Its scope must be broad enough to encompass all the items that need to be included in the various classes, and the definition of the classes must be exact enough to assure the existence of only one place for every item. Further, that place must be the same for every user of the classification. The underlying logic is simple; every question must have a unique, unambiguous binary answer: "yes" or "no"; "true" or "false"; "present" or "absent"; "included" or "excluded"; and so on.

Entities, types, and attributes change continuously in a dynamic world. A viable classification system which contains them must be flexible enough to accommodate such changes. Its classes must be expandable. To be comprehensive, new and mutually exclusive classes may have to be added to the structure. Old classes may in addition have to be modified or deleted.

Classification schemes are based on the viewpoint of particular people, called upon to do certain tasks at a specific point in time. As experience grows and circumstances change, the systems too must grow and change.

Decimal Codes. One of the most widely known classification codes is the Dewey Decimal System used primarily for indexing libraries or classifying written correspondence by subject matter. The following is a representative example:

300.	Sociology
400.	Philology
500.	Natural Science
510.	Mathematics
520.	Astronomy
530.	Physics
531.	Mechanics
531.1	Machines
531.11	Level and Balance
531.12	Wheel and Axle
531.13	Cord and Catenary
531.14	Pulley
531.141	Pulley, Compound

The decimal method of coding is designed to be used for identifying data in situations where the quantity of items to be coded cannot be limited to any specific anticipated volume. It is particularly well suited for classifying and filing abstracts of written material because it is able to handle an infinite number of items as they are added to any given classification.

Pure decimal code construction does not lend itself readily to mechanized data processing methods because fixed-code field definition is inconsistent with the decimal code expandability. A

number of devices may be used for machine processing of the decimal code, such as tagging variable length fields, special indentation and spacing, and blocked construction as in the following example.

Code	Subject
531000	Mechanics
531100	Machines
531110	Level and Balance
531120	Wheel and Axle
531130	Cord and Catenary
531140	Pulley
531141	Pulley, Compound

In this example, the decimal code has been converted to a six-digit, fixed-field block classification code.

The organization of the decimal code is retained, but the degree of expandability has been limited to ten subdivisions for each machine class. The next section describes block codes in greater detail.

Block Codes. The block codes dedicates each code position or groups of digits to some characteristic of the items to be coded. There are several variations of block coding. One of the simplest forms is the high order block. This form uses only the first digit in a blocking mode, the rest of the code is some other type. If several company locations are involved, for instance, employee identification numbers may be blocked like this:

First Digit	Location
1	New York
2	Chicago
3	Denver
4	San Francisco

Hence, "200001" might be the number of the first man hired at the Chicago location. This use of block coding is common when duplicate employee numbers which existed at several previously autonomous locations are incorporated in a central information processing system. The blocking first digit eliminates the duplicates. This technique also allows each location to continue issuing new numbers without the necessity of establishing a central number control point.

Dependent Codes. In most classification systems, classes are divided into subclasses, and subclasses are divided further into sub-subclasses. When coding these classes and subclasses, usually the code assigned to subclasses is unique only within the subclass since the same codes are used to code members of another subclass. By example, the following illustration demonstrates the dependency of the identification of the class for unique identification of the subclass.

Class: States of the United States

Members: Alabama—Coded 01
Arizona—Coded 04

Subclass: Counties of the States of the United States:

Alabama

Autauga County—Coded 001
Baldwin County—Coded 003
Barbour County—Coded 005

Arizona

Apache County—Coded 001
Cochise County—Coded 003
Coconino County—Coded 005

In this example, the code 001 as a county code represents two different entities (Autauga County, Alabama, and Apache County, Arizona). In order to be unambiguous, the county code must be used with the state code as 01001 for Autauga County, Alabama, and 04001 for Apache County, Arizona. In this example, the county code is dependent upon the state code in order to yield unique identification. The three character county code is also unique within a given state and can be used when the application is restricted or limited to counties of only one state.

When classified and coded in this way, the county code is a ***dependent code.*** When the county code is used with the state code, this collective code is also a significant code, because the code structure not only identifies the county, but also the state to which it belongs.

This concept of dependency is not limited solely to classes and subclasses. For example, in certain applications different transactions are identified by a code consisting of parts which represent the organization, the data of the transaction, and a serial number assigned to each transaction on that date. In this example, all three code segments must be employed to produce a unique transaction number derived from all other transaction numbers. This too, is a dependent significant code of the composite data element named "Transaction Number."

4.4.3. Mnemonic Codes (Constant Length Abbreviations). Mnemonic code construction is characterized by the use of either letters or numbers or letter-and-number combinations which describe the items coded, the combinations having been derived from descriptions of the items themselves.

The combinations are designed to be an aid to memorizing the codes and associating them with the items which they represent.

Unit of Measure codes are frequently mnemonic codes. For example:

FT—Foot or feet
BD—Board
BF—Board foot or feet

It should be noted that not all codes used by humans are truly fixed length. To facilitate computer processing, high- or low-order blanks or zeros must frequently be added to make the code values constant length.

There are some problems connected with the use of mnemonic codes to identify long, unstable lists of items. Wherever item names beginning with the same letters are encountered, there may be a conflict of mnemonic use. To overcome this, the number of code characters is necessarily increased, thus increasing the likelihood that the combinations will be less memory-aiding for code users. Also, since descriptions may vary widely, it is difficult to maintain a code organization which conforms with a plan of classification.

Mnemonic codes are used to best advantage for identifying relatively short lists of items (generally 50 or fewer unless the list is quite stable), coded for manual processing where it is necessary that the items be recognized by their code. A common problem, however, is that the code is likely to be misapplied when specific code values are subject to change and users rely too heavily on memory. Thus, to be effectively coded with mnemonics, entity sets must be relatively small and stable.

Acronyms

The acronym is a particular type of mnemonic representation formed from the first letter or letters of several words. An acronym often becomes a word in itself. For example:

RADAR = RAdio Detecting And Ranging
HEW = Department of Health, Education & Welfare

Only when they are of fixed length are acronyms considered data codes.

SECTION 5. PRINCIPLES OF DATA CODE DEVELOPMENT

5. Principles of Data Code Development

5.1 Introduction. The need to communicate with and by means of computers has made increasing demands on data systems designers and users to work out, work with, and understand computer codes and printouts. The difficulties of natural language, and particularly the English language, which were examined above, must be overcome in any efficient data code. But it must always be remembered that a data code will be used by human beings, including people who do not have much familiarity with data processing. Data codes should therefore be designed with two features in mind: optimum human-oriented use, and machine efficiency.

