BS IEC 61636-1:2016

BSI Standards Publication

Software interface for maintenance information collection and analysis (SIMICA): Exchanging test results and session information via the extensible markup language (XML)

bsi.

National foreword

This British Standard is the UK implementation of IEC 61636-1:2016.

The UK participation in its preparation was entrusted to Technical Committee EPL/501, Electronic Assembly Technology.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

Published by BSI Standards Limited 2016

ISBN 978 0 580 94225 9
ICS 25.040.01; 35.060

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 November 2016.

Amendments/corrigenda issued since publication

Date	Text affected

BS IEC 61636-1:2016

IEC 61636-1

Edition 1.0 2016-11

INTERNATIONAL STANDARD IEEE Std 1636.1™

Software interface for maintenance information collection and analysis (SIMICA): Exchanging test results and session information via the extensible markup language (XML)

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 25.040.01; 35.060

ISBN 978-2-8322-3684-0

Contents

SOFTWARE INTERFACE FOR MAINTENANCE INFORMATION COLLECTION AND ANALYSIS (SIMICA): EXCHANGING TEST RESULTS AND SESSION INFORMATION VIA THE EXTENSIBLE MARKUP LANGUAGE (XML)

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation.

   IEEE Standards documents are developed within IEEE Societies and Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of IEEE and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards. Use of IEEE Standards documents is wholly voluntary. IEEE documents are made available for use subject to important notices and legal disclaimers (see http://standards.ieee.org/IPR/disclaimers.html for more information).

   IEC collaborates closely with IEEE in accordance with conditions determined by agreement between the two organizations.

2) The formal decisions of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees. The formal decisions of IEEE on technical matters, once consensus within IEEE Societies and Standards Coordinating Committees has been reached, is determined by a balanced ballot of materially interested parties who indicate interest in reviewing the proposed standard. Final approval of the IEEE standards document is given by the IEEE Standards Association (IEEE-SA) Standards Board.

3) IEC/IEEE Publications have the form of recommendations for international use and are accepted by IEC National Committees/IEEE Societies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC/IEEE Publications is accurate, IEC or IEEE cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.

4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications (including IEC/IEEE Publications) transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC/IEEE Publication and the corresponding national or regional publication shall be clearly indicated in the latter.

5) IEC and IEEE do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC and IEEE are not responsible for any services carried out by independent certification bodies.

6) All users should ensure that they have the latest edition of this publication.

7) No liability shall attach to IEC or IEEE or their directors, employees, servants or agents including individual experts and members of technical committees and IEC National Committees, or volunteers of IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board, for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC/IEEE Publication or any other IEC or IEEE Publications.

8) Attention is drawn to the normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

9) Attention is drawn to the possibility that implementation of this IEC/IEEE Publication may require use of material covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. IEC or IEEE shall not be held responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patent Claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility.

International Standard IEC 61636-1/IEEE Std 1636.1 has been processed through IEC technical committee 91: Electronics assembly technology, under the IEC/IEEE Dual Logo Agreement.

The text of this standard is based on the following documents:

IEEE Std	FDIS	Report on voting
1636.1 (2013)	91/1360/FDIS	91/1371/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

The IEC Technical Committee and IEEE Technical Committee have decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IEEE Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA): Exchanging Test Results and Session Information via the eXtensible Markup Language (XML)

Sponsor

IEEE Standards Coordinating Committees on Test and Diagnosis for Electronic Systems (SCC20)

Approved 23 August 2013

IEEE-SA Standards Board

Abstract: This standard is intended to promote and facilitate interoperability between components of automatic test systems where test results need to be shared. The standard thus facilitates the capture of test results data in storage devices and databases, facilitating online and offline analysis. The test results schema becomes a class of information that can be used within the SIMICA family of standards. The exchange format utilizes the XML formats.

Keywords: automated test system (ATS), eXtensible markup language (XML), IEEE 1636.1™, session information, Software Interface for Maintenance Information Collection and Analysis (SIMICA), test results, XML schema

IEEE Introduction

This introduction is not part of IEEE Std 1636.1™-2013, IEEE Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA): Exchanging Test Results and Session Information via the eXtensible Markup Language (XML).

Maintainers of complex systems require the ability to capture and share test result information in a way that supports such activities as performance analysis, post-production product improvement, maintenance process improvement, and diagnostic maturation. Principal stakeholders of this project include but are not limited to maintenance organizations within various Departments/Ministries of Defense, the commercial airlines, the automotive industry, and the telecommunications industry. This standard is being developed as a component of the IEEE 1636™ Software Interface for Maintenance Information Collection and Analysis (SIMICA) project. SIMICA's purpose is to specify a software interface for access, exchange, and analysis of product diagnostic and maintenance information. Clause 4, Test results and session information, provides a subset of the data needed to satisfy SIMICA requirements.

The use of formal information models will facilitate exchanging historical test results between information systems and analysis tools. The models will facilitate creating open system software architectures for maturing system diagnostics.

The XML schema described in this standard where appropriate utilizes and references components of the IEEE Std 1671™ schema set.

It is anticipated that these schemas will be used throughout industries that utilize diagnostic and maintenance data as an exchange format that can be understood by humans or machines. In order to ensure wide acceptance throughout the user community, the schemas have been designed to encompass a broad range of use cases. To accommodate use cases beyond the released design, the schemas provide means for user extensibility.

It is anticipated that the IEEE Std 1636.1 schema will be used throughout the automatic test equipment (ATE) industry as an exchange format that can be understood by humans or machines. In order to ensure wide acceptance throughout the user community, the schemas have been designed to encompass a broad range of use cases.

Software Interface for Maintenance Information Collection and Analysis (SIMICA): Exchanging Test Results and Session Information via the eXtensible Markup Language (XML)

1. Overview

The XML schema and EXPRESS model described in this document are intended for the recording of the history of the execution and observations from a test or test session. This information includes results data directly generated by test equipment or by the test equipment operating software. The combination of this information will aid in the improvement of the test process.

The XML schema associated with this standard is based on World Wide Web Consortium (W3C)[1] XML eXtensible Markup Language (XML) 1.0 Proposed Edited Recommendation [B1].[2]

The EXPRESS model associated with this standard is based on ISO 10303-11:1994 [B9].

[1] W3C is a registered trademark of the World Wide Web Consortium.
[2] Information on references can be found in Annex C.

1.1 Scope

The scope of this standard is the definition of an exchange format, utilizing XML, for exchanging data resulting from executing tests of a unit under test (UUT) via a test program in an automatic test environment. The standard uses the information models of IEEE Std 1636™-2009[3] as a foundation.

1.2 Purpose

The purpose of this standard is to specify a software interface for access, exchange, and analysis of test result information. The standard enables the capture of test results data, facilitating data analysis to assess the effectiveness of test and diagnostic processes applied to complex systems. The test results information model and XML schema define the semantics and exchange format for information to be used among applications implementing the SIMICA family of standards.

1.3 Application

1.3.1 Of this document

This document provides formal specifications of the information required for the development of shared maintenance data and the results of testing. These are applicable to both the SIMICA family of standards and the ATML family of standards.

Anticipated users of this standard include the following:

a) System developers
b) System maintainers
c) Test program set (TPS) developers
d) TPS maintainers
e) Automatic test equipment (ATE) system developers
f) ATE system maintainers
g) Test instrument developers

1.3.2 Of this document's annexes

This document includes three annexes. Of these three, two are normative (Annex A and Annex B).

Annex A contains the description of each of the XML schema elements and types.

Annex B contains the description of the EXPRESS and EXPRESS-G model elements.

Annex C is informative, and thus are provided strictly as information, for both users and maintainers of this document.

[3] Information on references can be found in Clause 2.

1.4 Precedence

In the event of conflict between this document and an SIMICA family component standard, this document shall take precedence.

In the event of conflict between this document and a normatively referenced standard (See Clause 2), the normatively referenced standard, as it applies to the information being produced, shall take precedence.

In the event of conflict between this document's EXPRESS model definition and/or annotations and this document's XML schema definition and/or annotations, this document's EXPRESS model definition and/or annotations, as it applies to the information being produced, shall take precedence.

In the event of conflict between this document's EXPRESS model definition and/or annotations and an SIMICA family component standard and/or EXPRESS model, this document's EXPRESS model definition and/or annotations, as it applies to the information being produced, shall take precedence.

In the event of conflict between this document's XML schema definition and/or annotations and an SIMICA family component standard and/or XML schemas, this document's XML schema definition and/or annotations, as it applies to the information being produced, shall take precedence.

In the event of conflict between this document's XML schema definition and/or annotations and the ATML Common XML schema, this document's XML schema definition and/or annotations, as it applies to the information being produced, shall take precedence.

1.5 Conventions used in this document

1.5.1 General

All simple, complex types attribute groups and elements will be listed; explanatory information will be provided, along with examples if additional clarification is needed. The explanatory information shall include information on the intended use of the elements and/or attributes where the name of the entity does not clearly indicate its intended use. For elements derived from another source type (e.g., an abstract type), only attributes which extend the source type shall be listed; details regarding the base type shall be listed along with the base type.

The namespace prefix "c:" identifies that the type or attribute group is contained in Annex B of IEEE Std 1671™ (Schema-Common.xsd).

When referring to an attribute of an XML element, the convention of *[element]@[attribute]* shall be used. In cases where an attribute name is referred to with no associated element, the attribute name shall be enclosed in single quotes.

In tables that describe XML elements, the column "Use" indicates the occurrence constraints for each element.

a) "Required" indicates that the element shall appear exactly once.

b) "Optional" indicates that the element may appear once or not at all.

c) "1..∞" indicates that the element shall appear at least once and may appear multiple times.

d) "0..∞" indicates that the element may appear multiple times, once, or not at all.

All specifications for the EXPRESS language are given in the `Courier` type font which includes references to entity and attribute names in the supporting text.

1.5.2 Word usage

In this document, the word *shall* is used to indicate a mandatory requirement. The word *should* is used to indicate a recommendation. The word *may* is used to indicate a permissible action. The word *can* is used for statements of possibility and capability.

2. Normative references

The following referenced documents are indispensable for the application of this document (i.e., they must be understood and used, so each referenced document is cited in text and its relationship to this document is explained). For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments or corrigenda) applies.

IEEE Std 1636™-2009, IEEE Trial-Use Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA).[4, 5]

IEEE 1636.99™-2013, IEEE Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA): Common Information Elements.

IEEE Std.1671™-2010, IEEE Standard for Automatic Test Markup Language (ATML) for Exchanging Automatic Test Equipment and Test Information via XML.

3. Definitions, acronyms, and abbreviations

For the purposes of this document, the following terms and definitions apply. The *IEEE Standards Dictionary Online* [B2] should be consulted for terms not defined in this clause.[6]

3.1 Definitions

branch: In an eXtensible Markup Language (XML) document or schema, a specified element and all elements subordinate to that specified element.

component (in eXtensible Markup Language (XML) schema): The generic term for the building blocks that compose the abstract data model of the schema.

eXtensible Markup Language (XML) attribute: Name-value pair associated with an XML element.

eXtensible Markup Language (XML) document: A (text) data object that conforms to the XML requirements for being well-formed (as defined by W3C).

[4] IEEE publications are available from The Institute of Electrical and Electronics Engineers (http://standards.ieee.org/).
[5] The IEEE standards or products referred to in this clause are trademarks of The Institute of Electrical and Electronics Engineers, Inc.
[6] *IEEE Standards Dictionary Online* subscription is available at:
http://www.ieee.org/portal/innovate/products/standard/standards_dictionary.html

eXtensible Markup Language (XML) namespace: A method for distinguishing XML elements and attributes that may have the same name but different meanings. A URL is used as a prefix to a "local name." This combination ensures the uniqueness of the element or attribute name. The URL is used only as a way to create a unique prefix and does not have to resolve to a real page on the Internet.

NOTE—See Namespaces in XML 1.0 [B10] and Schenk and Wilson [B11][7].

eXtensible Markup Language (XML) schema: The structure or framework used to define a data record. This includes each field's name, type, shape, dimension, and mapping.

framework: A framework is a real or conceptual structure expressed as a set of abstract classes. The framework provides a context for the components to be used.

instance document: A textual information set grouped for some purpose that is governed by a single XML schema.

maintenance: Activity intended to keep equipment (hardware) or programs (software) in satisfactory working condition, including replacements, adjustments, repairs, software/firmware updates, and program improvements. Maintenance can be preventative or corrective. (Adapted from MIL-STD-1309D [B12].)

particle (in eXtensible Markup Language (XML) schema): A kind of component.

qualified name (in XML schema): The complete name of an XML element, attribute, or data type, including the local name and a prefix that identifies the namespace in which the local name is defined/declared.

sequence (in XML schema): A compositor for model group schema components which specifies that subordinate elements in an instance document must correspond, in order, to the specified particles.

3.2 Acronyms and abbreviations

AI-ESTATE	Artificial Intelligence Exchange and Service Tie to All Test Environments
ATE	automatic test equipment
ATML	Automatic Test Markup Language
ATS	automatic test system
DMC	Diagnostic and Maintenance Control
ISO	International Organization for Standardization
MAI	maintenance action information
SCC20	Standards Coordinating Committee 20
SIMICA	Software Interface for Maintenance Information Collection and Analysis
TPS	test program set
UUT	unit under test

[7] Notes in text, tables, and figures are given for information only and do not contain requirements needed to implement the standard.

URL	universal resource locator
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

4. Test results and session information

4.1 Background

Current automatic test system architectures are implemented with tight coupling between components. This tight coupling inhibits interoperability by requiring components of the automatic test system to be developed specific to that particular architecture. In many cases, this coupling can be reduced by developing the components that operate relative to standard interfaces.

This document will facilitate accomplishing several objectives. First, the document will serve as a single source for specifying essential test data with data elements related to the unit under test (UUT), the test station, and the test program. Second, the document will assist the automatic test equipment (ATE) industry to design and create compatible, interoperable tool sets such as data parsers and writers. Third, the standard will assist ATE users of such data (e.g., automotive, semiconductor, aerospace, and military) to process and display test results across a variety of systems.

This document has been developed as a “component standard” under IEEE Std 1636. SIMICA’s purpose is to specify software interfaces for access, exchange, and analysis of product diagnostic and maintenance information. Test results provide a subset of the data needed to satisfy SIMICA’s requirements.

This document also represents the test results component of IEEE Std 1671 (ATML). In defining its overall architecture, ATML references include both IEEE Std 1636 (SIMICA) and IEEE Std 1636.1 (this standard).

4.2 Introduction

This document’s XML schema and EXPRESS model provides a standard format for the transport of both quantitative (measured values) and qualitative (pass/fail determination) test results. The design is such that it is possible to store ancillary information such as environmental conditions and system/operator messages. This information, although not specifically “results,” is intended to permit use of an instance document for a variety of purposes, including statistical analysis and diagnostics. Some examples of this ancillary information include identifying information for the UUT, the test station, and the test program; ambient environmental conditions at the time of the test; test equipment calibration data; as well as test program input data and ancillary textual comments. This document establishes a hierarchical structure for results data to permit the grouping of a series of related test results in a single instance document.

4.3 Applicability

This document will permit test results data to be shared for a variety of purposes, including statistical analysis, diagnostics, and improvement of the unit under test (UUT) repair process.

4.4 Usage

This document presumes some knowledge of XML and the use of XML schemas. A variety of XML software tools are available in a number of computer programming languages. This document makes no presumption regarding the tool(s) being used or the specific test system(s) generating the test result information being captured in an XML instance document.

This document describes the TestResults.xsd schema and specifies the EXPRESS information model that conformant instance documents must follow. In general, this document serves as an enhancement to the annotations provided within the XML schema and EXPRESS model files.

4.4.1 XML schema representations

Within the body of this document, unless otherwise indicated, all syntax references relate to XML. Refer to XML eXtensible Markup Language (XML) 1.0 [B1] for detailed descriptions of XML data formats.

4.4.2 EXPRESS/EXPRESS-G representations

This document also uses the EXPRESS information modeling language to represent the information contained in the XML schema in a way that supports alternative exchange mechanisms and better defines the semantics of the elements of the XML schema. The information models are presented in a lexical form (EXPRESS) as well as in graphical form (EXPRESS-G) to facilitate understanding. The EXPRESS language is defined by ISO 10303-11:1994 [B9].

