Bellcore
Bell Communications Research

BELLCORE PRACTICE
BR 751–000–100
ISSUE 3, JUNE 1989

COMMON LANGUAGE® Codes, Abbreviations, And Acronyms - General Description

Prepared for Bellcore by:

L. Modrell

For further information, please contact:

L. Modrell

732-699-5281

To obtain copies of this document, Regional Company/BCC personnel should contact their company's document coordinator; Bellcore personnel should call (732) 699-5802.

Project funding year:1997.

Trademark Acknowledgements

COMMON LANGUAGE CODES, ABBREVIATIONS, AND ACRONYMS - GENERAL DESCRIPTION

Contents

List of Tables

NOTICE OF DISCLAIMER

This document is issued by Bell Communications Research, Inc. (Bellcore) to inform its clients of Bellcore's Practice on *COMMON LANGUAGE® Codes, Abbreviations, and Acronyms-General Description.*. Neither this document nor any of its contents should be disclosed to persons other than employees of those companies.

Bellcore reserves the right to revise this document for any reason, including but not limited to, conformity with standards promulgated by various agencies, utilization of advances in the state of the technical arts, or the reflection of changes in the design of any equipment, techniques, or procedures described or referred to herein.

BELLCORE MAKES NO REPRESENTATION OR WARRANTY, EXPRESS OR IMPLIED, WITH RESPECT TO THE SUFFICIENCY, ACCURACY, OR UTILITY OF ANY INFORMATION OR OPINION CONTAINED HEREIN. BELLCORE EXPRESSLY ADVISES THAT ANY USE OF OR RELIANCE UPON SAID INFORMATION OR OPINION IS AT THE RISK OF THE USER, AND THAT BELLCORE SHALL NOT BE LIABLE FOR ANY DAMAGE OR INJURY INCURRED BY ANY PERSON ARISING OUT OF THE SUFFICIENCY, ACCURACY, OR UTILITY OF ANY INFORMATION OR OPINION CONTAINED HEREIN.

This document does not represent any commitment by Bellcore or by any of its clients to purchase any product using the Codes, Abbreviations, and Acronyms - General Description described herein.

Each Bellcore Client Company may have requirements or specifications different from those described herein.

Nothing contained herein shall be construed as conferring by implication, estoppel or otherwise, any license or right under any patent, whether or not the use of any information herein necessarily employs an invention of any existing or later issued patent.

Bellcore does not recommend products, and nothing contained herein is intended as a recommendation of any product to anyone.

1. PURPOSE

1.0.1 This practice provides an explanation of the procedures and guidelines used by the Language Standards Department of Bellcore in the development, maintenance, and documentation of COMMON LANGUAGE standard codes, abbreviations, and acronyms.

2. SCOPE

2.0.1 This practice is primarily directed to people with a need to know the following information:

a. The basic functions and responsibilities of the Bellcore Language Standards Department.

b. A description of COMMON LANGUAGE coding procedures.

c. Descriptions of coding terms used throughout COMMON LANGUAGE documentation.

3. REASON FOR ISSUE

3.0.1 This practice is re-issued to reflect a change in trademark and the inclusion of the "Purpose," "Scope," and "Reason for Issue" sections.

4. GENERAL

4.0.1 Questions concerning this practice, or any of the codes and abbreviations described in this practice, should be directed to the Executive Director, Language Standards Department, Bellcore, through the appropriate Bellcore Client Company (BCC) COMMON LANGUAGE contact.

4.0.2 Historically, as manual record keeping systems came into use in the telecommunications industry, it became necessary to keep information as short and concise as possible. Background information had to be recorded in a minimal amount of space so that files could be kept to a manageable size. People using these systems found that by creating codes and abbreviations, they could save transcription time and reduce their work effort. Since the people processing this information could easily make translations and interpretations, a wide range of codes and abbreviations could be used with little regard for standardization.

