
**Information technology — Topic Maps —
Constraint Language (TMCL)**

*Technologies de l'information — Plans relatifs à des sujets —
Contraintes de langage (TMCL)*

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 19756:2011](https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011)

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19756:2011

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2011

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

Contents

	Page
Foreword.....	v
Introduction.....	vi
1 Scope.....	1
2 Normative references.....	1
3 Notation and Conventions.....	1
3.1 General.....	1
3.2 tmdm:subject.....	2
3.3 Following an Association.....	2
3.4 Following an Association to Roles of Given Type.....	2
3.5 Playing a Role.....	2
3.6 Being a Subtype.....	2
3.7 Being an Instance.....	2
3.8 Matching a Regular Expression.....	2
3.9 The Value of an Occurrence.....	2
3.10 Comparison of iso:ctm-integer values.....	3
4 Validation Semantics.....	3
5 TMCL Syntax.....	3
6 TMCL Declarations.....	4
6.1 General.....	4
6.2 Topic Type.....	4
6.3 Name Type.....	4
6.4 Occurrence Type.....	4
6.5 Association Type.....	4
6.6 Role Type.....	4
6.7 Overlap Declaration.....	5
7 TMCL Constraint Types.....	5
7.1 General.....	5
7.2 Abstract Topic Type Constraint.....	5
7.3 Subject Identifier Constraint.....	5
7.4 Subject Locator Constraint.....	6
7.5 Item Identifier Constraint.....	7
7.6 Topic Name Constraint.....	7
7.7 Variant Name Constraint.....	8
7.8 Topic Occurrence Constraint.....	9
7.9 Topic Role Constraint.....	9
7.10 Scope Constraint.....	10
7.11 Scope Required Constraint.....	11
7.12 Reifier Constraint.....	12
7.13 Topic Reifies Constraint.....	12
7.14 Association Role Constraint.....	13
7.15 Role Combination Constraint.....	14
7.16 Occurrence Data Type Constraint.....	15
7.17 Unique Value Constraint.....	15
7.18 Regular Expression Constraint.....	16
8 Additional TMCL Templates.....	16
9 User-defined Constraints.....	16
9.1 General.....	16
9.2 Denial Constraint.....	17
9.3 Requirement Constraint.....	17
10 Schema Documentation.....	17
10.1 General.....	17
10.2 The Schema Topic.....	17

10.3	Documentation Occurrences.....	18
10.4	The Topic Map Topic Type.....	19
11	Conformance.....	19
Annex A	(normative) TMCL Templates CTM.....	20
Annex B	(normative) TMCL meta-schema.....	25

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 19756:2011](https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011)

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national 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 and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 19756 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

TECH STANDARD PREVIEW

(standards.iteh.ai)

[ISO/IEC 19756:2011](https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011)

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

Introduction

TMCL is a constraint language for Topic Maps, allowing definitions of Topic Maps schemas to be written in a precise and machine-readable form. This makes it possible to validate a topic map against a TMCL schema to see if it conforms to the constraints in the schema, and also enables other uses, such as schema-driven editors, object mappings, and so on.

TMCL is defined as a Topic Maps vocabulary consisting of a number of topic, association, occurrence, and role types, identified by Published Subject Identifiers (PSIs), and defined using English prose. TMCL defines the concept of validation, by which a given topic map is valid according to a schema if it conforms to all the constraints in that schema and a number of global validation rules which apply to all topic maps independent of schema.

TMCL does not have any syntax of its own, since it is defined simply as a Topic Maps vocabulary. However, a number of CTM templates are defined in this International Standard in order to facilitate authoring of TMCL schemas using CTM.

iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 19756:2011](https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011)

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

Information technology — Topic Maps — Constraint Language (TMCL)

1 Scope

This International Standard defines a Topic Maps vocabulary for representing constraints on Topic Maps instance data and CTM templates for authoring TMCL schemas.

It does not define a syntax for representing constraints on Topic Maps instance 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.

NOTE Each of the following documents has a unique identifier that is used to cite the document in the text. The unique identifier consists of the part of the reference up to the first comma.

