INTERNATIONAL STANDARD

ISO 16642

First edition
2003-08-15

Computer applications in terminology — Terminological markup framework

Applications informatiques en terminologie — Plate-forme pour le balisage de terminologies informatisées

Reference number
ISO 16642:2003(E)

© ISO 2003

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO 2003

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

::
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

Contents

Page

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 16642 was prepared by Technical Committee ISO/TC 37, *Terminology and other language resources*, Subcommittee SC 3, *Computer applications for terminology*.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

© ISO 2003 – All rights reserved

Introduction

Terminological data are collected, managed and stored in a wide variety of systems, typically in applications, i.e. various kinds of database management system, ranging from personal computer applications for individual users to mainframe term-bank systems operated by major companies and governmental agencies. Termbases are comprised of various sets of data category and are based on various kinds of data model. Terminological data often need to be shared and reused in a number of applications, and this sharing is usually accomplished using intermediate formats. To facilitate co-operation and to prevent duplicate work, it is important to develop standards and guidelines for creating and using terminological data collections as well as for sharing and exchanging data.

The meta-model defined in this International Standard fits into an integrated approach to be used in analysing existing terminological data collections and in designing new ones, which are typically processed using relational or text-based data management systems. Terminological data collections can also be stored as structured documents with markup based on formats that are typically defined using Standard Generalized Markup Language (SGML), defined in ISO 8879 [12], or eXtensible Markup Language (XML), which is based on SGML but amended for use on the World Wide Web by the World Wide Web Consortium (W3C). An integrated approach eases the tasks of importing data from a flat file with markup into a database and of exporting data from a database to a structured document. Another motivation for an integrated approach, as opposed to entirely separate approaches for databases and structured documents, is that XML-based formats are now being processed in new ways, similar to traditional database management systems. For example, XML files are being queried and updated directly without importing data into traditional database environments.

This integrated approach to analysis and design consists of two levels of abstraction. The first (and most abstract) level of the integrated approach is the meta-model level. The meta-model level, which could also be called the abstract conceptual data model level, supports analysis and design at a very general level. The second level is the data model level.

At the data model level, the designer of the terminological data collection has the possibility to make various choices, based on real-life needs. First, designers must determine the form of representation most appropriate for their terminological data, addressing the following choices:

— whether to use a relational database or a flat file with markup;

— whether the data will be used primarily for queries and updates, and be represented in some database management system and, if this is the case, what system to use;

— whether the data will be used primarily for sharing and interchange, and be represented in a flat file with markup.

For the purposes of this International Standard it is assumed that all flat files will use XML markup.

Once the choice between a database management system and a flat file with XML markup has been made, a data model must be chosen. For a relational database, a typical method of describing a data model is an entity-relationship diagram. For an XML document, a typical method of describing a data model is a Document Type Definition (DTD). An alternative method, using what is called an “XML schema”, is provided by the W3C. In the future, it will be possible to use more abstract methods of describing an XML format.

A specific implementation of the meta-model for terminology markup expressed in XML is called a terminological markup language (or TML), which can be described on the basis of a limited number of characteristics, namely

— how the TML expresses the structural organization of the meta-model (i.e. the expansion trees of the TML),

— the specific data categories used by the TML and how they relate to the meta-model,

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

— the way in which these data categories can be expressed in XML and thus anchored on the expansion trees of the TML, i.e. the XML style of any given data category, and

— the vocabularies used by the TML to express those various informational objects as XML elements and attributes according to the corresponding XML styles.

Some of the examples in this International Standard are instances of the MSC (MARTIF with Specified Constraints) and Geneter formats as described in Annex B and Annex C respectively.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS

© ISO 2003 – All rights reserved

Not for Resale

Computer applications in terminology — Terminological markup framework

1 Scope

This International Standard specifies a framework designed to provide guidance on the basic principles for representing data recorded in terminological data collections. This framework includes a meta-model and methods for describing specific terminological markup languages (TMLs) expressed in XML. The mechanisms for implementing constraints in a TML are defined in this International Standard, but not the specific constraints for individual TMLs, except for the three TMLs defined in Annexes B to D.

This International Standard is designed to support the development and use of computer applications for terminological data and the exchange of such data between different applications. It does not standardize data categories and methods for the specification of data structures which are specified in ISO 12620 and other related International Standards.

This International Standard also defines the conditions that allow the data expressed in one TML to be mapped onto another TML and specifies a generic mapping tool (GMT) for this purpose (see Annex A).

In addition, this International Standard describes a generic model for describing linguistic data.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 1087-1, *Terminology work — Vocabulary — Part 1: Theory and application*

ISO 1087-2, *Terminology work — Vocabulary — Part 2: Computer applications*

ISO 12620:1999, *Computer applications in terminology — Data categories*

Extensible Markup Language (XML) 1.0, Second edition, BRAY, T., PAOLI, J., SPERBERG-MCQUEEN, C. M., and MALER, E. (eds.), W3C Recommendation 6 October 2000, available at <http://www.w3.org/TR/REC-xml>

Dublin Core Qualifiers, 2000-07-11, available at <http://dublincore.org/documents/2000/07/11/dcmes-qualifiers/>

XHTML™ 1.0 The Extensible HyperText Markup Language, 2nd edition, available at <http://www.w3.org/TR/xhtml1/>

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 1087-1 and ISO 1087-2, and the following apply.

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.1
CI
complementary information
information supplementary to that described in terminological entries and shared across the terminological data collection

NOTE Domain hierarchies, institution descriptions and bibliographical references are typical examples of complementary information.

3.2
data category
result of the specification of a given data field

[ISO 1087-2:2000, definition 6.14]

NOTE 1 A data category is a type of data field, such as Definition.

NOTE 2 ISO 12620 is an inventory of data categories, i.e. a “DCR” (see 3.3).

3.3
DCR
data category registry
data category specification used as a normative reference for the description of a TML

NOTE ISO 12620:1999 is a typical DCR in the context of this International Standard.

3.4
DCS
data category selection
component of a TML's specification that constrains its informational content

NOTE The informational content may be constrained, for example by specifying which data categories are allowed and how each data category can be used.

3.5
expansion tree
list of XML elements together with their organization that implement a level of the meta-model in a given TML

3.6
GMT
generic mapping tool
canonical representation of the terminological markup framework model in XML

3.7
GI
global information
technical and administrative information applying to the entire data collection

EXAMPLE Title of the data collection, revision history.

3.8
information unit
IU
elementary piece of information attached to a level of the meta-model

3.9
LS
language section
part of a terminological entry containing information related to one language

NOTE One terminological entry may contain information on one, two or more languages.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.10
object language
language being described

3.11
structural level
level of the meta-model to which one or more information units can be attached

3.12
structural skeleton
abstract description of an instance of a terminological database in conformity with the meta-model

3.13
style
properties relating to a data category that determine how it may be expressed in XML

3.14
TCS
term component section
part of a term section giving linguistic information about the components of a term

3.15
TS
term section
part of a language section giving information about a term

EXAMPLE Usage of a term, term elements.

3.16
TDC
terminological data collection
collection of data containing information on concepts of specific subject fields

[ISO 1087-2:2000, definition 2.21]

NOTE For the purposes of this International Standard, terminological data collections are assumed to contain GI and CI in addition to strictly terminological information.

3.17
TE
terminological entry
entry containing information on terminological units

EXAMPLE Subject-specific concepts, terms, etc.

NOTE Every element in the TE can be linked to CI, to other entries and to other elements in the same entry.

3.18
TML
terminological markup language
XML application for describing a TDC conforming to the constraints expressed in this International Standard

3.19
UML
unified modelling language
language for specifying, visualizing, constructing and documenting the artefacts of sofware systems

3.20
vocabulary
set of strings used to implement a data category according to a style

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.21
working language
language used to describe objects

3.22
XML outline
part of a terminological database corresponding to the XML implementation of the meta-model

4 General principles and interoperability principle

Describing a specific TML can be seen as a process involving several knowledge sources which interact with one another at various levels. This process leads to the required specification of two important aspects of a TML:

— the informational properties of the TML, i.e. its capacity to represent a given piece of information related to the terminological description;

— the way the TML can be expressed, for instance as an XML document.

Figure 1 represents the various knowledge sources that form the basis of this International Standard and that can lead to the full specification of a TML.

Two of those knowledge sources are shared by all TMLs and can be seen as reference material for this International Standard.

— The meta-model describes the basic hierarchy of structural levels to which any TML shall conform as defined in this International Standard.

— A DCR is a set of data category specifications on which any specific TML shall rely for creating its own data category set. For the application of this International Standard, ISO 12620 forms a reference DCR for any information unit to be used in the specification of a TML.

Two other knowledge sources are used to define the specific information units of a given TML from the point of view of both its informational properties and its representation in XML.

— The DCS describes the set of data categories that can be used within a given TML. The DCS can comprise both a subset of the DCR together with any idiosyncratic data categories needed for a specific application.

— The dialectal specification (Dialect) includes the various elements needed to describe a given TML as an XML document. These elements comprise expansion trees and data category instantiation styles, together with their corresponding vocabularies.

The combination of the meta-model and a given DCS is enough to define conditions of interoperability, encompassing the full informational properties of the TML from a terminological point of view. Any information structure that corresponds to such conditions has a canonical expression as an XML document using the GMT representation. The interoperability between two different TMLs depends solely on their compatibility at that level (see Figure 2).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

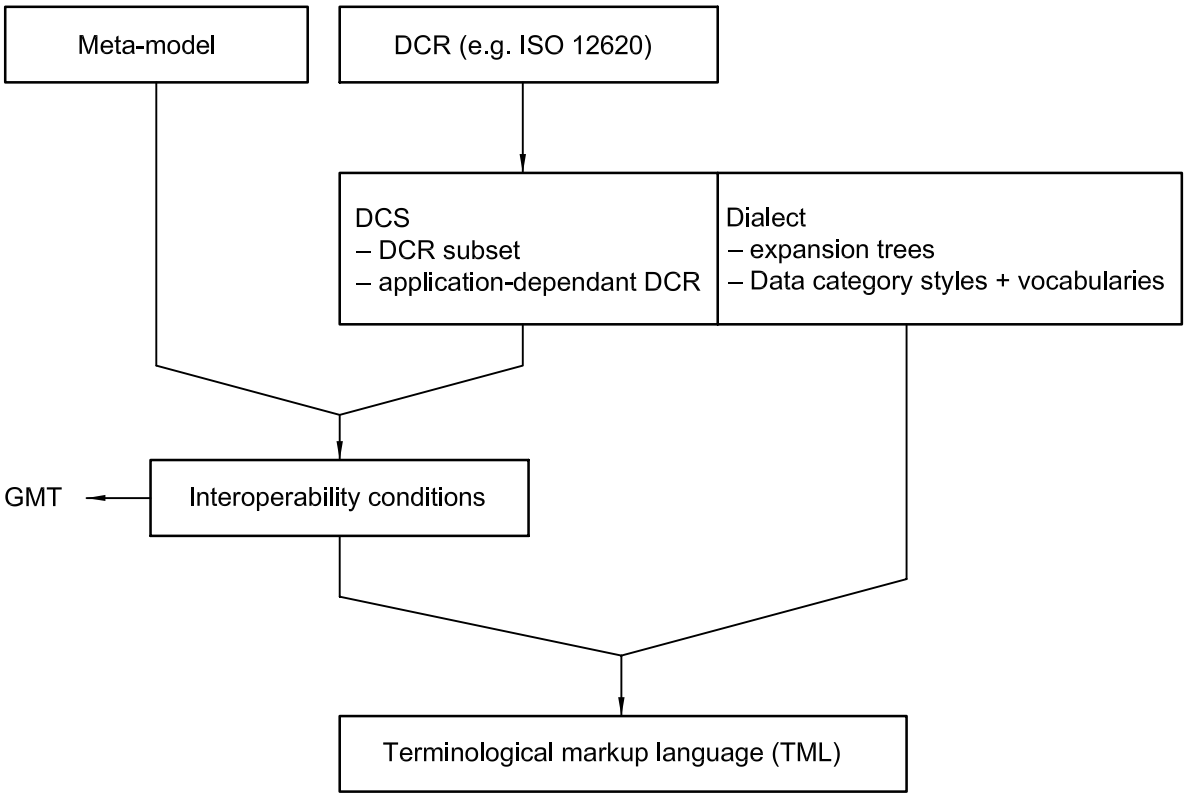


Figure 1 — The various knowledge sources involved in the description of a TML

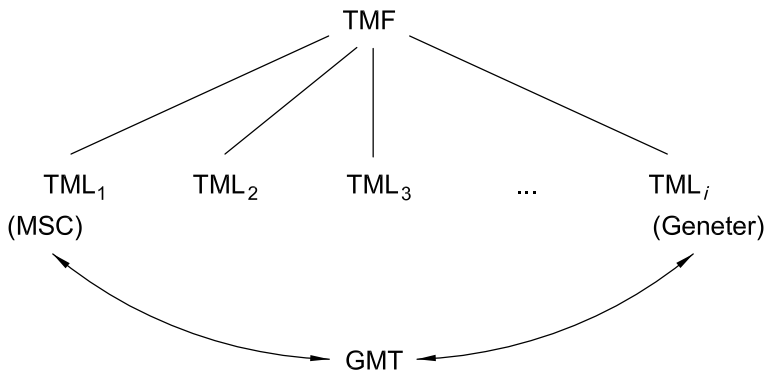


Figure 2 — Interoperability between two TMLs using the GMT

More precisely, the interoperability between two TMLs can be defined by comparing their specifications. Under the condition that two TMLs are based upon the same DCS, any terminological data collection expressed in either TML can be transformed, without any loss of information, into a collection represented using the other format. In particular, the additional specifications that are needed to express interoperability conditions in the

--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

form of an XML implementation of a TML do not influence the level of interoperability between two markup languages. The preceding interoperability principle leads to two immediate consequences:

— it justifies the role of the GMT as a pivot markup language when transforming any data expressed in a TML into data expressed in another TML;

— when two TMLs are based upon two different DCSs, it provides a framework for identifying what information can be transformed between one format and another, and what will be lost during the transformation (weak interoperability condition).

