

IEEE ABBET™

Trial-Use Standard for A Broad-Based Environment for Test (ABBET)

Overview and Architecture

Sponsor

**Standards Coordinating Committee 20,
Abbreviated Test Language for All Systems (ATLAS)**

Approved June 17, 1993

IEEE Standards Board

Abstract: The overall concept of A Broad-Based Environment for Test (ABBET) is defined, and mandatory requirements for implementations ABBET are specified. The elements of ABBET and their interrelationships are described. Guidelines and requirements governing the elements of the ABBET set of standards and guides are established, and common terms to be used throughout the set are defined.

Keywords: ABBET, ABBET layer, base document, component standard, reference document

ABBET is a trademark of the Institute of Electrical and Electronics Engineers, Inc.

The Institute of Electrical and Electronics Engineers, Inc.
345 East 47th Street, New York, NY 10017-2394, USA

Copyright © 1993 by the Institute of Electrical and Electronics Engineers, Inc.
All rights reserved. Published 1993. Printed in the United States of America

ISBN 1-55937-359-8

No part of this publication may be reproduced in any form, in an electronic retrieval system or otherwise, without the prior written permission of the publisher.

IEEE Standards documents are developed within the Technical Committees of the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Board. Members of the committees serve voluntarily and without compensation. They are not necessarily members of the Institute. The standards developed within IEEE represent a consensus of the broad expertise on the subject within the Institute as well as those activities outside of IEEE that have expressed an interest in participating in the development of the standard.

Use of an IEEE Standard is wholly voluntary. The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of all concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason IEEE and the members of its technical committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

IEEE Standards documents are adopted by the Institute of Electrical and Electronics Engineers without regard to whether their adoption may involve patents on articles, materials, or processes. Such adoption does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the standards documents.

Publication of this trial-use standard for comment and criticism has been approved by the Institute of Electrical and Electronics Engineers. Trial-use standards are effective for 24 months from the date of publication. Comments for revision will be accepted for 18 months after publication. Suggestions for revision should be directed to the Secretary, IEEE Standards Board, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08817-1331, and should be received no later than February 8, 1995. It is expected that following the 24-month period, this trial-use standard, revised as necessary, shall be submitted to the IEEE Standards Board for approval as a full-use standard.

Foreword

[This foreword is not a part of IEEE Std 1226-1993, IEEE ABBET Trial-Use Standard for A Broad-Based Environment for Test (ABBET), Overview and Architecture.]

This standard provides a conceptual overview and architecture of A Broad-Based Environment for Test (ABBET). This trial-use standard is being distributed to get concurrence on the concept level and direction of ABBET. It is anticipated that the wider audience of the trial-use standard will also bring comment on areas that are only partially defined in the architecture. The beneficial impact of this trial-use standard on interchangeability of test environment components built according to this trial-use standard awaits more detailed definition of system architecture and component standard interfaces.

This standard represents a major departure from the previous ATLAS standards, which focused on human-to-human and human-to-machine communications of test specifications. This trial-use standard emphasizes machine-to-machine communications to enable unprecedented automation in test program development, use, and maintenance. Along with this new scope comes the requirement to increase the precision, predictability, and reproducibility of the communications in a test environment where a reduction of human effort and an increase in machine effort is anticipated.

Introduction

[This introduction is not a part of IEEE Std 1226-1993, IEEE ABBET Trial-Use Standard for A Broad-Based Environment for Test (ABBET), Overview and Architecture.]

The benefits of using standards in civilian and military test-related applications have long been recognized. The scope for standardization extends from low-level standards associated with instrument control to high-level standards associated with specifying tests in an implementation-independent manner.

In the 1960s, Aeronautical Radio Incorporated (ARINC) started the development of the Abbreviated Test Language for Avionics Systems (ATLAS). In 1976, management of the ATLAS standard was passed to the IEEE and the meaning of the ATLAS acronym was changed to Abbreviated Test Language for All Systems to reflect its broader field of application.

Within the IEEE, development of ATLAS-related standards (IEEE Std 416-1984,* IEEE Std 716-1989, IEEE Std 771-1989, etc.) was vested in an ad hoc committee, which became Standards Coordinating Committee 20 (SCC20). This was also known as the ATLAS Committee. In the mid-1980s, SCC20 broadened the scope of its activities to include other potential standards related to testing, including Automatic Test Program Generation (ATPG) and Test Equipment Description Language (TEDL).