ISO/IEC 13250-2:2006, *Information technology — Topic Maps — Part 2: Data model*

ISO/IEC 18048, *Information technology — SGML applications — Topic Map Query Language (TMQL)*¹⁾

ISO/IEC 13250-6, *Information technology — Topic Maps — Part 6: Compact syntax*

XML Schema-2, *XML Schema Part 2: Datatypes Second Edition*, W3C Recommendation, 28 October 2004, available at <<http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>>

3 Notation and Conventions

3.1 General

The TMCL validation rules are defined in English prose using constructs from ISO/IEC 13250-2:2006 (the TMDM), and written according to certain conventions, whereby some of the phrases in the text have particular interpretations. These phrases and their interpretations are given in the following clauses.

Throughout this clause the defined phrases contain placeholders given as variables (written thus: *v*) in the definitions. When the phrases are used, these placeholders are defined either by reference to topics already defined in the context where the phrase is used, or using a QName (an identifying token of the form *foo:bar*). QNames are expanded into full IRIs using the prefix declarations in Clause 5, and refer to the topic item which has that IRI in its [subject identifiers] property.

The following namespace prefixes are used throughout this International Standard:

```
%prefix tmcl http://psi.topicmaps.org/tmcl/  
%prefix tmdm http://psi.topicmaps.org/iso13250/model/  
%prefix xsd http://www.w3.org/2001/XMLSchema#
```

Thus, *tmdm:bar* is a shorthand for the IRI <http://psi.topicmaps.org/iso13250/model/bar>.

Throughout this International Standard the syntax [foo], that is, a name in square brackets, not italicized, is used to refer to property names from ISO/IEC 13250-2:2006 (the TMDM).

1) To be published

3.2 tmdm:subject

The topic `tmdm:subject` (<http://psi.topicmaps.org/iso13250/model/subject>) represents the concept of a subject as defined in ISO/IEC 13250-2:2006. It is the type of which all topics, statements, and association roles are instances, and the common supertype of all types. In TMCL it is used in constraints to make it clear that any topic whatsoever may appear in a particular position in an ontology.

3.3 Following an Association

To follow an association of type **at** from a topic **t** means traversing all associations of type **at**. **A** is the set of all the association items **a** in the [parent] properties of all the role items in the [roles played] property of **t** where **a**'s [type] property contains **at** or some subtype of **at**.

The end result of the traversal is the set of topic items in the [player] property of each role item **r** in **a**'s [roles] property, except where **r** is contained in **t**'s [roles played] property.

3.4 Following an Association to Roles of Given Type

To follow an association of type **at** to roles of type **rt** from a topic **t** means traversing all associations of type **at**. **A** is the set of all the association items **a** in the [parent] properties of all the role items in the [roles played] property of **t** where **a**'s [type] property contains **at** or a subtype of **at**.

The end result of the traversal is the set of topic items in the [player] property of each role item **r** in **a**'s [roles] property, except where **r** is contained in **t**'s [roles played] property, or where **r**'s [type] property does not contain **rt** or some subtype of **rt**.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

3.5 Playing a Role

A topic **t** is said to play a role of type **rt** in an association of type **at** when **t**'s [roles played] property contains at least one role item **r** whose [type] property contains **rt** (or some subtype of **rt**) and **r**'s [parent] property contains an association item whose [type] property contains **at** or some subtype of **at**.

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

3.6 Being a Subtype

A topic type **t1** is said to be a subtype of the topic type **t2** if following the association type `tmdm:supertype-subtype` from **t1** to roles of type `tmdm:supertype` produces either **t2** or a subtype of **t2**. If **t2** is `tmdm:subject` then **t1** is a subtype of it even if the necessary `tmdm:supertype-subtype` associations are not present.

3.7 Being an Instance

A topic **t** is said to be an instance of the topic type **tt** if following the association type `tmdm:type-instance` from **t** produces either **tt**, a subtype of **tt**, or if **tt** is `tmdm:subject`.

A statement **s** (as defined in ISO/IEC 13250-2:2006) is said to be an instance of the statement type **st** if the **s**'s [type] property contains either **st**, a subtype of **st**, or if **st** is `tmdm:subject`.

3.8 Matching a Regular Expression

A string **s** matches a regular expression **r** if the string is a member of the set of strings $L(r)$ denoted by **r** as defined in appendix F of [XML Schema-2].