The combination of interoperability conditions with dialectal information provides the information necessary to express a given TML in XML.

5 Generic model for describing linguistic data and its application to terminology

5.1 Introduction

5.1.1 General principles

Linguistic structures exist in a wide variety of formats ranging from highly organized data (e.g. terminological data bases) to loosely structured information (e.g. text lightly annotated for names, dates, etc.). The representation of annotated corpora is not hierarchical, but based on the expression of multiple views representing various levels of linguistic information, usually pointing to primary data (e.g. part of speech tagging) and sometimes to one another (e.g. reference annotation based on basic phrase structure annotation).

The following model identifies a class of document structures which could be used to cover a wide range of linguistic annotation formats, and provides a framework which can be applied using XML.

Each type of document structure is described by means of a three-tiered information structure that describes:

— a *meta-model*, which represents a hierarchy of structural nodes which are relevant for linguistic description;

— specific *information units*, which can be associated with each structural node of the meta-model;

— relevant *annotations*, which can be used to qualify some parts of the value associated with a given information unit.

Each structural node can be qualified by a group of basic or compound information units. A basic information unit describes a property that can be directly expressed by means of a data category. A compound information unit corresponds to the grouping at one level of several basic information units, which, taken together, express a coherent unit of information.

For instance, a compound information unit can be used to represent the fact that a transaction can be a combination of a transaction type, a responsibility and the transaction date.

Basic information units, whether they are directly attached to a structural node or within a compound information unit, can take two non-exclusive types of value:

— an atomic value corresponding either to a simple type (in the sense of XML schemas) such as a number, string, element of a picklist, etc., or to a mixed content type in the case of annotated text;

— a reference to a structural node in order to express a relation between it and the current structural node.

Basic and compound information units can be abstractly represented as feature-value structures associated with specific structural nodes in the structural skeleton. For instance, a Geneter sub-document identified as `<Owner>UHB</Owner>` can be modelled as a basic information unit in the following way:

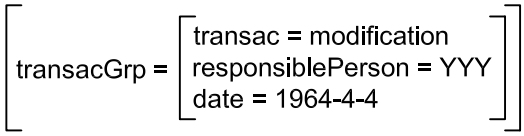
```
[owner = UHB]
```


--`,,`,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

Similarly, the following MSC sub-document

```
<transacGrp>
   <transac>modification</transac>
   <transacNote type="responsiblePerson">YYY</transacNote>
   <date>1964-4-4</date>
</transacGrp>
```

can be modelled as



The preceding model for information units is to be completed by a last level of information representation, which corresponds to the association of semantic information to subparts of information unit values. Such *annotations* typically occur when one wants to identify, within a terminological definition, specific references to information involving genus and/or differentia. See, for instance the following definition for lead pencil:

```
<definition>
   <broaderConcept>pencil</broaderConcept> whose
   <characteristic>casing</characteristic> is fixed around a central
   <characteristic>graphite</characteristic> medium which is
   <characteristic>used for writing or making marks</characteristic>
</definition>
```

Such information, also known as *mixed content* in XML, cannot be directly represented as a feature structure and will be directly expressed in the following GMT representation (<annot> element).

5.1.2 Example

To illustrate how a TDC can be analysed as an abstract structure, let us consider a simple terminological entry expressed as an XML document conforming to MSC specifications:

```
<?xml version="1.0"?>
<martif type="MSC" lang="en">
   <text>
      <body>
         <termEntry id="ID67">
            <descrip type="subjectField">manufacturing</descrip>
            <descrip type="definition">A value between 0 and 1 used in ...
            </descrip>
            <langSet lang="en">
               <tig>
                  <term>alpha smoothing factor</term>
                  <termNote type="termType">fullForm
                  </termNote>
               </tig>
            </langSet>
            <langSet lang="hu">
               <tig><term>Alfa ...</term></tig>
            </langSet>
         </termEntry>
      </body>
   </text>
</martif>
```

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

The XML document represented above can be mapped to the abstract model described in this clause by identifying a structural skeleton corresponding to the meta-model and by associating the corresponding information units with each structural node in the structural skeleton, as shown in Figure 3.

Here, data categories can be mapped onto the corresponding data categories specified in ISO 12620:

Data category	ISO 12620:1999 number	ISO 12620:1999 name
id	A.10.15	entry identifier
subjectField	A.4	subject field
definition	A.5.1	definition
lang	A.10.7.1	language identifier
term	A.1	term
termType	A.2.1	term type
fullForm	A.2.1.7	full form

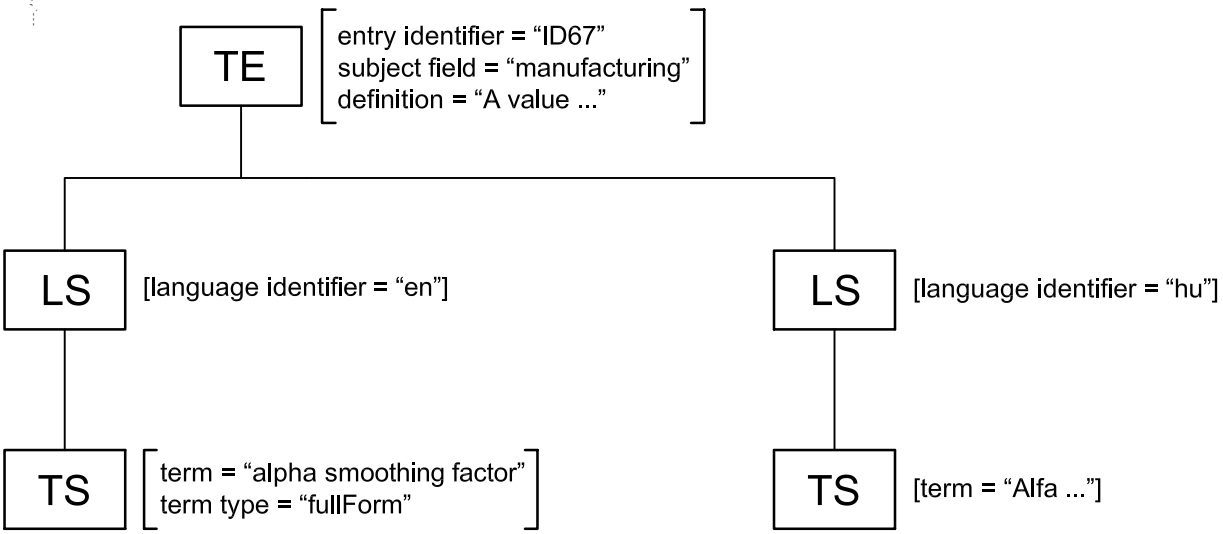


Figure 3 — Mapping an XML document to the abstract model

5.2 Generic representation of structural levels and information units

5.2.1 Linguistic data can be represented using a generic architecture that consists of a graph of elementary structural nodes to which one or more information units are attached. This architecture is shown in a UML diagram in Figure 4. The diagram expresses the relationship between the following defined classes:

— structural node: a class containing one attribute (LevelName) that identifies objects of this type in the context of a given Linguistic Resource (LR) format (e.g. TE/Terminological Entry for the representation of TermBanks);

— information unit: a class containing three attributes to identify objects of this type in relation to a given data category (IUName, e.g. Definition, PartOfSpeech, etc.), to describe a type for its content (C_type) and to provide the actual content value (C_value).

NOTE The value of C_type can either belong to the set of simple types as defined in XML Schema Part 2: Datatypes or be MIXED as described in 5.2.2.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Objects of these two classes can be related in the following ways.

— association: Indicates that a structural node is related to another structural node by a hierarchical link. There is no constraint on the number of links or the structure of the network that those links create (tree, directed acyclic graph, etc.) (0..*).

— hasContent: Relates a structural node to information units (e.g. a definition attached to a Terminological Entry). An instance of an information unit is attached to one and only one structural node (1..1).

— refinement: Relates information units that provide additional information to another information unit (IU) (e.g. a /note/ refining a /definition/). A refining IU is related to one and only one refined IU (1..1).

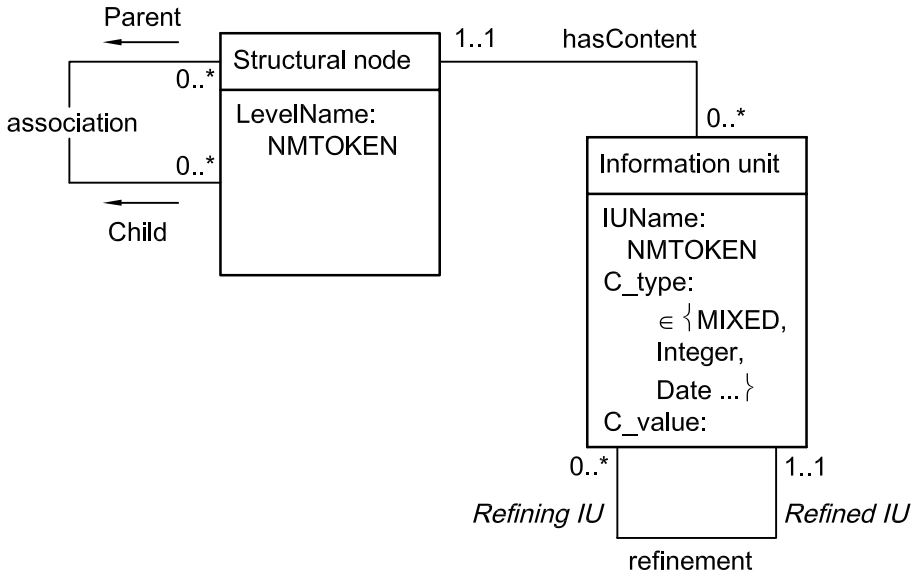


Figure 4 — UML diagram for structural nodes and information units

5.2.2 The MIXED type is an ordered combination of textual content and information units, corresponding to any kind of annotated content. It can be represented in a UML by means of the agglomeration operator, as shown in Figure 5.

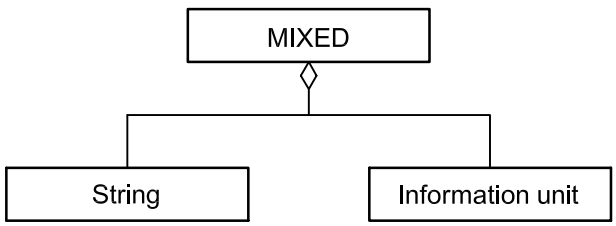


Figure 5 — MIXED object class

Keeping this rather general definition leads to the fact that annotations in textual content can be refined by other information units (for instance to indicate when and by whom the annotation has been made).

5.3 The terminological meta-model

The terminological meta-model is based on guidelines concerning the methods and principles of terminology management involving the production of terminological entries as described in ISO 704. One of the most important characteristics of a terminological entry — compared to a lexicographical entry — is its concept orientation. A terminological entry treats one concept in a given language and, in the case of multilingual

--`,,`-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

terminological entries, one or more totally or partially equivalent concepts in (an)other language(s), whereas a lexicographical entry contains one lemma (the base form of a single lexical unit) and one or more definitions (representing different meanings) in one or more languages. The meta-model presented in this International Standard gives guidelines for terminological entries. It is understood in this context that although lexicographical entries typically contain word or lexeme related information (such as part of speech, gender, etc.), some of this lexicographical information is frequently also included in terminological entries.

A terminological data collection comprises global information about the collection and a number of entries. Each entry performs three functions:

— it describes one concept, or two or more totally or partially equivalent concepts, in one or more languages;

— it lists the terms that designate the concept(s);

— it describes the terms themselves.

Each entry can have multiple language sections, and each language section can have multiple terminological units. Each data element in an entry can be associated with various kinds of descriptive and administrative information. In addition, there are various other resources that are not part of any one entry, but that can be linked to one or more entries. Such resources include bibliographic references, descriptions of ontologies, and binary data such as images that illustrate concepts.

By instantiating the generic architecture presented in 5.2, the terminological meta-model is described through seven instances from the structural node class.

— TDC (terminological data collection): Top level container for all information contained in a terminology system.

— GI (global information): Information that applies to all elements represented in a file, as opposed to information that may pertain to some but not to all components of the file. Usually contains, for example, the title of the (XML) file, the institution or individual originating the file, address information, copyright information, update information, etc.

— TE (terminological entry): Information that pertains to a single concept. Usually contains, for example, descriptive information pertinent to a concept, and administrative information concerning the concept. Can contain one or more language sections depending on whether the termbase is monolingual, bilingual, or multilingual.

— CI (complementary information): Usually contains, for example, textual bibliographical or administrative information residing in or external to the file, static or dynamic graphic images, video, audio, or virtually any other kind of binary data (i.e. blobs). Might also include references to other terminological resources or contextual links to related text corpora or to ontologies. These items are often designated as shared resources because they are available to all points in a termbase and are not repeated for different entries.

— LS (language section): Contains all the term sections for a terminological entry that are used in a given language, as well as information. Usually contains, for example, definitions, contexts, etc. associated with that language or the terms in that language.

— TS (term section): Information about terms. Usually contains, for example, a single term used to designate the concept that is the subject of the terminological entry, as well as any other information (e.g. definitions, contexts, etc.), associated with that term.

— TCS (term component section): Information about morphemic elements, words, or contiguous strings from which a polymorphemic (or multiword) term is formed. In some languages, such as German or English, it is frequently unnecessary to distinguish information about the individual components making up a polynomial term. In other languages, such as French or Spanish, it is important to be able to include information such as gender for the individual words used in constructing a multiword term because this information is necessary when using the term in texts.