4.0.3 The evolution from manual to mechanized systems did not change the need for coded information. The requirements for this information, however, became far

more stringent. Conciseness was still required; but now mechanization required that redundant information, translations, and interpretations be eliminated. Machines, while having greater speed, do not have the same flexibility as people using a manual system. The need for an abbreviated language or "common language" to be used throughout the Bell Operating Companies became manifest.

4.0.4 This COMMON LANGUAGE Code is defined as the standard representation of information, or data, used as input or output by people in a mechanized environment. It is the language people use to communicate with a machine in an ongoing operation and which the machine uses to communicate with people in the same operation. It is also the language that people use to communicate with each other as well as machine to machine communication.

4.0.5 The need for a Bell System approach to COMMON LANGUAGE Codes was identified in the early 1960s as a result of AT&T efforts to standardize the equipment ordering procedures and Operating Telephone Company efforts at mechanization in the trunk facilities area. Over the next few years, a series of AT&T sponsored committees and task forces developed some of the first COMMON LANGUAGE code sets. The most important of the early efforts was the work done by a COMMON LANGUAGE Task Force in the mid-1960s. The task force consisted of representatives from AT&T, several Operating Telephone Companies, Western Electric, and Bell Laboratories. It developed and published codes to be used for the identification of equipment, circuits, facilities, and locations. As a result of these efforts, a small COMMON LANGUAGE code group was established at AT&T.

4.0.6 When the Business Information Systems Programs (BISP) area of Bell Laboratories was established in 1967, it absorbed certain responsibilities of the AT&T COMMON LANGUAGE group. The Bell Laboratories Language Standards Department became responsible for the development and maintenance of codes required in BISP projects. In addition, it assumed responsibility for the continued development and maintenance of the codes previously developed by the Task Force. It soon became evident that there was a need for a centralized group to develop and maintain codes required by AT&T and operating company projects as well as the BISP projects. In April 1971, the charter of the Bell Laboratories Language Standards Department was expanded to include such development and maintenance. In 1978, the COMMON LANGUAGE department became a separate project of BISP. With divestiture on January 1, 1984, the COMMON LANGUAGE department became the responsibility of Bellcore. Shortly afterward, the COMMON LANGUAGE department changed

its name to the Language Standards Department. The applicable documents, formerly known as BSPs, are now referred to as Bellcore Practices.

5. CODING TERMINOLOGY

5.0.1 It is important to know some basic coding terminology in order to understand the development and maintenance of codes. This terminology, used throughout the COMMON LANGUAGE Code documentation, is described in the following paragraphs.

5.0.2 A **code** is defined as a shortened representation of a word or group of words using letters, numbers, and/or symbols. The word(s) being coded is known as a **data value.** A group of data values and codes related to a particular subject is known as a **code set.** Table A contains a sample code set for resistance limits.

Table A. Sample Code Set: Resistance Limits

DATA VALUE	CODE
0 - 885 ohms	8
886 - 1200 ohms	12
1201 - 1300 ohms	13
1301 - 1600 ohms	16
1601 - 1800 ohms	18
1801 - 2800 ohms	28
2810 - 3600 ohms	36

5.0.3 A code consists of letters, numbers, and symbols that are known as characters. The type of characters that appear in the code, and the positioning of these characters, are known as its character set. In describing a character set, "A" represents any alphabetic character, "N" a numeric character, "S" a special character (i.e., symbol or space), "X" an alpha, numeric, or special character. Thus, a code having a character set of "AANX" is a 4-character code with its first and second characters in alpha, its third character numeric, and its fourth character alpha, numeric, or special.

5.0.4 The **structure** of a code refers to its overall composition. In other words, the type of characters that constitute the code (without taking special characters into consideration) is critical. A code structure may be **alphabetic** (all alpha

characters), **numeric** (all numeric characters), or **alphanumeric** (alpha and numeric characters). The code example in 5.03 (AANX) is an alphanumeric code structure.