The initial motivation for ATLAS development was to provide unambiguous human-to-human communication of test requirements, rather than to provide for an Automatic Test Equipment (ATE) interpretable input. ATLAS evolved to become a test language that performs the functions of a specification language and a programming language. As a general programming language, it has some limitations; however, it fulfills an important role as a specification language with signal and event-oriented statements not found in other languages.

Although the test community had seen the virtue of a single high-level test language standard, the broader computing community had not come to any agreement on a single common language. As a result, the burgeoning computer power of the late-1970s was accompanied by proliferation of computer languages.

The United States Department of Defense (DoD) was concerned with the cost of supporting many computer languages and thus developed the Ada Programming Language (ANSI/MIL-STD-1815A-1983). High-quality commercial Ada compilers and other development tools are now widely available.

Ada was developed to support advanced software engineering practices. Specifically, Ada addresses the controlled preparation and life cycle support of large, complex programs. In addition, Ada supports the definition of standardized interfaces at various levels of abstraction. This capability can be used to develop a layered approach to the implementation of test software and provide programming capabilities beyond those supported by ATLAS.

By the late-1980s, there was within SCC20 a growing feeling that Ada might provide the framework needed to integrate the various test-related standards being developed.

Discussions about how Ada could and should be used ranged from using Ada only for implementing ATLAS compilers to complete replacement of ATLAS with Ada.

Eventually, it was generally agreed that an ideal standardized test domain would utilize the advantages of Ada programming and implementation features while retaining the high-level, UUT-oriented (UUT stands for Unit-Under-Test) test specification aspects of ATLAS.

*Information on references can be found in clause 2.

ATLAS would remain as a standard test specification language, with Ada being used as a standard implementation route for ATLAS. This would not eliminate the possibility of migrating to the exclusive use of Ada in the future. But this use would occur in an Ada environment that fully supports the semantic intent of ATLAS and preserves the abstract model of the ATE system that ATLAS provides the test engineer.

The standardized Ada test environment would also facilitate integration of other test-related standards.

At the 89-1 SCC20 meeting in Edinburgh, Scotland, the acronym ABET (ATLAS/Ada Based Environment for Test) was coined, a Project Authorization Request (PAR) was prepared, and the first of a standard draft was written, setting forth the requirements to be satisfied by the ABET standard. The ABET PAR was approved by the IEEE Standards Board on November 9, 1989. In that PAR, the scope of the proposed ABET standard is:

“To define an alternative environment for the implementation of ATLAS test specifications, using the programming capabilities of the Ada Language.”

At the 90-1 SCC20 meeting in San Diego, California, the ABET requirements document and an associated document giving the rationale for the various requirements were revised and approved by the ABET Subcommittee.

During this same period, a third document was written to describe the ABET architecture, that is, the elements of the ABET environment and the relationships between the elements. During the evolution of the architecture document, the ABET concept was broadened to encompass a complete integrated diagnostic environment from UUT design through life-cycle maintenance.

It was soon realized that due to the breadth of the ABET concept, multiple standards would be required to define the various ABET functions and interfaces. In such situations, the IEEE recommends the creation of a base standard, IEEE Std XXXX-19XX, and a set of component standards, numbered XXXX.1, XXXX.2, and so forth.

At the 91-2 SCC20 meeting in Toulouse, France, an amendment to the P1226 PAR was approved to change the project title to Ada Based Environment for Test (ABET). This amendment was approved by the Standards Board in March 1992.

At the 92-2 SCC20 meeting in San Antonio, Texas, an amendment to the P1226 PAR was approved to change the project title to A Broad-Based Environment for Test (ABBET). This amendment was approved by the Standards Board in March 1993.

As a result this standard, IEEE Std 1226-1993, is the base standard for the ABBET set of standards. It combines and expands on information from the requirements, rationale, and architecture documents.

