WITHDRAWN TAS
JULY 1998

ENDORSED BY SANZ
AS 1209—1978
UDC 681.3.048

Australian Standard 1209—1978

PROGRAMMING LANGUAGE COBOL

STANDARDS ASSOCIATION OF AUSTRALIA
Incorporated by Royal Charter

k

THE FOLLOWING SCIENTIFIC, INDUSTRIAL AND GOVERNMENTAL ORGANIZATIONS and departments were officially represented on the committee entrusted with the preparation of this standard:

Australian Banks Automation Committee

Australian Bureau of Statistics

Australian Computer Users Association

Australian Public Service Board

CSIRO, Division of Computing Research

Department of Defence

Life Offices Association of Australasia

Manufacturers of data processing equipment

National Library of Australia

Office Equipment Industry Association of Australia

Qantas Airways Limited

Telecom Australia

Universities and colleges

This standard, prepared by Committee MS/20 Computers and Information Processing, was approved on behalf of the Council of the Standards Association of Australia on 18 September 1978, and was published on 1 January 1979.

To keep abreast of progress in industry, Australian standards are regularly reviewed. Suggestions for improvements to published standards, addressed to the head office of the Association, are welcomed.

AUSTRALIAN STANDARD

PROGRAMMING LANGUAGE COBOL

AS 1209—1978

First published 1972
Revised 1978

PUBLISHED BY THE STANDARDS ASSOCIATION OF AUSTRALIA
STANDARDS HOUSE, 80 ARTHUR ST, NORTH SYDNEY, N.S.W.

ISBN 0 7262 3297 1

PREFACE

This revision of AS 1209—1972 was prepared by the Association's Committee on Computers and Information Processing.

It specifies the form and interpretation of data processing programs expressed in COBOL for the purpose of facilitating interchangeability of such programs for use in a variety of automatic data processing systems.

The 1972 edition of this standard was based on American National Standard COBOL, ANSI X3.23—1968. Similarly this revision has been based on ANSI X3.23—1974. Like its predecessor, this document provides specifications for both the form and interpretation of programs expressed in COBOL.

The organization of COBOL specifications in this standard is based on a functional processing concept. The standard defines a Nucleus and eleven functional processing modules: Table Handling, Sequential I-O, Relative I-O, Indexed I-O, Sort-Merge, Report Writer, Segmentation, Library, Debug, Inter-program Communication, and Communication. Each module contains two or three levels with nine modules having a null set as the lowest level. In all cases, lower levels are proper subsets of the higher levels within the same module. The minimum standard is defined as the low level of the Nucleus plus the low level of the Table Handling and Sequential I-O modules. Full Australian Standard COBOL is defined as the highest level of the Nucleus and the eleven processing modules. The major technical differences between this standard and its predecessor are detailed in Appendix B on pages XIV-9 through XIV-34.

COBOL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

No warranty, expressed or implied, is made by any contributor or by the CODASYL Programming Language Committee as to the accuracy and functioning of the programming system and language. Moreover, no responsibility is assumed by any contributor, or by the committee, in conjunction therewith.

The authors and copyright holders of the copyrighted material used herein, viz

> FLOW-MATIC (trademark of Sperry Rand Corporation), Programming for the UNIVAC I and II, Data Automation systems copyrighted 1958, 1959, by Sperry Rand Corporation; IBM Commercial Translator Form No F 28-8013, copyrighted 1959 by IBM; FACT, DSI 27A5260-2760, copyrighted 1960 by Minneapolis-Honeywell

have specifically authorized the use of this material in whole or in part, in the COBOL specifications. Such authorization extends to the reproduction and use of COBOL specifications in programming manuals or similar publications.

This standard is technically identical with ISO 1989—1978, since the ISO standard is the same as ANSI X3.23—1974. Grateful acknowledgement is made to the American National Standards Institute for permission to reproduce from the latter document.

TABLE OF CONTENTS

SECTION I: INTRODUCTORY INFORMATION

SECTION VIII: REPORT WRITER MODULE

SECTION IX: SEGMENTATION MODULE

STANDARDS ASSOCIATION OF AUSTRALIA

Australian Standard Specification
for

PROGRAMMING LANGUAGE COBOL

1. INTRODUCTION TO THE STANDARD

1.1 SCOPE AND PURPOSE

The scope of this standard is to specify both the form and interpretation of programs expressed in COBOL. Its purpose is to promote a high degree of machine independence in such programs in order to permit their use on a variety of automatic data processing systems.

1.2 STRUCTURE OF LANGUAGE SPECIFICATIONS

The organization of COBOL specifications in this standard is based on a functional processing module concept. The standard defines a Nucleus and eleven functional processing modules: Table Handling, Sequential I-O, Relative I-O, Indexed I-O, Sort-Merge, Report Writer, Segmentation, Library, Debug, Inter-Program Communication, and Communication. Each module contains either two or three levels. In all cases, the lower levels are proper subsets of the higher levels within the same module. Nine modules contain a null set as their lowest level.

This organization provides the flexibility necessary to tailor specifications in such a way that they will satisfy the requirements of a large variety of data processing applications. At the same time, inherent in this organization is the ability to determine, with a greater degree of certainty than previously possible, the elements of the standard that are included in a given compiler.

The following is a characterization of the contents of the component levels of each module.

The Nucleus contains language elements that are necessary for internal processing. This module is divided into two levels. The low level supplies elements necessary to perform basic internal operations, i.e., the more elementary options of the various clauses and verbs. The high level of the Nucleus provides more extensive and sophisticated internal processing capabilities.

The Table Handling module contains the language elements necessary for: (1) the definition of tables, (2) the identification, manipulation and use of indices, and (3) reference to the items within tables. This module is divided into two levels. The low level provides the ability to define fixed length tables of up to three dimensions, and to refer to items within them using either a subscript or an index. The high level provides for the definition of variable length tables. In addition, facilities for serial and nonserial lookup are provided by the SEARCH verb and its attendant Data Division clauses.

The Sequential I-O module contains the language elements necessary for the definition and access of sequentially organized external files. The module is divided into two levels. The low level contains the basic facilities for the definition and access of sequential files and for the specification of checkpoints. The high level contains more complete facilities for defining and accessing these files.