4.5 Relationships to other automatic test system (ATS) architectural elements

4.5.1 General

In the ATS context, a test is a procedure for evaluating or quantifying the operation of some device or system. The TestResults schema provides a standard format for the transport or storage of both quantitative (measured values) and qualitative (pass/fail determination) test results. The XML schema design is such that ancillary information such as environmental conditions and system/operator messages may also be stored in an XML instance document. This information, while not specifically “results,” is intended to permit use of an XML instance document for a variety of purposes, including statistical analysis and diagnostics. Some examples of this ancillary information includes identifying information for the UUT, the test station, and the test program; ambient environmental conditions at the time of the test; test equipment calibration data; test program input data and ancillary textual comments. The structure of the schema establishes a hierarchical structure for results data to permit the grouping of a series of related test results in a single instance document.

4.5.2 ATML test description instance documents

Within the context of an ATS, a test is any procedure for evaluating or quantifying the operation of a UUT. This test may be an implementation of an ATML test description XML instance document. In those cases where the test is an implementation of an ATML test description XML instance document, the relationships described in this clause apply.

The SIMICA test result can be qualitative (yes/no) or quantitative (a measured or calculated value). It can be a personal observation or the output of an ATS.

The SIMICA Test Results XML schema provides a standard format for exchanging and storing the measured values, pass/fail results, and accompanying data (including test operator, station information, and environmental conditions) associated with the test method implemented for the ATML Test Description XML instance documents defined test(s).

ATS architecture shall maintain a direct correlation between the ATML test description and the SIMICA test result. An example of this direct correlation is depicted by Figure 1.

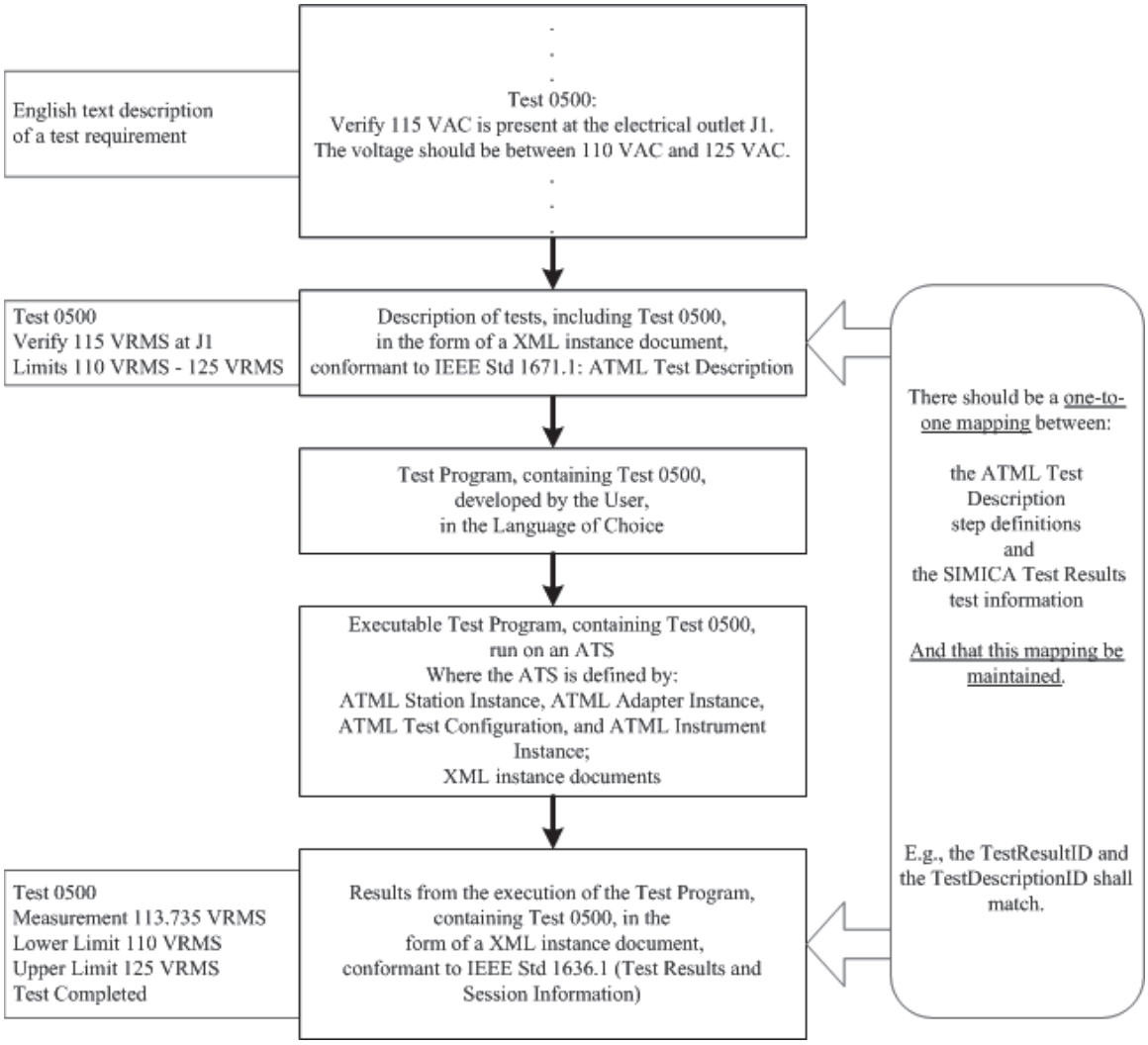

Figure 1—Direct relationship to a IEEE 1671.1 test description

Maintaining this direct correlation between the ATML test, the test method, and the SIMICA test results is vital to all interested parties (engineering, contracts, etc.) to both understand and agree upon:

— The methods of making measurements and,

— The method of obtaining the data.

5. EXPRESS model, EXPRESS-G diagram, and XML schema names and locations

The IEEE provides a download site for material published in association with IEEE Standards, presented in machine friendly format. This material is digital rights management restricted use material. The SIMICA family of standards utilizes this download site to allow easy accessibility to all of the SIMICA family EXPRESS models and XML schemas (and in some cases, example XML instance documents). As depicted by Figure 2, the IEEE download site (http://standards.ieee.org/downloads/) contains several folders, each folder labeled by an associated IEEE standards number (e.g., IEEE 1636 standards are in the 1636 folder). Each folder under the "base" IEEE standards number contains the material (XML schemas, etc) for that family member. Family members are identified by their "dot" standard number (if it is a "dot" standard) and the year in which that standard was published by the IEEE.

NOTE 1—Standards that are revised will contain a folder for the year in which the standard is reissued. Both folders (for each year the standard was published) will be present on the IEEE download Web site.

NOTE 2—Providing a particular standard has associated material that is to be made available via the download Web site, folders for that standard are not available until the standard is published by the IEEE.

Figure 2 depicts a portion of the IEEE download site, as it pertains to the SIMICA family of standards.

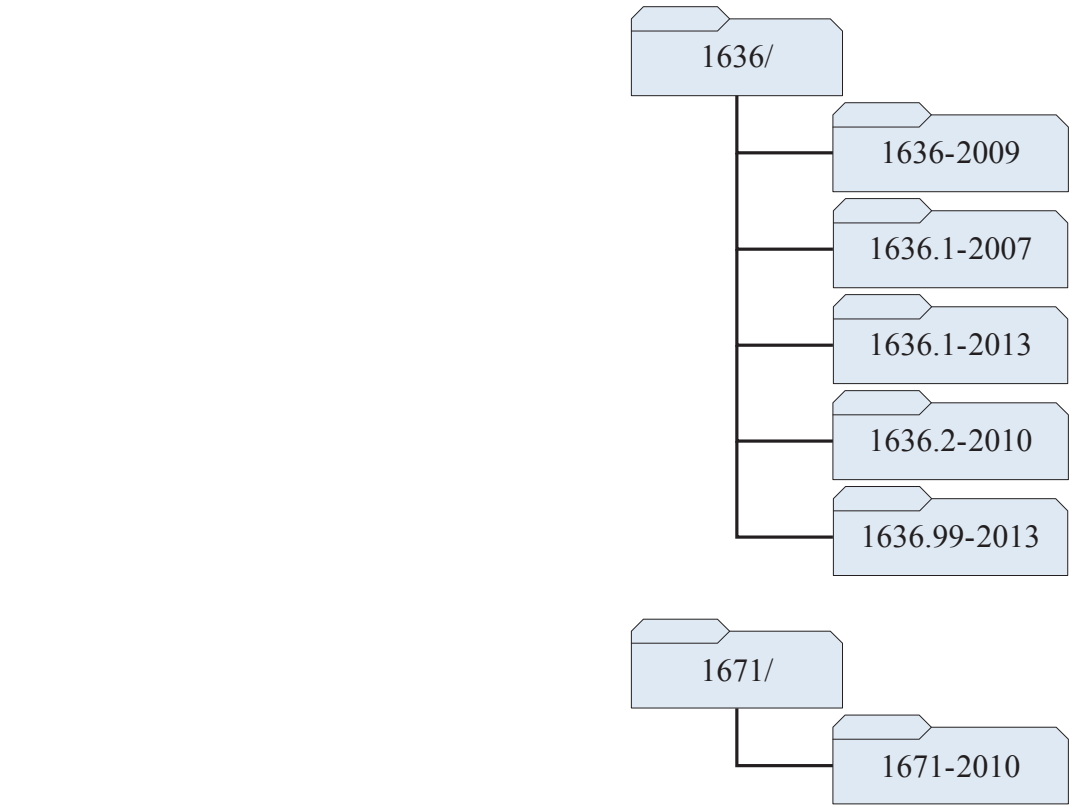

Figure 2—SIMICA-related IEEE download site structure

The IEEE Std 1636.1–associated XML schemas names, and the IEEE download site folder locations; where each of the XML schemas shall be located, is as defined in Table 1. Where the IEEE Std 1636.1–associated EXPRESS model shall be located, is as defined in Table 2.

Table 1—IEEE Std 1636.1 XML schema names and folder locations

Component	Defined in	XML schema name	IEEE download site folder (See Figure 2)
Test results and session information	Annex A.1	TestResults.xsd	1636/1636.1-2013
Test results collection	Annex A.2	TestResultsCollection.xsd	1636/1636.1-2013

Table 2—IEEE Std 1636.1 EXPRESS model and diagram names and folder locations

Component	Defined in	EXPRESS model name	IEEE download site folder (See Figure 2)
Test results and session information model	Annex B.1	1636.1.exp	1636/1636.1-2013

The XML schemas identified in Table 1 includes the ATML common XML schema and the SIMICA common XML schema. The XML schema name and the IEEE download site folder locations and where the XML schemas shall be located is as defined in Table 3.

Table 3—ATML and SIMICA common element XML schema names and locations

Component	Defined in	XML schema name	IEEE download site folder
ATML common	IEEE Std 1671-2010 Annex B.1	Common.xsd	1671/1671-2010
SIMICA common	IEEE Std 1636.99-2013 Annex A.1	SIMICACommon.xsd	1636/1636.99-2013

6. Conformance

The minimal expectation for XML instance documents conformant with this document shall be that a populated XML instance is considered valid if it complies with:

a) The Test Results XML schema (Defined in Annex A of this document, and available as described in Clause 5)

b) The Test Results EXPRESS model (Defined in Annex B of this document, and available as described in Clause 5)

c) The SIMICA Common XML schema (Defined in Annex A of IEEE Std 1636.99, and available as described in Clause 5)

d) The ATML Common XML schema (Defined in Annex B.1 of IEEE Std 1671, and available as described in Clause 5)

7. Extensibility

A provision in the XML schema of an extension mechanism is necessary to ensure the viability of the specification and allow producers and consumers of SIMICA XML instance documents to interoperate in those cases where there is a requirement to exchange relevant data that is not included in the TestResults.xsd schema. The use of the extensions shall be done in a way that ensures that a conformant consumer can utilize the extended file without error, discard, or otherwise sidestep the extended data and use the non-extended portions of the data as it is intended—without error or loss of functionality.

Extensions shall be additional information added to the content model of the element being extended.

Extensions shall not repackage existing information entities that are already supported by this standard.

An extended instance document shall be accompanied by the extension XML schema and documentation sufficient to explain the need for the extension as well as the underlying semantics and relationship(s) to the base schema.

TestResults.xsd supports two forms of extension:

a) Wildcard-based extensions allow for the extension of SIMICA schemas with additional elements.

b) Type derivation allows for extending the set of data types by deriving a new type from an existing type.

XML schemas control the location and type of extension allowed.

An element has an extensible content model if in instance documents that element can contain elements and data beyond that specified by the schema. SIMICA schemas should explicitly identify where they can be extended. Only elements from a namespace different from the document namespace should be allowed in an extension. The schema shall use the TestResults `<Extension>` type to identify where extension is allowed.

Allowing the extension of a schema using type substitution should be avoided. Schemas should mark elements defined via a simple or complex type with the block attribute set to #all if type substitution is to be avoided. Elements which use type substitution as their means of definition should set the abstract attribute to true.

Annex A

(normative)

XML schemas

Should the reader not have a general understanding of XML schemas, there are several XML schema tutorials available for reference. The *XML Schema Part 0: Primer* [B13], the *XML Schema Tutorial* [B14], and the *XML Schema Tutorial, Part 1* [B15] are three available on the World Wide Web. These tutorials will help with the understanding of the contents of the TestResults.xsd schema that this Annex is defining the elements of.

Prefixes utilized in this Annex are as follows:

a) The prefix “c:” represents that the element is defined by/is inherited from the IEEE Std 1671-2010 associated Common.xsd XML schema.

b) The prefix “sc:” represents that the element is defined by/is inherited from the IEEE Std 1636.99-2013 associated SIMICACommon.xsd XML schema.

c) The prefix “tr:” represents that the element is defined within the TestResults.xsd schema.

A.1 TestResults.xsd

attributeFormDefault	unqualified
elementFormDefault	qualified
targetNamespace	urn:IEEE-1636.1:2013:TestResults

A.1.1 element TestResults

type	tr:TestResults				
properties	**content** complex				
children	tr:Personnel tr:PreTestRepairs tr:References tr:ResultSet tr:Site tr:TestDescription tr:TestProgram tr:TestStation tr:UUT tr:WorkOrder tr:Extension				
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	uuid	c:Uuid	required			**documentation** A universal unique identifier for the element containing this attribute.
	classified	xs:boolean	optional			**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional			**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
	name	c:NonBlankString	optional			**documentation** The name of the instance document. Example: Acme Widget Test Results.
annotation	**documentation** The TestResults element shall be a container for all information entities utilized in the collection of the results of testing a UUT on (or within) a particular test station, executing a particular set of tests. This root element serves as the parent element for all other elements within the Test Results XML schema.					

A.1.2 complexType Action

properties	**abstract** true					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension					
used by	**complexTypes** SessionAction Test					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.
annotation	**documentation** The Action complex type shall be used to identify specific data related to any action(s) performed; either in/during the execution of a test. If an ATML Test Description instance document is identified by the /TestResults/TestDescription element, then the value of the 'referenceID' attribute shall be identical to the value of the 'ID' attribute of the Test Description Test Group, Test or SessionAction that was executed to generate the current Action. Otherwise, this referenceID attribute shall reference the test program entity that generated this Action, for example: line number, test number, step name, etc.					

A.1.3 element Action/Data

type	c:Value
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:Datum c:Collection c:IndexedArray
annotation	**documentation** This element shall be used to identify data associated with non-test actions.

A.1.4 element Action/Description

type	c:NonBlankString
properties	**minOcc** 0 **maxOcc** 1 **content** simple
facets	**Kind** **Value** **Annotation** minLength 1 whiteSpace replace
annotation	**documentation** This element shall be used to identify descriptive information for the parent element.

A.1.5 element Action/EnvironmentalData

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Environmental tr:Extension
annotation	**documentation** This element shall be used to identify environmental conditions which are related to a test.