Anticipated users of the ABBET set of standards include:

- a) UUT developers
- b) UUT maintainers
- c) Test program set developers
- d) Test program set maintainers
- e) ATE system developers
- f) ATE system maintainers
- g) Test instrument developers
- h) Developers of ABBET-based tools and systems
- i) Developers of prime mission equipment that uses the supported UUT as a component

At the time this standard was completed, the ABBET Architecture Working Group had the following membership:

Robert McGarvey, Richard Patrick, Dez Cass, *Co-chairs*
John Heiser, Glenn Lafflin, *Co-secretaries*

Tom Andrews	Gary Hardenburg	Didier Pasquier
Jean-Yves Barbier	Leonard Haynes	Pat Patel
Michel Bantegnie	Gary Hinkle	Dave Pawson
Jeffrey Barkley	Willis Horth	John Pereira
Steven Benedikt	Elton Howard	David Phinney
Eric Bergaud	Billy Jackman	Jean Pouilly
Michael Blair	Ulla Jacobsson	Narayanan Ramachandran
Ivo Bordignon	Gary Jenkins	James Reeder
Trina Bornejko	Richard Jenkins	Michael Rhodes
Patrice Brard	Patrick Johnson	Robert Rolfe
Alan Bridges	John Kausek	William Russell
Jerrold Brody	Sigurd Kimpel	Nick Salemi
Ron Brooks	Paul Knese	Albert Sanson
Terry Campbell	Kerry Kugler	Michael Seavey
Patrice Chassard	Michel Lacroix	Steve Shutt
Jim Clemson	Don Lang	William Simpson
Frank Conforti	John Leuthe	David Smith
James Cottrell	Terry Lindemann	Greg Smith
Michel Courtois	Fred Liquori	Philip Smith
Jack Cross	Peter Lord	Robert Smith
Doug Crowe	Bill Lutton	Doug Stinson
Greg Darnell	Jean Paul Marcelot	Richard Sueyda
Bernard Dathy	Gayle Matysek	Jacques Tete
Don Davenport	Pat McCown	Mark Thullen
Bernard Dugas	Scott McEndree	Paul Vicen
Bernard Dumont	Tom McGinley	Milt Weber
Christine Fisher	Harry McGuckin	J. Richard Weger
Jacques Flandrois	Charles Morris	Daniel Weiss
Al Gilman	William Neblett	William Whitaker
John Goddard	David L. Nichols	Butch Williams
Chris Gorringer	John Odak	Roger Williams
Jim Graves	Roy Oishi	Danny Yam
Arnold Greenspan	Charles Oliverio	Michael Yarbrough
T. Winston Griffin	Catherine Ozenfant	D. York
Randy J. Hansel	Kristine Parella	William Young
Samuel Harbaugh		Jehuda Ziegler

The following persons were on the balloting committee:

Guy Adam	Jerry Grasso	David L. Nichols
Michel Bantegnie	Arnold Greenspan	Goran Nordstrom
Jeffrey Barkley	T. Winston Griffin	Hans Obermeier
Eric Bergaud	Arthur Hann	Chuck Oliverio
Michael Blair	James Hanna	Les Orlidge
Ivo Bordignon	Randy J. Hansel	Kate O'Sullivan
Maria Bostic	Gary Hardenburg	Catherine Ozenfant
Patrice Brard	Damon Hart	Richard Patrick
Alan Bridges	Robert Hayes	Dave Pawson
Jerrold Brody	Leonard Haynes	John Pfalz
Ron Brooks	John E. Heiser	Jean Pouilly
Terry Campbell	Robert Hillman	Brian Raffaelli
Michael Carpenter	Gary L. Hinkle	Narayanan Ramachandran
Michael Carter	Raymond Hutter	Michael Ray
E. Elaine Cashar	Ulla Jacobsson	James Reeder
D. J. Cass	Gary Jenkins	William Reeves
Shawn Chapman	Steve Karlovic	Michael Rhodes
Patrice Chassard	Helmuth Kaunzinger	Robert Rolfe
Richard Churcher	Wayne Kim	William Russell
Frank Conforti	Paul Knese	Albert Sanson
Andrew Costen	Kerry Kugler	Michael Seavey
James Cottrell	SMSft Glenn Lafflin	John Sheppard
Jack Cross	Terry Lambert	Lee Shombert
Bernard Daniel	Lars Olov Larsson	William Simpson
Greg Darnell	John W. Leuthe	David Smith
Bernard Dathy	Terry Lindemann	Greg Smith
George Diab	Peter Lord	Philip Smith
Joseph DiGrazia	Rolf Lundin	Joseph Stanco
Bernd Dinklage	Michael S. Martin	Doug Stinson
Pascal Dormal	Gayle Matysek	William Taylor
Bernard Dugas	Kirk McClure	Vince Terpstra
Wolf Dieter Erdmann	Robert McGarvey	J. Richard Weger
Horst Ey	Harry McGuckin	Daniel Weiss
Christine Fisher	Martin Meth	William Whitaker
Jacques Flandrois	James Mills	Kenneth Wilkinson
Janet Floyd	Charles Morris	Glenn Wright
John Goddard	David Morris	William Young
Sharon Goodall	William Neblett	Jehuda Ziegler
	Richard Neese	