3.9 The Value of an Occurrence

The value of a topic **t**'s occurrence of type **ot** is referred to as **v**, and produced by finding the occurrence item **o** in **t**'s [occurrences] property whose [type] property contains **ot** or some subtype of **ot**. If no such occurrence exists there is no given value. It is an error if there is more than one such occurrence.

3.10 Comparison of iso:ctm-integer values

When comparing tmcl:card-min and tmcl:card-max values, in addition to the ordinary integers, one may encounter the special value * of datatype iso:ctm-integer. This special value is used to indicate unlimited cardinality, and compares as larger than any specific integer.

4 Validation Semantics

This International Standard defines two kinds of rules:

- Constraint validation rules, which specify how to validate the constraint represented by an individual topic that is an instance of some subclass of tmcl:constraint.
- Global validation rules, which are not tied to any specific constraint topic, and which apply to the entire topic map.

A schema is a set of constraint topics, declaration topics, topic types, association types, role types, name types, and occurrence types contained in a topic map.

A topic map is valid according to a schema if the topic map is valid according to each individual constraint in the schema, and if the topic map is valid according to all the global validation rules specified in this International Standard.

NOTE 1 This is a very strict form of validation, which may not be suitable for all situations. Implementations may allow users to turn off any set of constraints they choose in order to allow more flexible forms of validation.

A topic map is valid according to a constraint if the topic map satisfies the constraint validation rule defined for the constraint type of which the constraint topic is an instance.

NOTE 2 TMCL does not dictate when constraints should be applied, nor does it state what it means to a controlling application when it is found that a given constraint is violated.

The validation rules defined in this International Standard assume that the schema and the instance data are located in the same topic map. If this is not the case, implementations shall behave as though the schema and instance topic maps were merged. This includes any schemas included by using tmcl:includes-schema, even though these schemas may physically reside in other topic maps.

Implementations shall also behave as though the topic map in Annex B were merged into the topic map being validated.

5 TMCL Syntax

TMCL has no syntax of its own, but since TMCL schemas are represented as topic maps, any Topic Maps representation can be used to create a TMCL schema. The schema shall, however, be valid according to the TMCL meta-schema in Annex B. The values of all occurrences of type tmcl:regexp shall be valid regular expressions as defined in [XML Schema-2].

Throughout this International Standard all examples are given using the CTM syntax (defined in ISO/IEC 13250-6), and to facilitate the authoring of TMCL in CTM a number of templates are defined. These templates exist in a resource <http://www.isotopicmaps.org/tmcl/templates.ctm> that can be downloaded and included in any CTM file.

To include the TMCL templates the following CTM include directive can be used:

```
%include http://www.isotopicmaps.org/tmcl/templates.ctm
```

6 TMCL Declarations

6.1 General

TMCL provides a number of constructs known as declarations, which allow Topic Maps constructs to be used in ways which would otherwise be forbidden by the global validation rules. Declarations are in a sense the opposite of constraints, in that their effect is to relax constraints built into TMCL.

6.2 Topic Type

Making a topic an instance of `tmcl:topic-type` is a declaration that the topic may be used as a topic type.

EXAMPLE Declaring person a topic type:

```
person isa tmcl:topic-type .
```

Global Validation Rule: any topic **t** which plays a role of type `tmdm:type` in an association of type `tmdm:type-instance`, but which is not an instance of `tmcl:topic-type`, is invalid.

6.3 Name Type

Making a topic an instance of `tmcl:name-type` is a declaration that the topic may be used as a name type.

EXAMPLE Declaring nickname a name type:

```
nickname isa tmcl:name-type .
```

Global Validation Rule: any topic name item **n** whose `[type]` property contains a topic item **t** where **t** is not an instance of `tmcl:name-type` is invalid.

6.4 Occurrence Type

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

Making a topic an instance of `tmcl:occurrence-type` is a declaration that the topic may be used as an occurrence type.

EXAMPLE Declaring date-of-birth an occurrence type:

```
date-of-birth isa tmcl:occurrence-type .
```

Global Validation Rule: any occurrence item **o** whose `[type]` property contains a topic item **t** where **t** is not an instance of `tmcl:occurrence-type` is invalid.

6.5 Association Type

Making a topic an instance of `tmcl:association-type` is a declaration that the topic may be used as an association type.