A.1.6 element Action/EnvironmentalData/Environmental

type	extension of c:NamedValue
properties	**minOcc** 1 **maxOcc** unbounded **content** complex
children	c:Datum c:Collection c:IndexedArray
attributes	**Name** **Type** **Use** **Default** **Fixed** **annotation** name c:NonBlankString required **documentation** A descriptive or common name for the subject data value. timeStamp xs:dateTime **documentation** Date and time associated with the environmental data. This attribute should be used where the time the data was recorded is significant within the overall context of a particular action.
annotation	**documentation** This element shall be used as a collector for multiple sets of environmental data pertinent to a particular action. Identifies the environmental data that is pertinent to a particular action.

A.1.7 element Action/EnvironmentalData/Extension

type	Extension
properties	**content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.8 element Action/Events

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Event
annotation	**documentation** This element shall be used as a collector for session occurrences such as system or operator messages.

A.1.9 element Action/Events/Event

type	tr:Event					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	tr:Message tr:Data tr:Reference tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Event. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Event.
	severity	derived by: xs:int	optional			**documentation** An enumeration of 0 to 4 inclusive that shall indicate a severity level for the Event. It shall be presumed that a value of 0 indicates least severe. Successive values shall indicate increasing levels of severity with 4 indicating most severe.
	source	c:NonBlankString	required			**documentation** An identification of the source of the Event (e.g., operator or test system).
	timeStamp	xs:dateTime	optional			**documentation** The date and time of the Event occurrence.
annotation	**documentation** This element shall be used to identify non-result data or system/operator messages generated during a test.					

A.1.10 element Action/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.11 element Action/Parameters

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Parameter
annotation	**documentation** This element shall be used to identify test parameter data.

A.1.12 element Action/Parameters/Parameter

type	tr:Parameter					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	tr:Description tr:Data tr:Reference tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Parameter. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Parameter.
	timeStamp	xs:dateTime	optional			**documentation** The date and time associated with the Parameter. This shall be used where the time of application of a parameter is significant within the overall context of a test.
annotation	**documentation** This element shall be used to identify a particular test parameter.					

A.1.13 complexType Event

children	tr:Message tr:Data tr:Reference tr:Extension					
used by	**Element** Action/Events/Event					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Event. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Event.
	severity	derived by: xs:int	optional			**documentation** An enumeration of 0 to 4 inclusive that shall indicate a severity level for the Event. It shall be presumed that a value of 0 indicates least severe. Successive values shall indicate increasing levels of severity with 4 indicating most severe.
	source	c:NonBlankString	required			**documentation** An identification of the source of the Event (e.g., operator or test system).
	timeStamp	xs:dateTime	optional			**documentation** The date and time of the Event occurrence.
annotation	**documentation** This complex type shall identify non-result data or system/operator messages generated during a test.					

A.1.14 element Event/Data

type	c:NamedValue					
properties	**minOcc** 0 **maxOcc** unbounded **content** complex					
children	c:Datum c:Collection c:IndexedArray					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	name	c:NonBlankString	required			**documentation** A descriptive or common name for the subject data value.
annotation	**documentation** This element permits the recording of structured event data.					

A.1.15 element Event/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.16 element Event/Message

type	c:NonBlankString
properties	**minOcc** 0 **maxOcc** unbounded **content** simple
facets	**Kind** **Value** **Annotation** minLength 1 whiteSpace replace
annotation	**documentation** This element permits the recording of textual data relevant to the subject event.

A.1.17 element Event/Reference

type	c:Document
properties	**minOcc** 0 **maxOcc** unbounded **content** complex
children	c:URL c:Text c:Extension
attributes	**Name** **Type** **Use** **Default** **Fixed** **Annotation** uuid c:Uuid required **documentation** The universal unique identifier for the document. name c:NonBlankString required **documentation** A descriptive or common name for the document. controlNumber c:NonBlankString optional **documentation** A unique identifier for the document. version c:NonBlankString optional **documentation** The version identification of the document.
annotation	**documentation** This element shall be used to identify reference(s) to external items (e.g., documents).

A.1.18 complexType Indictments

children	tr:Indictment tr:Extension					
used by	**Element** TestResult/Indictments					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	retestTestGroup	c:NonBlankString	optional			**documentation** Identifies a group of tests to be run to verify the indicted components have been repaired.
	indictmentsDateTime	xs:dateTime	optional			**documentation** The date and time that the indictments were made.
annotation	**documentation** This complex type shall be used to indicate (1) that something is wrong with one or more subcomponents of the tested UUT or (2) which subcomponents of the tested UUT are the possible causes of a test failure.					

A.1.19 element Indictments/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.20 element Indictments/Indictment

properties	**minOcc** 1 **maxOcc** unbounded **content** complex
children	tr:RepairActionRecommended tr:ReferenceDesignator
annotation	**documentation** This element shall be used to identify the information for one subcomponent of the tested UUT, which is a possible cause of a test failure. Multiple Indictment siblings shall be used to specify a set of UUT subcomponents of which one or more is the cause of the failure.

A.1.21 element Indictments/Indictment/ReferenceDesignator

type	tr:ReferenceDesignator					
properties	**content** complex					
children	tr:Description tr:FailureModes tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	classLetterAndNumber	c:NonBlankString	required			**documentation** The IEEE Std 315 subclause 22.4 Class Designation Letter and a numerical number, for the reference designator. (e.g., R1 for Resistor component number 1, A1 for SRA number 1).
	type	c:NonBlankString	required			**documentation** Defines the type of the part or component being referenced (e.g., WRA, SRA, Component, etc.).
annotation	**documentation** A subcomponent of the UUT corresponding to one of the most likely causes of the test failure.					

A.1.22 element Indictments/Indictment/RepairActionRecommended

type	tr:RepairAction					
properties	**content** complex					
children	tr:Description tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	RepairCode	required			**documentation** The type of repair action that was taken.
	code	c:NonBlankString	optional			**documentation** The application specific code for the value attribute.
annotation	**documentation** Identifies the code and description used to describe the type of recommended repair.					

A.1.23 complexType Outcome

used by	**elements** Test/Outcome TestResult/Outcome					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	tr:OutcomeValue	required			**documentation** Shall contain one of the following enumerations: “Passed” shall indicate the results of a test were within specified limits. “Failed” shall indicate that the results of a test were not within specified limits. “Aborted” shall indicate that a test did not complete. “NotStarted” shall indicate that the test did not start. “UserDefined” shall indicate that the test outcome as been defined to be something other than one of the five enumerations available. “Unknown” shall indicate that the result of the test is not known.
	qualifier	c:NonBlankString	optional			**documentation** Additional descriptive data for the ‘value’ attribute. For example, (…value=“Failed” qualifier=“High” …). In the case of (…value=“Aborted” …), qualifier shall provide essential descriptive or explanatory information regarding the reason for the test not completing normally.
	referenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this outcome, or the ID of a Test Description outcome.
	forced	xs:boolean	optional			**documentation** A True/False indication as to whether or not the recorded outcome is a user override of the observed outcome.
annotation	**documentation** This complex type shall be used to identify test outcomes (i.e., Passed, Failed, or Aborted). If an ATML Test Description instance document is identified by the /TestResults/TestDescription element, then the value of the ‘referenceID’ attribute shall be identical to the value of the ‘ID’ attribute of the Test Description Outcome. Otherwise, this referenceID attribute shall reference the test program entity that generated this outcome, for example: line number, test number, step name, etc.					

A.1.24 complexType Parameter

children	tr:Description tr:Data tr:Reference tr:Extension					
used by	**Elements** Test/Calibration Action/Parameters/Parameter					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Parameter. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Parameter.
	timeStamp	xs:dateTime	optional			**documentation** The date and time associated with the Parameter. This shall be used where the time of application of a parameter is significant within the overall context of a test.
annotation	**documentation** This complex type provides a structure in which test parameters may be reported. Parameters are generally described as configuration or input values for a test.					

A.1.25 element Parameter/Data

type	c:Value
properties	minOcc 0 maxOcc 1 **content** complex
children	c:Datum c:Collection c:IndexedArray
annotation	**documentation** This element shall permit the recording of structured, restricted values of the Parameter.

A.1.26 element Parameter/Description

type	c:NonBlankString		
properties	**minOcc** 0 **maxOcc** 1 **content** simple		
facets	**Kind**	**Value**	**Annotation**
	minLength	1	
	whiteSpace	replace	
annotation	**documentation** This element shall be used to identify unstructured, unrestricted textual descriptions of the Parameter.		

A.1.27 element Parameter/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.28 element Parameter/Reference

type	c:Document					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:URL c:Text c:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	uuid	c:Uuid	required			**documentation** The universal unique identifier for the document.
	name	c:NonBlankString	required			**documentation** A descriptive or common name for the document.
	controlNumber	c:NonBlankString	optional			**documentation** A unique identifier for the document.
	version	c:NonBlankString	optional			**documentation** The version identification of the document.
annotation	**documentation** This element shall be used to identify any referential data. A typical example would be a reference to a document or location where run-time test parameters are stored. This would be useful in cases where such parameters are not permitted in the TestResults XML instance document due to security reasons.					

A.1.29 complexType ReferenceDesignator

children	tr:Description tr:FailureModes tr:Extension
used by	**Elements** Indictments/Indictment/ReferenceDesignator Repair/ReferenceDesignator
attributes	**Name** / **Type** / **Use** / **Default** / **Fixed** / **Annotation** classLetterAndNumber / c:NonBlankString / required / / / **documentation** The IEEE Std 315 subclause 22.4 Class Designation Letter, and a numerical number, for the reference designator (e.g., R1 for Resistor component number 1, A1 for SRA number 1). type / c:NonBlankString / required / / / **documentation** Defines the type of the part or component being referenced (e.g., WRA, SRA, Component, etc.).
annotation	**documentation** This element shall be used as the base type for any element representing a reference designator, with optional descriptions of the designator and failure modes of the item the designator represents. Reference designators are used for the purpose of unambiguously identifying a component in an electrical schematic (circuit diagram) or on a printed circuit board (PCB). The reference designator usually consists of one or two letters followed by a number, e.g., R13, C1002. IEEE Std 315 subclause 22.4 contains a list of Class Designation Letters to use for electrical and electronic assemblies. For example, the letter R is the designation letter for the resistors of an assembly, C for capacitors, K for relays.

A.1.30 element ReferenceDesignator/Description

type	c:NonBlankString
properties	**minOcc** 0 **maxOcc** 1 **content** simple
facets	**Kind** / **Value** / **Annotation** **minLength** 1 **whiteSpace** replace
annotation	**documentation** The description that corresponds to the reference designator abbreviation.

A.1.31 Element ReferenceDesignator/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.32 element ReferenceDesignator/FailureModes

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:FailureMode
annotation	**documentation** The specific failure modes associated with the component indicated by the reference designator.

A.1.33 element ReferenceDesignator/FailureModes/FailureMode

type	c:NonBlankString
properties	**minOcc** 1 **maxOcc** unbounded **content** simple
facets	**Kind** **Value** **Annotation** **minLength** 1 **whiteSpace** replace
annotation	**documentation** The specific failure mode associated with the component indicated by the reference designator.

A.1.34 complexType Repair

children	tr:RepairActionTaken tr:ReferenceDesignator tr:ComponentDescription tr:ComponentInstance tr:Procedure tr:Extension				
used by	**Elements** TestResults/PreTestRepairs/Repair				
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	preventive	xs:boolean	optional			**documentation** A True/False indication as to whether or not the repair was performed as part of preventative maintenance.
annotation	**documentation** This complex type shall be used to capture the type of repair(s) conducted.					

A.1.35 element Repair/ComponentDescription

type	c:ItemDescriptionReference
properties	**content** complex
children	c:DescriptionDocumentReference c:Definition
annotation	**documentation** This element identifies the specific item repaired when the serial number of the item is not known.

A.1.36 element Repair/ComponentInstance

type	c:ItemInstanceReference
properties	**content** complex
children	c:InstanceDocumentReference c:Definition
annotation	**documentation** This element identifies the specific item repaired when the serial number of the item is known.

A.1.37 element Repair/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.38 element Repair/Procedure

type	c:Document					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:URL c:Text c:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	uuid	c:Uuid	required			**documentation** The universal unique identifier for the document.
	name	c:NonBlankString	required			**documentation** A descriptive or common name for the document.
	controlNumber	c:NonBlankString	optional			**documentation** A unique identifier for the document.
	version	c:NonBlankString	optional			**documentation** The version identification of the document.
annotation	**documentation** This element shall be used to identify the document containing the procedure required to perform a repair.					

A.1.39 element Repair/ReferenceDesignator

type	tr:ReferenceDesignator					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	tr:Description tr:FailureModes tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	classLetterAndNumber	c:NonBlankString	required			**documentation** The IEEE Std 315 subclause 22.4, Class Designation Letter, and a numerical number, for the reference designator (e.g., R1 for Resistor component number 1, A1 for SRA number 1).
	type	c:NonBlankString	required			**documentation** Defines the type of the part or component being referenced (e.g., WRA, SRA, Component, etc.).
annotation	**documentation** This element identifies the subcomponent of the UUT that was associated with the repair.					

A.1.40 element Repair/RepairActionTaken

type	tr:RepairAction					
properties	**content** complex					
children	tr:Description tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	RepairCode	required			**documentation** The type of repair action that was taken.
	code	c:NonBlankString	optional			**documentation** The application specific code for the value attribute.
annotation	**documentation** This element identifies the code and description used to describe the type of repair action taken.					

A.1.41 complexType RepairAction

children	tr:Description tr:Extension
used by	**Elements** Indictments/Indictment/RepairActionRecommended Repair/RepairActionTaken
attributes	**Name** **Type** **Use** **Default** **Fixed** **Annotation** value RepairCode required **documentation** The type of repair action that was taken. code c:NonBlankString optional **documentation** The application specific code for the value attribute.
annotation	**documentation** This element shall be used as the base type of any element providing information on the work performed to restore a part to an operational state.

A.1.42 element RepairAction/Description

type	c:NonBlankString
properties	**minOcc** 0 **maxOcc** 1 **content** simple
facets	**Kind** **Value** **Annotation** **minLength** 1 **whiteSpace** replace
annotation	**documentation** Text providing additional descriptive data for the value and code attributes.

A.1.43 element RepairAction/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.44 complexType SessionAction

type	extension of tr:Action					
properties	base tr:Action					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:ActionOutcome					
used by	**element** TestGroup/SessionAction					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.
annotation	**documentation** This complex type shall be used to record any action other than a test occurring within the context of a TestResults session, having a specified ActionOutcome.					

A.1.45 element SessionAction/ActionOutcome

type	tr:SessionActionOutcome					
properties	**content** complex					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	tr:SessionActionOutcomeValue	required			**documentation** Shall indicate either “Done”, “Aborted”. “NotStarted”, “UserDefined”, or Unknown”
	qualifier	c:NonBlankString	optional			**documentation** A user-friendly textual enhancement of the value attribute string.
	referenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this outcome, or the ID of a Test Description outcome.
	forced	xs:boolean	optional			**documentation** A True/False indication as to whether or not the outcome was overwritten to achieve a desired state.
annotation	**documentation** This element shall be used to identify the completion status of a given action.					

A.1.46 complexType SessionActionOutcome

used by	**Element** SessionAction/ActionOutcome					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	tr:SessionActionOutcomeValue	required			**documentation** Shall indicate either “Done”, “Aborted”. “NotStarted”, “UserDefined”, or “Unknown”
	qualifier	c:NonBlankString	optional			**documentation** A user-friendly textual enhancement of the value attribute string.
	referenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this outcome, or the ID of a Test Description outcome.
	forced	xs:boolean	optional			**documentation** A True/False indication as to whether or not the outcome was overwritten to achieve a desired state.
annotation	**documentation** This complex type shall be used to record the outcome of all actions that took place during the session. These include actions that are not directly part of the test executed, such as human activity. If an ATML Test Description instance document is identified by the /TestResults/TestDescription element, then the value of the ‘referenceID’ attribute shall be identical to the value of the ‘ID’ attribute of the Test Description Outcome. Otherwise, this referenceID attribute shall reference the test program entity that generated this outcome, for example: line number, test number, step name, etc.					