When the IEEE Standards Board approved this standard on June 17, 1993, it had the following membership:

Wallace S. Read, *Chair*

Donald C. Loughry, *Vice Chair*

Andrew G. Salem, *Secretary*

Gilles A. Baril
Clyde R. Camp
Donald C. Fleckenstein
Jay Forster*
David F. Franklin
Ramiro Garcia
Donald N. Heirman
Jim Isaak

Ben C. Johnson
Walter J. Karplus
Lorraine C. Kevra
E. G. "Al" Kiener
Ivor N. Knight
Joseph L. Koepfinger*
D. N. "Jim" Logothetis

Don T. Michael*
Marco W. Migliaro
L. John Rankine
Arthur K. Reilly
Ronald H. Reimer
Gary S. Robinson
Leonard L. Tripp
Donald W. Zipse

*Member Emeritus

Also included are the following nonvoting IEEE Standards Board liaisons:

Satish K. Aggarwal
James Beall
Richard B. Engelman
David E. Soffrin
Stanley Warshaw

Rachel Auslander
IEEE Standards Project Editor

Contents

CLAUSE	PAGE
1. Overview.....	1
1.1 Scope.....	1
1.2 Purpose.....	1
1.3 Application.....	1
2. References.....	2
3. Definition of terms and acronyms.....	4
4. ABBET overview.....	5
4.1 Purpose and objectives of ABBET	5
4.2 Overview	5
4.3 ABBET layer model	6
4.4 Open architecture	8
4.5 ABBET architecture.....	9
4.6 Role of reference documents	16
4.7 ABBET document set	16
4.8 Conformance.....	17
5. Requirements on the ABBET architectural framework.....	18
5.1 General requirements	18
5.2 Test and diagnostic services.....	19
5.3 Information services.....	23
5.4 User interface services	28
5.5 Common services.....	29
6. Requirements on all component standards	30
6.1 Component standard organization	30
6.2 Content requirements	30
7. Requirements on specific component standards.....	35
7.1 Product Description Interface for ABBET component standard	37
7.2 Test Strategy/Requirements Interface for ABBET component standard.....	38
7.3 Maintenance Data Collection and Retrieval Interface for ABBET component standard.....	39
7.4 ATLAS-Level Test Procedure Interface for ABBET component standard.....	40
7.5 Test Equipment Configuration Interface for ABBET component standard	41
7.6 Test Resource Interface for ABBET component standard	42
7.7 Bus Interfaces for ABBET component standard	43
7.8 Common Ada Packages for ABBET component standard	44
8. Requirements on the guides.....	46
8.1 Introductory Guide to ABBET.....	46
8.2 Guide to the Implementation of ABBET	46
8.3 Guide to the Use of ABBET	46

CLAUSE	PAGE
9. Specification techniques	47
9.1 Headers	47
9.2 Formal Specifications	48

Trial-Use Standard for A Broad-Based Environment for Test (ABBET)

Overview and Architecture

1. Overview

The set of A Broad Based Environment for Test (ABBET) standards and guides was developed by the ABBET Subcommittee of the IEEE Standards Coordinating Committee 20 (SCC20) to serve as international standards for the application of the ISO/ANSI/MIL-STD-1815A-1983 Ada programming language¹ to automatic testing and integrated diagnostics. ABBET uses the features of Ada to specify a comprehensive environment for integrating design data, test strategies and requirements, test procedures, test results management and Automatic Test Equipment (ATE) system implementations. The **IEEE ABBET** standards consist of this base standard and a series of component standards. A number of guides are associated with the **IEEE ABBET** standards.

1.1 Scope

This standard defines the overall concept of ABBET and specifies mandatory requirements for implementing ABBET. This standard is the base document for the family of standards and guides that together comprise ABBET. This standard spans the complete capability of ABBET and its family of standards and guides.