5.0.5 A fourth type of code structure is known as **mnemonic.** A mnemonic code is designed to aid memory. For example, a user can quickly associate the code "NC" for the state of North Carolina. The other three types of codes are all random assignments of characters that do not aid memory. Almost all of the codes developed by the Language Standards Department are mnemonic.[1]

5.0.6 Table B, Page 9 illustrates the various types of code structures and offers an example and description of each.

5.0.7 A code may be further described as either a code element or a code chain. A **code element** is a character or set of continuous characters that represent a basic item of data. A place code, for example, is considered a code element because it represents a basic "unit" of data, such as SPFD for the city of Springfield. A combination of two or more code elements is known as a code chain. For example, the location-place state code, SPFDIL, for Springfield, Illinois, is a code chain consisting of two code elements: the place code and a state of the States of the United States code. The order of the elements in a code chain is referred to as its **sequence.**

5.0.8 **Code length** refers to the minimum and maximum number of characters in the code. A code may be either fixed length or variable length.[2]

6. CODE DEVELOPMENT PROCEDURE

6.0.1 The COMMON LANGUAGE **code development procedure** is used by the Language Standards Department in the development of new code sets. The code development procedure begins when a coding request is received from a Bellcore client. A **coding request** is a list of specifications required by the requestor to meet a coding need.

6.0.2 The information supplied on the coding request indicates the use the code will serve, the type of people who will use it and their tasks, and a list of all known data values and references. It also indicates the growth potential of the data values, existing code sets which may satisfy the coding requirements, constraints

1. Mnemonic codes usually consist of alpha characters only. In a few cases, however, they may consist of numeric characters or combinations of alpha, numeric, or special characters. Mnemonic codes may be specifically identified as alpha-mnemonic, numeric-mnemonic, or alphanumeric-mnemonic.

2. The basic characteristics of a code (i.e., length, structure, character set, and sequence) are collectively known as the code make-up.

which will affect the code make-up, forms on which it will be used, the due date, and any other pertinent information or remarks.

6.0.3 When the request for a new code is received, a code analyst in the Bellcore Language Standards Department begins a code study. The code study involves, briefly, gathering all pertinent information, analyzing this information, preparing a code recommendation, and documenting the findings.

6.0.4 In a code study, an important tool used in gathering data is the department's **Reference Library.** The library is a collection of documents from Bellcore organizations, other industries, and from the United States government that contains information related to coding. These documents are scrutinized to determine if any codes exist that can satisfy the request.

6.0.5 The code analyst must next analyze the data gathered and, by applying the various coding principles and guidelines, make a code recommendation. There are twenty-six basic **Code Design Principles** used as a guide by the analyst (see Table C, Page 10). The code design principles are the result of detailed studies of the human/code relationship conducted by the new Language Standards Department in conjunction with the Human Performance Technology Center and consulting psychologists from the American Institute for Research (a private outside consulting firm).

6.0.6 The analyst then prepares a code recommendation and documents the code study. The documentation, known as a **"code study package,"** consists of a code recommendation page(s), significant information supporting the recommendation, statistical analysis, evidence of code usage found during the study, exhibits, and a bibliography.

6.0.7 When the code study package is completed, it is given a permanent identifying number by the Language Standards Department. This number is known as the Bellcore Standard Reference Number. After the necessary approvals, the code study package is sent to the BCCs COMMON LANGUAGE Coordinators for final review and comments.

6.0.8 When all comments concerning the code have been answered, the appropriate Language Standards Technical Advisory Group (TAG) approves the code and the Language Standards Department has it published as a Bellcore Practice (BR). A copy of each code study package prepared by the department is indexed and placed in the Language Standards Department's Reference Library.

6.0.9 BR 751-000-101 contains examples of codes and abbreviations that have been approved by the TAG. This practice gives the code name, description, length, structure, Bellcore Practice reference, and an example for each code or abbreviation. It also includes the organization responsible for assigning individual codes for which there is a COMMON LANGUAGE requirement (e.g., U. S. Postal Service assigns ZIP Codes®; Bellcore Language Standards

Department assigns place codes; local telephone companies assign telephone exchange codes).