A.1.47 complexType Test

type	extension of tr:Action					
properties	**base** tr:Action					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:Outcome tr:Calibration tr:TestLimits tr:TestResult tr:Extension					
used by	**element** TestGroup/Test **complexType** TestGroup					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.
	entryPoint	xs:boolean	optional			**documentation** A True/False indication as to whether or not the test represents a test entry point.
	operatingMode	c:NonBlankString	optional			**documentation** A user-defined value describing the operating mode of the system when generating the specific instance of the test.

	classified	xs:boolean	optional	**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional	**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
annotation	**documentation** This complex type shall be used to identify all relevant information related to a single test. This includes input parameters, system or operator events, calibration values and an outcome.			

A.1.48 element Test/Calibration

type	tr:Parameter					
properties	**minOcc** 0 **maxOcc** unbounded **content** complex					
children	tr:Description tr:Data tr:Reference tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Parameter. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Parameter.
	timeStamp	xs:dateTime	optional			**documentation** The date and time associated with the Parameter. This shall be used where the time of application of a parameter is significant within the overall context of a test.
annotation	**documentation** This element shall be used to identify operational calibration information when it is necessary to record calibration values for the system associated with the execution of the test.					

A.1.49 element SessionAction/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.50 element Test/Outcome

type	tr:Outcome					
properties	**content** complex					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	tr:OutcomeValue	required			**documentation** Shall contain one of the following enumerations: “Passed” shall indicate the results of a test were within specified limits. “Failed” shall indicate that the results of a test were not within specified limits. “Aborted” shall indicate that a test did not complete. “NotStarted” shall indicate that the test did not start. “UserDefined” shall indicate that the test outcome as been defined to be something other than one of the five enumerations available. “Unknown” shall indicate that the result of the test is not known.
	qualifier	c:NonBlankString	optional			**documentation** Additional descriptive data for the ‘value’ attribute. For example, (…value=“Failed” qualifier=“High” …). In the case of (…value=“Aborted” …), qualifier shall provide essential descriptive or explanatory information regarding the reason for the test not completing normally.
	referenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this outcome, or the ID of a Test Description outcome.
	forced	xs:boolean	optional			**documentation** A True/False indication as to whether or not the recorded outcome is a user override of the observed outcome.
annotation	**documentation** This element shall be used to identify the discrete result of the subject test (i.e., Passed, Failed, Aborted, Not Started, User Defined, or Unknown).					

A.1.51 element Test/TestLimits

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Limits
annotation	**documentation** This element shall be used to identify comparison limits that apply to all test and/or test groups. If more than one Limits child element exists, then the logical expression that combines all limits shall be constructed by taking the first limit (without its operator) and then appending subsequent limits in the order in which they appear in the instance document, each prefixed by its operator. When an instance document contains (1) this element and (2) TestLimits children of any TestResult sibling of this element, then the limits of this element shall apply; the limits defined under TestResult siblings are irrelevant. When an instance document contains (1) a Limits child of a TestGroup element, and (2) Limits descendants of any Test belonging to the TestGroup, then the limits defined for the TestGroup element shall apply; the limits defined by descendants of Tests (Test/TestLimits or Test/TestResult/TestLimits) are irrelevant.

A.1.52 element Test/TestLimits/Limits

type	c:Limit					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	c:Expected c:SingleLimit c:LimitPair c:Mask c:Description c:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	operator	c:LogicalOperator	optional			**documentation** The comparison with the two boundary limits may be for a value between the limits or outside the limits. The LogicalOperator AND explicitly indicates a between comparison; OR explicitly indicates an outside comparison. Example: GT 3 AND LT 7 (between) vs. GT 10 OR LT 3 or GT 5 OR GT 10 (outside). While the logical operator may be inferred from the combination of limit values and comparison types, the c:LogicalOperator attribute permits better definition and less possibility for misinterpretation.
	name	c:NonBlankString	optional			**documentation** A descriptive or common name for the limit expressed in the element.
annotation	**documentation** This element shall be used to identify the limits against which test data (tr:TestData) is compared to in order to arrive at a test outcome (tr:testOutcome).					

A.1.53 element Test/TestResult

type	tr:TestResult
properties	**minOcc** 0 **maxOcc** unbounded **content** complex
children	tr:Outcome tr:Description tr:Indictments tr:TestData tr:TestLimits tr:Extension
attributes	**Name** **Type** **Use** **Default** **Fixed** **Annotation** ID xs:ID required **documentation** An identifying name for the TestResult. The value of "ID" shall be unique within the context of the containing XML instance document. name c:NonBlankString optional **documentation** A user-friendly textual name for the TestResult.
annotation	**documentation** This element shall be used for the recording of results and other information related to a single occurrence of a test.

A.1.54 complexType TestGroup

type	extension of tr:Test					
properties	**base** tr:Test					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:Outcome tr:Calibration tr:TestLimits tr:TestResult tr:Extension tr:Test tr:TestGroup tr:SessionAction					
used by	**elements** TestResults/ResultSet TestGroup/TestGroup					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.
	entryPoint	xs:boolean	optional			**documentation** A True/False indication as to whether or not the test represents a test entry point.

	operatingMode	c:NonBlankString	optional	**documentation** A user-defined value describing the operating mode of the system when generating the specific instance of the test.
	classified	xs:boolean	optional	**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional	**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
	callerName	c:NonBlankString	optional	**documentation** This attribute is a place to record which step called this test group in a Test Results document.
annotation	**documentation** This complex type shall be used to identify a grouping of related tests, test groups, or session actions. TestGroup provides a hierarchical structure for the aggregation of test results data within an Test Results XML instance document. TestGroup is recursive; that is, a TestGroup optionally contains subordinate TestGroup elements. The TestGroup structure shall be used to contain a collection of multiple iterations of a single test, or a related set of tests that the user desires to be reported or captured as a unit. When multiple Test, TestGroup, or SessionAction elements appear, the order of appearance of these elements should correspond to the time sequence order in which the test(s) or action(s) occurred. When the optional Outcome element appears, it shall represent a summary outcome of all subordinate TestGroup elements. Each subordinate TestGroup element may have a separate and distinct Outcome.			

A.1.55 element TestGroup/SessionAction

type	tr:SessionAction					
properties	**content** complex					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:ActionOutcome tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.
annotation	**documentation** This element shall be used to identify the actions that occurred during a test session that are not actual test results.					

A.1.56 element TestGroup/Test

type	tr:Test					
properties	**content** complex					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:Outcome tr:Calibration tr:TestLimits tr:TestResult tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.

	entryPoint	xs:boolean	optional	**documentation** "1" A True/False indication as to whether or not the test represents a test entry point.
	operatingMode	c:NonBlankString	optional	**documentation** A user-defined value describing the operating mode of the system when generating the specific instance of the test.
	classified	xs:boolean	optional	**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional	**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
annotation	**documentation** This element shall be used to identify a single test.			

A.1.57 element TestGroup/TestGroup

type	tr:TestGroup					
properties	**content** complex					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:Outcome tr:Calibration tr:TestLimits tr:TestResult tr:Extension tr:Test tr:TestGroup tr:SessionAction					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.

	entryPoint	xs:boolean	optional	**documentation** "1" A True/False indication as to whether or not the test represents a test entry point.
	operatingMode	c:NonBlankString	optional	**documentation** A user-defined value describing the operating mode of the system when generating the specific instance of the test.
	classified	xs:boolean	optional	**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional	**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
	callerName	c:NonBlankString	optional	**documentation** This attribute is a place to record which step called this test group in a Test Results document.
annotation	**documentation** This element shall be used to identify a related set of single tests.			

A.1.58 complexType TestResult

children	tr:Outcome tr:Description tr:Indictments tr:TestData tr:TestLimits tr:Extension					
used by	**element** Test/TestResult					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	xs:ID	required			**documentation** An identifying name for the TestResult. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the TestResult.
annotation	**documentation** This complex type shall be used to capture test result data, comparison limits, and indicted components.					

A.1.59 element TestResult/Description

type	c:NonBlankString
properties	**minOcc** 0 **maxOcc** 1 **content** simple
facets	**Kind** **Value** **Annotation** minLength 1 whiteSpace replace
annotation	**documentation** This element shall be used to identify any descriptive information for the TestResult.

A.1.60 element TestResult/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.61 element TestResult/Indictments

type	tr:Indictments					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	tr:Indictment tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	retestTestGroup	c:NonBlankString	optional			**documentation** Identifies a group of tests to be run to verify the indicted components have been repaired.
	indictmentsDateTime	xs:dateTime	optional			**documentation** The date and time that the indictments were made.
annotation	**documentation** This element shall be used to identify the group of UUT subcomponents that are the most likely cause of the test failure indicated by the given test result. The final indictment collection in a XML TestResult instance document is the union of all indictments at all levels.					

A.1.62 element TestResult/Outcome

type	tr:Outcome					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	value	tr:OutcomeValue	required			**documentation** Shall contain one of the following enumerations: "Passed" shall indicate the results of a test were within specified limits. "Failed" shall indicate that the results of a test were not within specified limits. "Aborted" shall indicate that a test did not complete. "NotStarted" shall indicate that the test did not start. "UserDefined" shall indicate that the test outcome as been defined to be something other than one of the five enumerations available. "Unknown" shall indicate that the result of the test is not known.
	qualifier	c:NonBlankString	optional			**documentation** Additional descriptive data for the 'value' attribute. For example, (…value="Failed" qualifier="High" …). In the case of (…value="Aborted" …), qualifier shall provide essential descriptive or explanatory information regarding the reason for the test not completing normally.
	referenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this outcome, or the ID of a Test Description outcome.
	forced	xs:boolean	optional			**documentation** A True/False indication as to whether or not the recorded outcome is a user override of the observed outcome.
annotation	**documentation** This element shall be used to identify the qualitative result of evaluating collected test data against quantitative limits.					

A.1.63 element TestResult/TestData

type	extension of c:Value					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:Datum c:Collection c:IndexedArray					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	acquisitionTimeStamp	xs:dateTime	optional			**documentation** The date and time associated with when the test data was acquired from the test equipment.
annotation	**documentation** This element shall be used to capture data from the test equipment. This may be post-processed data.					

A.1.64 element TestResult/TestLimits

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Limits
annotation	**documentation** This element shall be used to identify the limits against which test data is compared in order to arrive at a test outcome. If more than one “Limits” child element exists, then the logical expression that combines all limits shall be constructed by taking the first limit (without its operator) and then appending subsequent limits in the order in which they appear in the instance document, each prefixed by its operator.

A.1.65 element TestResult/TestLimits/Limits

type	c:Limit					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	c:Expected c:SingleLimit c:LimitPair c:Mask c:Description c:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	operator	c:LogicalOperator	optional			**documentation** The comparison with the two boundary limits may be for a value between the limits or outside the limits. The LogicalOperator AND explicitly indicates a between comparison; OR explicitly indicates an outside comparison. Example: GT 3 AND LT 7 (between) vs. GT 10 OR LT 3 or GT 5 OR GT 10 (outside). While the logical operator may be inferred from the combination of limit values and comparison types, the c:LogicalOperator attribute permits better definition and less possibility for misinterpretation.
	name	c:NonBlankString	optional			**documentation** A descriptive or common name for the limit expressed in the element.
annotation	**documentation** This element shall be used to identify the limits against which TestData is compared to arrive at an outcome. Within an XML instance document, these limit values shall only apply to the TestResults branch containing this element.					

A.1.66 complexType TestResults

children	tr:Personnel tr:PreTestRepairs tr:References tr:ResultSet tr:Site tr:TestDescription tr:TestProgram tr:TestStation tr:UUT tr:WorkOrder tr:Extension
used by	**element** TestResults

attributes

Name	Type	Use	Default	Fixed	Annotation
uuid	c:Uuid	required			**documentation** A universal unique identifier for the element containing this attribute.
classified	xs:boolean	optional			**documentation** An indication that the element is or is not classified.
securityClassification	c:NonBlankString	optional			**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
name	c:NonBlankString	optional			**documentation** The name of the instance document. Example: Acme Widget Test Results.

annotation	**documentation** This complex type shall be used to capture all information entities utilized in the collection of the results of testing a UUT on (or within) a particular test station, executing a particular set of tests.

A.1.67 element TestResults/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.68 element TestResults/Personnel

properties	**content** complex
children	tr:CustomerRepresentative tr:QualityAssurance tr:SystemOperator tr:Extension
annotation	**documentation** The Personnel element is a collector for CustomerRepresentative, QualityAssurance, and SystemOperator. These child elements shall be used in any use case requiring the identification of these individuals in the Test Results instance document. If the Personnel element appears in an instance document, at least one child element must appear.

A.1.69 element TestResults/Personnel/CustomerRepresentative

type	c:Person					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:OtherData c:Address					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A user-defined string uniquely identifying the contact.
	name	c:NonBlankString	optional			**documentation** A descriptive or common name for the operator.
	affiliation	c:NonBlankString	optional			**documentation** The organization the person represents.
	email	c:NonBlankString	optional			**documentation** The persons e-mail address.
	phoneNumber	c:NonBlankString	optional			**documentation** The persons telephone number.
annotation	**documentation** This element shall be used to identify the customer representative whom is witnessing testing or will be signing off on the test results.					

A.1.70 element TestResults/Personnel/Extension

type	Extension
properties	**minOcc** 0 **maxOcc** 1 **content** complex
annotation	**documentation** This element shall provide a specific extension point for use cases that require elements not provided in the basic structure.

A.1.71 element TestResults/Personnel/QualityAssurance

type	c:Person					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:OtherData c:Address					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A user-defined string uniquely identifying the contact.
	name	c:NonBlankString	optional			**documentation** A descriptive or common name for the operator.
	affiliation	c:NonBlankString	optional			**documentation** The organization the person represents.
	email	c:NonBlankString	optional			**documentation** The persons e-mail address.
	phoneNumber	c:NonBlankString	optional			**documentation** The persons telephone number.
annotation	**documentation** This element shall be used to identify the quality assurance representative whom will be signing off on the test results.					

A.1.72 element TestResults/Personnel/SystemOperator

type	c:Person					
properties	**content** complex					
children	c:OtherData c:Address					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A user-defined string uniquely identifying the contact.
	name	c:NonBlankString	optional			**documentation** A descriptive or common name for the operator.
	affiliation	c:NonBlankString	optional			**documentation** The organization the person represents.
	email	c:NonBlankString	optional			**documentation** The persons e-mail address.
	phoneNumber	c:NonBlankString	optional			**documentation** The persons telephone number.
annotation	**documentation** This element shall be used to identify the operator performing the testing of the UUT.					

A.1.73 element TestResults/PreTestRepairs

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Repair tr:MaintenanceActionInformationDocumentReference
annotation	**documentation** This element shall be used to identify any repairs made to the UUT prior to performing test(s) on the UUT and is used in the diagnostic process.

A.1.74 element TestResults/PreTestRepairs/MaintenanceActionInformationDocumentReference

type	c:DocumentReference					
properties	**content** complex					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A user-defined string uniquely identifying the document.
	uuid	c:Uuid	required			**documentation** The universal unique identifier for the document.

A.1.75 element TestResults/PreTestRepairs/Repair

type	tr:Repair					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	tr:RepairActionTaken tr:ReferenceDesignator tr:ComponentDescription tr:ComponentInstance tr:Procedure tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	preventive	xs:boolean	optional			**documentation** A True/False indication as to whether or not the repair was performed as part of preventative maintenance.
annotation	**documentation** This element shall be used to identify a particular repair made to the UUT prior to the performance of test(s).					

A.1.76 element TestResults/References

properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	tr:Reference
annotation	**documentation** This element shall be used to identify any external reference document relevant to a particular test, test group, or complete test program as defined or required by the user.

A.1.77 element TestResults/References/Reference

type	extension of c:Document					
properties	**minOcc** 1 **maxOcc** unbounded **content** complex					
children	c:URL c:Text c:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	uuid	c:Uuid	required			**documentation** The universal unique identifier for the document.
	name	c:NonBlankString	required			**documentation** A descriptive or common name for the document.
	controlNumber	c:NonBlankString	optional			**documentation** A unique identifier for the document.
	version	c:NonBlankString	optional			**documentation** The version identification of the document.
	type	c:NonBlankString	optional			**documentation** This attribute provides a way to define what the document is that is being referenced.
annotation	**documentation** This element shall be used to identify a particular externally referenced item.					