1.2 Purpose

This standard describes the elements of ABBET and their interrelationships, establishes guidelines and requirements to govern the elements of the **IEEE ABBET** set of standards and guides, and defines common terms to be used throughout the set. The purpose of the guidelines and requirements in this standard is to ensure that the elements in the **IEEE ABBET** set of standards and guides fit together into a cohesive whole.

This standard focuses on mandatory requirements for the component standards.

1.3 Application

This standard provides an overview of the **IEEE ABBET** set of standards and guides for developing:

- a) The component standards and guides
- b) ABBET conformant test systems
- c) ABBET-based test program sets and test objects
- d) Design data for use in test
- e) ABBET environment for automated tools
- f) Shared usage of maintenance data and test results

¹ Information on references can be found in clause 2.

2. References

This standard shall be used in conjunction with the following publications. When the following standards are superseded by an approved revision, the revision will be considered for incorporation into this standard.²

ANSI/ASME Y14.26M-1989, Digital Representation for Communication of Product Definition Data.³

ARINC Spec 626-2-1992, Standard ATLAS for Modular Testing.⁴

EIA/TIA-232-E, Interface Between Data Terminal Equipment and Data Circuit Terminating Equipment Employing Serial Binary Data Interchange (Commonly known as RS-232).⁵

EIA-548 EDIF, Electronic Design Interchange Format, Version 2 0 0.

EIA EDIF, Electronic Design Interchange Format, Test View Version 2 0 301 Draft 1992.

First Extension (January 1991) to the Recommendation T.410 Series (1988) of Recommendations Contained in the CCITT Blue Book, Fascicle VII.6 Recommendation I.6, I, Tiled raster graphics.⁶

IEEE Std 100-1992, The New IEEE Standard Dictionary of Electrical and Electronics Terms (ANSI).⁷

IEEE Std 268-1992, American National Standard for Metric Practices (ANSI).

IEEE Std 416-1984 (Reaff 1991), IEEE Standard ATLAS Test Language (ANSI).

IEEE Std 488.1-1987, IEEE Standard Digital Interface for Programmable Instrumentation (ANSI).

IEEE Std 488.2-1992, IEEE Standard Codes, Formats, Protocols, and Common Commands for Use with IEEE Std 488.1-1987, IEEE Standard Digital Interface for Programmable Instrumentation (ANSI).

IEEE Std 716-1989, IEEE Standard C/ATLAS Test Language.

IEEE Std 771-1989, IEEE Guide to the Use of the ATLAS Specification (ANSI).

IEEE P981/D12.1, April 1989, Trial-Use Standard Resource Description Language for Programmable Instrumentation.⁸

IEEE Std 993-1990, IEEE Trial-Use Standard for Test Equipment Description Language (TEDL).

IEEE Std 1029.1-1991, IEEE Standard for Waveform and Vector Exchange Specification (WAVES) (ANSI).

IEEE P1029.2, September 1990, Standard for Automatic Test Program Generation Information.

² The capabilities of evolving reference standards will be integrated as they serve to improve the ABBET environment.

³ ANSI publications are available from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

⁴ ARINC publications are available from the Aeronautical Radio, Inc., 2551 Riva Rd., Sales Dept., Annapolis, MD 21401, USA.

⁵ EIA publications are available from Global Engineering, 1990 M Street NW, Suite 400, Washington, DC, 20036, USA.

⁶ CCITT publications are available from the CCITT General Secretariat, International Telecommunications Union, Sales Section, Place des Nations, CH-1211, Genève 20, Switzerland/Suisse, and in the United States from National Technical Information Service, Department of Commerce, 5285 Port Royal Road, Springfield VA 22161, USA.

⁷ IEEE publications are available from the Institute of Electrical and Electronics Engineers, Service Center, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA.

⁸ This authorized standards project was not approved by the IEEE Standards Board at the time this went to press. It is available from the IEEE Service Center.

IEEE P1029.3, September 1991, TRSL, Standard for the Representation of the Unit Under Test (UUT) Design Data for Test Generation.

IEEE Std 1076-1987, IEEE Standard VHDL Language Reference Manual (ANSI).

IEEE Std 1155-1992, IEEE Standard for VMEbus Extensions for Instrumentation: VXIbus (ANSI).