EXAMPLE Declaring that works-for is an association type:

```
works-for isa tmcl:association-type .
```

Global Validation Rule: any association item **a** whose `[type]` property contains a topic item **t** where **t** is not an instance of `tmcl:association-type` is invalid.

6.6 Role Type

Making a topic an instance of `tmcl:role-type` is a declaration that the topic may be used as a role type.

EXAMPLE Declaring containee a role type:

```
containee isa tmcl:role-type .
```

Global Validation Rule: any association role item *r* whose [type] property contains a topic item *t* where *t* is not an instance of `tmcl:role-type` is invalid.

6.7 Overlap Declaration

The `tmcl:overlap-declaration` is used to declare that the sets of instances of two or more topic types are non-disjoint (that is, that they may overlap). The default is that the instance sets of different topic types are disjoint.

EXAMPLE The following declares that the topic types `person` and `employee` overlap:

```
person isa tmcl:topic-type;
overlaps(employee).
```

The CTM template for the overlaps declaration is defined as:

```
def overlaps($t1, $t2)
  ?c isa tmcl:overlap-declaration.
  tmcl:overlaps(tmcl:allows : ?c, tmcl:allowed : $t1)
  tmcl:overlaps(tmcl:allows : ?c, tmcl:allowed : $t2)
end
```

Global Validation Rule: the set of all the topic types of which a given topic *t* is an instance is referred to as *TT*. *t* is invalid if there exists a pair of different topics *t1* and *t2* in *TT* such that:

- *t1* is not a subtype of *t2* (or vice versa), and
- there does not exist a *c* which is an instance of `tmcl:overlap-declaration` and where following the `tmcl:overlaps` association from *c* produces both *t1* and *t2*.

7 TMCL Constraint Types

7.1 General

<https://standards.iteh.ai/catalog/standards/sist/5bfdeed6-2676-45fc-96ea-e878241b6592/iso-iec-19756-2011>

The `tmcl:constraint` topic type is used as the base type for all topic types that are considered constraint types. It is an abstract type used simply to group the constraint types for ease of schema introspection.

7.2 Abstract Topic Type Constraint

The `tmcl:abstract-constraint` states that a given topic type shall not have any direct instances.

EXAMPLE The following states that `creature` is an abstract topic type:

```
creature isa tmcl:topic-type;
is-abstract().
```

The CTM template for the abstract constraint is defined as:

```
def is-abstract($t)
  ?c isa tmcl:abstract-constraint.
  tmcl:constrained-topic-type(tmcl:constraint : ?c, tmcl:constrained : $t)
end
```

Constraint Validation Rule for all constraints *c* of type `tmcl:abstract-constraint`: *c* applies to a topic type *t*, which can be found by following associations of type `tmcl:constrained-topic-type` from the *c* topic. *t* violates the constraint if it plays the role of type `tmdm:type` in any association of type `tmdm:type-instance`.

7.3 Subject Identifier Constraint

A subject identifier constraint constrains the subject identifiers of instances of a given topic type. The constraint has the following properties:

- card-min, indicating the minimum number of subject identifiers a valid instance shall have
- card-max, indicating the maximum number of subject identifiers a valid instance can have, and
- regexp, a regular expression the subject identifier shall match.

EXAMPLE The following states that topics of type person shall have zero or one subject identifier:

```
person isa tmcl:topic-type;
has-subject-identifier(0, 1, ".*").
```

The CTM template for the subject identifier constraint is defined as:

```
def has-subject-identifier($tt, $min, $max, $regexp)
?c isa tmcl:subject-identifier-constraint;
tmcl:card-min: $min;
tmcl:card-max: $max;
tmcl:regexp: $regexp.
tmcl:constrained-topic-type(tmcl:constraint : ?c, tmcl:constrained : $tt)
end
```

Constraint Validation Rule for all constraints **c** of type tmcl:subject-identifier-constraint: **c** applies to a topic type **t**, which can be found by following the tmcl:constrained-topic-type association from **c**. The constraint has the following properties:

- **r**, which is the value of **c**'s occurrence of type tmcl:regexp if given, and ".*" if not,
- **min**, which is the value of **c**'s tmcl:card-min occurrence, if given, and 0, if not, and
- **max**, which is the value of **c**'s tmcl:card-max occurrence, if given, and undefined, if not.