These instances of structural levels implement the “association” relation with constraints on cardinality (see Figure 6), which can also be schematized by the sketch shown in Figure 7.

— A TE can contain any number of LSs (0..*).

— An LS can contain any number of TSs (0..*).

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

— A TS can contain any number of TCSs (0..*).

— A TDC must contain exactly one GI (1..1), at most one CI-Level (0..1) and any number of TEs (0..*).

Hierarchical organization is ensured by the 1..1 limitations expressed for the dual cardinalities for each relation.

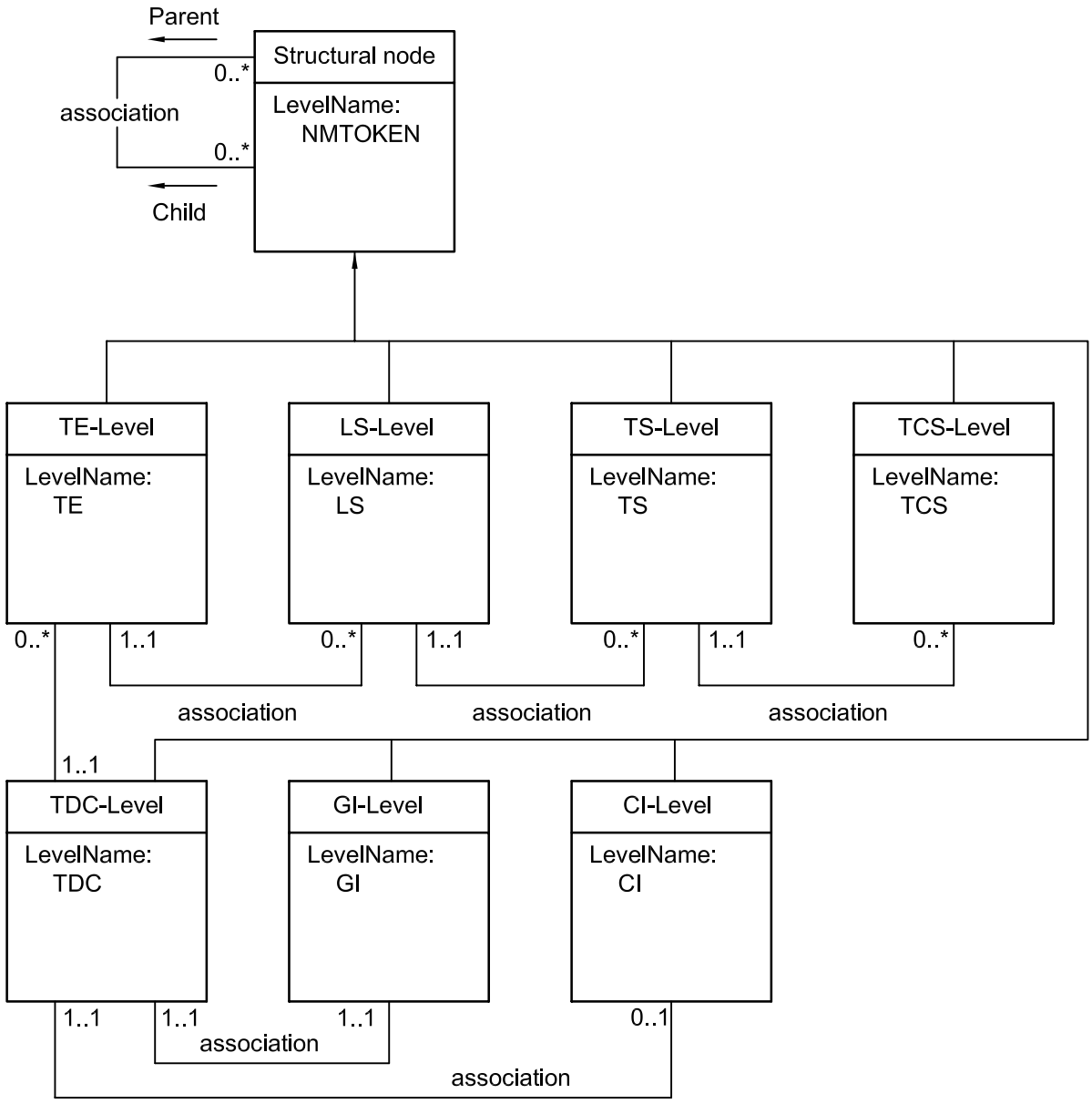


Figure 6 — Terminological meta-model — UML diagram

--`,,`,-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

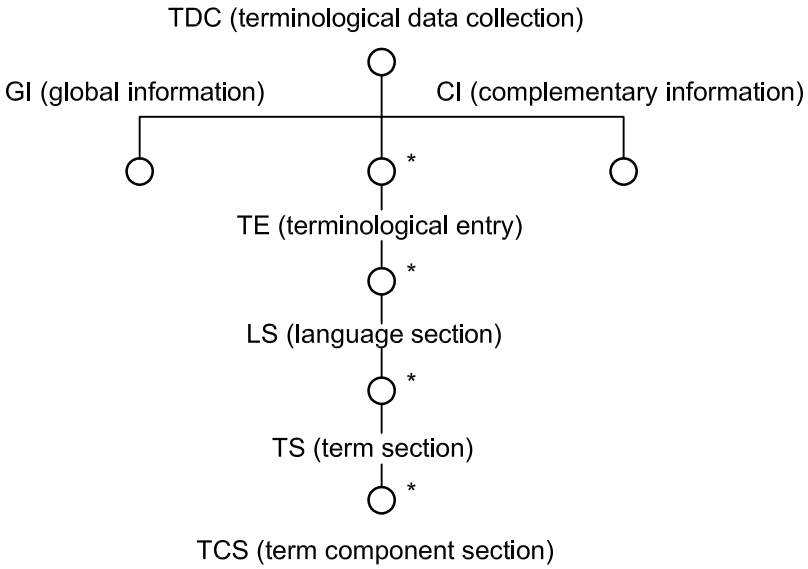


Figure 7 — Terminological meta-model — Schematic view

5.4 Designing representations of terminological data on the basis of the meta-model

Each DCS shall be configured in response to the real-world needs of some user group and consist of a list of data categories from the DCR (e.g. ISO 12620) and constraints on each of those data categories. Constraints include restrictions on the values each data category may take (ranging from “text with markup” for contextual examples to a “picklist” for grammatical gender, including specific data types, for example, defined in XML Schema Part 2: Datatypes, which can be used to describe numbers or dates, for instance). Constraints on descriptive data categories also include restrictions on where a particular data category can appear in an entry, selected from the options provided by the core structure of the meta-model (namely TE, LS, TS and TCS).

The meta-model provides for a vast range of possible data models, particularized to a real-world application by selecting a structure based on the meta-model and restricted to certain data categories and data category values applied to the TE, LS, TS and TCS as object classes of the meta-model.

All XML formats conforming to this International Standard are based on

a) the meta-model,

b) subsets of DCSs that are essentially derived from ISO 12620, and

c) XML DTDs or XML schemas.

Database applications for terminological data conforming to this International Standard shall be based on

a) the meta-model,

b) DCSs that are essentially derived from ISO 12620, and

c) data models defined by means of entity relationship diagrams.

5.5 Interchange, dissemination and interoperability

Interchange involves a transfer of information between two computer systems and is often bidirectional, but need not be. Interchange is accomplished using intermediate formats. Dissemination is unidirectional and can be either for use by another computer system or for human viewing. Formats structured in compliance with the meta-model and the DCSs of ISO 12620 should be interoperable, i.e. it should be possible to convert data from one format into another format and back without loss of information (sometimes referred to as a “lossless round trip”).

--`,,`,-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

If data are so rigorously defined that it is unnecessary for the importer to establish contact with the originators of the data in order to interpret them, interchange is said to be blind. When there are only two interchange partners and they are known to each other, blindness is not an issue. But when there are multiple sources of terminological data that must be imported by a single routine, especially if it is desirable to add more sources without modifying the import routine, blindness becomes very important.

In bidirectional interchange, the objective is usually to maximise the preservation of information. But in the case of dissemination, representation can be intentionally partial, leaving out some information that was in the original data collection. For example, a dissemination-oriented representation for human translators does not necessarily include some administrative information that is only relevant to terminologists maintaining the database.

The specifics of a particular XML format will be influenced by the purpose of the format (i.e. for dissemination or for interchange) and by whether there is a need for blindness. Whatever the purposes and real-world needs that guide the design of a database application or an XML format, once designed, the database structure or the format takes on a life of its own as it is used to represent a variety of data, some of which may not have been anticipated by the designer. By following the integrated approach described here, it is more likely that the resulting format will be adaptable to varied circumstances and will be compatible with other database structures or formats.

5.6 XML canonical representation of the generic model

5.6.1 Introduction

The hierarchical organization of the meta-model and the qualification of each structural level can be realized in XML by instantiating the abstract structure described in 5.3 and associating information units to this structure.

The meta-model can be represented by means of a generic element `<struct>` (for structure) which can recursively express the embedding of the various representation levels of a terminological data collection. Each structural node in the meta-model shall be identified by means of a `type` attribute associated with the `<struct>` element. The possible values of the `type` attribute shall be the identifiers of the levels in the meta-model, i.e. TDC, GI, CI, TE, LS, TS, TCS.

Basic information units associated with a structural node can be represented using the `<feat>` (for feature) element.

Compound information units can be represented using the `<brack>` (for bracket) element, which can itself contain a `<feat>` element followed by any combination of `<feat>` elements and `<brack>` elements. Each information unit must be qualified with a `type` attribute, which shall take as its value the name of an ISO 12620 data category or that of a user-defined data category. Finally, the content model of the `<feat>` element can contain annotations expressed by means of an `<annot>` (for annotation) element. This element is also qualified by a `type` attribute referencing an ISO 12620 data category or an equivalent user-defined data category.

5.6.2 Example

The following example illustrates how the information contained within the MSC terminological entry presented in 5.1.2 can be encoded in the GMT format. This entry only contains basic information units and, as seen before, only maps three levels of the meta-model, namely TE, LS and TS.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<tmf>
   <struct type="TE">
         <feat type="entry identifier">ID67</feat>
         <feat type="subject field">manufacturing</feat>
         <feat type="definition">A value between 0 and 1 used in ...</feat>
         <struct type="LS">
               <feat type="language identifier">en</feat>
               <struct type="TS">
```

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
                    <feat type="term">alpha smoothing factor</feat>
                    <feat type="term type">fullForm</feat>
              </struct>
        </struct>
              <struct type="LS">
                    <feat type="language identifier">hu</feat>
                    <struct type="TS">
                          <feat type="term">Alfa ...</feat>
                    </struct>
              </struct>
        </struct>
   </struct>
</tmf>
```

5.6.3 Description of the GMT format

5.6.3.1 The <tmf> element

5.6.3.1.1 Description

The <tmf> element is the root element for any valid GMT document. It contains both the GI that corresponds to a TDC (the collection itself) and the CI comprising, in particular, external resources which are needed for describing the various TEs.

5.6.3.1.2 Content model (DTD)

```
<!ELEMENT tmf (struct)>
```

5.6.3.2 The <struct> element

5.6.3.2.1 Description

The <struct> element can be used to represent a structural node in a given structural skeleton. One such structural node will be represented by exactly one <struct> element. The <struct> element is recursive and may also contain <feat> and/or <brack> elements to express information units belonging to the corresponding level of the meta-model.

The <struct> element accepts the following XML attributes:

— type: categorizes the <struct> element by identifying a structural node in the meta-model (TDC, GI, TE, CI, LS, TS or TCS) represented by the element;

— id: allows unique identification of the corresponding information unit in the structural skeleton;

— target: a pointer to another <struct> element where the actual content of the current one is provided; the semantics of this pointer is that of a unification of the current <struct> element with the one which is pointed to.

5.6.3.2.2 Content model (DTD)

```
<!ELEMENT struct ((feat|brack)*, struct*)>

<!ATTLIST struct
      type (TDC|GI|CI|TE|LS|TS|TCS) #REQUIRED
      id ID #IMPLIED
      target CDATA #IMPLIED>
```


--`,,`,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

5.6.3.3 The <feat> element

5.6.3.3.1 Description

The <feat> element can be used to express any information unit that is either directly attached to a structural node in the structural skeleton (represented by a <struct> element), or grouped together with other information units within a <brack> element. The <feat> element can contain any type of tagged data, whether it corresponds to strict annotation by means of the <annot> element or whether a module involving external markup is used for a specific application. In the latter case, the elements from this module shall be qualified according to a specific namespace.

The <feat> element accepts the following XML attributes:

— type: categorizes the <feat> element through the reference to the name of the corresponding data category — the name should be taken either from ISO 12620 or from a data category described in the context of a specific application and not present in ISO 12620;

— target: a pointer to a <struct> element in the case the information unit expresses a relation between the current structural node and another structural node in the structural skeleton;

— source: a pointer to a <struct> element in cases where the information unit is described external to the structural node to which it is supposed to be attached — this approach can be used, for instance, to describe a database of conceptual links external to a TDC proper.

5.6.3.3.2 Content model (DTD)

```
<!ELEMENT feat (#PCDATA | annot)*>

<!ATTLIST feat
      type CDATA #REQUIRED
      target CDATA #IMPLIED
      source CDATA #IMPLIED>
```

EXAMPLE

The following elements constitute valid expressions of information units.