IEEE Std P1232, March 1993, Standard for Artificial Intelligence and Expert System Tie to Automatic Test Equipment (AI-ESTATE).

ISO 1000 : 1981, SI units and recommendations for the use of their multiples and of certain other units.⁹

ISO 8632-1 : 1987 Information processing systems—Computer graphics—Metafile for the storage and transfer of picture description information—Part 1 : Functional specification.

ISO 8879 : 1986, Information processing—Text and office systems—Standard Generalized Markup Language (SGML).

ISO/ANSI/MIL-STD-1815A-1983, Ada Programming Language.

ISO/DIS 10303, Industrial automation systems and integration—Product data representation and exchange.

ISO/DIS 10303-11, The EXPRESS Language Reference Manual.

ISO/IEC 8802-3 : 1993 [ANSI/IEEE Std 802.3, 1993 Edition], Information technology—Local and metropolitan area networks—Carrier sense multiple access with collision detection (CSMA/CD) and physical layer specifications.

MATE-STD-CIL, 1 August 1989, Control Interface Intermediate Language (CIIL).¹⁰

MATE-STD-SDF, Simulation Data Format Standard Handbook.

MIL-STD-1388-2, DOD Requirements for a Logistics Support Analysis Record.¹¹

NISTIR 4412, The Initial Graphics Exchange Specification (IGES) Version 5.1, National Institute of Standard and Technology.¹²

SCPI consortium - Standard Commands for Programmable Instruments (SCPI), 1993¹³

⁹ISO publications are available from the ISO Central Secretariat, Case Postale 56, 1 rue de Varembé, CH-1211, Genève 20, Switzerland/Suisse. ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 11 West 42nd Street, 13th Floor, New York, NY 10036, USA.

¹⁰MATE publications are available from Kelly Air Force Base, SA-ALC/LDAE, 404 George Road, Texas 78241-4401, USA.

¹¹MIL publications are available from the Director, U.S. Navy Publications and Printing Service, Eastern Division, 700 Robbins Avenue, Philadelphia, PA 19111, USA.

¹²NIST publications are available from the NIST office in Gaithersburg, MD 20899. NIST will provide a source as to where to obtain a copy of this specification.

¹³SCPI publications are available from SCPI consortium, 8380 Hercules Drive, Suite P3, La Mesa, CA 91942, USA.

3. Definition of terms and acronyms

A clear understanding of the following terms with respect to ABBET is particularly important in order to understand this standard.

- 3.1 ABBET application:** An end-use program constructed using one or more ABBET components.
- 3.2 ABBET component:** An implementation of the services defined in an **IEEE ABBET** component standard.
- 3.3 ABBET implementation:** The installation and utilization of one or more ABBET applications.
- 3.4 ABBET layer:** A natural grouping to the ABBET services that is recognized by the ABBET layer model. See 4.3.
- 3.5 AI-ESTATE:** Artificial Intelligence - Expert System Tie to ATE. It provides standard representation of dependency information used to drive intelligent diagnostic reasoning systems.
- 3.6 ATLAS:** Abbreviated Test Language for All Systems. The signal-oriented semantics of ATLAS are used by ABBET for definitions of test procedures in Ada.
- 3.7 IGES:** Initial Graphics Exchange Specification. A specification for representing product data, defined by ANSI/ASME Y14.26M-1989.
- 3.8 integrated diagnostics:** A process that covers the entire spectrum of diagnostic activities in all phases of weapon system acquisition.
- 3.9 PDES:** Product Data Exchange using STEP. A specification for representing product data using ISO/DIS 10303 STEP.
- 3.10 RDL:** Resource Description Language. A standardized computer language used to describe test instrument capabilities and communication sequences.
- 3.11 reference document:** A standard that shall be on hand and available to the user in order to help in the implementation of another standard.
- 3.12 service:** The lowest unit of behavior defined in ABBET. A service is specified in terms of an interface defined in Ada and its behavior.
- 3.13 TEDL:** Test Equipment Description Language. A standardized computer language used to describe the configuration of ATE systems.
- 3.14 test object:** An encapsulation of a test procedure and its associated information.
- 3.15 TRSL:** Test Requirement Specification Language. A standardized computer language used to specify test requirements.
- 3.16 test strategy:** The approach taken, given a set of ATE instrumentation, to satisfy the test requirements of a device.
- 3.17 VHDL:** VHSIC Hardware Description Language. A language format for a simulatable product description for digital systems.
- 3.18 virtual resource:** An abstract representation of a test capability, independent of its physical realization.
- 3.19 WAVES:** Waveform and Vector Exchange Specification. A standardized computer language used to describe digital test vectors.