A.1.78 element TestResults/ResultSet

type	tr:TestGroup					
properties	**content** complex					
children	tr:Description tr:Events tr:Parameters tr:Data tr:EnvironmentalData tr:Extension tr:Outcome tr:Calibration tr:TestLimits tr:TestResult tr:Extension tr:Test tr:TestGroup tr:SessionAction					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	ID	c:NonBlankString	required			**documentation** A descriptive or common name for the Action. The value of "ID" shall be unique within the context of the containing XML instance document.
	name	c:NonBlankString	optional			**documentation** A user-friendly textual name for the Action.
	userDefinedType	c:NonBlankString	optional			**documentation** A user defined textual "type" that will describe the non-test action.
	cost	xs:double	optional			**documentation** The monetary cost value associated with performing the action.
	simulated	xs:boolean	optional			**documentation** A True/False indication as to whether or not the Action occurred during a simulated test.
	startDateTime	xs:dateTime	required			**documentation** The beginning date and time of the action.
	endDateTime	xs:dateTime	optional			**documentation** The ending date and time of the action.
	testReferenceID	c:NonBlankString	optional			**documentation** A reference to the test program entity that generated this Action or the ID of a Test Description Test, Test Group, or Session Action.
	documentRequirementID	c:NonBlankString	optional			**documentation** When an ATML Test Description instance document does not exist, this attribute can be used to track that a given result in the report maps to a given requirement from a requirements document.

	entryPoint	xs:boolean	optional	**documentation** A True/False indication as to whether or not the test represents a test entry point.
	operatingMode	c:NonBlankString	optional	**documentation** A user-defined value describing the operating mode of the system when generating the specific instance of the test
	classified	xs:boolean	optional	**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional	**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
	callerName	c:NonBlankString	optional	**documentation** This attribute is a place to record which step called this test group in a Test Results document.
annotation	**documentation** This element shall be used to identify the quantitative and/or qualitative results, as well as non-test session data; collected during execution of a test or group of tests.			

A.1.79 element TestResults/Site

type	c:Organization					
properties	**minOcc** 0 **maxOcc** 1 **content** complex					
children	c:Address c:Contacts c:FaxNumber c:URL c:WorkCenter					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	name	c:NonBlankString	required			**documentation** A descriptive or common name for the manufacturer.
	cageCode	c:NonBlankString	optional			**documentation** The CAGE code for the company indicated by the name attribute.
annotation	**documentation** This element shall be used to identify the physical location at which the test results were collected.					

A.1.80 element TestResults/TestDescription

type	c:ItemDescriptionReference
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:DescriptionDocumentReference c:Definition
annotation	**documentation** This element shall be used to identify an instance document conforming to IEEE Std 1671.1, which describes the tests that were performed to generate the test results collected in the current Test Results document.

A.1.81 element TestResults/TestProgram

type	SoftwareInstance
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:DescriptionDocumentReference c:Definition c:SerialNumber c:ReleaseDate IssueDate Warranty
annotation	**documentation** This element shall be used to identify the specific test program which was used to generate the results data in the TestResults XML instance document.

A.1.82 element TestResults/TestStation

type	HardwareInstance
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:DescriptionDocumentReference c:Definition c:SerialNumber c:ManufactureDate c:Calibration c:Components c:ParentComponent c:PowerOn IssueDate Warranty
annotation	**documentation** The element is used to identify the specific test station or host system upon which the tests were conducted.

A.1.83 element TestResults/UUT

type	c:ItemInstance
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:DescriptionDocumentReference c:Definition c:SerialNumber
annotation	**documentation** This element shall be used to uniquely identify the UUT upon which the test(s) is/are performed.

A.1.84 element TestResults/WorkOrder

type	WorkOrder
properties	**minOcc** 0 **maxOcc** 1 **content** complex
children	c:WorkOrderNumber c:WorkItemNumber c:MaintenanceLevel c:Description c:Extension
annotation	**documentation** This element shall be used to identify information characterizing the authorization of the testing or maintenance of a system instance.

A.1.85 simpleType OutcomeValue

type	restriction of **xs:string**
properties	**base** xs:string
used by	**attribute** Outcome/@value
facets	**Kind** **Value** **Annotation** enumeration Passed enumeration Failed enumeration Aborted enumeration NotStarted enumeration UserDefined enumeration Unknown
annotation	**documentation** This shall be used as the base type for any XML schema attribute or element that represent permitted values for a test outcome. In use, “Passed” shall indicate the results of a test were within specified limits. “Failed” shall indicate that the results of a test were not within specified limits. “Aborted” shall indicate that a test did not complete. “NotStarted” shall indicate that the test did not start. “UserDefined” shall indicate that the test outcome as been defined to be something other than one of the five enumerations available. “Unknown” shall indicate that the result of the test is not known.

A.1.86 simpleType SessionActionOutcomeValue

type	restriction of **xs:string**
properties	**base** xs:string
used by	**attribute** SessionActionOutcome/@value
facets	**Kind** **Value** **Annotation** enumeration Done enumeration Aborted enumeration NotStarted enumeration UserDefined enumeration Unknown
annotation	**documentation** This shall be used as the base type for any XML schema attribute or element that represents permitted session action codes.

A.2 TestResultsCollection.xsd

attributeFormDefault: unqualified
elementFormDefault: qualified
targetNamespace: urn:IEEE-1636.1:2013:TestResultsCollection

A.2.1 element TestResultsCollection

properties	**content** complex
children	trc:TestResults trc:Extension
annotation	**documentation** The TestResultsCollection element shall be a container for a collection of test results instance documents.

A.2.2 element TestResultsCollection/TestResults

type	tr:TestResults					
properties	**content** complex					
children	tr:Personnel tr:PreTestRepairs tr:References tr:ResultSet tr:Site tr:TestDescription tr:TestProgram tr:TestStation tr:UUT tr:WorkOrder tr:Extension					
attributes	**Name**	**Type**	**Use**	**Default**	**Fixed**	**Annotation**
	uuid	c:Uuid	required			**documentation** A universal unique identifier for the element containing this attribute.
	classified	xs:boolean	optional			**documentation** An indication that the element is or is not classified.
	securityClassification	c:NonBlankString	optional			**documentation** A use-case determined string declaring the security classification level of the element containing this attribute and the subordinate branch of the XML document.
	name	c:NonBlankString	optional			**documentation** The name of the instance document. Example: Acme Widget Test Results.
annotation	**documentation** The TestResults element shall be a container for all information entities utilized in the collection of the results of testing a UUT on (or within) a particular test station, executing a particular set of tests.					

A.2.3 element TestResultsCollection/Extension

type	Extension
properties	**content** complex
annotation	**documentation** The Extension element shall provide a specific extension point for use cases that require elements not provided.

Annex B

(normative)

EXPRESS models

Should the reader not have a general understanding of EXPRESS, there are several overviews of EXPRESS available for reference. Annex B of IEEE Std 1232 [B5] and Schenk and Wilson [B11] are two available. These overviews will help with the understanding of the contents of the Test Results information model that this Annex is defining.

B.1 TEST_RESULTS_MODEL

The Test Results information model provides a semantic definition for information consisting of both quantitative (measured values) and qualitative (pass/fail determination) test results. The model design is such that ancillary information such as environmental conditions and system/operator messages may also be included in an instance document. This information, while not specifically "results," is intended to permit use of an instance document for a variety of purposes, including statistical analysis and diagnostics. Some examples of this ancillary information includes identifying information for the UUT, the test station, and the test program; ambient environmental conditions at the time of the test; test equipment calibration data; test program input data and ancillary textual comments. The model defines a hierarchical structure for results data to permit the grouping of a series of related test results in a single instance document.

EXPRESS specification:

```
*)
  SCHEMA TEST_RESULTS_MODEL;
    USE FROM SIMICA_COMMON_MODEL_DOT_99
      (WorkOrder,
      NamedValue,
      DateTime,
      SystemInstance,
      HardwareInstance,
      SoftwareInstance,
      UUID,
      ReferenceDesignator,
      TypeDescription,
      Organization,
      Document,
      Contact,
      Limit,
      ItemDescription,
      CommonValue,
      ActionOutcome,
      TestOutcome,
      RepairAction,
      ItemInstanceReference,
      ItemDesignReference,
      ClassifiedAttributes);
    REFERENCE FROM SIMICA_MAI_MODEL
      (MaintenanceActionInformationDocument);
(*
```

B.1.1 CostType

Defines a data element for specifying the classification or organization of cost elements associated to some activity.

EXPRESS specification:

```
*)
  TYPE CostType = Double;
  END_TYPE;
(*
```

B.1.2 DateTimeType

Type “dateTimeType” defines a data element that conforms to an XML date-time schema.

EXPRESS specification:

```
*)
  TYPE DateTimeType = STRING;
  END_TYPE;
(*
```

B.1.3 DescriptionType

Type “descriptionType” defines a data element for providing a text-based description of some entity.

EXPRESS specification:

```
*)
  TYPE DescriptionType = STRING;
  END_TYPE;
(*
```

B.1.4 Double

Defines a data element corresponding to the XML “double” data type. It is assumed to be a floating point type with double precision (usually requiring 32 bits).

EXPRESS specification:

```
*)
  TYPE Double = BINARY (32) FIXED;
  END_TYPE;
(*
```

B.1.5 Identifier

Type “identifier” defines a data element that represents a string-based ID for some entity.

EXPRESS specification:

```
*)
  TYPE Identifier = STRING;
  END_TYPE;
(*
```

B.1.6 MessageType

Defines a string data element for associating an arbitrary text comment (or message) with a document or portion of a document.

EXPRESS specification:

```
*)
  TYPE MessageType = STRING;
  END_TYPE;
(*
```

B.1.7 ModeType

Defines a data element for representing various operating modes of a system or unit (such as the UUT or the test station).

EXPRESS specification:

```
*)
  TYPE ModeType = STRING;
  END_TYPE;
(*
```

B.1.8 NameType

Type “NameType” defines a data element used to name some entity.

EXPRESS specification:

```
*)
  TYPE NameType = STRING;
  END_TYPE;
(*
```

B.1.9 SourceType

User-defined text string identifying the source of the event or information (e.g., “operator” or “system”).

EXPRESS specification:

```
*)
  TYPE SourceType = STRING;
  END_TYPE;
(*
```

B.1.10 DesignOrInstance

Chooses between an item design reference or an item instance reference.

EXPRESS specification:

```
*)
  TYPE DesignOrInstance = SELECT
    (ItemInstanceReference,
    ItemDesignReference);
  END_TYPE;
(*
```

B.1.11 GroupMember

Corresponds to either a test, action, or another test group.

EXPRESS specification:

```
*)
  TYPE GroupMember = SELECT
    (TestGroup,
    SessionAction,
    Test);
  END_TYPE;
(*
```

B.1.12 RepairReference

Select Type “RepairReference” chooses between referring to an MAI document or referring to a RepairElement that was the object of a pre-test action.

EXPRESS specification:

```
*)
  TYPE RepairReference = SELECT
    (MaintenanceActionInformationDocument,
    MaintenanceActionInstance);
  END_TYPE;
(*
```

B.1.13 TestDescriptionSource

The TestDescriptionSource branch provides elements and attributes sufficient to uniquely identify an IEEE Std 1671.1 Test Description instance document for the test(s) contained in the TestResults instance document.

EXPRESS specification:

```
*)
  TYPE TestDescriptionSource = SELECT
    (Document,
    ItemDescription);
  END_TYPE;
(*
```

B.1.14 Action

Used to identify specific data related to any action performed, either during the execution of a test or as part of the repair process.

EXPRESS specification:

```
*)
  ENTITY Action
    ABSTRACT SUPERTYPE OF (ONEOF(SessionAction, Test));
    ID                                :Identifier;
    name                              :OPTIONAL NameType;
    userDefinedType                   :OPTIONAL TypeDescription;
    cost                              :OPTIONAL CostType;
    simulated                         :OPTIONAL BOOLEAN;
    startDateTime                     :DateTimeType;
    endDateTime                       :OPTIONAL DateTimeType;
    testReferenceID                   :OPTIONAL Identifier;
    description                       :DescriptionType;
    events                            :OPTIONAL SET [1:?] OF Event;
    actionData                        :OPTIONAL CommonValue;
    environmentalData                 :OPTIONAL Environmental;
    documentRequirementID             :OPTIONAL Identifier;
    inputParameter                    :OPTIONAL SET [1:?] OF Parameter;
  UNIQUE
    oneID                             :ID;
END_ENTITY;
(*
```

Attribute definitions:

ID : A descriptive or common name for the Action. The value of ID shall be unique within the context of the containing model.

name : A user-friendly textual name for an action.

userDefinedType : Provides a user-defined textual ‘type’ that will provide a categorization of the non-test action.

cost : The monetary cost value associated with performing the action.

simulated : Boolean that may be used to indicate that the action occurred during a simulated test.

startDateTime : Beginning time and date of the recorded action.

endDateTime : Ending time and date of the recorded action.

testReferenceID : The reference ID of the TestGroup, Test, or SessionAction that was executed to generate the test result.

description : A string element that shall be used to capture any descriptive information for the Action.

events : A collector element for session occurrences such as system or operator messages.

actionData : Permits the capture of data associated with the non-test action(s).

environmentalData : Identifies the environmental conditions which are related to a test.

documentRequirementID : This attribute is used to track and record the ID of the requirement that is met by a given test result in the test results document.

inputParameter : This element shall be used to record parameter data.

B.1.15 ComponentProcedure

A combination of repair procedure and information about the thing being repaired.

EXPRESS specification:

```
*)
  ENTITY ComponentProcedure;
    repairProcedure                 :OPTIONAL Document;
    repairedItem                    :DesignOrInstance;
  END_ENTITY;
(*
```

Attribute definitions:

repairProcedure : Identifies the procedure required to perform the repair.

repairedItem : Information on the item being worked on through the associated maintenance action.

B.1.16 Environmental

Entity “Environmental” is a collector for multiple sets of environmental data pertinent to a particular action.

EXPRESS specification:

```
*)
  ENTITY Environmental;
    environmentalValues              :SET [1:?] OF TimedDatum;
  END_ENTITY;
(*
```

Attribute definitions:

environmentalValues : Identifies the environmental data that is pertinent to a particular action.

B.1.17 Event

Provides a sequence of one or more child elements to capture non-result system messages and/or references to external entities.

EXPRESS specification:

```
*)
  ENTITY Event;
    ID                               :Identifier;
    name                             :OPTIONAL NameType;
    severity                         :OPTIONAL INTEGER;
    source                           :SourceType;
    timeStamp                        :OPTIONAL DateTimeType;
    message                          :OPTIONAL MessageType;
    namedData                        :OPTIONAL SET [1:?] OF NamedValue;
    eventReference                   :OPTIONAL SET [1:?] OF Document;
  UNIQUE
    oneID                            :ID;
  WHERE
    rangeLimit                       :(SELF.severity >= 0) AND
                                     (SELF.severity <= 4);
  END_ENTITY;
(*
```

Attribute definitions:

ID : A descriptive or common name for the event. The value of ID shall be unique within the context of the containing model.

name : A user-friendly textual name for the event.

severity : An enumeration of 0 to 4 inclusive that shall be used to indicate a severity level for the Event. A value of 0 shall indicate least severe.

source : Identifies the source of the event (e.g., operator or test system).

timestamp : Date and time of event occurrence.

message : Permits the recording of textual data relevant to the subject event.

namedData : Permits the recording of structured event data.

eventReference : Identifies references to external items, for example documents.

Formal propositions:

rangeLimit Specifies the valid range is 0 to 4.

B.1.18 Indictment

This element shall be used to identify the information for one subcomponent of the tested UUT that is a possible cause of a test failure. Multiple Indictment siblings shall be used to specify a set of UUT subcomponents of which one or more is the cause of the failure.