— Basic information unit attached directly to a structural node (level TE):

```
<struct type="TE">
      <feat type="entry identifier">ID67</feat>
</struct>
```

— Basic information unit whose value is a reference to a structural node in the structural skeleton and whose id attribute has the value “TE24”:

```
<struct type="TE">
      <feat type="partitive relation" target="TE24"/>
</struct>
```

— Basic information unit anchored at the structural node in the structural skeleton whose id attribute value is “TE24”:

```
<struct type="TE">
      <feat type="entry identifier" source="TE24">ID67</feat>
</struct>
```

— Compound information unit anchored at the structural node in the structural skeleton whose id attribute value is “TE23” and which makes reference to a structural node in the structural skeleton whose id attribute value is “TE24”:

```
<feat type="partitive relation" source="TE23" target="TE24"/>
```

--`,,`,,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

5.6.3.4 The <brack> element

5.6.3.4.1 Description

The <brack> element can be used to express any group of information units whose meaning is interrelated. It contains a list comprising at least one <feat> element followed by any combination of <brack> and <feat> elements. The <brack> element accepts the following attribute:

— source: a pointer to a <struct> element in cases where the group of information units involved is described external to the structural node to which it is supposed to be attached. The source attribute is thus inherited by all the <feat> elements contained in the current <brack> element.

5.6.3.4.2 Content model (DTD)

```
<!ELEMENT brack (feat, (feat|brack)+)>
<!ATTLIST brack
      source CDATA #IMPLIED>
```

EXAMPLE

The following elements constitute valid expressions of information units.

— Compound information unit comprising two basic features:

```
<brack>
      <feat type="classification code">xxx</feat>
      <feat type="classification system">Lenoch</feat>
</brack>
```

— Compound information unit anchored at a structural node in the structural skeleton where the id attribute value is "TE24":

```
<brack source="TE24">
      <feat type="classification code">xxx</feat>
      <feat type="classification system">Lenoch</feat>
</brack>
```

5.6.3.5 The <annot> element

5.6.3.5.1 Description

The <annot> element shall be used to tag any portion of the content of a given <feat> element, provided this procedure is allowed by the content type of the corresponding data category. The <annot> element accepts the following attributes:

— type: categorizes the <annot> element by referencing the name of the corresponding data category — the name should be taken either from ISO 12620 or from a data category specified in the context of a specific application and not present in ISO 12620;

— target: a pointer to a <struct> element in cases where the annotation expresses a relation between the current information unit and another structural node in the structural skeleton.

5.6.3.5.2 Content model (DTD)

```
<!ELEMENT annot (#PCDATA)>

<!ATTLIST annot
      type CDATA #REQUIRED
      target CDATA #IMPLIED>
```

--`,,`,,`-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

EXAMPLE

The following constitutes a valid expression of information units.

```
<feat type="definition">
      <annot type="broader concept generic">pencil</annot> whose
      <annot type="characteristic">casing</annot> is fixed around a central
      <annot type="characteristic">graphite</annot> medium which is
      <annot type="characteristic">used for writing or making marks</annot>
</feat>
```

5.7 Representing languages in a terminological data collection

Any terminological data collection conforming to this TMF should clearly distinguish between the working language and the object language, which are the two types of language information that can be attached to any level of the collection.

The working language is the language used to express any given textual content in the data collection. This information shall be represented using the `xml:lang` attribute as defined in the Extensible Markup Language (XML) recommendation of the W3C and used accordingly. In particular, the scope of the working language is the whole sub-document starting from the element where the information has been declared, unless it is superseded by another working language declaration for some element in this sub-document.

The object language is the language of the terminological information which is being described at some level in the terminological data collection (typically at the language section level). As such, it is represented in the TMF as a data category (“language identifier” in ISO 12620) and may be represented in a given TML using any style among those described in this International Standard. Its possible values are those allowed by the reference data category in ISO 12620 or a reduced set defined for a given TML.

The following example shows how the two types of language can be used within a LS expressed in GMT:

```
<struct type="LS" xml:lang="fr">
      <feat type="language identifier">en</feat>
      <feat type="definition">Une valeur entre 0 et 1 utilisée...</feat>
      <struct type="TS">
            <feat type="term" xml:lang="en">alpha smoothing factor</feat>
            <feat type="term type">fullForm</feat>
      </struct>
</struct>
```

This same example can be represented in MSC as follows:

```
<langSet lang="en" xml:lang="fr">
      <descrip type="definition">Une valeur entre 0 et 1 utilisée...
      </descrip>
      <tig>
            <term xml:lang="en">alpha smoothing factor</term>
            <termNote type="termType">fullForm</termNote>
      </tig>
</langSet>
```

and in Geneter:

```
<languageGrp value="en" xml:lang="fr">
      <Definition>Une valeur entre 0 et 1 utilisée...</Definition>
      <termGrp>
            <Term formType='fullForm'>alpha smoothing factor</Term>
      </termGrp>
</languageGrp>
```

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

6 Defining a TML

6.1 General

The specification of a TML shall be considered as a sequence consisting of two phases.

— A first phase consists of specifying those data categories required for this TML, i.e. a DCS. This can be done by selecting a subset of the data categories specified in ISO 12620 and, if necessary, specifying additional data categories that are needed for the current TML but do not belong to ISO 12620. This phase shall lead to the definition of interoperability conditions needed for interaction with other TMLs.

— A second phase corresponds to the realization of the TML as an XML format. This is achieved by providing expansion trees associated with the different structural nodes in the meta-model and by instantiating the necessary information styles and vocabularies required for the data categories that occur in those trees. This phase provides the minimal information to specify fully the XML schemas controlling the valid instances of the TML as well as the filters which can transform a TML instance into an GMT instance and vice versa.

These two steps are more precisely defined in 6.2 and 6.3.

6.2 Defining interoperability conditions

The definition of interoperability conditions is based upon the specification of the set of data categories that are valid for a given TML. This specification relies upon the provision, for each data category, of a set of properties, which can be modelled as a Resource Description Framework (RDF) representation. These properties are as follows:

— a namespace (in the sense of XML namespaces) that is either the namespace of a DCR (e.g. ISO 12620) from which the data category is taken or a local namespace associated with the application-defined data categories;

— a unique name (DCName property in RDF) within the namespace;

— a type (DCType) which indicates whether the data category describes a possible information unit for the TML (DCType='complex') or is one possible value of an information unit (DCType='simple');

— the list of possible structural nodes (DCLevel) where the data category may occur for the TML;

— the list of the values (Content) that are allowed for the category in the case of a complex data category.

If a data category has been selected from a DCR, the following constraints apply.

— The content description for the data category is subsumed by the one in ISO 12620. For instance, if the content is defined by a data type, it should be a sub-type of the one described in ISO 12620, or, if the content is described as a picklist, it should be a subset of the corresponding picklist in ISO 12620.

— The category can be applied to a list of structural nodes, which is a subset of the list of authorized structural nodes expressed in ISO 12620.

6.3 Implementing a TML

6.3.1 Introduction

Realizing interoperability conditions as a TML requires the specification of the XML structures that can be used in instances that will describe the corresponding terminological data collection. This requires an XML outline, which is the set of XML elements that will implement the structural skeleton of the instance and anchoring mechanisms for the various information units that are described in the DCS.

6.3.2 Implementing the meta-model

The structural part of a TML shall be defined by associating an XML sub-tree (or expansion tree) with each structural node in the meta-model. For each structural node having a parent in the meta-model (i.e. for which

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

there exists a higher level in the meta-model) an anchor shall also be defined which comprises a node in its parent's expansion tree and to which its own expansion tree can be attached.

The XML outline of an instance of a TML comprises all the expansion trees associated with its structural skeleton.

6.3.3 Anchoring data categories on the TML XML outline

6.3.3.1 General

The expansion tree associated with a structural node consists of a set of XML element nodes. Each of these nodes is a potential anchor for the implementation of an information unit that is allowed at that structural node. According to the anchoring style associated with the information unit and in conjunction with the corresponding vocabulary used in the actual TML, it is possible to specify how the data category will be expressed as an XML sub-structure of its anchor. The corresponding properties (anchor, style and vocabulary) shall be additional descriptions included with the corresponding data category in the full DCS associated with a TML.

6.3.3.2 Styles and vocabulary

Any information unit attached to the structural skeleton of a TML can be implemented using one of the five styles Attribute, Element, TypedElement, ValuedElement and TypedValuedElement. These styles correspond to the way a feature-value pair is expressed in XML.

The Attribute style implements an information unit as an XML attribute of a given anchor. The vocabulary represents the name of the XML attribute. The value associated with a specific information unit is realized as the content of this XML attribute.

EXAMPLES

GMT representation	**MSC representation** (anchor: `<termEntry>`)

`<struct type="TE">`
`  <feat type="entry identifier">ID67</feat>`
`  ...`
`</struct>`
	`<termEntry id="ID67">`
	`  ...`
	`</termEntry>`

GMT representation	**Geneter representation** (anchor: `<ldl>`)

`<struct type="LS">`
`  <feat type="language identifier">en`
`  </feat>`
`  ...`
`</struct>`
	`<ldl language="en">`
	`  ...`
	`</ldl>`

The Element style implements an information unit as an XML element, which itself is a child of a given anchor. The vocabulary represents the name of this XML element. The value associated with a specific information unit is realized as the content of this XML element.

EXAMPLES

GMT representation	**MSC representation** (anchor: `<tig>`)

`<struct type="TS">`
`  <feat type="term">alpha smoothing factor`
`  </feat>`
`</struct>`
	`<tig>`
	`  <term>alpha smoothing factor</term>`
	`</tig>`

GMT representation	**Geneter representation** (anchor: `<tl>`)

`<struct type="TS">`
`  <feat type="term">barbed wire</feat>`
`</struct>`
	`<tl>`
	`  <Term>barbed wire</Term>`
	`</tl>`

--`,,`,,-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

The TypedElement style implements an information unit as an XML element, which itself is a child of a given anchor, and which is further specified by an XML attribute `type`. The vocabulary represents the name of this XML element and a value for the XML attribute `type`. The value associated with a specific information unit is realized as the content of this XML element.

EXAMPLES

GMT representation	**MSC representation** (anchor: `<termEntry>`)

`<struct type="TE">`
`  <feat type="subject field">manufacturing`
`  </feat>`
`</struct>`
	`<termEntry ... >`
	`  <descrip type="subject field">`
	`  manufacturing</descrip>`
	`</termEntry>`

GMT representation	**Geneter representation**

`<struct type="TE">`
`  <feat type="subject field">manufacturing`
`  </feat>`
`</struct>`
	`<terminologicalEntry ... >`
	`  <free type="subject field">manufacturing`
	`  </free>`
	`</terminologicalEntry>`

The ValuedElement style implements an information unit as an XML element, which is itself a child of a given anchor, and which is further specifed by an XML attribute `value`. The vocabulary represents the name of this XML element. The value associated with a specific information unit is realized as the content of the XML attribute `value`.

The TypedValuedElement style implements an information unit as an XML element, which is itself a child of a given anchor, and which is further specified by means of an XML attribute `type`. The vocabulary represents the name of this XML element and the name of the XML attribute. The value associated with a specific information unit is realized as the content of the XML attribute `type`.

6.3.3.3 Constraints on datatypes for information units

Whereas information units implemented using the Element or TypedElement style can take values containing additional markup (in particular markup resulting from the implementation of annotations), information units implemented using either the Attribute or ValuedElement style shall not contain any such markup. In this respect, consistency checking is required when defining the DCS for a given TML.

6.3.3.4 External markup modules

DCSs can include reference to external markup modules that can be used in the content model of information units or to reference external objects corresponding to CI in the meta-model (e.g. bibliographical references). These modules are referenced using a registered namespace.

As an example, the following XML schema declaration can be used to define a content model comprising any element coming from the XHTML The Extensible HyperText Markup Language recommendation.

```
<complexType name="xhtmlContent">
      <any namespace="http://www.w3.org/1999/xhtml"
      minOccurs="0"
      maxOccurs="unbounded"
      processContents="skip"/>
</complexType>
```

6.3.4 Implementing annotations

Annotations (expressed in GMT with an `<annot>` element) must be implemented in the same way as information units that are attached to the structural skeleton of a given TML except that only the Element and TypedElement styles may be used for this purpose.

--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

6.3.5 Implementing brackets

Bracketed information units (expressed in GMT with a <brack> element) must be implemented by providing the name of an element which is associated to the main information unit of the group and whose content is the set of the realizations of the information units of the group.

6.3.6 Namespaces

The DCS may comprise the description of an XML namespace that references the various XML objects (element or attributes) resulting from the specification of an actual TML. This description becomes mandatory when additional markup modules are associated with the TML.

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex A
(normative)

XML schema of the GMT format

This Annex comprises the specification of the GMT format using the XML Schema Part 2: Datatypes syntax. This schema shall be used as a reference to check the conformity of any data represented in GMT in the case it does not contain any additional markup module. In any other case, the schema shall be modified to incorporate the definition of the namespaces to be associated with the external markup to be used.