4. ABBET overview

4.1 Purpose and objectives of ABBET

The primary purpose of ABBET is to provide for a broad-based environment for test that encompasses the total product life cycle. The Ada language is used to support the information and implementation needs of this test environment for automated testing applications.

ABBET is intended to accomplish the following objectives:

- a) Facilitate the use of existing Ada related software engineering tools in test software development
- b) Specify a modular architecture for designing ATE test software within a framework that supports development of reusable software assets
- c) Define formal relationships between information that conforms with prescribed, test-related, reference documents and the appropriate ABBET standardized interfaces
- d) Define standard mechanisms for implementation of test programs on automatic test systems using the Ada language
- e) Support the integration of ABBET capabilities into strategies for achieving integrated diagnostics objectives (This is accomplished through the development and application of standardized product models and linkages that permit the establishment of information transfers related to diagnostics at various levels.)
- f) Specify expanded capabilities based on existing test tools and environments that support the reference documents
- g) Define functions, services, and interfaces documented in a methodology independent of a specific implementation language

4.2 Overview

ABBET is a set of software interface standards for the test domain to facilitate the transition of test information and implementation of automated tests. These software interface standards are defined to support software component portability, reusability, exchangeability, and interoperability, and to serve as targets for test-related software development tools.

Each ABBET interface is defined in one of the **IEEE ABBET** component standards. The functionality of these standards is related to existing reference documents consisting of international standards and other widely accepted documents. The **IEEE ABBET** component standards define the Ada packages, data types, and data structures that are used to give a capability comparable to the associated reference documents. They also define mappings between the reference documents and the Ada packages and data structures.

Figure 1 illustrates the possible scope of ABBET in terms of interfaces and associated reference documents. The open architecture of ABBET supports the integration of these and additional interfaces and reference documents. This open architecture also supports interfaces and reference documents from new technologies. It should be noted that the primary relationships among the segments in figure 1 are radial. For example ABBET is related to *instruments* through *CIIL/SCPI*. However, there is not necessarily a relationship between adjacent segments. For example, the placement of *BIT I/O format* next to *X-windows/GKS* does not connote an interface.

As a minimum, ABBET integrates the capabilities of the following reference documents:

- a) IEEE Std 716-1989
- b) IEEE P981/D12.1, April 1989
- c) IEEE Std 993-1990
- d) ISO/ANSI/MIL-STD-1815A-1983

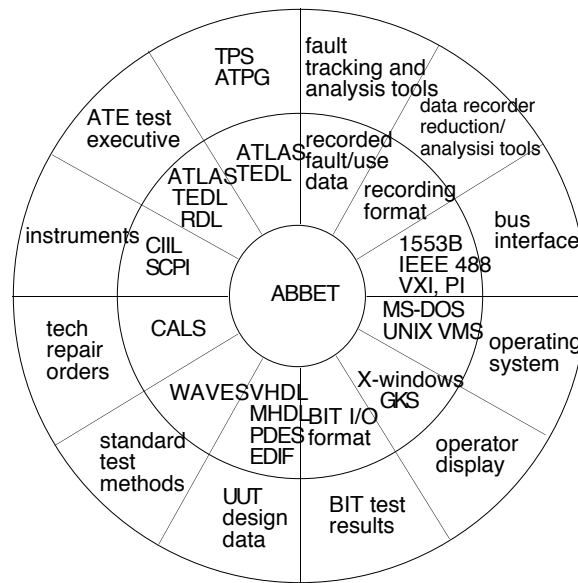


Figure 1—Illustration of ABBET's scope

4.3 ABBET layer model

There is a natural grouping to the ABBET interfaces that is recognized by the definition of the ABBET layer model.

The layer model is a conceptual model, not an implementation model. No implementation hierarchy is required by the model. ABBET applications may use any consistent combination of interfaces.