This section provides guidelines to assist in the design and development of data codes which support both features.

5.2. Ten Characteristics of a Sound Coding System. The most viable and useful coding system is one which contains the greatest number of the following ten features:

(1) ***Uniqueness.*** The code structure must ensure that only one value of the code with a single meaning may be correctly applied to a given entity, although that entity may be described or named in various ways.

(2) ***Expandability.*** The code structure must allow for growth of its set of entities, thus providing sufficient space for the entry of new items within each classification. The structure must also allow existing classifications to be expanded and others added as required. Generally considered, at least a doubling of the original set must be accommodatable, with normal expansion between presently assigned positions; an anticipated life span, depending upon the collection and the dynamics of the environment, should be scheduled.

(3) ***Conciseness.*** The code should require the fewest possible number of positions to adequately describe each item. Brevity is advantageous for human recording, communication line transmission, and computer storage efficiencies.

(4) ***Uniform Size and Format.*** Uniform size and format is highly desirable in mechanized data processing systems. The unauthorized addition of prefixes and suffixes to the root code is a common problem and is incompatible with the first trait—uniqueness. Because such prefixes and suffixes are often of variable length and do not always appear, inconsistencies and confusion result.

(5) ***Simplicity.*** The code must be simple to apply and easily understood by each user, particularly workers with the least experience.

(6) ***Versatility.*** The code should be easily modified to reflect necessary changes in conditions, characteristics, and relationships of the encoded entities. However, every change in the nature of the defined entities must be accompanied by a corresponding change in the code or coding structure.

(7) ***Sortability.*** It is desirable to obtain reports in a predetermined format or order. Reports are most valuable when sorted for optimum human efficiency. Although data must be collatable and sortable, the representative code for the data does not have to be sortable, if it can be correlated with another code which is sortable.

(8) ***Stability.*** Code users need codes which require infrequent updating. Individual code assignments for a given entity should be made with a minimal likelihood of change, either in the specific code or in the entire coding structure. Changes are costly, laborious, and cause errors, and can damage the system when uncontrolled.

(9) ***Meaningfulness.*** Meaningfulness should accompany the codes to the greatest extent possible. To instill greater meaning, the code values should reflect characteristics of the encoded entities, such as mnemonic features, unless such a procedure results in inconsistency or inflexibility.

(10) ***Operability.*** The code should be adequate for present and anticipated data processing both geared to machine and human use. Care must be exercised to minimize the clerical effort or computer update and maintenance time required to continue operations.

5.3. Code Design Principles. This summary of data codification principles is intended to serve as a checklist for system designers. Its use may help them to avoid the potentially expensive results of inadequately conceived and developed data codes.

It should be noted that, in many instances, these traits may be conflicting. For example, if a coding structure is to have sufficient expandability for future needs, it may have to sacrifice conciseness to some degree. Hence, all trade-offs must be appropriately considered to enable optimum efficiency within a given structure.

5.3.1. General.

Planning a Coding System. Sufficient effort and, if need be, time must be spent in preliminary study, definition and planning, when designing a new coding scheme. Potential problems must be anticipated and all design alternatives thoroughly evaluated ***prior*** to implementation of the new system.

(1) ***Code Significance.*** When properly used, significant codes provide a basis for additional information and tend to be easier and more reliable for human use than non-significant codes. However, caution must be exercised in the development of significant codes to assure that significant parts are connected to stable entities. For example, a significant code for an organization should not be associated with the location of the organization when a change in location would result in a change in the code. Excessively significant codes can become unmanageable and lack expandability, and should thus be avoided. For extremely simple tasks, numeric characters are preferable. However, alpha characters are more meaningful and thus better suited to complex tasks.

(2) ***Use of Standard Codes.*** Existing codes should be used wherever possible. New codes should not be designed unless absolutely necessary. In all cases, the preference of the code users should be taken into consideration. It is advantageous to consider all code systems employed by the intended users of a new coding system.

(3) ***Multiple Code Set Compatibility.*** More than one code or representation is necessary, in some instances, to meet most systems requirements. A single code is the ideal objective, but is not always the most practicable solution. Multiple codes, if needed, should be translatable from one code to another, i.e., the data items remain unchanged, only the codes are variable.

(4) ***Mnemonic Codes.*** Mnemonic codes may be used to aid association and memorization, thus increasing human processing efficiency, provided they are not used for identification of very long, unstable lists of items. Mnemonic structures must be carefully chosen, however, to insure that flexibility is not sacrificed. Mnemonics should generally be avoided if the potential code set exceeds 50 entries, because the effectiveness of the mnemonic feature decreases as the number of items to be coded increases. Where mnemonic or otherwise meaningful codes cannot be provided for all codes in the system, preference should be given to codes having the highest use frequency.

(5) ***Code Naming.*** All independent data code segments must be individually named with standard, unique, consistently applied labels.

(6) ***Calculation of Code Capacity.*** When calculating the capacity of a given code for covering all situations while maintaining code uniqueness, the following formula applies (assuming 24 alpha characters and 10 numeric digits are used because the letters I and O should be avoided whenever possible):

$C = (24^A)\ (10^N)$
where
C = total available code combinations possible
A = number of alpha positions in the code
N = number of numeric positions in the code
($A + N$, when combined, equal the total positions of the code.)

NOTE: The above formula assumes that a given code position is *either* alpha *or* numeric—never both. If a given position can have *both* alpha and numeric characters, the formula becomes $C = (36)^{A+N}$ or $(34)^{A+N}$ when the letters "I" and "O" are not used.

5.3.2. Code Length

(1) ***Conciseness.*** Codes should be of minimum length to conserve space and reduce data communication time, but at the same time optimized in terms of the code users capabilities.

(2) ***Fixed Length.*** A code of a fixed length (e.g., always three characters, not one, two, or three) is more reliable and easier to use than a variable length code.

(3) ***Segmentation.*** Codes longer than four alphabetic or five numeric characters should be divided into smaller segments for purposes of reliable recording, e.g., XXX-XX-XXXX is more reliable than XXXXXXXXX. The code designer should take advantage of common English usage to divide or link long code phrases.