A description in HTML format of the elements and complexTypes is provided in the following file: Schema gmt.html

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified">
      <xs:complexType name="annotType">
            <xs:simpleContent>
                  <xs:restriction base="xs:string">
                        <xs:attribute name="type" type="xs:string"/>
                        <xs:attribute name="target" type="xs:string"/>
                  </xs:restriction>
            </xs:simpleContent>
      </xs:complexType>
      <xs:complexType name="brackType">
            <xs:sequence>
                  <xs:element name="feat" type="featType"/>
                  <xs:choice maxOccurs="unbounded">
                        <xs:element name="feat" type="featType"/>
                        <xs:element name="brack" type="brackType"/>
                  </xs:choice>
            </xs:sequence>
            <xs:attribute name="source" type="xs:string"/>
      </xs:complexType>
      <xs:complexType name="featType" mixed="true">
            <xs:choice minOccurs="0" maxOccurs="unbounded">
                  <xs:element name="annot" type="annotType"/>
            </xs:choice>
            <xs:attribute name="type" type="xs:string" use="required"/>
            <xs:attribute name="target" type="xs:string" use="required"/>
            <xs:attribute name="source" type="xs:string"/>
      </xs:complexType>
      <xs:complexType name="structType">
            <xs:sequence>
                  <xs:choice minOccurs="0" maxOccurs="unbounded">
                        <xs:element name="feat" type="featType"/>
                        <xs:element name="brack" type="brackType"/>
                  </xs:choice>
                  <xs:element name="struct" type="structType" minOccurs="0"
                  maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:attribute name="type" use="required">
                  <xs:simpleType>
                        <xs:restriction base="xs:NMTOKEN">
                              <xs:enumeration value="TDC"/>
                              <xs:enumeration value="GI"/>
```


--`,,`,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

```
                              <xs:enumeration value="CI"/>
                              <xs:enumeration value="TE"/>
                              <xs:enumeration value="LS"/>
                              <xs:enumeration value="TS"/>
                              <xs:enumeration value="TCS"/>
                        </xs:restriction>
                  </xs:simpleType>
            </xs:attribute>
      </xs:complexType>
      <xs:element name="tmf">
            <xs:complexType>
                  <xs:sequence>
                        <xs:element name="struct" type="structType"/>
                  </xs:sequence>
            </xs:complexType>
      </xs:element>
</xs:schema>
```

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex B
(normative)

The MSC TML

B.1 Introduction

A TML is defined by specifying a set of data categories and an XML implementation of the meta-model elaborated in this International Standard combined with these data categories. (This meta-model is hereinafter simply referred to as “the meta-model”.) The TML called MSC (MARTIF with Specified Constraints) is defined by a specified set of data categories selected from ISO 12620, and which are provided in B.4.

MSC is designed to represent terminological data for the processes of analysis, dissemination and exchange of information from human-oriented terminological databases (termbases).

The data categories in MSC have been selected with the objective of supporting “blind” representation. A blind representation supports interpretation without consulting the data provider. For example, consider the data category *grammatical gender*. Suppose that a TML specifies that the possible values of this data category are *masculine*, *feminine* and *neuter*, with their normal meanings in grammars of Western European languages. An instance of the data category grammatical *gender* with the value *feminine* can be interpreted without consulting the data provider. All information necessary for the interpretation of this data category is in the specification of the TML. On the other hand, suppose that another TML does not specify the possible values of the data category *grammatical gender*. Instead, these values are to be determined separately by each data provider. A user who receives a terminological data collection including the data category *grammatical gender* with the value *356* will not be able to interpret this data category without consulting the data provider concerning the meaning of the value *356*. Blindness is not an absolute property of a format but is instead a matter of degree.

The implementation of the meta-model in MSC is based on ISO 12200, which is commonly called MARTIF. Each implementation of the meta-model specifies expansion trees for every structural node of the meta-model and, for each data category, an XML style and vocabulary. For example, the structural node *language section* is expanded in MSC as the XML element `<langSet>`. And the data category *definition* uses the TypedElement style with the vocabulary `descrip` for the tag name and `definition` for the value of the `type` attribute (`<descrip type="definition">...</descrip>`). This implementation has been chosen in order to support multiple modes of compliance checking. All TMLs support the mode of compliance checking that uses a comprehensive XML schema for the TML as input to a general-purpose XML parser. This mode produces error messages suitable for an expert user of MSC who is thoroughly familiar with both XML and the comprehensive MSC schema. The extensive use in MSC of TypedElement styles also makes it possible to report MSC compliance of an XML document using a two-stage rather than a single-stage mode. In the first stage, an MSC XML document is checked against the proper use of typed elements and structural nodes; in the second stage, the document is checked against specific data categories. This alternative mode of compliance checking supports the generation of error messages that are more readily understandable to a terminologist with a limited familiarity with XML and no experience in reading XML schemas.

Any number of additional TMLs, besides MSC, can be defined using the same meta-model implementation as MSC together with a different selection of data categories. Any given set can be either more limited than the data categories of MSC (i.e. constitute a subset of MSC) or more extensive (i.e. constitute a superset), depending on the application. A terminologist-friendly compliance checker can be designed to adjust automatically to the particular set of data categories chosen for that TML.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.2 An example of an MSC XML document

The following is an example of a simple but complete MSC document. This sample MSC entry has several properties:

a) it corresponds directly to the meta-model in this International Standard;

b) it is well-formed and core-structure-valid;

c) it adheres to the default MSC extensible constraint specification (XCS).

The numbers at the left are line numbers. They are not part of the MSC document but serve as identifiers for the comments which follow the MSC document.

```
1  <?xml version='1.0'?>
2  <!DOCTYPE martif SYSTEM "./MSCcoreStructureDTD-v-1-0.DTD.TXT">
3  <martif type='MSC' xml:lang='en' >
4      <martifHeader>
5            <fileDesc><sourceDesc><p>from an Oracle corporation termBase</p>
             </sourceDesc></fileDesc>
6            <encodingDesc><p type='DCSName'>MSCdefaultXCS-v-1-0.XML</p>
             </encodingDesc>
7      </martifHeader>
8      <text> <body>
9         <termEntry id='eid-Oracle-67'>
10             <descrip type='subjectField'>manufacturing</descrip>
11             <descrip type='definition'>A value between 0 and 1 used in ...
               </descrip>
12             <langSet xml:lang='en'>
13                  <tig>
15                       <term tid='tid-Oracle-67-en1'>alpha smoothing factor
                         </term>
15                       <termNote type='termType'>fullForm</termNote>
16                  </tig>
17             </langSet>
18             <langSet xml:lang='hu'>
19                  <tig>
20                       <term tid='tid-Oracle-67-hu1'>
                         Alfa simítási tényezõ</term>
21                  </tig>
22             </langSet>
23        </termEntry>
24      </body> </text>
25  </martif>
```

Only a minimal acquaintance with XML is assumed in the following explanation. Indeed, an acquaintance with HTML from building simple web pages, along with the knowledge that XML allows user-defined tag names whereas HTML comes with a set of pre-defined tag names, should be sufficient to allow an understanding of the following explanation. For key MSC elements, the correspondence to the structural component of the meta-model in this International Standard is indicated.

Lines 1 and 2 `<?xml` …: These lines state that the following lines constitute an XML document that conforms to version 1.0 of the definition of XML by the World Wide Web consortium (W3C) and to the MSC DTD.

Line 3 `<martif` …: This line states that this particular XML document is an MSC document and thus can be validated against a specification of the MSC core structure which, for this document, is an XML DTD called MSCdefaultXCS-v-1-0.XML and against the XCS file referred to in the encodingDescription element (line 6). Alternatively, the core structure can be validated against a schema version of the description of the core structure. The `xml:lang` attribute indicates that the default language for text in this document is English (ISO 639 code 'en'). The `xml:lang` attribute can take an ISO 639 code as its value but can also take a two-part value, e.g. “fr-CA” for Canadian French.

--`,,`,,-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Lines 4 to 7 `<martifHeader` …: These lines provide global information about the collection: specifically, a file description indicating that the example was derived from an entry in a termbase used at Oracle corporation and that the MSC XCS is being used.

Line 8 `<text> <body>`: The text element surrounds the body element, which contains the collection of concept-oriented "TE" (`<termEntry>`) elements, and, optionally, `<front>` and `<back>` elements.

Line 9 `<termEntry` …: Each `termEntry` element is one instance of the "TE" object class as illustrated by the data model in this International Standard. The `id` attribute has a value that is unique throughout the document, making it possible for other elements to point unambiguously to this element. The `id` 'eid-Oracle-67' consists of the following information: eid [entry identifier] + the name of the database [Oracle] + the serial number of the entry [67].

Line 10 `<descrip type='subjectField'` …: The subject field data category is authorized by the XCS (Data Constraint Specification) mentioned above. It consists of a meta-data category element (`descrip`) with the specific data category indicated in the value of the `type` attribute.

Line 11 `<descrip type='definition'` …: This piece of descriptive information is also associated with the concept.

Line 12 `<langSet lang='en'>`: The `<langSet>` element corresponds to a "LS" object class, according to which a TE consists of associated information and LSs. This line begins the English LS.

Lines 13 and 14 `<tig> <term>` …: The meta-model in this International Standard states that a LS consists of instances of a "TS" object class which, in MSC, corresponds to a `<tig>` (or `<ntig>`) element. An instance of a TS consists of a term and associated information, which in this case is the `termType`. The name `tig` stands for **t**erm **i**nformation **g**roup. The `id` 'tid-Oracle-67-en1' consists of the following information: eid [entry identifier] + the name of the database [Oracle] + the serial number of the entry [67] + the language code [en] + the serial number of the `tig` within that language group [1].

Line 15 `<termNote type='termType'` …: This piece of information associated with the term is the ISO 12620 data category "term type". Its value in this case is "fullForm". A `<termNote>` tag is used since the information is closely associated with the term itself rather than with the concept being described.

Line 16 `</tig>`: This element simply ends the current TS.

Line 17 `</langSet>`: This element ends the English LS.

Line 18 `<langSet lang='hu'>`: This element begins the Hungarian LS.

Lines 19 to 21 `<tig>` ...: These lines consist of a TS with a Hungarian term but no definition and no explicit term type. Each character of the term that is not found in ISO/IEC 646 is represented as a hex character reference corresponding directly to a Unicode character. The actual Hungarian term is "Alfa simítási tényezõ". Note that the final character "õ" (o-tilde) should more properly be an o-double-acute, which is represented by the following Unicode hex character reference: "ő", a character not available in a typical Latin 1 font. In XML, a Unicode hex character reference consists of "&#x" + four hex digits from the Unicode standard + a semicolon.

Line 22 `</langSet>`: This element ends the Hungarian LS.

Line 23 `</termEntry>`: This element ends the current TE.

Line 24 `</body> </text>`: These elements end the set of TEs, which in this case consist of only one entry, and the MSC text element, which is the composite of TEs and other resources called CI in the meta-model (see 5.3). In this MSC document, there are no resources outside the TE. If there were, they would be in the MSC element `<back>` or `<front>`.

Line 25 `</martif>`: This element ends the entire MSC document.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.3 Expansion trees

The expansion trees are illustrated in Figure B.1.

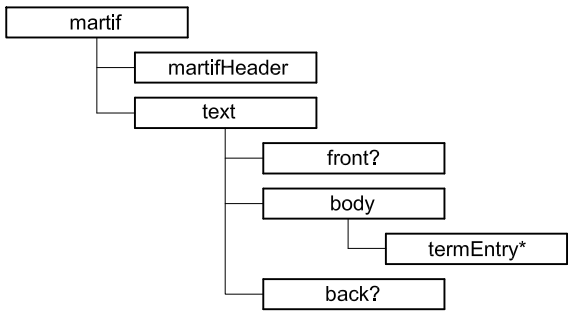
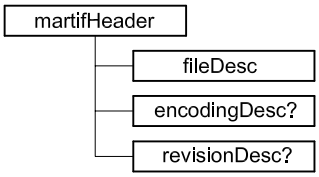
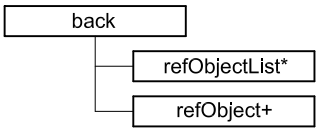
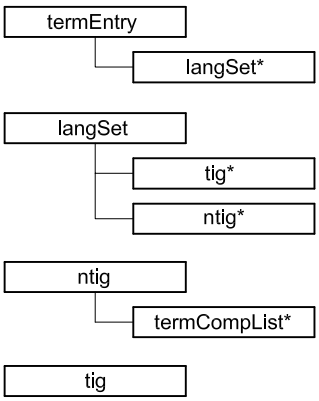
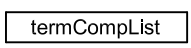
Level	Structure	Comments
TDC		The following codes apply: ? occurs 0 or 1 times * repeatable element + occurs at least once
GI		There are more elements which are descendants of fileDesc, encodingDesc and revisionDesc which cannot be shown because of the limited space
CI		The type values for refObject are expressed in the XCS file but are not data categories included in ISO 12620
TE LS TS TS		The element <ntig> differs from <tig> in that it allows for a termComponent level. In this way <ntig>s are more robust than <tig>s
TCS		

Figure B.1 — Expansion trees

B.4 Data categories

The MSC data categories in HTML format are provided in the following file: MSC data categories.html

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

B.5 Conclusion

MSC qualifies as a TML in compliance with this International Standard. Furthermore, related formats that are also TMLs can be defined simply by selecting a different set of data categories, without varying the XML expansion trees, styles and vocabularies. Hence the TMLs thus defined share a common core structure. As a result, MSC forms the basis for the MSC-type subset of all possible TMLs. This subset constitutes a family of related formats, some of which do not involve blind interchange, which can in many cases be processed by a single software tool.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex C
(normative)

The Geneter TML

C.1 Introduction

This Annex specifies the characteristics of the Geneter TML that can be realized using the TMF as defined in this International Standard.

Geneter provides an XML-based implementation of the “meta-model” as defined by this International Standard for TDCs.

Geneter is a format that describes data categories and their relationships in a TDC. It is “generic” in as far as it takes into account data categories that have been defined in ISO 12620 (data categories) or identified in real applications.

More restricted subsets corresponding to the needs of specific groups of users can be derived from this format. These subsets can be “closed” so that they can allow for “blind” interchange. A “Geneter subset” shall comply with the “subsetting rules” as defined in C.6.2.

To be “Geneter conformant” a Geneter instance shall conform to the Geneter format or to a Geneter subset. A conformant instance shall be a valid XML document as defined in Extensible Markup Language (XML) (“An XML document is valid if it has an associated document type declaration and if the document complies with the constraints expressed in it.”)

This Annex provides an example illustrating the conformance of the Geneter TML to the TMF defined in this International Standard (see C.2). The Geneter implementation of each meta-model level is specified in the following subclauses: GI (C.3); TE (C.4); CI (C.5). C.6 deals with methods for restricting or extending the Geneter format.

C.2 Example: specification of a Geneter subset as a TML

A Geneter subset corresponding to a specific data structure related to user needs is specified in C.6.4. The resulting model, an encoded example and the vocabulary and style specifications of this TML are illustrated in the following paragraphs and Table C.1.

The following model is derived from the Geneter subset given in C.6.4