7. BELLCORE COMMON LANGUAGE PRACTICES

7.0.1 Once a code is approved by the TAG, it is ready for publication as a Bellcore Practice. This is accomplished by preparing a draft practice and forwarding it to the BCCs for approval. After their approval, the practice is issued and distributed by means of the new Bellcore publication distribution procedure.

7.0.2 Bellcore COMMON LANGUAGE practices are published in two divisions, with the numerical designations 795 and 751. The codes for the identification of equipment; circuits, both message and special services; facilities; and locations are published in the 795 Division. All other COMMON LANGUAGE codes and abbreviations are published in the 751 Division. Refer to the Division Indexes, BR 795-000-000 and 751-000-000, respectively, for complete lists of the COMMON LANGUAGE standard practices.

8. CODE MAINTENANCE

8.0.1 Once a code has been approved as a COMMON LANGUAGE standard code, the Language Standards Department establishes procedures for its maintenance. Some codes, such as those representing date and time, require little maintenance. Others, such as the location identification and equipment identification codes, require extensive maintenance procedures to keep them current. Changes to existing codes should be minimized or avoided, if possible. The individual COMMON LANGUAGE practices contain specific maintenance procedures.

8.0.2 Changes, deletions, or additions to existing code sets should be sent to the Language Standards Document Administrator, through the BCC COMMON LANGUAGE Coordinator, using the established updating procedures specified in COMMON LANGUAGE Practice BR 751-000-102. Inquiries from other

than the BCCs should be directed through normal organizational channels to the Language Standards Department.

8.0.3 The Language Standards Department maintains the COMMON LANGUAGE Code sets and re-issues the practices as required.

9. ABBREVIATIONS AND ACRONYMS

9.0.1 The Language Standards Department is responsible for the development, maintenance, and documentation of standard abbreviations.

9.0.2 An abbreviation is a shortened representation of a word or a group of words. COMMON LANGUAGE abbreviations are categorized as single word abbreviations, multi-word abbreviations, system acronyms, or symbols. Refer to BR 751-410-100, "COMMON LANGUAGE® Standard Abbreviations-Description," for additional information concerning abbreviations. Some sample acceptable abbreviations are listed below:

Height = HGT (simple abbreviation)

Direct Circuit = DIR CKT (multi-word abbreviation)

CLONES (system acronym)

Percent = % (symbol)

9.0.3 It is important to note that the terms abbreviation and code are not synonyms. A code can be considered more structured and subject to more constraints than an abbreviation. Each code within a certain code set, for example, pertains to a particular element that is unique and is of a specific length and character set. Abbreviations cover a wide range of subjects; they are not unique, and vary greatly in length and character set. To illustrate, the COMMON LANGUAGE standard abbreviation for the word transmission is "TRMSN." A three-character, alpha-mnemonic code was required to indicate transmission trouble in the Trouble-Pair and Binding-Post code set. Since the standard abbreviation could not be used because of its size, the code "TMS" was assigned.

9.0.4 The COMMON LANGUAGE abbreviations are contained in the **Abbreviations Master List.** This lists all abbreviations (and the words they represent) in common use throughout the BCCs. The Master List is published in BR 751-410-

101 (alphabetical order according to word) and in BR 751-410-102 (alphabetical order according to abbreviation).

9.0.5 The Language Standards Department will develop new abbreviations and revise and publish abbreviation practices as required. Abbreviations are developed according to a set of rules established by the Language Standards Department.

9.0.6 An **acronym** is a word formed from the initial letter or letters of each of the successive parts or major parts of a compound term. An acronym may, or may not, form a pronounceable word. The following examples illustrate acronym formation:

Circuit **D**esign **S**ystem = (CDS)
Plug-in **I**nventory **C**ontrol **S**ystem = (PICS)

Table B. Types Of Code Structures

STRUCTURE	EXAMPLE	DESCRIPTION
Alphabetic	EAF	Facility Type Code: Lenkert 42C Carrier Terminal
Numeric	1591	Typical Account-General ledger code
Alphanumeric	D2106	Typical Equipment Measurement Code (EMC)
Mnemonic		
Alpha	JAN	Month-Mnemonic Code: January
Numeric	31	Day of Month Code: 31st Day of Month
Alphanumeric	JAN 31 84	Calendar Date-Mnemonic Code: January 31, 1984