EXPRESS specification:

```
*)
  ENTITY Indictment;
    repairActionRecommended          :Action;
    indictedRefDes                   :ReferenceDesignator;
  END_ENTITY;
(*
```

Attribute definitions:

repairActionRecommended : Identifies the code and description used to describe the type of recommended repair.

indictedRefDes : A subcomponent of the UUT corresponding to one of the most likely causes of the test failure.

B.1.19 Indictments

Entity “Indictments” defines a collector of individual “Indictment” elements. A separate collector is defined to support associating attributes with the collector.

Note that multiple such elements may exist within an instance document, specifying multiple sets of UUT subcomponents. In this case, one or more subcomponents from each set is the cause of a failure. For example, the presence of one Indictments element specifying subcomponents A1 and A2 along with a second Indictments element specifying subcomponents A3 and A4 means “(A1 is defective OR A2 is defective) AND (A3 is defective OR A4 is defective)”.

EXPRESS specification:

```
*)
  ENTITY Indictments;
    retestTestGroup                  :OPTIONAL NameType;
    indictmentsDateTime              :OPTIONAL DateTime;
    callout                          :SET [1:?] OF Indictment;
  END_ENTITY;
(*
```

Attribute definitions:

retestTestGroup : Identifies a group of tests to be run to verify the indicted components have been repaired.

indictmentsDateTime : Provides a date/time stamp for when the indictments were made.

callout : The set of Indictment elements called out by the test procedure/test program.

B.1.20 MaintenanceActionInstance

Entity “MaintenanceActionInstance” shall be used to capture the type of maintenance action conducted.

EXPRESS specification:

```
*)
  ENTITY MaintenanceActionInstance;
    repairActionTaken               :RepairAction;
    testFailureCause                :OPTIONAL ReferenceDesignator;
    actionTarget                    :SET [1:?] OF ComponentProcedure;
  END_ENTITY;
(*
```

Attribute definitions:

repairActionTaken : Identifies the code and description used to describe the type of repair action taken.

testFailureCause : Identifies the subcomponent of the UUT that was associated with the repair.

actionTarget : The items maintained and the procedure used to maintain them.

B.1.21 Parameter

Provides a structure in which test parameters may be reported. Parameters are generally described as configuration or input values for a test.

EXPRESS specification:

```
*)
  ENTITY Parameter;
    ID                              :Identifier;
    name                            :OPTIONAL NameType;
    timeStamp                       :OPTIONAL DateTimeType;
    description                     :DescriptionType;
    parameterData                   :CommonValue;
    parameterReference              :OPTIONAL Document;
  UNIQUE
    oneID                           :ID;
  END_ENTITY;
(*
```

Attribute definitions:

ID : Descriptive or common name for the parameter. The value of ID shall be unique within the context of the containing model.

name : A user-friendly textual name for the parameter.

timestamp : Date and time associated with the captured Parameter. This attribute shall be used in cases where the time of application of a parameter is significant within the overall context of a Test.

description : Provides an unstructured, unrestricted textual description of the Parameter.

parameterData : Permits the recording of structured, restricted values of the Parameter.

parameterReference : Permits the recording of any referential data. A typical example would be a reference to a document or location where run-time test parameters may be stored. This would be useful in cases where such parameters are not permitted in the Test Results instance document due to security reasons.

B.1.22 Personnel

The Personnel element is a collector for CustomerRepresentative, QualityAssurance, and SystemOperator. These child elements shall be used in any use case requiring the identification of these individuals in the Test Results instance document. If the Personnel element appears in an instance document, at least one child element must appear.

EXPRESS specification:

```
*)
  ENTITY Personnel;
    systemOperator                  :OPTIONAL Contact;
    qualityAssuranceRepresentative  :OPTIONAL Contact;
    customerRepresentative          :OPTIONAL Contact;
  WHERE
    atLeastOne          :EXISTS(SELF.customerRepresentative) OR
                         EXISTS(SELF.qualityAssuranceRepresentative) OR
                         EXISTS(SELF.systemOperator);
  END_ENTITY;
(*
```

Attribute definitions:

systemOperator : Identifies the operator performing the testing of the UUT.

qualityAssuranceRepresentative : Identifies the quality assurance representative who will be signing off on the test results.

customerRepresentative : Identifies the customer representative who is witnessing testing or signing off on the test results.

Formal propositions:

atLeastOne Specifies that, if personnel exist, at least one of the types must exist as well.

B.1.23 SessionAction

Entity “SessionAction” is any action other than a test occurring within the context of a TestResults session, having a specified ActionOutcome.

EXPRESS specification:

```
*)
  ENTITY SessionAction
    SUBTYPE OF(Action);
    sessionActionOutcome            :OPTIONAL ActionOutcome;
  END_ENTITY;
(*
```

Attribute definitions:

sessionActionOutcome : Identifies the completion status of the session action.

B.1.24 Test

This entity provides for the capture of all relevant information related to a single test. This includes input parameters, system or operator events, calibration values, and an outcome.

EXPRESS specification:

```
*)
  ENTITY Test
    SUBTYPE OF(Action);
    testClassification              : ClassifiedAttributes;
    entryPoint                      : OPTIONAL BOOLEAN;
    operatingMode                   : OPTIONAL ModeType;
    hasOutcome                      : TestOutcome;
    calibrationFactor               : OPTIONAL SET [1:?] OF Parameter;
    testLimit                       : OPTIONAL SET [1:?] OF Limit;
    result                          : OPTIONAL SET [1:?] OF TestResult;
  WHERE
    resultExists                    :(EXISTS(SELF.result)) OR
                                    (EXISTS(SELF.hasOutcome));
  END_ENTITY;
(*
```

Attribute definitions:

testClassification : Associates a security classification with the test.

entryPoint : Boolean that may be used (as necessary) to indicate whether the Test being reported represents a test entry point.

operatingMode : User-defined value describing the operating mode of the ATS for the specific instance of a Test.

hasOutcome : The discrete result of the subject test.

calibrationFactor : This element shall be used to identify operational calibration information when it is necessary to record calibration values for the system associated with the execution of the test.

testLimit : The element shall be used to identify comparison limits that apply to all tests or test groups. When an instance document contains this element as well as child tests with additional limits, then the limits of the test element shall take precedence. In the event the test is part of a test group, then the limits on the test group shall take precedence.

result : This element shall be used for the recording of results and other information related to a single occurrence of a test.

Formal propositions:

resultExists Specifies that either an outcome exists, the set of results exists, or both. At least one must appear.

B.1.25 TestGroup

Entity “TestGroup” shall be used to identify a grouping of related tests, test groups, or session actions. TestGroup provides a hierarchical structure for the aggregation of test results data within an Test Results XML instance document. TestGroup is recursive; that is, a TestGroup optionally contains subordinate TestGroup elements. The TestGroup structure shall be used to contain a collection of multiple iterations of a single test, or a related set of tests that the user desires to be reported or captured as a unit. When multiple Test, TestGroup, or SessionAction elements appear, the order of appearance of these elements should correspond to the time sequence order in which the test(s) or action(s) occurred. When the optional Outcome element appears, it shall represent a summary outcome of all subordinate TestGroup elements. Each subordinate TestGroup element may have a separate and distinct Outcome.

EXPRESS specification:

```
*)
  ENTITY TestGroup
    SUBTYPE OF(Test);
    containsMember                  :SET [1:?] OF GroupMember;
    callerName                      :OPTIONAL NameType;
  END_ENTITY;
(*
```

Attribute definitions:

containsMember : Provides a set of tests, actions, or test groups.

callerName : This attribute is a place to record which step called this test group in a test results document.

B.1.26 TestProgram

The TestProgram entity provides for the unique identification of a software test program that was executed to generate the results data in the TestResult instance.

EXPRESS specification:

```
*)
  ENTITY TestProgram
    SUBTYPE OF(SoftwareInstance);
    configuration                   :OPTIONAL SET [1:?] OF NamedValue;
  END_ENTITY;
(*
```

Attribute definitions:

configuration : Attribute “configuration” is provided to record test program settings and other relevant data that is required to help explain the test results.

B.1.27 TestResult

Entity “TestResult” shall be used to capture test result data, comparison limits, and indicted components associated with a test.

EXPRESS specification:

```
*)
  ENTITY TestResult;
    ID                              :Identifier;
    name                            :OPTIONAL NameType;
    description                     :DescriptionType;
    testData                        :CommonValue;
    componentCallouts               :OPTIONAL Indictments;
    testLimits                      :OPTIONAL SET [1:?] OF Limit;
  UNIQUE
    oneID                           :ID;
  END_ENTITY;
(*
```

Attribute definitions:

ID : A descriptive or common name for the test result. The value of ID shall be unique within the context of the containing model.

name : Permits a more user-friendly textual name for the test result.

description : A string element that may be used to capture any descriptive information for the TestResult.

testData : This element shall be used to capture data from the host system. This data may be either raw or post-processed data. It is presumed that a transform will be provided for raw data.

componentCallouts : Identifies the group of UUT subcomponents that are the most likely cause of the test failure indicated by the given test result.

testLimits : Identifies the limits against which test data is compared in order to arrive at a test outcome.

B.1.28 TestResults

Entity “TestResults” is the root element of the schema and defines a container for all information entities used to collect test results on a unit under test within a particular operational context running a particular test program.

EXPRESS specification:

```
*)
  ENTITY TestResults;
    ID                                :UUID;
    testResultsClassification         :ClassifiedAttributes;
    name                              :OPTIONAL NameType;
    staff                             :Personnel;
    preTestRepair                     :OPTIONAL LIST [1:?] OF
                                                RepairReference;
    references                        :OPTIONAL SET [1:?] OF
                                                TypedDocument;
    resultSet                         :SET OF TestGroup;
    site                              :OPTIONAL Organization;
    testDescription                   :OPTIONAL TestDescriptionSource;
    activeTestProgram                 :OPTIONAL TestProgram;
    station                           :OPTIONAL TestStation;
    unitUnderTest                     :OPTIONAL SystemInstance;
    testAuthorization                 :OPTIONAL WorkOrder;
  UNIQUE
    oneID                             :ID;
  END_ENTITY;
(*
```

Attribute definitions:

ID : Provides the universally unique identifier for the entity containing this attribute.

testResultsClassification : Associates a security classification with the test results document should one exist.

name : The name of the instance document.

staff : This element shall be used to identify information regarding the personnel associated with performance of the testing.

preTestRepair : This element shall be used to identify any repairs made to the UUT prior to performing tests on the UUT and is used in the diagnostic process.

references : This element shall be used to identify any external reference document relevant to a particular test, test group, or complete test program as defined or required by the user.

resultSet : This element shall be used to identify the quantitative and/or qualitative results, as well as non-test session data collected during execution of a test or group of tests.

site : Attribute “site” indicates the location where the test results were collected.

testDescription : This element shall be used to identify an instance document conforming to IEEE Std 1671.1, which defines the tests that are directly associated the test results collected in the current Test Results document.

activeTestProgram : The element is used to identify the specific test program used to generate the results data in the test results instance document.

station : The element is used to identify the specific test station or host system upon which the tests were conducted.

unitUnderTest : Attribute “unitUnderTest” identifies the specific UUT upon which the tests were performed.

testAuthorization : Identifies the work order related or authorizing the testing of the UUT.

B.1.29 TestStation

This element shall be used to capture information describing or identifying the specific Test Station or equipment on which the subject test(s) was/were conducted.

EXPRESS specification:

```
*)
  ENTITY TestStation
    SUBTYPE OF(HardwareInstance);
  END_ENTITY;
(*
```

B.1.30 TimedDatum

Entity “TimedDatum” extends NamedValue to enable associating a time stamp with the data collected.

EXPRESS specification:

```
*)
  ENTITY TimedDatum
    SUBTYPE OF(NamedValue);
    timeStamp                        :OPTIONAL DateTime;
  END_ENTITY;
(*
```

Attribute definitions:

timeStamp : Date and time associated with the environmental data. This attribute should be used where the time the data was recorded is significant within the overall context of a particular action.

B.1.31 TypedDocument

This entity identifies a document of a specific type and gives the associated type of that document.

EXPRESS specification:

```
*)
  ENTITY TypedDocument
       SUBTYPE OF(Document);
      docType                                    :OPTIONAL NameType;
  END_ENTITY;
(*
```

Attribute definitions:

docType : A textual name for the type of reference, e.g., image file, text file, etc.

```
*)
END_SCHEMA;
(*
```