For each instance **i** of **t** the set of its subject identifiers which match **r** is referred to as **S**. **i** violates **c** if the cardinality of **S** is lower than **min** or greater than **max** (provided **max** is not undefined).

7.4 Subject Locator Constraint

A subject locator constraint constrains the subject locators of instances of a given topic type. The constraint has the following properties:

- card-min, indicating the minimum number of subject locators a valid instance shall have
- card-max, indicating the maximum number of subject locators a valid instance can have, and
- regexp, a regular expression that the subject locator shall match.

EXAMPLE The following states that topics of type document shall have at least one subject locator:

```
document isa tmcl:topic-type;
has-subject-locator(1, *, ".*").
```

The CTM template for the subject locator constraint is defined as:

```
def has-subject-locator($tt, $min, $max, $regexp)
?c isa tmcl:subject-locator-constraint;
tmcl:card-min: $min;
tmcl:card-max: $max;
tmcl:regexp: $regexp.
tmcl:constrained-topic-type(tmcl:constraint : ?c, tmcl:constrained : $tt)
end
```

Constraint Validation Rule for all constraints **c** of type tmcl:subject-locator-constraint: **c** applies to a topic type **t**, which can be found by following the tmcl:constrained-topic-type association from **c**. The constraint has the following properties:

- **r**, which is the value of **c**'s occurrence of type tmcl:regexp if given, and ".*" if not,
- **min**, which is the value of **c**'s tmcl:card-min occurrence, if given, and 0, if not, and

- **max**, which is the value of **c**'s `tmcl:card-max` occurrence, if given, and undefined, if not.

For each instance *i* of **t** the set of its subject locators which match **r** is referred to as **S**. *i* violates **c** if the cardinality of **S** is lower than **min** or greater than **max** (provided **max** is not undefined).

7.5 Item Identifier Constraint

An item identifier constraint constrains the item identifiers of topic map constructs of a given type. The constraint has the following properties:

- `card-min`, indicating the minimum number of item identifiers a valid instance shall have
- `card-max`, indicating the maximum number of item identifiers a valid instance can have, and
- `regexp`, a regular expression that the item identifier shall match.

EXAMPLE The following states that topics of type document shall have at least one item identifier:

```
document isa tmcl:topic-type;
has-item-identifier(1, *, ".*").
```

The CTM template for the item identifier constraint is defined as:

```
def has-item-identifier($tt, $min, $max, $regexp)
  ?c isa tmcl:item-identifier-constraint;
  tmcl:card-min: $min;
  tmcl:card-max: $max;
  tmcl:regexp: $regexp.
  tmcl:constrained-construct(tmcl:constraint : ?c, tmcl:constrained : $tt)
end
```

iTech STANDARD PREVIEW
(standards.iteh.ai)

Constraint Validation Rule for all constraints **c** of type `tmcl:item-identifier-constraint`: **c** applies to a Topic Maps construct **t**, which can be found by following the `tmcl:constrained-construct` association from **c**. The constraint has the following properties:

- **r**, which is the value of **c**'s occurrence of type `tmcl:regexp` if given, and `".*"` if not,
- **min**, which is the value of **c**'s `tmcl:card-min` occurrence, if given, and 0, if not, and
- **max**, which is the value of **c**'s `tmcl:card-max` occurrence, if given, and undefined, if not.

For each instance *i* of **t** the set of its item identifiers which match **r** is referred to as **S**. *i* violates **c** if the cardinality of **S** is lower than **min** or greater than **max** (provided **max** is not undefined).

7.6 Topic Name Constraint

A topic name constraint constrains the type and cardinality of topic names for instances of a given topic type. The constraint has the following properties:

- `card-min`, indicating the minimum number of names a valid instance shall have, and
- `card-max`, indicating the maximum number of names a valid instance shall have.

EXAMPLE The following states that topics of type person shall have exactly one name of type `tmdm:topic-name`:

```
person isa tmcl:topic-type;
has-name(tmdm:topic-name, 1, 1).
```

The CTM template for the topic name constraint is defined as:

```
def has-name($tt, $nt, $min, $max)
  ?c isa tmcl:topic-name-constraint;
  tmcl:card-min: $min;
  tmcl:card-max: $max.
```