(4) ***Potential Expansion.*** The code structure should provide for adding new items without having to recode existing items or extending the code length.

5.3.3. Code Format.

(1) ***User Considerations.*** Code components and phrases should be formatted according to user needs for information, considering greatest ease of scanning for accuracy and completeness, and compactness of the message. Message formatting should be coordinated among system users.

(2) ***Alphabetic versus Numeric.***[4] Human recording of numeric codes is generally more reliable than that of alphabetic (all letters) or alphanumeric codes (letters and numbers) where no mnemonic characteristics exist. Controlled alphanumeric codes (i.e., where certain positions are always alphabetic or numeric) are more reliable than random alphanumeric codes. For example, AA001 (where the first two characters are always letters and the last three are numbers) is a more reliable code than when letters or numbers can appear in any position.

(3) ***Character Grouping.*** In cases where the code is structured with both alpha and numeric characters, similar character types should be grouped and not dispersed throughout the code. For example, fewer errors occur in a three character code where the structure is alpha-alpha-numeric (i.e., HWS) than in the sequence alpha-numeric-alpha (i.e., H5W).

(4) ***Code Position Sequence.*** If a code divides an entire entity set into smaller groupings, the high-order positions should be broad, general categories; and low-order positions should be the most selective and discriminating (including any prefixes and suffixes). An example is the date (YYMMDD). If a descriptive code is formulated consisting of two or more existing independent codes, the individual code segment occupying the higher-order position will be based on usage requirements and processing efficiency considerations.

(5) ***Separation of Code Segments.*** Code segments should be separated by a hyphen (when displayed) or exist in complete separation (when stored and displayed) if the positions or segments are completely independent and can stand alone (i.e., no other code is required for complete meaning).

(6) ***Check Characters.*** When the number of characters of a proposed code exceeds four characters and when this code will be for purposes of identification of major subjects (e.g., organizations, projects, materials, individuals, etc.) consideration should be given to the addition of an error-detecting character to avoid errors in recording. Employment of a self checking code avoids many unnecessary problems of posting data to the wrong record and providing misinformation.

5.3.4. Character Content

(1) ***Special Characters.*** Familiar characters should be used, and characters other than letters or numbers (such as the hyphen, period, space, asterisk, etc.) are to be avoided in code structures (except for separating code segments, where a hyphen may be used). Upper case letters only, i.e., ABC . . . Z (not abc . . . z), are to be used in data codes. Names and abbreviations may use both upper and lower case letters and other characters. The vocabulary for a given code system should contain the fewest possible character classes. Wherever possible, the character set used for data standards should conform to the American National Standard Code for Information Interchange (ASCII).[5]

[4] Cardozo, B. L., and Leopold, F. F., Human Code Transmission, Ergonomics, 133-141 (1963).
[5] ANS X3.4-1968.

(2) ***Visual Similarities.*** When it is necessary to use an alphanumeric random code structure, characters that are easily perceived as, or confused with, other characters should be avoided. Some examples are: letter I vs. number 1; letter O vs. number zero; letter Z vs. number 2; slash or virgule, / vs. number 1; and letters O and Q.

(3) ***Acoustical Similarities.*** Nonsignificant codes should avoid characters that can be confused when pronounced (acoustically homogeneous); for example, the letters B, C, D, G, P, and T or the letters M and N.

(4) ***Vowels.*** Avoid the use of vowels (A, E, I, O, and U) in alpha codes or portions of codes having three or more consecutive alpha characters to preclude inadvertent formation of recognizable English words.

(5) ***Collating Considerations.*** Any specific character position should be ***either*** letters ***or*** decimal digits in order to avoid collating sequence incompatibility.

5.3.5. Assignment Conventions

(1) ***Meaningfulness Reduces Errors.*** Significant or meaningful data codes are preferred over nonsignificant or random codes. This facilitates use by the human coder and reduces errors. For example, in coding the counties of the States of the United States, fewer errors may be expected when the code structure is SSCCC—where the first two characters are the code for a State and the last three characters are the code for a county within that State—than in a code such as XXXX that is randomly assigned to each county.

In this connection, mnemonic data codes produce fewer errors than other types of codes where the number of items to be coded is relatively small and stable. For example, M and F are more reliable codes for male and female than 1 and 2. Y and N are preferred for Yes and No over 1 and 2.

(2) The rules of the data code structure and its derivation should be clearly stated and consistently applied. For example, a mnemonic abbreviation may be formed by deleting all vowels from the names of the coded items as DT for date or GRN for green, or the first letters of the words of the coded items may be used as EOF for End of File or DO for Due Out.

(3) ***Codes for Numeric Categories.*** Quantities or numbers should not be coded since this introduces additional translation and a loss of preciseness. For example, the numbers 1 to 99 could be coded A, 100-199 coded B, etc. This may be desirable for purposes of categorization, but statistical value is lost since the actual numbers can not be derived once they are coded. Categorizations can be performed during later phases of data processing rather than in precoding of the input data.

(4) ***Use of "Natural" Data.*** A code structure should not be developed if the specific data in its natural form (such as specific percentage amounts) is appropriate and adequate.

(5) ***Sequence Code Numbering.*** To maintain fixed code length and avoid confusing leading zeros, codes assigned in sequence may be assigned beginning with "101," "102" or "1001," "1002," etc. rather than with "1." Another advantage of this practice is keeping unauthorized persons from determining the quantity of data in the total entity set from knowledge of a single code (e.g., Product Serial Numbers). Code numbers with lower values may be used to identify miscellaneous or special situations, if so desired, or may be left unassigned. (This procedure does reduce code set capacity, however.)

(6) ***Use of "0000" and "9999" as code values.*** One should not use all "0's" (implies nothing) or all "9's" (implies the end) as assigned code values. These values should be reserved for special situations or for use as processing indicators.

(7) ***"Miscellaneous" Codes.*** A code category for "Miscellaneous' or "Other" varieties must be used with great discretion. One should not allow the placement of entities in this category which actually belong in a more specific class.

SECTION 6. GUIDELINES FOR DEVELOPMENT OF DATA STANDARDS

6. Guidelines for Development of Data Standards

6.1. Introduction. A data standardization project may be initiated at the international or national level, within a trade or professional association, or within an industrial organization. The task may begin when the people responsible for information or data within the organization find difficulty in obtaining and interchanging the data needed to conduct their necessary functions, and recognize the need for standards. At the national level, a data standardization project may be established by the American National Standards Institute, whenever it has been determined that a specific standard should be developed.