B.2 TestResults model EXPRESS-G diagrams

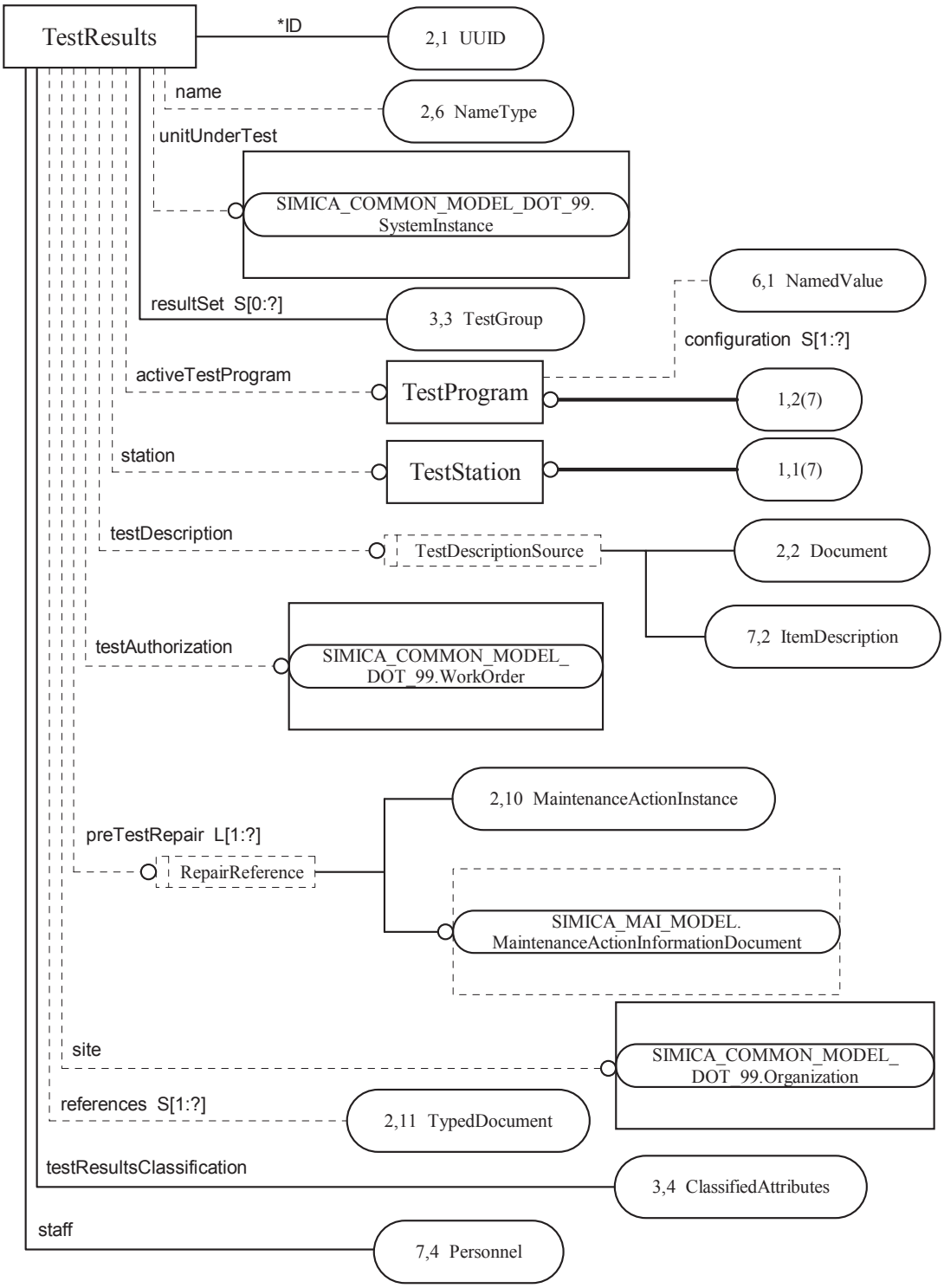

Figure B.1—TEST_RESULTS_MODEL EXPRESS-G, diagram 1 of 7

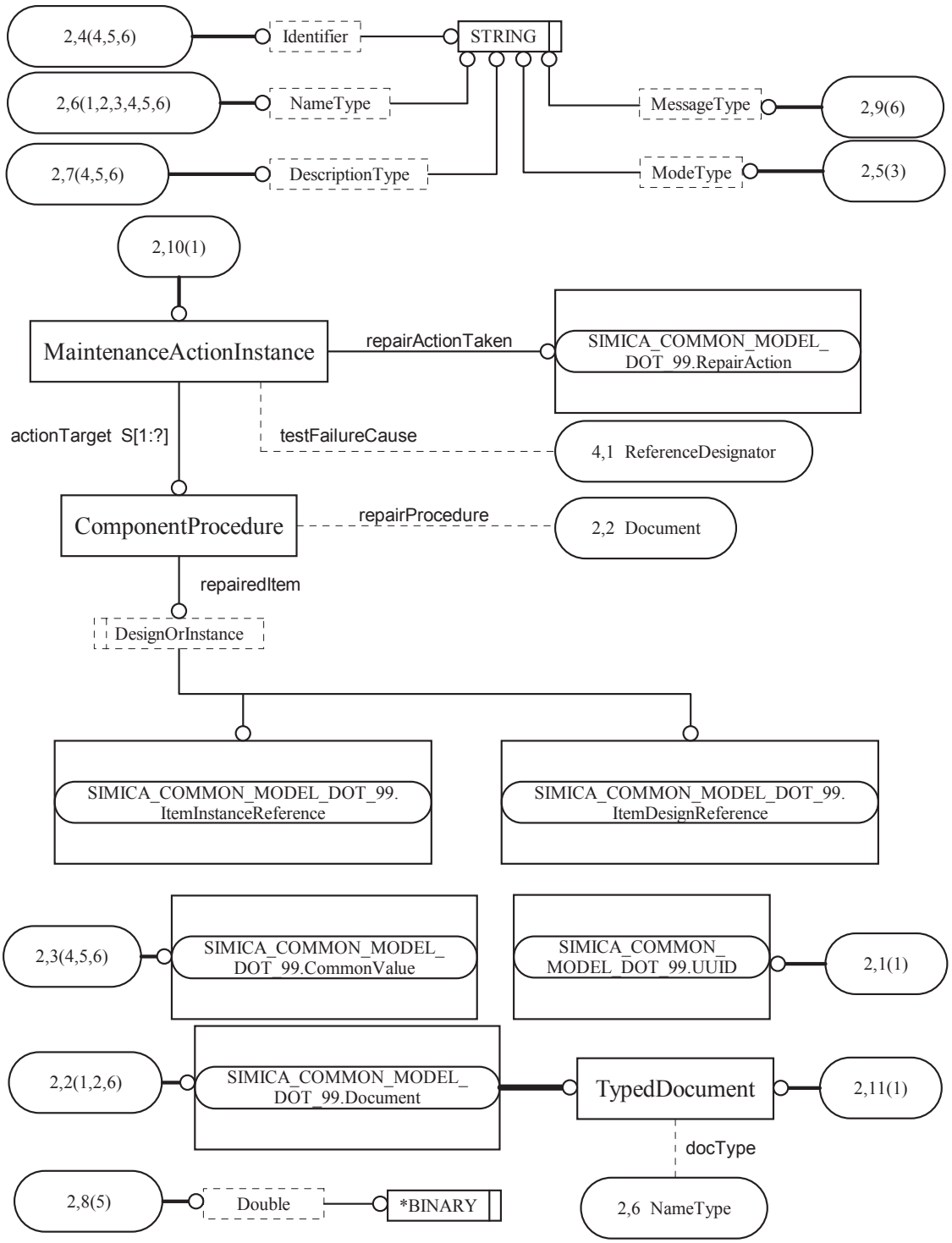

Figure B.2—TEST_RESULTS_MODEL EXPRESS-G, diagram 2 of 7

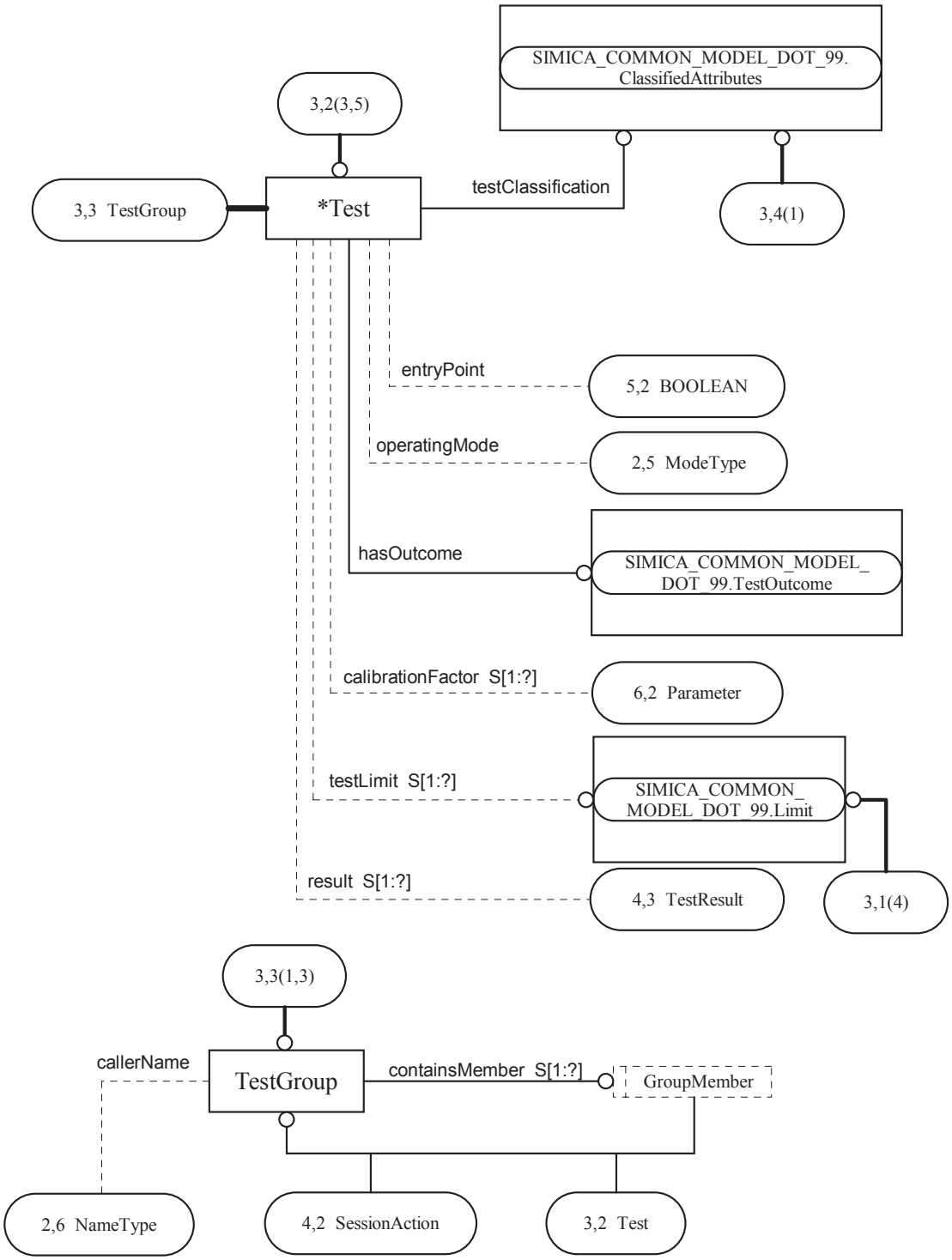

Figure B.3—TEST_RESULTS_MODEL EXPRESS-G, diagram 3 of 7

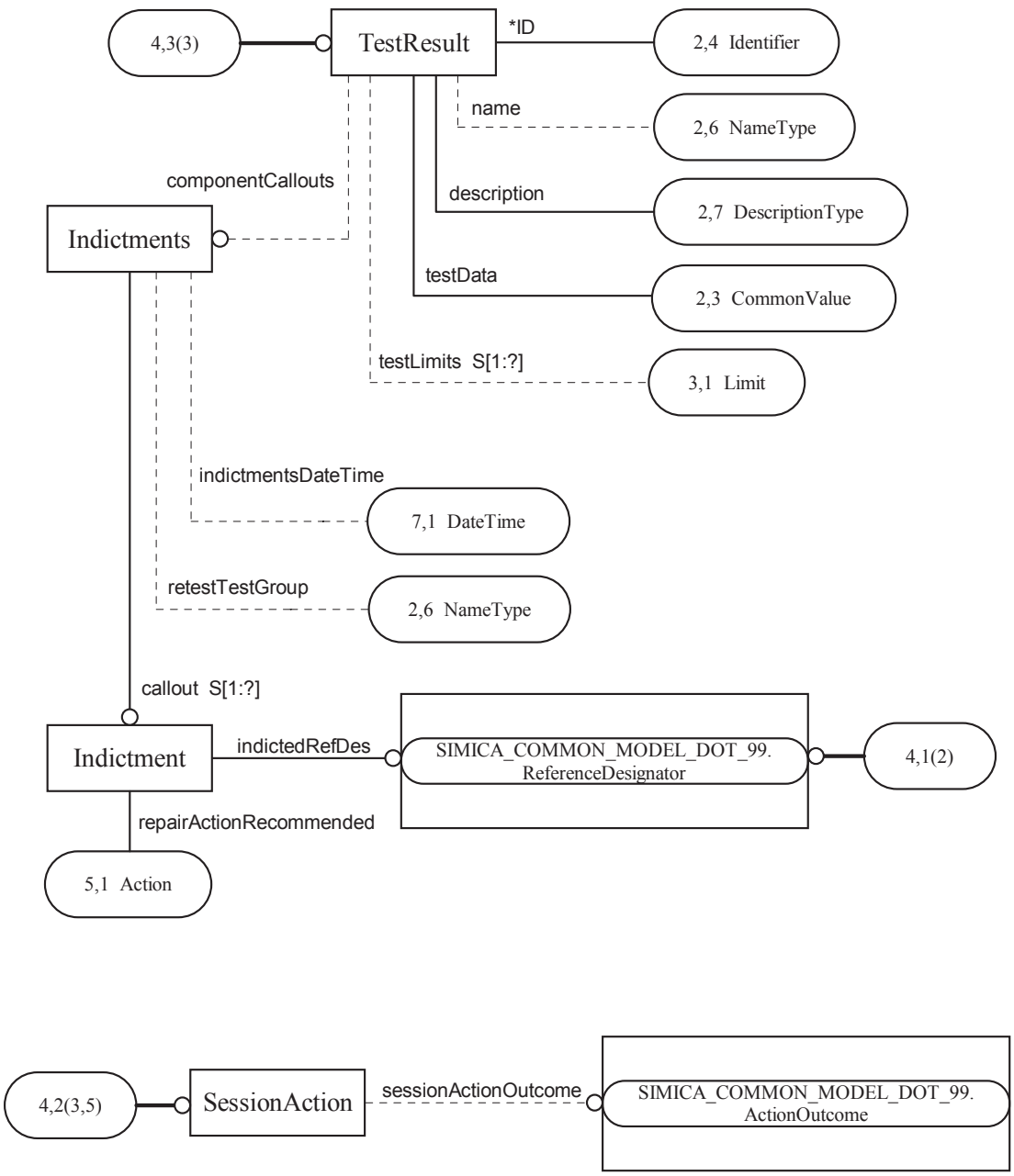

Figure B.4—TEST_RESULTS_MODEL EXPRESS-G, diagram 4 of 7

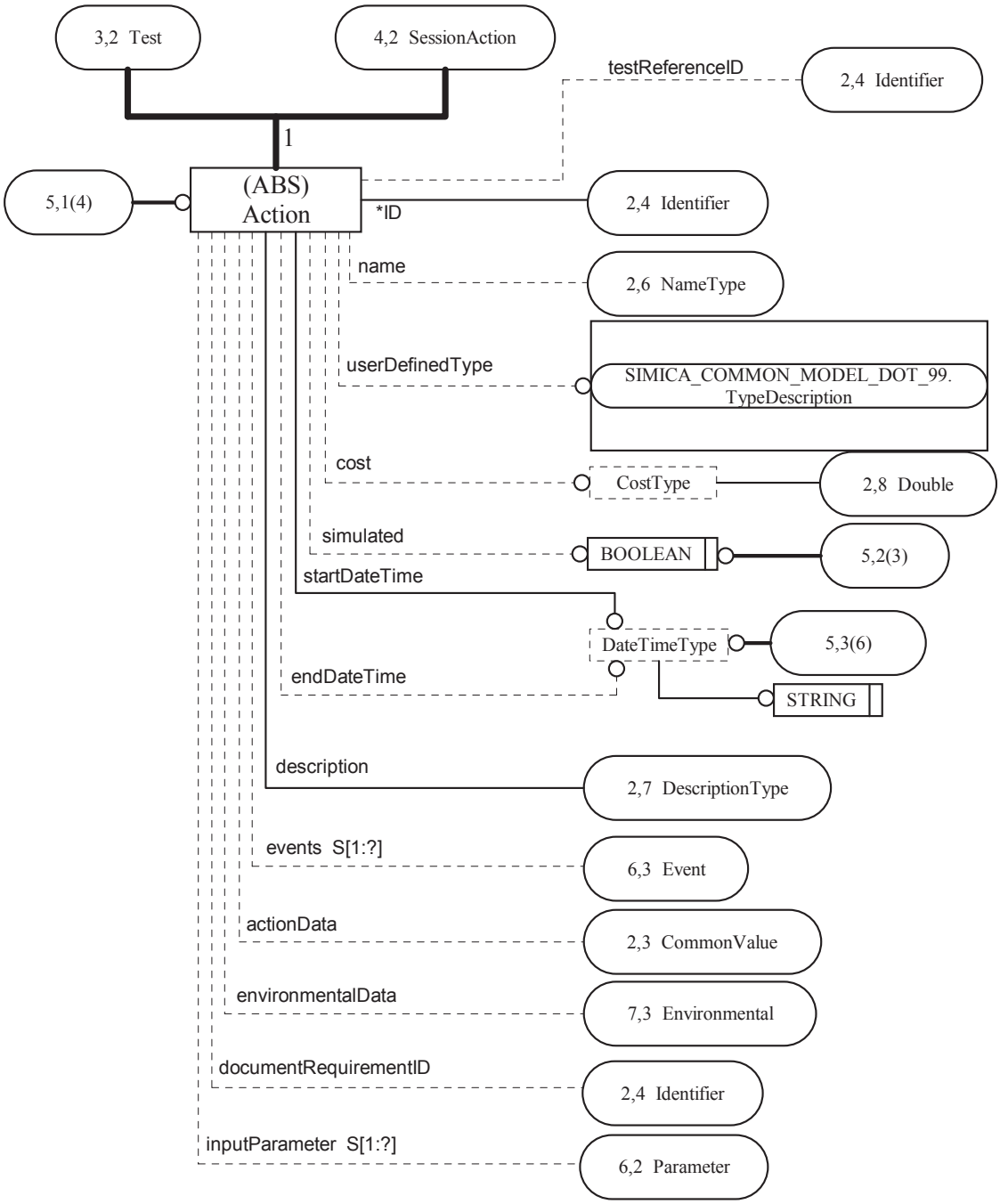

Figure B.5—TEST_RESULTS_MODEL EXPRESS-G, diagram 5 of 7

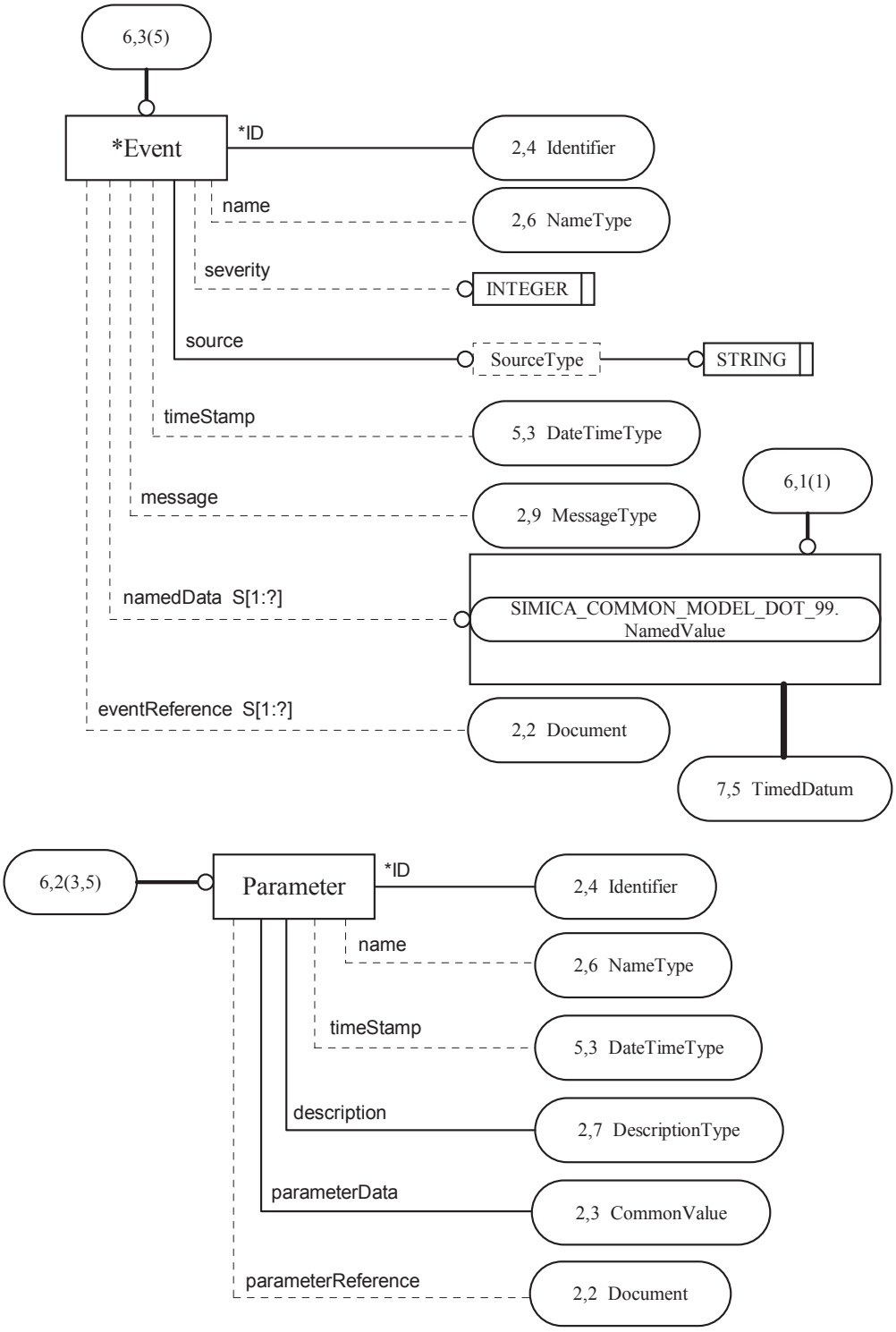

Figure B.6—TEST_RESULTS_MODEL EXPRESS-G, diagram 6 of 7

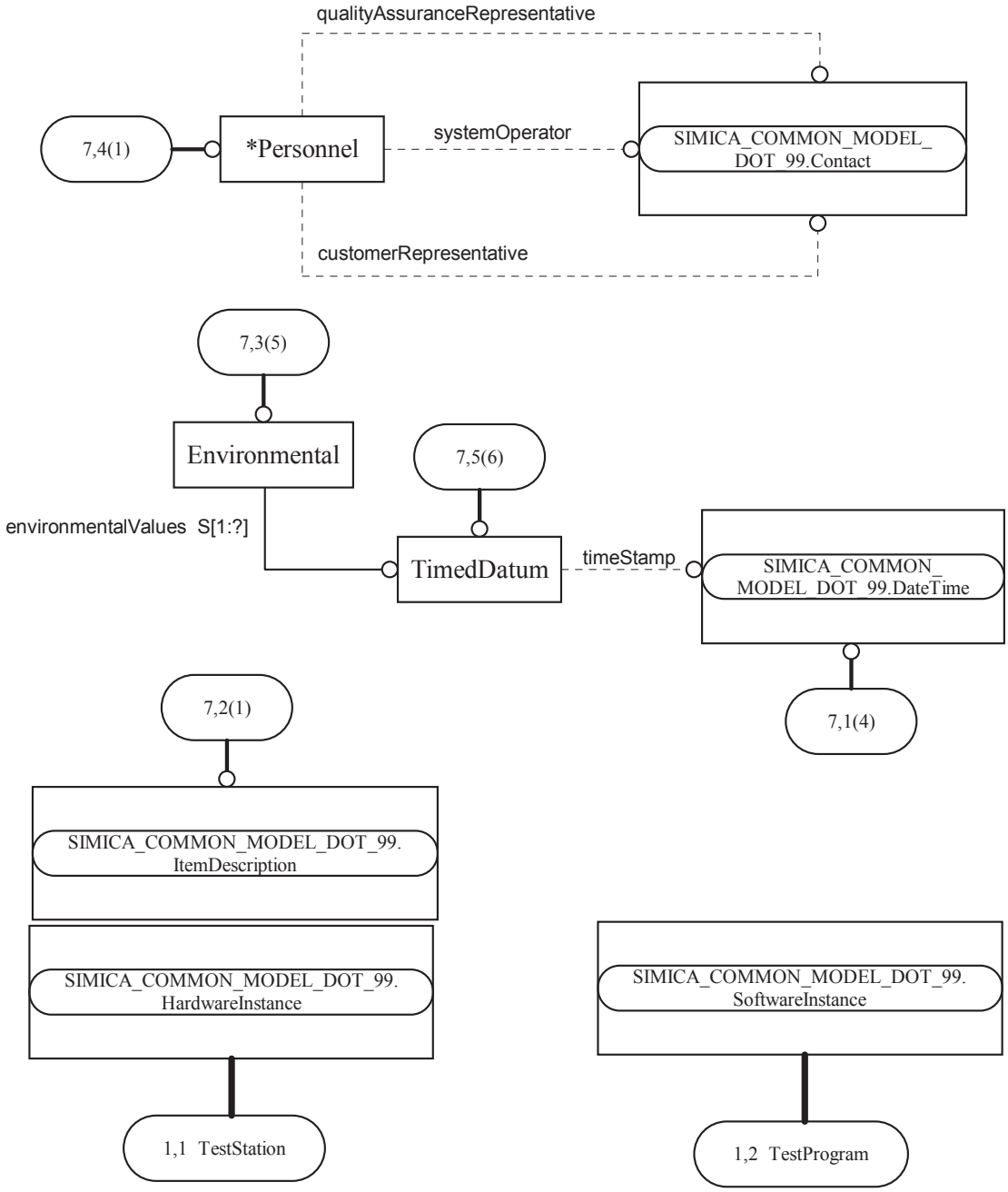

Figure B.7—TEST_RESULTS_MODEL EXPRESS-G, diagram 7 of 7

Annex C

(informative)

Bibliography

Bibliographical references are resources that provide additional or helpful material but do not need to be understood or used to implement this standard. Reference to these resources is made for informational use only.

[B1] Extensible Markup Language (XML) 1.0, 5th ed. W3C Proposed Edited Recommendation 05 February 2008.[8]

[B2] *IEEE Standards Dictionary Online*.[9]

[B3] *IEEE Standards Style Manual*.[10]

[B4] IEEE Std 315™-1975(Reaff. 1993), Graphic Symbols for Electrical and Electronic Diagrams (Including Reference Designation Letters).[11]

[B5] IEEE Std 1232™-2010, IEEE Standard for Artificial Intelligence Exchange and Service Tie to All Test Environments (AI-ESTATE).

[B6] IEEE Std 1636.1™-2007, IEEE Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA): Exchanging Test Results and Session Information via the Extensible Markup Language (XML).

[B7] IEEE Std 1636.2™-2009, IEEE Standard for Software Interface for Maintenance Information Collection and Analysis (SIMICA): Exchanging Maintenance Action Information via the Extensible Markup Language (XML).

[B8] IEEE Std 1671.1™-2009, IEEE Standard for Automatic Test Markup Language (ATML) for Exchanging Automatic Test Equipment and Test Information via XML: Exchanging Test Descriptions.

[B9] ISO 10303-11:1994, *Industrial Automation Systems and Integration—Product Data Representation and Exchange—Part 11: Description Methods: The EXPRESS Language Reference Manual.*[12]

[B10] Namespaces in XML 1.0, 3rd ed. World Wide Web Consortium Recommendation 8 December 2009. Available: http://www.w3.org/TR/REC-xml-names/.

[B11] Schenk, D. A., and P. R. Wilson, *Information Modeling: The EXPRESS Way*. New York: Oxford University Press, 1994.

[B12] U.S. Navy, *Definitions of Terms for Test, Measurement and Diagnostic Equipment*, MIL-STD-1309D. Washington, DC: Naval Electronics Systems Command (ELEX-8111), 12 February 1992.

[B13] XML Schema Part 0: Primer. Available: http://www.w3.org/TR/xmlschema-0/.

[8] Available from World Wide Web: http://www.w3.org/TR/2008/PER-xml-20080205.
[9] *IEEE Standards Dictionary Online* subscription is available at: http://www.ieee.org/portal/innovate/products/standard/standards_dictionary.html.
[10] Available from World Wide Web: https://development.standards.ieee.org/myproject/Public/mytools/draft/styleman.pdf
[11] IEEE publications are available from The Institute of Electrical and Electronics Engineers (http://standards.ieee.org/).
[12] ISO publications are available from the ISO Central Secretariat, Case Postalė 56, 1 rue de Varembė, CH-1211, Genevė 20, Switzerland/.Suissė (http://www.iso.ch/). ISO publications are also available in the United States from the Sales Department, American National Standards Institute, 25 West 43rd Street, 4th floor, New York, NY 10036, USA (http://www.ansi.org/).

[B14] XML Schema Tutorial. Available: http://www.xfront.com.

[B15] XML Schema Tutorial, Part 1. Available: www.liquid-technologies.com/Tutorials/XmlSchemas/XsdTutorial_01.aspx.

Annex D

(informative)

IEEE list of participants

At the time this IEEE standard was completed, the Diagnostic and Maintenance Control (SASB/SCC20/DMC_WG) Working Group had the following membership:

Mike Seavey, *Chair*

Chris Gorringe
Anand Jain
Teresa Lopes
Ion Neag
John Sheppard
Timothy Wilmering

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Michael Bodkin
Bill Brown
Malcolm Brown
Keith Chow
Ray Davis
David Droste
H. Glickenstein
Chris Gorringe
Randall Groves
Werner Hoelzl
Noriyuki Ikeuchi
Anand Jain
Teresa Lopes
Greg Luri
William Maciejewski
Scott Misha
Mukund Modi
Ion Neag
Charles Ngethe
Leslie Orlidge
Peter Richardson
Robert Robinson
Bartien Sayogo
Mike Seavey
John Sheppard
Gil Shultz
Joseph Stanco
Walter Struppler
Ronald Taylor
Benton Vandiver
John Vergis
Timothy Wilmering
Oren Yuen
Daidi Zhong

When the IEEE-SA Standards Board approved this standard on 26 August 2013, it had the following membership:

John Kulick, *Chair*
David J. Law, *Vice Chair*
Richard H. Hulett, *Past President*
Konstantinos Karachalios, *Secretary*

Masayuki Ariyoshi
Peter Balma
Farooq Bari
Ted Burse
Wael William Diab
Stephen Dukes
Jean-Philippe Faure
Alexander Gelman
Mark Halpin
Gary Hoffman
Paul Houzé
Jim Hughes
Michael Janezic
Joseph L. Koepfinger*
Oleg Logvinov
Ron Petersen
Gary Robinson
Jon Walter Rosdahl
Adrian Stephens
Peter Sutherland
Yatin Trivedi
Phil Winston
Yu Yuan