```
<!-- structure definition -->
<!ELEMENT geneter (terminologicalEntry+) >
<!ATTLIST geneter profile (oracle) 'oracle'>
<!ELEMENT terminologicalEntry (SubjectField, Definition, languageCtn+) >
<!ATTLIST terminologicalEntry

     identifier CDATA #IMPLIED>

<!ELEMENT languageCtn (Term*)>
<!ATTLIST languageCtn value (en|hu) #IMPLIED>
<!-- data categories definition -->
<!ELEMENT SubjectField (#PCDATA)>
<!ELEMENT Definition (#PCDATA)>
<!ATTLIST Definition xml:lang (en) 'en' >
```

--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<!ELEMENT Term (#PCDATA) >
<!ATTLIST Term formType (fullForm|abbreviation) #IMPLIED
               workingStatus (working|consolidated) #REQUIRED>
```

The XML encoding of an entry is as follows.

```
<?xml version ="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE geneter SYSTEM 'oracle.dtd' []> <geneterprofile = "oracle">

     <terminologicalEntry identifier = 'ID67' workingStatus='consolidated'>
           <SubjectField>Manufacturing</SubjectField>
           <Definition xml:lang ='en'>A value between 0 and 1 used in ...
           </Definition>
           <languageCtn value='en'>
                 <Term formType='fullForm'>alpha smoothing</Term>
           </languageCtn>
           <languageCtn value='hu'>
                 <Term> Alfa simítási tényezõ</Term>
            </languageCtn>
    </terminologicalEntry>

</geneter>
```

Table C.1 — Geneter style and vocabulary

ISO 12620:1999	Style	Geneter element
entry type		terminologicalEntry
entry identifier	**Level**: TE **Type**: attribute **Anchor information** : terminologicalEntry **Value**: txt	terminologicalEntry identifier
subject field	**Level**: TE **Type**: element **Value**: txt	SubjectField
definition	**Level**: TE **Type**: element **Value**: txt	Definition
?	**Level**: TE **Type**: attribute **Anchor information**: Definition **Value**: txt	xml:lang
?		languageCtn
language identifier	**Level**: LS **Type**: attribute **Anchor information**: languageCtn **Value**: txt	languageCtn value
term	**Level**: TS **Type**: element **Value**: txt	Term
term related information	**Level**: TS **Type**: attribute **Anchor information**: Term **Value**: picklist	Term formType
element working status	**Level**: TS **Type**: attribute **Anchor information**: Term **Value**: picklist	Term workingStatus

--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

C.3 GI

The GI of a Geneter TDC contains information (meta-data) about the collection. It is represented by a `<header>` element. Meta-data are defined by two namespaces according to the Dublin Core Qualifiers.

The namespaces declaration for a Geneter header is a follows:

```
xmlns:dc="http://purl.oclc.org/dc#"xmlns:dcq="http://purl.org/dc/qualifiers/1.0/"
```

An example of a Geneter header is

```
<header>

      <meta name = "DC.Type" scheme = "DCMIType" content = "Dataset">
      <meta name = "DC.Date.Issued" scheme = "ANSI.X3.X30-
      1985" content = "20011511">

</header>
```

C.4 TE

C.4.1 General

A TE consists of data categories and containers for further data categories.

C.4.2 Data category types

The Geneter format is based on three types of data category as follows.

a) *Structural* data categories (for instance, `<Definition>`, `<Term>` or `<PartOfSpeech>`) used to describe the terminological information. They are listed with their name, attributes and content model in column 2 of the synopsis given in C.4.10.1.

b) *Embedded* data categories used within the content of a structural data category to incorporate terminology related information (see Table C.2). Their content model is `%Line;` (see C.4.6.2 for mixed content model types).

Table C.2 — Embedded elements

Name	Attributes	Explanation
Annotation	**Type**: CDATA **Scheme**: CDATA **Value**: CDATA %GeneralAttributes;	Additional linguistic information
Characteristic	**Type**: CDATA %GeneralAttributes;	ISO 12620:1999, A.5.8
EntailedTerm	**Type**: CDATA %GeneralAttributes;	ISO 12620:1999, A.10.6.1
ForeignText	**Type**: CDATA **Language**: CDATA %GeneralAttributes;	ISO 12620:1999, A.10.8
Keyword	%GeneralAttributes;	ISO 12620:1999, A.9.4

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

c) *Basic* (non-specific) elements used within the content of a structural data category or of an embedded data category (see Table C.3). (The `<Ptr>` element is described in C.4.7).

Table C.3 — Basic elements

Name	Attributes	Content model	Explanation
Data	%GeneralAttributes;	#PCDATA	Simple data category for basic text
Date	**Type:** CDATA **Scheme:** CDATA **Calendar:** CDATA %GeneralAttributes;	CDATA	ISO 12620:1999, A.10.2.1
Fpi	**Type**: CDATA %GeneralAttributes;	%URI;	ISO 12620:1999, A.10.21.2
Quantity	**Type**: CDATA **Value:** CDATA **Unit:** CDATA %GeneralAttributes;	EMPTY	
Range	**Type**: CDATA **Min**: CDATA **Max**: CDATA %GeneralAttributes;	EMPTY	ISO 12620:1999, A.5.7
Segment	**Type**: CDATA %GeneralAttributes;	<seg>+ (CDATA)	Segmented element (hyphenation, syllabification,...)
Unit	**Type**: CDATA %GeneralAttributes;	%basicLine;	ISO 12620:1999, A.5.6
Url	**Type**: CDATA %GeneralAttributes;	%URI;	ISO 12620:1999, A.10.21.1
Where	**Type**: CDATA %GeneralAttributes;	%basicLine;	A location indicator
Who	**Type**: CDATA **Role**: CDATA %GeneralAttributes;	%basicLine;	Person or corporate body involved in some action

The following is an example showing the three types of data category.

```
<Definition>A liquid with a <Characteristic>boiling temperature</Characteristic>
of <Quantity value="100", Unit = "Celsius"/></Definition>
```

C.4.3 Characteristics of a Geneter data category: name, attributes and content model

For each data category, the Geneter format specifies its *name* (an XML generic identifier), its *attributes* and its *content model*.

For practical reasons, structural data categories are grouped into blocks corresponding respectively to

— data *function* (administrative information, description of a property, description of a relation), and

— data *position* in the tree structure of a TE (corresponding to the four levels TE, LS, TS and TCS of the meta-model).

For instance, at the TE level (language independent section), elements are grouped in the `%lisAdminDatCat;`, `%lisPropDatCat;` and `%lisRelDatCat;` blocks. The prefixes for the other levels are `lds` (LS), `term` (TS) and `component` (TCS).

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

C.4.4 General attributes

The attributes for any data category are

— the core attributes (`id`, `class`, `style`, `title`) and the internationalization attributes (`xml:lang`, `dir`) defined in XHTML1, and

— specific attributes such as `security` or `workingStatus` corresponding respectively to ISO 12620 `element Working Status` (ISO 12620:1999, 10.11) and `security subset` (ISO 12620:1999, 10.3.9).

C.4.5 Container

A container is a structure used to refine a data category (i.e. to supply additional information). A container begins with the data category to be refined. This data category is unique and compulsory. The name of a container is formed by adding the suffix `Ctn` to the name of this data category as shown in the following examples.

Example of refinement by the `<Source>`:

```
<termCtn>

      <Term>barbed wire</Term>
      <Source>source</Source>

</termCtn>
```

Example of container for an empty element:

```
<registerCtn value='neuter'>

      <Note>...</Note>

</registerCtn
```

Containers can be embedded:

```
<termCtn>

      <Term>barbed wire</Term>
      <sourceCtn>
            <Source>source</Source>
            <Note>note on source</Note>
      </sourceCtn>

</termCtn>
```

A data category can be recursive (i.e. self-refining):

```
<subjectFieldCtn>

      <SubjectField>chemistry</SubjectField>
      <subjectFieldCtn>
            <SubjectField>petrol</SubjectField>
      </subjectFieldCtn>

</subjectFieldCtn>
```

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

C.4.6 Content models

C.4.6.1 General

The content model of an element can be expressed as either mixed content or composite content. These models are described in C.4.6.2 and C.4.6.3 respectively.

C.4.6.2 Mixed content

There are four kinds of mixed content named respectively `%Flow;`, `%Inline;`, `%Line;` and `%basicLine;`.

The first two are defined as an extension of XHTML 1.0 using Modularization of XHTML, the `%inline;` entity of the HTML "Text Module" is redefined to incorporate embedded data categories (`%embeddedDataCategories;`) and basic data categories (`%basicDataCategories;`). So that they can occur anywhere, the HTML entity `%inline`; is used as shown in the following examples.

Original XHTML `%inline` declaration:

```
<!ENTITY % inline "a | %special; | %fontstyle; | %phrase; | %inline.forms;">
```

Modified XHTML `%inline` declaration:

```
<!ENTITY % inline "a | %special; | %fontstyle; | %phrase;| Data | Ptr |
%embeddedDataCategories; | %basicDataCategories;">
```

The original `%inline;` entity augmented with all the basic elements is renamed as `%Line;`. It is used as a content model for embedded elements.

The original `%inline;` entity is renamed as `%basicLine;`. It is used as a content model for basic elements.

The following is an example of enriched mixed content (with an embedded data category in a table):

```
<Example>

      boiling points

      <table><tr><td>
            <Characteristic value= "100" unit = "Celsius" />
      </td></tr></table>

</Example>
```

C.4.6.3 Composite content

Composite content is created using an ordered set of enumerated elements. These elements are either structural data categories (in the case of recursivity for instance) or basic data categories.

Example of a composite element:

```
<Importation>

      <Date>01-02-2001</Date>

      <Where>EUROTERMS</Where>

</Importation>
```


Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

C.4.7 Pointers

Relations between a TE and CI (see C.5) are expressed by the <Ptr> element. This element conforms to the XML Linking Language (XLink).

The following is an example of a pointer using the <Ptr> element:

```
<Owner>

      <Ptr xml:lang = 'en' xlink:embed = 'none' xlink:href = 'person01'> BROWN, J
      </Ptr>

</Owner>
```

C.4.8 Structure of a TE

The structure of a Geneter TE consists of three embedded containers (languageCtn, termCtn and termComponentCtn). Their correspondence with the meta-model is shown in Table C.4.

Table C.4 — Correspondence between TMF anchor levels and Geneter

Meta-model	Geneter
TE	terminologicalEntry
LS	languageCtn
TS	termCtn
TCS	termComponentCtn

C.4.9 Example of TE

The following instance is explained in detail in Table C.5.

```
<terminologicalEntry identifier='07'>

      <Owner><Ptr xlink:href = 'person01'>xxx</Ptr></Owner>

      <languageCtn value ='en'>
            <Definition>wire with short, sharp points on it</Definition>
            <termCtn>
                  <Term>barbed wire</Term>
                  <componentCtn rank = 1>
                        <Word>barbed</Word>
                        <PartOfSpeech>adj</PartOfSpeech>
                  </componentCtn>
            </termCtn>
      </languageCtn>

</terminologicalEntry>
```

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Table C.5 — Detailed explanation

Geneter encoding	Explanation	Level
`<terminologicalEntry identifier='TE07'>`	beginning of the TE with an attribute for the identifier	TE
`<Owner><Ptr xlink:href = 'person01'>xxx</Ptr></Owner>`	data category `<Owner>` with a link towards the description of a person	
`<languageCtn value ='en'>`	container for a LS with a language attribute	LS
`<Definition>wire with short, sharp points on it</Definition>`	data category `<Definition>`, content = "wire with ... "	
`<termCtn>`	container for the description of a term and its complements	TS
`<Term>barbed wire</Term>`	data category `<Term>`, content = "barbed wire"	
`<componentCtn>`	container for a Component group with an attribute `rank` indicating the position of the component inside the term	TCS
`<Word>barbed</Word>`	data category `<Word>`, content = "barbed"	
`<PartOfSpeech>adj</PartOfSpeech>`	data category `<PartOfSpeech>`, content = "adj"	
`</componentCtn>`	end of the component container	
`</termCtn>`	end of the term container	
`</languageCtn>`	end of the language container	
`</terminologicalEntry>`	end of the TE	

C.4.10 The tree structure of a terminological entry

C.4.10.1 Geneter synopsis

The tree given in the following HTML file represents the Geneter name, attributes, content model, position of any data category in the Geneter structure as well as the ISO 12620 position from which it is derived: Geneter synopsis.html.

Non ISO 12620 elements and entities (the name given to repetitive information) are defined in C.4.10.2.

C.4.10.2 Non ISO 12620 data categories

The non ISO 12620 data categories listed in the Geneter synopsis in C.4.10.1, fifth column [e.g. (1), (2), etc.] are explained below.

(1) Contributor = Any person or organization having a role in the production of the item.
(2) Coverage = The extent or scope of the content of the resource. Coverage will typically include spatial location.
(3) LastModification = Responsibility and date of the last modification of data.
(4) SourceLanguage = In an entry, the language in which a concept has been designated originally.
(5) TargetLanguage = In an entry, the languages in which equivalent designations are provided.
(6) Scope = Further indications about the field of application of a concept.
(7) CausalRelation = Associative relation between a cause and its effect. [ISO 1087-1:2000, definition, 3.2.26].
(8) RelatedDescription = Link with a non terminological description of a term (dictionary, lexicon) or a concept (thesaurus, ontology).
(9) Free = see C.6.5.
(10) FreeVal = see C.6.5.
(11) languageCtn = A container describing a concept in one language.
(12) ExternalLanguageSection = A language container located on a remote device.
(13) Derivation = Process of new word formation through the modification (addition, deletion or remplacement) of a morpheme (suffix) or a stem (root).
(14) Inflection = Modification of a word with elements that express some grammatical aspects and relations.
(15) SyntacticalFunction = Function of a term or a word in the relationships between linguistic units or in the grammatical construction.