There are several steps that must be taken to complete the task of standardization, beginning with the precise definition of the project and a thorough inquiry into the background and available resources to undertake this effort.

6.2. Project Definition. The first step to be taken is to define the purpose and scope of the project. The objectives need to be identified and a program of work developed. After these are prepared, a project chairman should be appointed and a task group formed. If a new ANSI project is to be established, the scope statement and program of work must be coordinated with the X3 Standards Planning and Requirements Committee (SPARC) and approved by the X3 Committee on Computers and Information Processing. The planned project should be documented in accordance with X3 procedures.[6]

6.3. Formation of Task Groups. It is important that the proper interests and talents be represented in the standards development. Identifying persons with the interest, the resources, and the expertise to assist in the work is often difficult. A letter can be sent to individuals and organizations requesting participation. This letter should request the type of person or expertise needed, and provide an estimate of the time involved and duration of the project.

The size of the group will depend on the particular project. Generally, a task group should have at least four members.

When the task group members are known, the first meeting should be planned. At the initial meeting, the objectives and planned work should be reviewed, administrative details should be discussed, and meeting schedules planned.

6.4. Information Collection. The development of coded representations for a particular class of subjects should begin with the following questions:

a. What are the requirements of the code, and what uses of it are anticipated?

b. Are codes really needed, and if so why?

c. What and how many items are to be included in the class of subjects to be coded?

d. What is the most effective code structure?

e. What rules or procedures are necessary for making code assignments?

Certain basic information needs to be collected to answer these questions. This includes seeking answers to further questions:

a. Will the users of the information produced by the systems accept data codes on the output document?

b. How critical is the coded data to the system? What are tolerable error rates? Should a check character be employed to reduce errors?

c. How will the data codes be maintained?

d. Are there codes currently in wide use that are acceptable?

e. What are the machine factors to be considered? (e.g., computer processing and storage capabilities, input media and method of recording—i.e., punched cards, punched paper tape, magnetic tape, on-line terminals, optically read forms, and transmission time.)

[6] Document X3/SD-3, Format and Instructions for Initiating a Standards Development Project (X3/SD-3). Available from the X3 Secretariat, CBEMA, 1828 L Street, NW., Washington, D.C. 20036.

f. How and by whom are the data collected or obtained?

g. What human factors (limitations and capabilities) need to be considered?

h. Have the code design criteria in 5.2. and 5.3. been consulted?

These factors are not listed in any particular order of significance. Trade-offs usually are necessary before final decisions are made because not all factors can be satisfied.

6.5. Criteria for Development of Standard Representations. The discussion of basic coding methods in Section 4 and the data coding principles in Section 5 are provided to assist in the development of specific standard data representations. It must be recognized, however, that some of the criteria in Section 5 conflict. The development task group must analyze the use of the particular representation and decide which criteria are more important to its particular situation.

The relative ease or difficulty users of a data code can be expected to experience can be estimated by the "Information Load Method." This method takes into account the length of the code and the structure of each character in the code. The "information load" of a given code is defined as the sum of the "character load" of each character of the code. The character load is a value equal to $\log^2$ of the total number of different characters that could appear in that character position. For example, the character load for a numeric character code position that could have values of 0 through 9 is the $\log^2$ of 10, or 3.32, and for an alpha character position where the values could range from A through Z, the character load is the $\log^2$ of 26, or 4.70. The information load of a three-character numeric code would thus be: 3.32+3.32+3.32, or 9.96. For a three character alpha code, the information load would be: 4.70+4.70+4.70, or 14.10. A code having two numeric characters and one alpha character would have an information load of: 3.32+3.32+4.70, or 11.34.

This technique is most usefully applied to nonsignificant codes where no secondary meaning can be derived from the code. Nonsignificant codes are used only to uniquely identify the coded subjects in the class. For example, the number 80 would be a nonsignificant code for the month of December, whereas 12 would be a significant code since December is the twelfth month of the year.

When longer codes are broken into smaller units, the information load applies to the smaller units. Whenever the information load exceeds 20, the error rate of data recording can be expected to increase. This rule is stated simply in principle number 3, Section 5.3.2.

6.6 Technical Specifications. The task group should develop the technical specifications of the proposed standard to include:

—a list of the data items by name (or as appropriate, the characteristics of the data items if these are not names, e.g., Social Security Account Number);

—definitions of those data items where explanation is necessary;

—abbreviations (as needed), and

—a unique data code (or codes) for each item.

There shall not be any duplicate codes on the list (or duplicate abbreviations). Names and definitions should be reviewed to insure that each data item is sufficiently different in name and meaning from any other item so that ambiguities are avoided. A concise name for the proposed standard should be determined, e.g., "Calendar Date," "States of the United States," etc.

When a proposed American National Standard is being prepared, applicable procedures and formats should be followed. Applicable ISO (International Organization for Standardization) procedures and guides should be followed if an ISO Recommendation is to be the end product. The task group chairman should obtain the most current procedures and guides either from the appropriate Standards Sectional Committee Chairman or from the American National Standards Institute.

SECTION 7. GUIDELINES FOR IMPLEMENTATION OF DATA STANDARDS

7. Guidelines for Implementation of Data Standards

7.1. Interchange. Data standards are developed and approved in order to facilitate the interchange of information between and among independent data systems. Data standards should be employed in these interchanges.

It is recommended that use of the standard be specified when data is requested from another organization. The transmitter is urged to consider converting the data to the standard form, especially if the receiving organization so requests.

7.2. Internal Files and Records. The determination of whether to incorporate data standards into internal files and records is a decision which should be left to the installation manager. When a conversion cost can be offset by the continuing cost of translation of data, the use of the standard in internal files and records can be justified on the basis of cost effectiveness. In other instances, the large investment in current systems and files is such that translation of data (especially, if there is an infrequent or limited amount of interchange) is justified. However, in the redesign of existing systems and in the design of new data systems, the use of data standards should be considered and employed to the maximum extent possible.