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Patrick Gibbons
IEEE Standards Program Manager, Document Development

This page deliberately left blank

British Standards Institution (BSI)

BSI is the national body responsible for preparing British Standards and other standards-related publications, information and services.

BSI is incorporated by Royal Charter. British Standards and other standardization products are published by BSI Standards Limited.

About us

We bring together business, industry, government, consumers, innovators and others to shape their combined experience and expertise into standards -based solutions.

The knowledge embodied in our standards has been carefully assembled in a dependable format and refined through our open consultation process. Organizations of all sizes and across all sectors choose standards to help them achieve their goals.

Information on standards

We can provide you with the knowledge that your organization needs to succeed. Find out more about British Standards by visiting our website at bsigroup.com/standards or contacting our Customer Services team or Knowledge Centre.

Buying standards

You can buy and download PDF versions of BSI publications, including British and adopted European and international standards, through our website at bsigroup.com/shop, where hard copies can also be purchased.

If you need international and foreign standards from other Standards Development Organizations, hard copies can be ordered from our Customer Services team.

Copyright in BSI publications

All the content in BSI publications, including British Standards, is the property of and copyrighted by BSI or some person or entity that owns copyright in the information used (such as the international standardization bodies) and has formally licensed such information to BSI for commercial publication and use.

Save for the provisions below, you may not transfer, share or disseminate any portion of the standard to any other person. You may not adapt, distribute, commercially exploit, or publicly display the standard or any portion thereof in any manner whatsoever without BSI's prior written consent.

Storing and using standards

Standards purchased in soft copy format:

- A British Standard purchased in soft copy format is licensed to a sole named user for personal or internal company use only.
- The standard may be stored on more than 1 device provided that it is accessible by the sole named user only and that only 1 copy is accessed at any one time.
- A single paper copy may be printed for personal or internal company use only.

Standards purchased in hard copy format:

- A British Standard purchased in hard copy format is for personal or internal company use only.
- It may not be further reproduced – in any format – to create an additional copy. This includes scanning of the document.

If you need more than 1 copy of the document, or if you wish to share the document on an internal network, you can save money by choosing a subscription product (see 'Subscriptions').

Reproducing extracts

For permission to reproduce content from BSI publications contact the BSI Copyright & Licensing team.

Subscriptions

Our range of subscription services are designed to make using standards easier for you. For further information on our subscription products go to bsigroup.com/subscriptions.

With **British Standards Online (BSOL)** you'll have instant access to over 55,000 British and adopted European and international standards from your desktop. It's available 24/7 and is refreshed daily so you'll always be up to date.

You can keep in touch with standards developments and receive substantial discounts on the purchase price of standards, both in single copy and subscription format, by becoming a **BSI Subscribing Member**.

PLUS is an updating service exclusive to BSI Subscribing Members. You will automatically receive the latest hard copy of your standards when they're revised or replaced.

To find out more about becoming a BSI Subscribing Member and the benefits of membership, please visit bsigroup.com/shop.

With a **Multi-User Network Licence (MUNL)** you are able to host standards publications on your intranet. Licences can cover as few or as many users as you wish. With updates supplied as soon as they're available, you can be sure your documentation is current. For further information, email subscriptions@bsigroup.com.

Revisions

Our British Standards and other publications are updated by amendment or revision.

We continually improve the quality of our products and services to benefit your business. If you find an inaccuracy or ambiguity within a British Standard or other BSI publication please inform the Knowledge Centre.

Useful Contacts

Customer Services
Tel: +44 345 086 9001
Email (orders): orders@bsigroup.com
Email (enquiries): cservices@bsigroup.com

Subscriptions
Tel: +44 345 086 9001
Email: subscriptions@bsigroup.com

Knowledge Centre
Tel: +44 20 8996 7004
Email: knowledgecentre@bsigroup.com

Copyright & Licensing
Tel: +44 20 8996 7070
Email: copyright@bsigroup.com

BSI Group Headquarters

389 Chiswick High Road London W4 4AL UK

bsi.