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

(16) TermComplement = Ancillary part of a term (the “to” preposition for an English verb for instance).
(17) TermDisplay = A displayable or printable form of a term (including embedded grammatical information for instance).
(18) Homonym = Terms having an identical pronunciation and/or spelling but referring to different concepts.
(19) Homophone = Terms having an identical pronunciation but different spellings and referring to different concepts.
(20) Polysemy = Characteristic of a sign that has several contents, several values and several meanings.

C.4.10.3 Entities for content models

The entities for the content models are listed below. The references in square brackets contain an ISO 12620 position or a short explanation.

`%act;` 'date?, who*' [date and responsibility of a transaction]

`%adminAgent;` 'BusinessUnit | businessUnitCtn | Contributor | contributorCtn | Customer | customerCtn | Owner | ownerCtn' [administrative information about persons or organizations]

`%adminItem;` 'Project | projectCtn | Product | productCtn | Application | applicationCtn | Environment | environmentCtn' [administrative information about applications]

`%cpt;` (Note | noteCtn | Source | sourceCtn)* [complements to a data category inside a container]

`%free;` (Free | freeCtn)* [used for negotiated interchange of extra data categories]

`%URI;` [content type for a Uniform Resource Locator (http://www.w3.org/TR/uri-clarification)]

C.4.10.4 Entities for suggested picklists

The entities for suggested picklists are listed below. The references in square brackets contain an ISO 12620 position or a short explanation.

`%AnimacyValue;` 'animate | inanimate' [ISO 12620:1999, A.2.2.4]

`%AntonymType;` 'antonymComplement | antonymContrast' [ISO 12620:1999, A.10.18.6]

`%CausalRelationType;` 'cause | consequence' [causal relation between concepts]

`%ComplementType;` 'ante | pos' [ancillary part of a term]

`%ContextType;` 'definingContext | explanatoryContext | associativeContext | linguisticContext | metalinguisticContext' [ISO 12620:1999, A.5.3]

`%ContributorRole;` 'expert | proposer' [role of contributor with respect to a work]

`%DefinitionType;` 'intensionalDefinition | extensionalDefinition | partitiveDefinition' [ISO 12620:1999, A.5.1]

`%DegreeOfEquivalenceDirectionality;` 'bidirectional | monodirectional' [ISO 12620:1999, A.3.3]

`%DegreeOfEquivalenceValue;` 'narrower | equivalent | quasiEquivalent | broader | nonEquivalent' [ISO 12620:1999, A.3.1]

`%DegreeOfSynonymyValue;` 'narrower | synonymous | quasiSynonymous | broader | nonsynonymous' [ISO 12620:1999, A.2.10]

`%DerivationType;` 'regressive | learned | improper'

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

%FormOfTermType;	'fullForm | abbreviation | shortFormOfTerm | initialism | acronym | clippedTerm' [ISO 12620:1999, A.2.1.7, A.2.1.8]
%FrequencyValue;	'commonly | infrequently | rarely' [ISO 12620:1999, A.2.3.4]
%GenericRelationType;	'superordinateConcept | subordinateConcept | coordinateConcept' [ISO 12620:1999, A.6.1]
%GeographicalUsageType;	'used | nonUsed' [ISO 12620:1999, A.2.3.2]
%GrammaticalGenderValue;	'masculine | feminine | neuter' [ISO 12620:1999, A.2.2.2]
%GrammaticalNumberValue;	'singular | plural | dual | massNoun' [ISO 12620:1999, A.2.2.3]
%IllustrationMediaType;	'image | audio | video' [ISO 12620:1999, A.5.5]
%IllustrationType;	'symbol | formula | equation | logicalExpression | figure' [ISO 12620:1999, A.2.1.13 to A.2.1.16]
%InflectionType;	'root | verbal | nominal | pronominal' [type of modification of a word]
%LanguagePlanningQualifier;	'recommendedTerm | nonstandardizedTerm | proposedTerm | newTerm' [ISO 12620:1999, A.2.9.2]
%NormativeAuthorizationValue;	'standardizedTerm | preferredTerm | admittedTerm | deprecatedTerm | prohibitedTerm | superseded Term | legalTerm | regulatedTerm' [ISO 12620:1999, A.2.9.1]
%NoteType;	'linguisticNote | technicalNote | userNote | workingNote | transferComment' [ISO 12620:1999, A.8]
%PartitiveRelationType;	'broaderConcept | narrowerConcept' [ISO 12620:1999, A.6.2]
%ProcessStatusValue;	'unprocessed | provisionallyProcessed | finalized' [ISO 12620:1999, A.2.9.4]
%ProprietaryRestrictionValue;	'trademark | tradeName' [ISO 12620:1999, A.2.3.7]
%RegisterValue;	'neutral | technical | benchLevel | slang | vulgar | familiar' [ISO 12620:1999, A.2.3.3]
%RelatedDescriptionList;	'ontology | thesaurus | documentaryLanguage | dictionary | lexicon | translationMemoryData' [non terminological description of terms or concepts]
%ResponsibilityType;	'person | corporateBody'
%SpatialRelationType;	'backward | forward | contiguous' [spatial relation between concepts]
%SubjectFieldType;	'classificationNumber | indexHeading' [ISO 12620:1999, A.4]
%TemporalQualifierValue;	'archaicTerm | outdatedTerm | obsoleteTerm' [ISO 12620:1999, A.2.3.5]
%TemporalRelationType;	'Preceding | Succeeding | Coincident' [temporal relation between concepts]
%TermDesignationType;	'term | formula | symbol | equation | logicalExpression' [ISO 12620:1999, A.2.1.13 to A.2.1.16]

--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

%TermDegreeOfSynonymy;	'narrower | broader' [ISO 12620:1999, A.2.10]
%TermFormType;	'fullForm | abbreviation | shortFormOfTerm | initialism | acronym | clippedTerm' [ISO 12620:1999, A.2.1.7, A.2.1.8]
%terminologicalEntryType;	'conceptEntry | standardizedEntry | collocation | phrase | setPhrase | standardText | synonymousPhrase | neologism | geographicalName | commonName | properName | collectiveName | officialDenomination | parallelSegment | managementUnit | partNumber' [ISO 12620:1999, A.10.10]
%TermLayout;	'main | secondary' [administrative status of a term]
%TermProvenanceType;	'transdisciplinaryBorrowing | translingualBorrowing | loanTranslation | shiftInMeaning' [ISO 12620:1999, A.2.4.1]
%TermStatus;	'neologism | wordCreation | foreignDesignation' [status of a new term]
%TermType;	'collocation | formula | phrase | setPhrase | standardText | synonymousPhrase | internationalism | internationalScientificTerm | geographicalName | commonName | properName | collectiveName | officialDenomination | managementUnit | partNumber' [ISO 12620:1999, A.2]
%TermVariantType;	'orthographical | grammatical' [ISO 12620:1999, A.2.1.9]
%TransScript;	'transcribedForm | transliteratedForm | romanizedForm' [ISO 12620:1999, A.2.1.10 to A.2.1.12]
%VariantDirectionality;	'isVariantOf | hasForVariant' [directionality for variants]

C.5 CI

C.5.1 Geneter CI types

The Geneter CI types are as follows:

— bibliographical information based on ISO 690 (<monographEntry>, <partOfMonographEntry>, <contributionEntry>, <serialEntry>, <articleEntry>, <patentEntry>, <standardEntry>, <audioVideoEntry>);

— bibliographical information based on ISO 690-2 for electronic documents (<eMonographEntry>, <ePartOfMonographEntry>, <eContributionEntry>, <eSerialEntry>, <eArticleEntry>, <eElectronicBoardEntry>, <eMessageEntry>);

— bibliographical information based on ISO 12083 (<refDocEntry>);

— description of persons (<person>) based on ISO 12083 bibliographic description;

— description of corporate bodies (<corporateBody>) based on ISO 12083 bibliographic description;

— description of thesaurus based on ISO 2788 and ISO 5964 (<thesaurus>);

— description of machine readable dictionaries based on ISO 1951;

— description of ontologies based on a specialization of the Ontology Inference Layer;

— XHTML documents (<xymlPage>);

— transitory language containers for exchanging information about one concept in one language (<transitoryLanguageCtn>);

— collating sequences [ISO 12620:1999, A.10.9];

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

— encoded binary data for exchanging image or sound or any other non XML document (doc, pdf, html, ...) (<encodedFile>);

— other objects (<freeObject>).

C.5.2 Mechanism for extending CI

By using XML namespaces, other types of linguistic description can be included in a Geneter collection. This mechanism can be used to manage lexicons (OLIF format), parallel segments for machine translation (TMX and XLIFF) and specialized ontologies (OIL). The <RelatedDescription> element links terminological entries with these descriptions.

C.6 Geneter restriction and extension

C.6.1 Creation of subsets

C.6.1.1 For particular needs, it is possible to create subsets based on the Geneter format. Any instance of a Geneter subset must be valid against the Geneter DTD. A Geneter subset must have a required "profile" attribute giving the Uniform Resource Locator of the subset model. To be compatible with the general model, a subset must comply with the general rules of XML specified in C.6.1.2 and C.6.1.3

C.6.1.2 For data elements the following rules apply:

a) any element which has an occurrence indicator ? or * can be deleted;

b) any element occurrence indicator (?, *, +) can be deleted;

c) when two elements are combined by OR connector " | " in a content model, one of the two elements can be deleted;

d) the occurrence indicator * can be replaced by the occurrence indicators ? or +.

C.6.1.3 For attributes the following rules apply:

a) the attributes whose default value is not the key word #REQUIRED can be deleted;

b) when the attribute value comes from an enumerated list, the list can be reduced but it shall contain at least one value;

c) when the attribute value is CDATA type, CDATA can be replaced by an enumerated list.

C.6.2 Different types of subset

A subset which contains neither an element nor an attribute free is a "strict" subset of Geneter.

If a subset contains <Free> elements whose type and value are literal or taken from enumerated lists, the subset is "closed". Such a subset could be called a "jargon" of Geneter.

If a subset contains <Free> elements whose type and value is CDATA, the subset is "open".

C.6.3 Blind subset

By applying rule a) in C.6.1.2 to fuzzy data categories like <Grammar>, and by applying it to all the <Free> data categories and to the <free> content element, it is possible to design a more concise Geneter model for blind interchange purpose. The subset mentioned in C.6.2 is such a blind subset.

--`,,`,`-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

C.6.4 Building a subset: an example

C.6.4.1 This example is based on a flat source structure in which all the “fields” of data are delimited by a comma. In order to create a Geneter subset corresponding to the original structure and a Geneter instance of these data, the four steps are as follows:

— identification of the type of each element;

— mapping of each element to a Geneter position;

— design of a Geneter subset able to host these positions and no others;

— encoding of the sample.

C.6.4.2 The first and second steps of the analysis for the following data sample (flat source structure in which all the “fields” are delimited by a comma) are shown in Table C.6 and Table C.7 respectively.

```
67, Manufacturing,,Standard,alpha smoothing factor,Approved,A value between 0
and 1 used in statistical forecasting calculations for smoothing demand
fluctuations. ORACLE Inventory uses the factor to determine how much weight
to give to current demand when calculating a forecast.,Alfa simitási tényezõ
```

Table C.6 — First step: Identification of the type of each element

Data category	Data	ISO 12620:1999 correspondence
EntryNumber	67	entry identifier (A.10.15)
Domain	Manufacturing	subject field (A.4)
Product		product subset (A.10.3.5)
Datatype (a full form as opposed to an abbreviation)	Standard	term type (A.2.1)
English	alpha smoothing factor	term (A.1)
Status (an indication of the administrative status of the Hungarian term)	Approved	process status (A.2.9.4)
Definition	A value between 0 and 1 used in statistical forecasting calculations for smoothing demand fluctuations.	definition (A.5.1)
Hungarian term	Alfa simitási tényezõ	term (A.1)

Table C.7 — Second step: Mapping of each element to a Geneter position

Data category	Geneter equivalent from synopsis in C.4.10.1	
	Number	Element name
EntryNumber	1	terminologicalEntry (identifier attribute)
Domain	1.1.2.9	terminologicalEntry/SubjectField
Product	1.1.1.20	terminologicalEntry/Product
Datatype[a]	1.2.5.2	terminologicalEntry/languageCtn/Term (formType attribute)
English term	1.2.5.2	terminologicalEntry/languageCtn/Term
Status of the Hungarian term	1.2.5.2	terminologicalEntry/languageCtn/Term (Status attribute)
Definition[b]	1.1.2.1	terminologicalEntry/Definition
Hungarian term	1.2.5.2	terminologicalEntry/languageCtn/Term

[a] Datatype is a property, not a relation, so it is encoded as an attribute (formType).

[b] Definition has been put in the Language Independent Section because it applies to the whole entry.

C.6.4.3 The third and fourth steps, designing and encoding a subset, are given in C.2.

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

C.6.5 Geneter extensions and negotiated interchange

For specific needs, new data categories can be added to the Geneter model at each level of the structure or inside the content models. If XML validity is required for an interchange transaction, these elements must be transformed into the meta-data category `<Free>` or into a container `<freeCtn>` which are defined in the Geneter format. The negotiation process consists of exchanging the semantics of these free elements with the partner receiving the data.

For instance, an extension (in this case a structural data category for indicating the unit rate for a data item) can be defined in the Geneter model by the statement:

```
<!ELEMENT Rate (Quantity)>
```

This element (in this example `Rate`) has to be added at some level of the Geneter tree (the `%lisAdminDatCat;` block for instance because it is an administrative information characterizing the whole entry). A possible instance (i.e. extension for local management) of this element will be:

```
<Rate><Quantity value = "5" unit = "US Dollar"/></Rate>
```

For exchange purposes (i.e. extension for negotiated interchange) this extra element will be transformed as follows (by an XSLT style-sheet for instance):

```
<Free type = "Rate"><Quantity value = "5" unit = "US Dollar"/></Free>
```

This encoding is conformant to the Geneter definition of a `<Free>` element. It will validate against the Geneter model.