SECTION 8. GUIDELINES FOR MAINTENANCE OF DATA STANDARDS

8. Guidelines for Maintenance of Data Standards

8.1 General. Maintenance of the information that makes up a data standard may be viewed from two distinct viewpoints. The first considers the unique administration of the specialized vocabulary or code set which requires peculiar updating and dissemination techniques. The second view sees the problem from the perspective of maintaining and managing a distinctively designed data base, perhaps one responsive to the accounting, inventory, report, and control needs of present large-scale management information systems.

Insofar as the second consideration has recently come under the scrutiny of ANSI Committee X3 in its deliberations concerning standard data base management methods and systems, only the first viewpoint will concern us here.

8.2 Maintenance and Information Relevant to Current Data Standards. There are currently at least five kinds of formal data standards in use:

International Standards—which have broad acceptance and the approval of such international groups as the International Organization for Standardization (ISO) and regional groups such as the European Computer Manufacturers Association (ECMA). These are intended for voluntary use and adoption within the national standards of the community of nations. (See Appendix B.)

American National Standards—which include a variety of standards on computer software, data representations such as code sets and structures, and formatting procedures which have been approved and published by the American National Standards Institute. These are intended for the voluntary acceptance and use of industry and government on a nation-wide scale. (See Appendix A.)

U.S. Federal Government Standard Data Elements and Codes for General Use—include Federal general standards for use in the executive branch of government. They embrace such standards as those for countries, states, counties, places, organizations, individuals, and elements of time. They are intended for general use by agencies.

U.S. Federal Standard Data Elements and Codes for Program Use—are intended for use in particular related programs concerning more than one agency of the Federal Government. These standards apply to data elements and codes usually limited to applications in weather, personnel, supply, and other unique systems. The same source data are generally used by several agencies, while the information contained in numerous data bases are aggregated and exchanged on a program basis.

Local Standards for data elements and codes—which are maintained for the use of individual disciplines, industries, or limited program applications and are either not applicable to international, national, or governmental implementation or not yet incorporated into standards with such broad-scale validity.

Existing data standards, which have been approved at the international and highest national levels, are announced, published, and distributed by the national standards organization in each country. In the United States of America, information about such standards may be obtained from the

American National Standards Institute, Inc.
1430 Broadway
New York, New York 10018

The responsibility for announcement, storage, and dissemination of information relevant to international, national, and Federal data standards may also be carried by certain national information centers, or such announcement media as the Federal Information Processing Standards Publication (FIPS PUB) Series, published by the U.S. National Bureau of Standards.

Local standards are generally maintained by the special group which designed the data base for its proper discipline or purpose-oriented applications. Information concerning the standards and maintenance operations is ordinarily available from the specific organization, trade association or

agency involved. An example of an international special purpose standard is the International List of Post Offices, obtainable from the Universal Postal Union.

8.3 Updating and Improvement of Current Data Standards. The maintenance of a data standard must be assigned to an appropriate organization. For the data standard may require any one of a great variety of data bases for its upkeep, control, and dissemination.

The data standard can apply to:

Literals—self-identifiable constants such as exact numbers (e.g., dates), serial entities, etc.

Small semi-permanent lists—such as states, counties, countries.

Mission-oriented codes—dynamic lists such as industrials or commodities.

Program or Discipline-oriented codes—as in technical data lists or transaction codes, e.g., for census districts.

Classified Structures—large, semi-constant hierarchically ordered lists such as the Federal Industrial Classification, or the Universal Decimal Classification.

Dynamic lists—such as the Social Security Number files.

Management Information or Command and Control System data elements—file headings required for intelligence or management analysis and report generation.

The files which contain such data must be seen as structures with more or less dynamic features. Depending upon their applications and many internal as well as environmental conditions, these files may often change in content and occasionally in structure, sequence, or storage medium.

Appropriate organizations must be entrusted with the data collection, selection, and posting of new entries to the existing files. The efficiency and effectiveness of these maintenance transactions can determine not only the cost but the feasibility of the entire standard data system.

The updating and improvement of current data standards must be channeled through the proper standards body. National Standards must be updated and reviewed by the appropriately appointed groups within the American National Standards Institute. This national organization will forward suggestions for modification and improvement to the proper groups within ISO for updating and revising international data standards. Similar proper governmental, industrial, and professional organizational channels should be used to improve existing data standards at these particular levels.

The American National Standards are periodically reviewed and updated when necessary, and at least once every five years.

8.4. Criteria for the Maintenance of Standards. To initiate a data standard it is necessary to question whether maintenance of the code can be justified from the viewpoints of:

cost effectiveness

comprehensive coverage

organizational mandate and competence

user needs

It must be determined in advance who should maintain the standard, and by what means of control: centralized, decentralized, or by a carefully designed balance of the two modes.

User needs must be established and a feedback mechanism must be built into the maintenance system. This may require continuous liaison between the maintaining organization and representatives of concerned user groups. This may involve other representatives of industry, commerce, professional organizations as well as Federal, State, and local governments.

Periodic review procedures must be established in advance and scrupulously implemented.

Simple file updating procedures must be instituted with special attention given to:

—timeliness of updating;

—periodic publication;

—efficient and effective promotion and distribution of the basic data base, periodic updates and relevant services, using appropriate media.

Periodic review of the administration and financing of the code or vocabulary data base maintenance is essential.

8. **Guidelines for Maintenance of Data Standards (Summary)**

8.1. **General**

8.2. **Maintenance and Information Relevant to Current Data Standards.** Existing data standards approved at the national and international levels are announced, published, stored, and distributed by the national standards organization or by national information centers, or announcement media such as the Federal Information Processing Standards Publication (FIPS PUB) Series.

8.3. **Updating and Improvement of Current Data Standards.** Channeled through ANSI, the U.S. national standards organization, national data standards are reviewed and updated at least once every five years. Suggestions for improvement may be sent to ANSI for distribution to the proper technical committee for action.

8.4. **Criteria for the Maintenance of Standards.** Suggestions are given on the maintenance of representational forms . . . vocabularies, abbreviation sets, and code structures. Housekeeping and control measures are required to accommodate the changes required in large dynamic lists.

APPENDIX A

SCOPE AND PROGRAM OF WORK OF AMERICAN NATIONAL STANDARDS INSTITUTE SUBCOMMITTEE X3L8, REPRESENTATIONS OF DATA ELEMENTS

(As approved by X3 Sectional Committee, January 23, 1970)

Background

The need for a program of data standardization arose with difficulties in interchanging data among the data systems of business and governments. The difficulties stemmed from different organizations using a great variety of representations for the same subject matter, such as places, dates, individuals, organizations and commodities, and using the same representations with completely different meanings, as well as from the lack of a common method for describing the data that was to be interchanged.