Table C. Code Design Practices

NUMBER	DESCRIPTION
1	Codes should be designed for the least skilled workers within the group of code users.
2	Preferences of code users should be considered. For example, workers prefer to work with alphabetic codes, rather then numeric or alphanumeric codes.
3	Code sets currently being used by the workers should be given preference. For example, workers using mostly numeric codes will make more errors in handling a newly introduced code containing alphabetic characters than a code with numeric characters.
4	A code should uniquely identify the information or data value it represents.
5	In determining the length of a code, room must be allowed for future growth of the code set.
6	Numeric characters should be given preference in codes designed for simple tasks.
7	Alphabetic characters should be given preference in codes designed for complex tasks. It is easier to convey meaning with alpha characters than with numeric characters.
8	Meaningfulness should be built into codes by whatever means and in as many ways as possible. Thus, the use of mnemonic codes significantly decreases the number of errors.
9	When meaning cannot be given equally to all codes within a set, preference should be given to the most frequently used codes.
10	In determining the length of a code, the capabilities of the code user should be considered. Users tend to work best with short codes, i.e., three to five characters in length.
11	Common English usage practices should be considered in dividing or forming long code chains. For example, a long code chain should be divided into elements in order to aid memory and recall.
12	Techniques for "psychologically" shortening long codes should be used. For example, users who are told the meaning of codes recall them more efficiently than users not told the meaning of the codes.

Table C. Code Design Practices (Continued)

NUMBER	DESCRIPTION
13	Cost tradeoffs should be carefully considered before lengthening a code with a check code or check digit(s). In other words, the addition of an error-check code to a code may actually increase the number of errors.
14	Codes should consist of characters already in the worker's vocabulary. Use alpha and numeric characters and commonly used symbols. Symbols, if possible, should be given the same meaning they have in the English language.
15	The group of characters from which a code set is drawn should be as small as possible, but should still consider the uniqueness requirements and anticipated growth. The larger the group of characters from which the code is drawn, the more difficult it is to remember the code.
16	Avoid, whenever possible, mixing different type characters within a code set. For example, codes containing both alphabetic and numeric characters cause more errors than codes containing all alphabetic or all numeric characters.
17	Rules for the design of a code set should be clearly defined and consistently followed. Rules for code design provide the user with a means for decoding and aid in both code learning and use.
18	Symbols should be used consistently for the same meaning or function in all departments and companies handling the code.
19	Characters that are easily confused visually or acoustically (e.g., letter I and number 1) should be avoided.
20	When necessary to use characters that are easily confused, distinguishing marks should be used to make differences more evident. However, differences in type and placement of distinguishing marks may increase the number of errors.
21	Simple characters should be given preference over complex characters. Military studies on symbology suggest that certain characters, (simple characters "L" and "A", for example) cause fewer errors than complex characters, like "G" and "Q."
22	If possible, characters should be positioned in a code in order to improve performance. For example, studies have shown that using a three-character code consisting of two alpha characters and one numeric character or two numeric characters and one alpha character, errors are reduced significantly when the odd character is not in the middle. For example, GA2 causes fewer errors than G2A.

Table C. Code Design Practices (Continued)

NUMBER	DESCRIPTION
23	Code elements and chains should be arranged or formatted according to the user's order of need for information. Codes should be formatted sequentially in the order they will be used at the output point.
24	The arrangements, or format, of code elements and chains should be coordinated among several code users. Inconsistencies among departments and/or groups in the arrangement of the elements of a code increase coding error.
25	The length of a code should be kept to the minimum required by users. Errors increase disproportionately with increases in the length of a code.
26	Codes should be formatted so as to facilitate ease of scanning for accuracy and completeness. In other words, an easily detected pattern in a code enhances meaningfulness and reduces errors.