The ABBET layer model spans from UUT product descriptions to programmable instrument commands. The model groups interfaces that are associated with UUT product descriptions, test strategies and requirement specifications, test procedure code development, ATE system descriptions, and programmable instrument commands. This model serves as a framework for coupling data supplied through various ABBET interfaces to produce outputs at other ABBET interfaces.

The layers of the ABBET layer model are as follows:

- a) Product description layer
- b) Test strategy/requirements layer
- c) Test procedure layer
- d) Test resource management layer
- e) Instrument control layer

Each layer focuses on specific objects (e.g., the product, the ATE, etc.) within the test domain, and collects related standard information, functions, services, and interfaces. The architecture and specific requirements of the framework defined by this base standard and its component standards are derived from analysis of the domain defined by the layer model.

4.3.1 Product description layer

The ABBET interfaces at this layer support the bridge from design-oriented product description information to test-oriented product information.

This layer encompasses standardized interfaces to UUT product design data prepared in accordance with a variety of reference documents. The design data may include descriptions of the UUT's physical and circuit design, its internally and externally measurable characteristics, and its responses and the stimuli needed to elicit them.

The standardization of these interfaces is intended to support analysis of test and maintenance issues during the product design phase. It also supports the development of automated tools to generate test requirements or procedures directly from UUT product descriptions.

4.3.2 Test strategy/requirements layer

The ABBET interfaces at this layer support the specification of:

- a) UUT test requirements
- b) Test strategies
- c) Automatic test generation
- d) Diagnostic models, including use of expert system interfaces
- e) Maintenance data collection and retrieval

The standardization of these interfaces is intended to support the development of automated tools to generate test procedures from UUT test requirements. It is also intended to capture test information during development for later use during maintenance.

4.3.3 Test procedure layer

The ABBET interfaces at this layer support the coding of test procedures.

The interfaces specified at this layer include the full signal-oriented vocabulary, semantics, and the UUT-oriented virtual resource model of ATLAS, and the ability to relate UUT test requirements to virtual test resources that are independent of any particular ATE.

The standardization of these interfaces is intended to support reuse and portability of test program components between different ATE and testing of different UUTs. The interfaces are also intended to serve as targets for automated tools that generate test procedures from UUT product descriptions or test strategies/requirements.

4.3.4 Test resource management layer

The ABBET interfaces at this layer specify the following:

- a) Mapping virtual resources to real resources
- b) Signal routing from:
 - 1) Test resource to test resource
 - 2) Test resource to UUT pin
 - 3) UUT pin to UUT pin
- c) Accessing to non-UUT signal-oriented test functions
- d) Maintaining and reporting test resource status

The interfaces at this layer support the capability and functionality specified by TEDL and RDL.

By supporting the functionality of TEDL, ABBET greatly enhances the transportability of higher-level test specifications by making them independent of signal interconnection and device differences.

4.3.5 Instrument control layer

The ABBET interfaces at this layer standardize access to various command and communications protocols used by test resources. This includes control of generic test functions common among many test instruments and access to instrument-unique functions.

The interfaces at this layer support a variety of instrument control data buses.

The intent of the interfaces specified at this layer is to allow test resources with common functionality, made by the same or different manufacturers, to be used to perform identical functions.

4.4 Open architecture

As product technology and associated test needs evolve, so shall ABBET. Rather than restricting ABBET implementations to current technology, ABBET shall provide an open architecture or framework that allows continual technology growth and evolution. Areas where change can be expected are:

- a) Processes associated with the development of tests and test-related data
- b) Types of product-related information
- c) Tools and applications
- d) Technologies

An open architecture for ABBET means the following for an ABBET implementation:

- It can be assembled from multiple vendor-supplied interface components
- It will not restrict its extension capabilities to the original supplier
- It will execute applications written by a number of independent vendors

For example, Vendor A offers an ABBET implementation consisting of a Test Resource Management interface and an Instrument Control interface. Vendor B offers an ABBET implementation consisting of a Test Procedure interface. Providing that the implementation of the Test Procedure interface from Vendor B uses the standard ABBET interfaces provided by Vendor A's implementation, the two implementations can be joined to form a more capable composite ABBET implementation. This implementation could then be expanded at a later date with the addition of a Test Strategy/Requirement interface implemented by Vendor C.

When an ABBET interface implementation needs a capability that has been allocated to another ABBET interface, it should use the services provided by the other interface. A service that requires the support of another service should gain that support only through the specified interface (see figure 2).