The need for standard representations and ways of describing interchanged data had been recognized earlier by particular industries, such as air transportation in the area of passenger reservations. To satisfy this need, programs to establish and maintain data interchange capabilities were initiated. In addition, agencies of the Federal Government initiated standardization programs to facilitate data interchange between agencies. Standardized representations, formats, and format descriptions are required among the several needs that must be satisfied for different organizations to interchange data. Early in the 1960's, a standardization program was initiated by the Computer and Business Equipment Manufacturers Association (CBEMA) and the American Standards Association, now the American National Standards Institute (ANSI), to establish standards related to systems, computers, equipments, devices, and media for information processing. This resulted in the formation of the ANSI Committee for Computers and Information Processing, designated X3, with representatives drawn from producer, consumer, and general interest groups. In 1966, Subcommittee X3L8 was established as part of the X3 organization and was given the responsibility for standardization of representations of data elements commonly used in inter-

change. The X3L8 Subcommittee has concentrated on development of standard representations for subject matter of common interest, including standards for times, individuals, organizations, places, and numeric values. Interest has expanded to cover other data elements involved in data interchange and to enlist in this program organizations with interest and experience in each of the areas involved.

Definitions

Data Element—A basic unit of identifiable and definable information. In information processing systems, a data element occupies the space provided by fields in a record or blocks on a form. It has an identifying name and a value or values for expressing a specific fact. Examples: Employee number, Employee name, Date of birth, Mailing address, Color of eyes, Height, and Weight.

Representations—Names, Abbreviations, Codes, and Numeric Values used to express a data element.

Scope

1. To develop standards for (1) describing the representations of data elements involved in data interchange; and (2) representing data elements of common interest, such as the elements concerned with the representations of times, locations, individuals, organizations, and materials.

2. To develop recommended procedures, criteria, and guidelines in order to provide an organized approach to the standardization of the representations of data elements.

Program of Work

1. To develop recommended procedures and criteria for the development, maintenance, issuance, and use of American National Standards for representations of data elements.

2. To develop proposed standards for the following items:

a. Representation of time elements to include dates, times, and time zones.

b. For identifying organizations, individuals, and accounts to include standards for name formatting.

c. Representations for States, Counties, Places, and Congressional District of the United States, Countries of the World and their Subdivisions, Shipping and Mailing Addresses, and Points Locations,

d. Representing quantitative numeric expressions.

3. To represent the interests of the United States through the X3 International Advisory Committee and the American National Standards Institute in the development of international recommendations for representations of data elements by the International Organization for Standardization (ISO) or other standardization bodies.

4. To act as the focal point within the American Standards Institute for reviewing proposed representations of data element standards that have been developed by other organizations and which are submitted for adoption as American National Standards and forwarding these with appropriate recommendations through established channels for subsequent standardization actions.

5. To assist, as necessary and resources allow, industry, government, and other groups in the development of proposed standards for representations of data elements.

Other Factors Bearing on the Work of X3L8

1. It is not feasible for one organization to develop representation standards for all the data elements involved in interchange. Accordingly, the most practicable approach is to have a single group develop and establish common procedures and criteria to guide other organizations in developing standards for their particular subject matter or application area. When the results of such developments by other organizations are submitted to ANSI for consideration as American National

Standards, X3L8 would review these and prepare recommendations concerning their acceptability or conflict with other established standards and forward these for appropriate standardization actions.

2. Many of the potential standards for representations of data elements are of such a magnitude that their maintenance is beyond the capabilities of X3L8 or the American National Standards Institute. Examples of such standards are those for representing those data elements concerned with identification of organizations and places (i.e., cities, towns, townships, boroughs, etc.) Accordingly, it is essential to depend upon some other organization outside the ANSI structure for this necessary maintenance. This situation does not necessarily forbid the development and establishment of American National Standards. These can be accomplished through agreements with the outside organization as to the procedures and criteria to be used in maintaining the standard. These procedures, criteria, and other considerations then form the basis for the proposed American National Standard.

APPENDIX B

SCOPE AND PROGRAM OF WORK

(As adopted by ISO/TC 97 on June 20, 1972)

Title ISO/TC 97/SC 14, Representations of Data Elements

Scope

Standardization of the representations of commonly interchanged data elements to facilitate information interchange and information processing.

Program of Work

1. To develop international recommendations for describing data elements and their representations involved in data interchange.

2. To develop international recommendations for representing data elements of common interest to include representations for:

a. Dates and time
b. Countries
c. Languages
d. Identification of Individuals
e. Identification of Organizations
f. Identification of Accounts
g. Mailing and shipping address
h. Point locations such as longitude and latitude
i. Units of measure
j. Numeric expressions

3. To develop recommended guidelines and criteria to provide for an orderly approach to the standardization and description of data elements involved in international information interchange.

4. To provide liaison with other organizations and ISO Committee for the coordination of data standards intended for information interchange.

APPENDIX C

BIBLIOGRAPHY

Aume, N. M., and Topmiller, D. A., An evaluation of experimental how-malfunctioned codes, Human Factors, 261–269 (1970).

Bell, J. R., The quadratic quotient method: A hash code eliminating secondary clustering, Commun. ACM. 107–109 (Feb. 1970).

Bell, J. R., and Kaman, C. H., The linear quotient hash code, Commun. ACM, 675–677 (Nov. 1970).

Blankenship, A. B., Memory span: A review of the literature, Psychol. Bull. 1–25 (1938).

Bonn, T. H., A standard for computer networks, IEEE Computer magazine. (May-June 1971).

Brown, J., Some facts of the decay theory of immediate memory, Quarterly Journal of Experimental Psychology, 12–21 (1958).

Cardozo, B. L., and Leopold, F. F., Human code transmission, Ergonomics, 133–141 (1963).

Carson, J. G. H., Item identification and classification in management operating systems, Production and Inventory Management (June 1971).

Cherry, C., On Human Communication (Massachusetts Institute of Technology Press, Cambridge, Mass., 1966).

Conrad, R., Experimental psychology in the field of telecommunications, Ergonomics, 289–295 (1960).

Conrad, R., The location of figures in alpha-numeric codes, Ergonomics, 403–406 (1962).

Conrad, R., and Hille, B. A., Memory for long telephone numbers, Post Office Telecommunications Journal, 37–39 (1957).

Conrad, R., and Hull, A. J., Copying alpha and numeric codes by hand: An experimental study, J. App. Psychol., 444–448 (1967).

Crossman, E. R. F. W., Information and Serial Order in Human Immediate Memory, in Proceedings of 4th London Symposium on Information Theory, 1961.

Crowley, E. T., and Crowley, R. C., Acronyms and Initialisms Dictionary, 3rd. ed. (Gale Research Company, Detroit, Mich., 1970).

Crowley, E. T., New Acronyms and Initialisms (Gale Research Company, Detroit, Mich., 1971).

Field, M. M., et al, Guidelines for Constructing Human Performance—Based Codes, Bell Telephone Laboratories Technical Report, 1971.

Frink, W. J., In coding it's structure that counts, Control Eng. (October 1962).

Gilbert, E. N., A comparison of signaling alphabets, Bell Sys. Tech. J., 504–522 (May 1952).

Gilbert, E. N., Information theory after eighteen years, Science, 320–326 (Apr. 15, 1966).

Goldman, S., Information Theory (Prentice-Hall, Englewood Cliffs, N. J., 1953).

Gombinski, J., and Hyde, W. F., Classification and coding, Graphic Science (March 1968).

Gombinski, J., Industrial classification and coding, Eng. Mater. & Des. (Sept. 1964).

Hall, R. A., Linguistics and Your Language (Doubleday & Company Inc., Garden City, N. Y., 1960).

Hamming, R. W., Error detecting and error correcting codes, Bell Syst. Tech. J., 147–160 (Apr. 1950).

Harris, D. H. et al, Wire sorting performance with color and number coded wires, Human Factors, 127–131 (Apr. 1964).

Hauck, E. J., Be kind to your data codes, Journal of Systems Management, (Dec. 1972).

Hare, V. C., Systems Analysis: A Diagnostic Approach (Harcourt, Brace, and World, New York, N. Y., 1969).

Heron, A., Immediate memory in dialing performance with and without simple rehersal, Quarterly Journal of Experimental Psychology, 94–103 (1962).

Hitt, W. D., An evaluation of five different abstract coding methods—experiment IV, Human Factors, 120–130 (Jul. 1961).

Hodge, M. H., and Field, M. M., Human Coding Processes, University of Georgia, 1970.

Hull, T. E., and Dobel, A. R., Random Number Generators, SIAM Rev. (July 1962).

Jackson, R. L., "Dial 911" Setup Qualified Success, Gannett News Service, July 1972.

Jones, B. W., Modular Arithmetic (Blaisdell Publishing Company, 1964).

Klemmer, E. T., Grouping of printed digits for manual entry, Human Factors, 397–400 (1969).

Klemmer, E. T., Grouping of Printed Digits for Telephone Entry, in Proceedings of Fourth International Conference on Human Factors in Telephony, Munich, 1968.

Klemmer, E. T., Keyboard entry, App. Ergonomics, 2–6 (1971).

Klemmer, E. T., Numerical error checking, J. App. Psychol., 316–320 (1959).

Klemmer, E. T., and Stocker, L. P., Optimum Grouping of Printed Digits, American Psychological Association Proceedings, 1972, 689–690.

Konz, S., et al, Human transmission of numbers and letters, J. Ind. Eng., 219–224 (May 1968).

Laden, H. N., and Gildersleeve, T. R., System Design for Computer Applications (John Wiley & Sons, New York, N. Y., 1963).

L'Insalata, B. B., COBOL Module for the Generation and Verification of a Check-bit Using the Modulus 10 Method, Western Electric Co. Technical Report, June 9, 1971.

Little, J. L., Some evolving conventions and standards for character information coded in six, seven and eight bits, Nat. Bur. Stand. (U.S.), Tech. Note 478, 30 pages (May 1969).

Little, J. L., and Mooers, C. N., Standards for user procedures and data formats in automated information systems and networks, AFIPS Conf. Proc. Spring Joint Computer Conf., Atlantic City, N. J., Apr. 30–May 2, 1968, 32, 89–94 (Thompson Book Co., Washington, D. C., 1968).

Mackworth, J. F., The effect of display time upon the recall of digits, Can. J. Psychol., 48–55 (1962).

Maurer, W. D., An improved hash cod for scatter storage, Commun. ACM, 35–38 (Jan. 1968).

Mayzner, M. S., and Gabriel, R. F., Information "chunking" and short-term retention, Journal of Psychology, 161–164 (1963).

Meltzer, H. S., and Ickes, H. F., Information interchange between dissimilar systems, Modern Data (Apr. 1971).

Miller, G. A., The magical number seven, plus or minus two: Some limits on our capacity for processing information, The Psychological Review (Mar. 1956).

Miller, G. A., The Psychology of Communication, (Basic Books, Inc., New York, N. Y., 1967) ; (Pelican Publishing House, Gretna, La., 1969).

Miller, G. A., and Nicely, P. E., An analysis of perceptual confusions among some English consonants, Journal of the Acoustical Society of America, 338–352 (1955).

Morris, R., Scatter storage techniques, Commun. ACM, 38–44 (Jan. 1968).

Oberly, H. S., A comparison of the spans of attention and memory, Am. J. Psychol. 295–302 (1928).

O'Reagan, R. T., Computer-assigned codes from verbal responses, Commun. ACM, 455–459 (June 1972).

Owsowitz, S., and Sweetland, A., Factors Affecting Coding Errors, The Rand Corporation, 1965.

Peterson, W. W., Error-Correcting Codes, 2nd ed. (Massachusetts Institute of Technology Press, Cambridge, Mass., 1972).

Pollack, I., Assimilation of sequentially coded information, Am. J. Psychol., 421–435 (1953).

Radke, C. E., The use of quadratic residue research, Commun. ACM, 103–105 (Feb. 1970).

RAND Corporation, A Million Random Digits with 100,000 Normal Deviates (The Free Press, Glencoe, Ill. 1955).

Rocke, M. G., Data Codification Principles and Methods, Caterpillar Tractor Company, 1971.

Severin, F. T., and Rigby, M. K., Influence of digit grouping on memory for long telephone numbers, J. App. Psychol., 117-119 (1953).

Shannon, C. E., Prediction and entropy of printed english, Bell Syst. Tech. J., 50–64 (Jan. 1951).

Shannon, C. E., and Weaver, W., The Mathematical Theory of Communication, Bell Syst. Tech. J. (1948); (University of Illinois Press, Urbana, Ill. 1949).

Sonntag, L., Designing human-oriented codes, Bell Lab. Rec. (Feb. 1971).

Stallcup, K. L., New growth in linguistics produces clarity, confusion and controversy, The New York Times, p. 90 (Jan. 8, 1973).

Talbot, J. E., The human side of data input, Data Process. Mag. (Apr. 1971).

Thorpe, C. E., and Rowland, G. E., The effect of "natural" grouping of numerals on short-term memory, Human Factors, 38–44 (1965).

Wicklegrew, W. A., Size of rehersal group and short-term memory, J. of App. Psychol., 413–419 (1964).

Woodbury, M. A., and Lipkin, M., Coding of medical histories for computer analysis, Commun. ACM, (Oct. 1972).

Wooldridge, D. E., The Machinery of the Brain (McGraw-Hill Publications, New York, N. Y., 1963).

Woznick, A. M., Item Identification Standards, Systems Engineering CASO Development Report 56904010, Western Electric Company, February 9, 1971.

U. S. GOVERNMENT PRINTING OFFICE : 1976 O - 220-174

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH reports National Bureau of Standards research and development in physics, mathematics, and chemistry. It is published in two sections, available separately:

• **Physics and Chemistry (Section A)**
Papers of interest primarily to scientists working in these fields. This section covers a broad range of physical and chemical research, with major emphasis on standards of physical measurement, fundamental constants, and properties of matter. Issued six times a year. Annual subscription: Domestic, $17.00; Foreign, $21.25.

• **Mathematical Sciences (Section B)**
Studies and compilations designed mainly for the mathematician and theoretical physicist. Topics in mathematical statistics, theory of experiment design, numerical analysis, theoretical physics and chemistry, logical design and programming of computers and computer systems. Short numerical tables. Issued quarterly. Annual subscription: Domestic, $9.00; Foreign, $11.25.

DIMENSIONS/NBS (formerly Technical News Bulletin)—This monthly magazine is published to inform scientists, engineers, businessmen, industry, teachers, students, and consumers of the latest advances in science and technology, with primary emphasis on the work at NBS. The magazine highlights and reviews such issues as energy research, fire protection, building technology, metric conversion, pollution abatement, health and safety, and consumer product performance. In addition, it reports the results of Bureau programs in measurement standards and techniques, properties of matter and materials, engineering standards and services, instrumentation, and automatic data processing.

Annual subscription: Domestic, $9.45; Foreign, $11.85.

NONPERIODICALS

Monographs—Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

Handbooks—Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications—Include proceedings of conferences sponsored by NBS, NBS annual reports, and other special publications appropriate to this grouping such as wall charts, pocket cards, and bibliographies.

Applied Mathematics Series—Mathematical tables, manuals, and studies of special interest to physicists, engineers, chemists, biologists, mathematicians, computer programmers, and others engaged in scientific and technical work.

National Standard Reference Data Series—Provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated. Developed under a world-wide program coordinated by NBS. Program under authority of National Standard Data Act (Public Law 90-396).

NOTE: At present the principal publication outlet for these data is the Journal of Physical and Chemical Reference Data (JPCRD) published quarterly for NBS by the American Chemical Society (ACS) and the American Institute of Physics (AIP). Subscriptions, reprints, and supplements available from ACS, 1155 Sixteenth St. N.W., Wash. D. C. 20056.

Building Science Series—Disseminates technical information developed at the Bureau on building materials, components, systems, and whole structures. The series presents research results, test methods, and performance criteria related to the structural and environmental functions and the durability and safety characteristics of building elements and systems.

Technical Notes—Studies or reports which are complete in themselves but restrictive in their treatment of a subject. Analogous to monographs but not so comprehensive in scope or definitive in treatment of the subject area. Often serve as a vehicle for final reports of work performed at NBS under the sponsorship of other government agencies.

Voluntary Product Standards—Developed under procedures published by the Department of Commerce in Part 10, Title 15, of the Code of Federal Regulations. The purpose of the standards is to establish nationally recognized requirements for products, and to provide all concerned interests with a basis for common understanding of the characteristics of the products. NBS administers this program as a supplement to the activities of the private sector standardizing organizations.

Consumer Information Series—Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

Order **above** *NBS publications from: Superintendent of Documents, Government Printing Office, Washington, D.C. 20402.*

Order **following** *NBS publications—NBSIR's and FIPS from the National Technical Information Services, Springfield, Va. 22161.*

Federal Information Processing Standards Publications (FIPS PUBS)—Publications in this series collectively constitute the Federal Information Processing Standards Register. Register serves as the official source of information in the Federal Government regarding standards issued by NBS pursuant to the Federal Property and Administrative Services Act of 1949 as amended, Public Law 89-306 (79 Stat. 1127), and as implemented by Executive Order 11717 (38 FR 12315, dated May 11, 1973) and Part 6 of Title 15 CFR (Code of Federal Regulations).

NBS Interagency Reports (NBSIR)—A special series of interim or final reports on work performed by NBS for outside sponsors (both government and non-government). In general, initial distribution is handled by the sponsor; public distribution is by the National Technical Information Services (Springfield, Va. 22161) in paper copy or microfiche form.

BIBLIOGRAPHIC SUBSCRIPTION SERVICES

The following current-awareness and literature-survey bibliographies are issued periodically by the Bureau:

Cryogenic Data Center Current Awareness Service. A literature survey issued biweekly. Annual subscription: Domestic, $20.00; Foreign, $25.00.

Liquified Natural Gas. A literature survey issued quarterly. Annual subscription: $20.00.

Superconducting Devices and Materials. A literature survey issued quarterly. Annual subscription: $20.00. Send subscription orders and remittances for the preceding bibliographic services to National Bureau of Standards, Cryogenic Data Center (275.02) Boulder, Colorado 80302.

U.S. DEPARTMENT OF COMMERCE
National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161

OFFICIAL BUSINESS

PRINTED MATTER

AN EQUAL OPPORTUNITY EMPLOYER

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE

COM 211

SPECIAL THIRD-CLASS RATE
BOOK

U.S.MAIL