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

Annex D
(informative)

Conformance of terminological data to TMF

D.1 General

This Annex discusses how XML-based terminological data can be made conformant to TMF by analysing the structure and content of the data and performing certain transformations of these data. The end result of this analysis is the specification of a TML that both represents the terminological data without loss of information and is interoperable with other TMLs as specified in this International Standard.

D.2 Example terminological data

Consider the following example XML-based representation of a terminological entry from an automotive engineering terminology database.

```
<termBank>
    <tbid>00aa</tbid>
    <tbDescription>Automotive Engineering</tbDescription>
    <conceptEntry>
        <domainOfConcept>ABS</domainOfConcept>
        <conceptLastModified>21-08-2001</conceptLastModified>
        <termGroup>
            <languageCode>Deutsch</languageCode>
            <termDefinition> Bauteile, die die elektronischen Steuer- und
             Regelvorgänge für die Blockierregelung und die
             Antriebsschlupfregelung übernehmen.</termDefinition>
            <termString>ABS/ASR-Steuerung</termString>
            <usageDescriptors>
                  <usedIn>Germany</usedIn>
                  <usedIn>Switzerland</usedIn>
            </usageDescriptors>
            <wordClass>n</wordClass>
            <wordGender>f</wordGender>
            <termLastModified>21-08-2001</termLastModified>
        </termGroup>
        <termGroup>
            <languageCode>English</languageCode>
            <termString>ABS/ASR control</termString>
            <usageDescriptors>
                  <usedIn>Britain</usedIn>
            </usageDescriptors>
            <wordClass>n</wordClass>
            <termLastModified>20-08-2001</termLastModified>
        </termGroup>
    </conceptEntry>
</termBank>
```

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

D.3 Description of content of elements

Table D.1 describes the information contained in the example in D.2.

Table D.1 — Description of content of elements

XML element	Description	Description of content
<tbid>	Unique identifier of this terminology database	Alphanumeric code
<tbDescription>	Text describing this terminology database	Text
<domainOfConcept>	Subject field of this concept entry	Selected value related to concept
<conceptLastModified>	Date that information pertaining to this concept was last changed	Date
<languageCode>	Language in which the term is used	Value selected from ISO 639-1 represented in the language of the term
<termDefinition>	Definition of the term	Text
<termString>	The term itself	Text
<usedIn>	Country in which this term is used in this language	Value selected from ISO 3166-1 represented as an English text descriptor
<wordClass>	Grammatical class of the term	Typically noun represented by n
<wordGender>	Grammatical gender of the term	*Masculine* represented by m, *feminine* represented by f, or *neuter* represented by n
<termLastModified>	Date that information pertaining to this term was last changed	Date

Other XML elements represent containers for this information.

NOTE In the above example, the implication of the description of <languageCode> along with the text content of the <termString> and <termDefinition> elements means that the XML attribute xml:lang should be introduced into the markup to show, for example, that both the language code and the language used to represent this code is German; for example, <languageCode **xml:lang="de"**>Deutsch</languageCode>. The introduction of this attribute should occur at the topmost point at which it is required to override the value of xml:lang propagated from elements higher in the structure.

D.4 Conformance to TMF

D.4.1 Meta-model specification

By comparison of the XML outline of this example with the structural nodes of the meta-model, the degree of conformance to the meta-model can be evaluated. Table D.2 shows this comparison.

Table D.2 — Comparison of XML outline with structural nodes of meta-model

Meta-model identifier	Vocabulary
TDC	<termBank>
GI	
TE	<conceptEntry>
LS	<termGroup>
TS	
CI	

For this example, there is no equivalent to the TS. The TS can, however, be introduced without loss of information. The example contains no CI, while the GI can be created out of the <tbid> and <tbDescription> elements. The result of these alterations is shown below. Bold XML elements denote the

--`,,`,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
© ISO 2003 – All rights reserved
Not for Resale

structural nodes, with bold italics denoting newly introduced sections. The `xml:lang` attribute has also been added, in italics, where needed.

```
<termBank xml:lang="en">
    <globalInformation>
        <tbid>00aa</tbid>
        <tbDescription>Automotive Engineering</tbDescription>
    </globalInformation>
    <conceptEntry>
        <domainOfConcept>ABS</domainOfConcept>
        <conceptLastModified>21-08-2001</conceptLastModified>
        <termGroup xml:lang="de">
            <languageCode>Deutsch</languageCode>
            <termDefinition> Bauteile, die die elektronischen Steuer- und
             Regelvorgänge für die Blockierregelung und die
             Antriebsschlupfregelung übernehmen.</termDefinition>
            <termSection>
                <termString>ABS/ASR-Steuerung</termString>
                <usageDescriptors xml:lang="en">
                    <usedIn>Germany</usedIn>
                    <usedIn>Switzerland</usedIn>
                </usageDescriptors>
                <wordClass>n</wordClass>
                <wordGender>f</wordGender>
                <termLastModified>21-08-2001</termLastModified>
            </termSection>
        </termGroup>
        <termGroup>
            <languageCode>English</languageCode>
            <termSection>
                <termString>ABS/ASR control</termString>
                <usageDescriptors>
                    <usedIn>Britain</usedIn>
                </usageDescriptors>
                <wordClass>n</wordClass>
                <termLastModified>20-08-2001</termLastModified>
            </termSection>
        </termGroup>
    </conceptEntry>
</termBank>
```

D.4.2 DCS

Based on the description of the content of elements given above, the following table shows example mappings of information units to data categories in ISO 12620 and to required data categories outside ISO 12620 such as ISO 639-1: Data category specification.html.

Many of the information units in the example map directly to ISO 12620 data categories. Ideally, this would be true for all information units.

There are exceptions to this rule in the example presented which need to be addressed. Firstly, the XML element `<usageDescriptors>` does not itself have content. For a TML, this grouping is unnecessary and hence can be dropped. Secondly, the XML elements with the suffix `LastModified` do not have direct equivalents in ISO 12620. To complete the mapping, appropriate encoding is required. `LastModified` contains a date that refers to the last time a modification was made. There are, in fact, two information units encoded here: a main unit that denotes that a terminology management process, in this case a modification, has occurred, and a date on which it occurred. These two information units do map to ISO 12620. As the date is a refinement information unit to the terminological management process, this information should be grouped accordingly, for example:

© ISO 2003 – All rights reserved

Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
<termManProcGrp>
    <termManProc>modification</termManProc>
    <modifiedDate>20-08-2001</modifiedDate>
<termManProcGrp>
```

The application of <brack> is appropriate to the GMT representation of this data.

D.4.3 Content mappings

For interoperability, where specific lists of data are expected to form the content of certain XML elements, mapping such shared identifiers can simplify these processes. As an example, consider a translation of the content of the <languageCode> element to a code based on ISO 639-1. For example, *English* could become en.639-1. Similarly, country codes can be mapped to ISO 3166-1.

D.4.4 TMF-conforming XML representation (GMT)

The result of this analysis and substitution of identifiers for those in this International Standard and in ISO 12620 produces the following GMT formatted data which can be considered as a TMF-conforming TML.

This TML could be transformed automatically using, for example, XSLT to the formats specified in normative Annexes B and C of this International Standard, and back, without loss of information.

In the following, data categories from ISO 12620 are denoted in bold. Bold italics denote references to other data categories such as those in ISO 639-1 and ISO 3166-1. By making reference to such stable systems, greater degrees of interoperability are assured.

```
<struct type="TDC" xml:lang="en">
    <struct type="GI">
         <feat type="subsetIdentifier-12620A.10.3">00aa</feat>
         <feat type="projectSubset-12620A.10.3.3">Automotive Engineering</feat>
    </struct>
    <struct type="TE">
         <feat type="subjectField-12620A.4">ABS</feat>
         <brack>
               <feat type="terminologyManagementTransactions-12620A.10.1">
                modification-12620A.10.1.3</feat>
               <feat type="modificationDate-12620A.10.2.1.3">21-08-2001</feat>
         </brack>
         <struct type="LS" xml:lang="de">
               <feat type="languageIdentifier-12620A.10.7.1">de-639.1</feat>
               <feat type="definition-12620A.5.1">Bauteile, die die
                elektronischen Steuer und Regelvorgänge für die Blockierregelung
                und die Antriebsschlupfregelung übernehmen.</feat>
               <struct type="TS">
                     <feat type="term-12620A.1">ABS/ASR-Steuerung</feat>
                     <feat type="geographicalUsage-12620A.2.3.2" xml:lang="en">
                      DE-3166.1</feat>
                     <feat type="geographicalUsage-12620A.2.3.2" xml:lang="en">
                      CH-3166.1</feat>
                     <feat type="partOfSpeech-12620A.2.2.1">n</feat>
                     <feat type="grammaticalGender-12620A.2.2.2">
                      feminine-12620A.2.2.2.2</feat>
                     <brack>
                         <feat type="terminologyManagementTransactions-
                          12620A.10.1">modification-12620A.10.1.3</feat>
                         <feat type="modificationDate-12620A.10.2.1.3">
                          21-08-2001</feat>
                     </brack>
               </struct>
```


--`,,`,-`-`,,`,,`,`,,`---
© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
        </struct>
        <struct type="LS">
              <feat type="languageIdentifier-12620A.10.7.1">en-639.1</feat>
              <struct type="TS">
                    <feat type="term-12620A.1">ABS/ASR control</feat>
                    <feat type="geographicalUsage-12620A.2.3.2" xml:lang="en">
                     GB-3166.1</feat>
                    <feat type="partOfSpeech-12620A.2.2.1">n</feat>
                    <brack>
                        <feat type="terminologyManagementTransactions-
                         12620A.10.1">modification-12620A.10.1.3</feat>
                        <feat type="modificationDate-12620A.10.2.1.3">21-08-2001
                        </feat>
                    </brack>
              </struct>
        </struct>
    </struct>
</struct>
```

--`,,`,-`-`,,`,,`,`,,`---

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Bibliography

[1] ISO 639-1, *Codes for the representation of names of languages — Part 1: Alpha-2 code*

[2] ISO 639-2, *Codes for the representation of names of languages — Part 2: Alpha-3 code*

[3] ISO/IEC 646, *Information technology — ISO 7-bit coded character set for information interchange*

[4] ISO 690, *Documentation — Bibliographic references — Content, form and structure*

[5] ISO 690-2, *Information and documentation — Bibliographic references — Part 2: Electronic documents or parts thereof*

[6] ISO 704, *Terminology work — Principles and methods*

[7] ISO 1951, *Lexicographical symbols and typographical conventions for use in terminography*

[8] ISO 2788, *Documentation — Guidelines for the establishment and development of monolingual thesauri*

[9] ISO 3166-1, *Codes for the representation of names of countries and their subdivisions — Part 1: Country codes*

[10] ISO 5964, *Documentation — Guidelines for the establishment and development of multilingual thesauri*

[11] ISO 8601, *Data elements and interchange formats — Information interchange — Representation of dates and times*

[12] ISO 8879, *Information processing — Text and office systems — Standard Generalized Markup Language (SGML)*

[13] ISO/IEC 10646-1, *Information technology — Universal Multiple-Octet Coded Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane*

[14] ISO 12083, *Information and documentation — Electronic manuscript preparation and markup*

[15] ISO 12200, *Computer applications in terminology — Machine-readable terminology interchange format (MARTIF) — Negotiated interchange*

[16] *XML Schema Part 2: Datatypes*, BIRON, P.V. and MALHOTRA, A. (eds.), W3C Recommendation 02 May 2001, available at <http://www.w3.org/TR/xmlschema-2/>

[17] *Modularization of XHTML™*, ALTHEIM, M., BOUMPHREY, F., DOOLEY, S., MCCARRON, S., SCHNITZENBAUMER, S. and WUGOFSKI, T. (eds.), W3C Recommendation 10 April 2001, available at <http://www.w3.org/TR/xhtml-modularization/>

[18] *XML Linking Language (XLink) Version 1.0*, DEROSE, S., MALER, E. and ORCHARD, D. (eds.), W3C Recommendation 27 June 2001, available at <http://www.w3.org/TR/xlink/>

[19] *URIs, URLs, and URNs: Clarifications and Recommendations 1.0*, URI Planning Interest Group, W3C Note 21 September 2001, available at <http://www.w3.org/TR/uri-clarification/>

[20] *Ontology Inference Layer (OIL)*, available at <http://www.ontoknowledge.org/oil/>

[21] *Open Lexicon Interchange Format (OLIF)*, available at <http://www.olif.net/>

[22] *XML Localisation Interchange File Format*, available at <http://www.xliff.org/>

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

[23] *Translation Memory eXchange*, Open Standards for Container/Content Allowing Re-use (OSCAR) committee, LISA Special Interest Group, available at <http://www.lisa.org/tmx/>

[24] *XSL Transformations (XSLT) Version 1.0*, CLARK, J. (ed.), W3C Recommendation 16 November 1999, available at <http://www.w3.org/TR/xslt>

[25] *XPointer Framework*, GROSSO, P., MALER, E., MARSH, J. and WALSH, N. (eds.), W3C Recommendation 25 March 2003, available at <http://www.w3.org/TR/xptr-framework/>

[26] *XML Path Language (XPath) Version 1.0*, CLARK, J. and DEROSE, S. (eds.), W3C Recommendation 16 November 1999, available at <http://www.w3.org/TR/xpath>

© ISO 2003 – All rights reserved
Copyright International Organization for Standardization
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

ICS 01.020; 35.240.30

Price based on 49 pages

Copyright International Organization for Standardization ved
Provided by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale