

Australian/New Zealand Standard™

Geographic information—Methodology for feature cataloguing



AS/NZS ISO 19110:2006

This Joint Australian/New Zealand Standard was prepared by Joint Technical Committee IT-004, Geographical Information/Geomatics. It was approved on behalf of the Council of Standards Australia on 11 January 2006 and on behalf of the Council of Standards New Zealand on 27 January 2006.
This Standard was published on 20 February 2006.

The following are represented on Committee IT-004:

ACT Planning and Land Authority
ANZLIC - the Spatial Information Council
Association of Crown Research Institutes New Zealand
Australasian Fire Authorities Council
Australian Antarctic Division
Australian Bureau of Statistics
Australian Hydrographic Office
Australian Map Circle
Australian Spatial Information Business Association
CSIRO Exploration & Mining
Department for Administrative and Information Services, SA
Department of Conservation, New Zealand
Department of Defence (Australia)
Department of Infrastructure, Planning and Environment, NT
Department of Natural Resources and Mines, Qld
Department of Primary Industries, Water and Environment, Tas
Geoscience Australia
Inter Governmental Committee on Surveying and Mapping
Land Information New Zealand
Land Victoria
Mapping Sciences Institute, Australia
Telecom New Zealand
Western Australian Land Information System

Keeping Standards up-to-date

Standards are living documents which reflect progress in science, technology and systems. To maintain their currency, all Standards are periodically reviewed, and new editions are published. Between editions, amendments may be issued. Standards may also be withdrawn. It is important that readers assure themselves they are using a current Standard, which should include any amendments which may have been published since the Standard was purchased.

Detailed information about joint Australian/New Zealand Standards can be found by visiting the Standards Web Shop at www.saiglobal.com.au or Standards New Zealand web site at www.standards.co.nz and looking up the relevant Standard in the on-line catalogue.

For more frequent listings or notification of revisions, amendments and withdrawals, Standards Australia and Standards New Zealand offer a number of update options. For information about these services, users should contact their respective national Standards organization.

We also welcome suggestions for improvement in our Standards, and especially encourage readers to notify us immediately of any apparent inaccuracies or ambiguities. Please address your comments to the Chief Executive of either Standards Australia or Standards New Zealand at the address shown on the back cover.

This Standard was issued in draft form for comment as DR 05521.

Australian/New Zealand Standard™

Geographic information—Methodology for feature cataloguing

First published as AS/NZS ISO 19910:2006.
Reissued incorporating Amendment No. 1 (December 2011).

COPYRIGHT

© Standards Australia Limited/Standards New Zealand

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher, unless otherwise permitted under the Copyright Act 1968 (Australia) or the Copyright Act 1994 (New Zealand).

Jointly published by SAI Global Limited under licence from Standards Australia Limited, GPO Box 476, Sydney, NSW 2001 and by Standards New Zealand, Private Bag 2439, Wellington 6140.

ISBN 0 7337 7255 2

PREFACE

This Standard was prepared by the Joint Standards Australia/Standards New Zealand Committee IT-004, Geographical Information/Geomatics.

This Standard incorporates Amendment No. 1 (December 2011). The changes required by the Amendment are indicated in the text by a marginal bar and amendment number against the clause, note, table, figure or part thereof affected.

A1

This Standard is identical with, and has been reproduced from ISO 19110:2005, *Geographic information—Methodology for feature cataloguing* and its amendment ISO 19110:2005/Amd.1:2011.

The objective of this Standard is to provide designers and analysts with a specification of how the classification of feature types is organized into a feature catalogue and presented to the users of a set of geographic data.

The terms ‘normative’ and ‘informative’ are used to define the application of the annex to which they apply. A normative annex is an integral part of a standard, whereas an informative annex is only for information and guidance.

As this Standard is reproduced from an international standard, the following applies:

- (a) Its number appears on the cover and title page while the international standard number appears only on the cover.
- (b) In the source text ‘this International Standard’ should read ‘this Australian/New Zealand Standard’.
- (c) A full point substitutes for a comma when referring to a decimal marker.

References to International Standards should be replaced by references to Australian or Australian/New Zealand Standards, as follows:

<i>Reference to International Standard</i>		<i>Australian/New Zealand Standard</i>	
ISO		AS/NZS ISO	
19109	Geographic information—Rules for application schema	19109	Geographic information—Rules for application schema
19115	Geographic information—Metadata	19115	Geographic information—Metadata
ISO/TS			
19103	Geographic information—Conceptual schema language	19103	Geographic information—Conceptual schema language

CONTENTS

	<i>Page</i>
1 Scope.....	1
2 Conformance	1
3 Normative references	2
4 Terms and definitions	2
5 Abbreviations	3
6 Principal requirements	4
6.1 Feature catalogue	4
6.2 Information elements.....	4
Annex A (normative) Abstract test suite	6
Annex B (normative) Feature catalogue template.....	16
Annex C (informative) Feature cataloguing examples.....	31
Annex D (informative) Feature cataloguing concepts	46
Bibliography	55

INTRODUCTION

Geographic features are real world phenomena associated with a location relative to the Earth, about which data are collected, maintained, and disseminated. Feature catalogues defining the types of features, their operations, attributes, and associations represented in geographic data are indispensable to turning the data into usable information. Such feature catalogues promote the dissemination, sharing, and use of geographic data through providing a better understanding of the content and meaning of the data. Unless suppliers and users of geographic data have a shared understanding of the kinds of real world phenomena represented by the data, users will be unable to judge whether the data supplied are fit for their purpose.

The availability of standard feature catalogues that can be used multiple times will reduce costs of data acquisition and simplify the process of product specification for geographic datasets.

This International Standard provides a standard framework for organizing and reporting the classification of real world phenomena in a set of geographic data. Any set of geographic data is a greatly simplified and reduced abstraction of a complex and diverse world. A catalogue of feature types can never capture the richness of geographic reality. However, such a feature catalogue should present the particular abstraction represented in a given dataset clearly, precisely, and in a form readily understandable and accessible to users of the data.

Geographic features occur at two levels: instances and types. At the instance level, a geographic feature is represented as a discrete phenomenon that is associated with its geographic and temporal coordinates and may be portrayed by a particular graphic symbol. These individual feature instances are grouped into classes with common characteristics: feature types. It is recognized that geographic information is subjectively perceived and that its content depends upon the needs of particular applications. The needs of particular applications determine the way instances are grouped into types within a particular classification scheme. ISO 19109, *Geographic information — Rules for application schema* specifies how data shall be organized to reflect the particular needs of applications with similar data requirements.

NOTE The full description of the contents and structure of a geographic dataset is given by the application schema developed in compliance with ISO 19109. The feature catalogue defines the meaning of the feature types and their associated feature attributes, feature operations and feature associations contained in the application schema.

The collection criteria used to identify individual real world phenomena and to represent them as feature instances in a dataset are not specified in this International Standard. Because they are not included in the standards, collection criteria should be included separately in the product specification for each dataset.

A standard way of organizing feature catalogue information will not automatically result in harmonization or interoperability between applications. In situations where classifications of features differ, this International Standard may at least serve to clarify the differences and thereby help to avoid the errors that would result from ignoring them. It may also be used as a standard framework within which to harmonize existing feature catalogues that have overlapping domains.

AUSTRALIAN/NEW ZEALAND STANDARD

Geographic information — Methodology for feature cataloguing**1 Scope**

This International Standard defines the methodology for cataloguing feature types. This International Standard specifies how the classification of feature types is organized into a feature catalogue and presented to the users of a set of geographic data. This International Standard is applicable to creating catalogues of feature types in previously uncatalogued domains and to revising existing feature catalogues to comply with standard practice. This International Standard applies to the cataloguing of feature types that are represented in digital form. Its principles can be extended to the cataloguing of other forms of geographic data.

This International Standard is applicable to the definition of geographic features at the type level. This International Standard is not applicable to the representation of individual instances of each type. This International Standard excludes spatial, temporal, and portrayal schemas as specified in ISO 19107, ISO 19108, and ISO 19117, respectively. It also excludes collection criteria for feature instances.

This International Standard may be used as a basis for defining the universe of discourse being modelled in a particular application, or to standardize general aspects of real world features being modelled in more than one application.

2 Conformance

Because this International Standard specifies a number of options that are not required for all feature catalogues, this clause specifies 12 conformance classes. These classes are differentiated on the basis of three criteria:

- a) What elements of a feature type are required in a catalogue:
 - 1) feature attributes only?
 - 2) feature attributes and feature associations?
 - 3) feature attributes, feature associations, and feature operations?
- b) Is there a requirement to link feature attributes, feature associations, and feature operations to only one feature type or may they be linked to multiple feature types?
- c) Is there a requirement to include inheritance relationships in the feature catalogue?

Annex A specifies a test module for each of the conformance classes, as shown in Table 1.

Table 1 — Conformance classes

Attributes only	Attributes and associations	Attributes, associations and operations	Properties associated with multiple features	Inheritance relationships included	Test module
X	—	—	—	—	A.17
—	X	—	—	—	A.18
—	—	X	—	—	A.19
X	—	—	X	—	A.20
—	X	—	X	—	A.21
—	—	X	X	—	A.22
X	—	—	—	X	A.23
—	X	—	—	X	A.24
—	—	X	—	X	A.25
X	—	—	X	X	A.26
—	X	—	X	X	A.27
—	—	X	X	X	A.28

3 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/TS 19103:—¹⁾, *Geographic information — Conceptual schema language*

ISO 19109:—¹⁾, *Geographic information — Rules for application schema*

ISO 19115:2003, *Geographic information — Metadata*

4 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

4.1

feature

abstraction of real world phenomena

[ISO 19101]

EXAMPLE The phenomenon named 'Eiffel Tower' may be classified with other similar phenomena into a feature type 'tower'.

NOTE A feature may occur as a type or an instance. Feature type or feature instance should be used when only one is meant.

1) To be published.

4.2

feature association

relationship that links instances of one **feature** (4.1) type with instances of the same or a different **feature** type

4.3

feature attribute

characteristic of a **feature** (4.1)

[ISO 19101]

EXAMPLE 1 A feature attribute named 'colour' may have an attribute value 'green' which belongs to the data type 'text'.

EXAMPLE 2 A feature attribute named 'length' may have an attribute value '82,4' which belongs to the data type 'real'.

NOTE A feature attribute has a name, a data type, and a value domain associated to it. A feature attribute for a feature instance also has an attribute value taken from the value domain.

4.4

feature catalogue

catalogue containing definitions and descriptions of the **feature** (4.1) types, **feature attributes** (4.3), and **feature associations** (4.2) occurring in one or more sets of geographic data, together with any feature operations that may be applied

4.5

feature operation

operation that every instance of a **feature** (4.1) type may perform

EXAMPLE A feature operation upon a 'dam' is to raise the dam. The results of this operation are to raise the height of the 'dam' and the level of water in a 'reservoir'.

NOTE Sometimes feature operations provide a basis for feature type definition.

4.6

functional language

language in which **feature operations** are formally specified

NOTE In a functional language, feature types may be represented as abstract data types.

5 Abbreviations

DIGEST Digital Geographic Information Exchange Standard

FACC Feature and Attribute Coding Catalogue

GFM General Feature Model

HTTP Hyper Text Transfer Protocol

IHO International Hydrographic Organization

TS Technical Specification

UML Unified Modeling Language

URI Uniform Resource Identifier

XML eXtensible Markup Language

6 Principal requirements

6.1 Feature catalogue

A feature catalogue shall present the abstraction of reality represented in one or more sets of geographic data as a defined classification of phenomena. The basic level of classification in a feature catalogue shall be the feature type. A feature catalogue shall be available in electronic form for any set of geographic data that contains features. A feature catalogue may also comply with the specifications of this International Standard independently of any existing set of geographic data.

6.2 Information elements

6.2.1 Introduction

The following clauses specify general and specific requirements for feature catalogue information elements. Annex B specifies detailed requirements. Annex C illustrates the application of these requirements. Annex D discusses the application of feature operations as the conceptual basis for determining feature types in a feature catalogue.

6.2.2 Completeness

A template for the representation of feature classification information is specified in Annex B. A feature catalogue prepared according to this template shall document all of the feature types found in a given set of geographic data. The feature catalogue shall include identification information as specified in Annex B. The feature catalogue shall include definitions and descriptions of all feature types contained in the data, including any feature attributes and feature associations contained in the data that are associated with each feature type, and optionally including feature operations that are supported by the data. To ensure predictability and comparability of feature catalogue content across different applications, it is recommended that the feature catalogue should include only the elements specified in Annex B. To maximize the usefulness of a feature catalogue across different applications, the use of a conceptual schema language to model feature catalogue information is recommended.

NOTE Natural-language definitions, feature-type aliases, criteria for the birth and death of feature instances, and other semantic elements of the feature catalogue may be included in a conceptual schema as structured comments or as attributes.

6.2.3 General requirements

6.2.3.1 Form of names

All feature types, feature attributes, feature associations, association roles, and feature operations included in a feature catalogue shall be identified by a name that is unique within that feature catalogue. If the name of a feature type, feature attribute, feature association, association role, or feature operation appears more than once in that feature catalogue, the definition shall be the same for all occurrences.

6.2.3.2 Form of definitions

Definitions of feature types, feature attributes, feature attribute listed values, feature associations, association roles, and feature operations shall be given in a natural language. These definitions shall be included in the catalogue, unless the catalogue specifies a separate definition source. If the same term appears in both the definition source and the feature catalogue, the definition in the feature catalogue shall apply.

6.2.4 Requirements for feature types

Each feature type shall be identified by a name and defined in a natural language. Each feature type may also be identified by an alphanumeric code that is unique within the catalogue and it may have a set of aliases. The feature catalogue shall also include, for each feature type, its feature operations and associated feature

attributes, feature associations and association roles, if any. The use of functional language specifications to help define feature types is recommended.

6.2.5 Requirements for feature operations

Feature operations, if any, shall be identified and defined for each feature type. Feature attributes involved in each feature operation shall be specified well as any feature types affected by the operation. The definition shall include a natural language definition and may be formally specified in a functional language.

6.2.6 Requirements for feature attributes

Feature attributes, if any, shall be identified and defined for each feature type. The definition shall include a natural language definition and a specified data type for values of the attribute. Each feature attribute may also be identified by an alphanumeric code that is unique within the catalogue.

6.2.7 Requirements for feature attribute listed values

Feature-attribute listed values, if any, shall be labelled for each feature attribute. The label shall be unique within the feature attribute of which it is a listed value. Each listed value may also be identified by an alphanumeric code that is unique within the feature attribute of which it is a listed value.

6.2.8 Requirements for feature associations

Feature associations, if any, shall be named and defined. Each feature association may also be identified by an alphanumeric code that is unique within the catalogue. The names and roles of the feature types that participate in the association shall be specified.

6.2.9 Requirements for association roles

Association roles, if any, shall be named and defined. The name of the feature type that holds the role and the association in which it participates shall be specified.

Annex A (normative)

Abstract test suite

A.1 Introduction

This normative annex presents the abstract test suite for evaluating conformance to this International Standard. This abstract test suite contains fifteen test cases and twelve test modules: a test case for the existence and form of feature catalogue information (A.2); a test case for general feature catalogue requirements (A.3); test cases for each principal feature catalogue information class (A.4 through A.16); and test modules for specific subsets of the functionality of a feature catalogue (A.17 through A.28).

Test cases are based on each principal-feature catalogue information class specified in Annex B, Tables B.1 through B.15. Each class-based test case consists of examining each class element (attribute or role) and verifying that:

- the obligation/condition specification for the presence of the element is met;
- the maximum number of occurrences of the element is not exceeded;
- the type of the value of the element is correct;

NOTE Unless otherwise stated, e.g., by specifying a type in a well-known package or ISO standard, the type specifications as will be given in ISO/TS 19103 may be applied.

- the value of the element is in accordance with the element description;
- any specified constraint on the element is met.

Tests on the class as a whole are specified as class-constraints as specified in Annex B, Tables B.1 through B.15, and/or in test modules.

Test modules are based on useful subsets of the functionality of the feature catalogue template specified in Annex B. Subsets are organized starting with core functionality capable of representing feature types and feature attributes that are unique to a feature type (see A.17). The core representation functionality is extended by supporting one or more of the following

- additional feature property types (for association roles, see A.18; for both association roles and feature operations, see A.19);
- additional relationships among feature catalogue information elements (for multiple-use feature attributes, see A.20; for inheritance, see A.23; for both multiple-use feature attributes and inheritance, see A.26);
- both additional feature property types and additional relationships among feature catalogue information elements (see A.21, A.22, A.24, A.25, A.27 and A.28).

Test modules are specified by the test cases that apply. Test modules A.18 through A.28 extend the core representation functionality specified by test module A.17. These extensions are summarized in Table 1.

To check that a feature catalogue conforms to this International Standard, verify that all of the requirements in at least one test module are satisfied.

A.2 Test case for existence and form of feature catalogue information

Information for the test case is as follows:

- a) test purpose: verify the existence and form of a feature catalogue;
- b) test method: check whether the feature catalogue exists and can be obtained in electronic form, by obtaining a copy of the feature catalogue such as on a computer disk or through a file transfer;
- c) reference: 6.1;
- d) test type: basic.

A.3 Test case for general feature catalogue requirements

Information for the test case is as follows:

- a) test purpose: verify that general feature catalogue requirements are met;
- b) test method: check
 - 1) if the feature catalogue is specified as applying to a given set of geographic data, then the feature catalogue documents all of the feature types found in that set of geographic data,
 - 2) whether all feature types, feature attributes, feature associations, association roles, and feature operations are identified by a name that is unique within the feature catalogue,
 - 3) whether all feature attribute listed values are identified by a label that is unique within the feature attribute of which it is a listed value,
 - 4) whether all feature types, feature attributes, feature associations, association roles, and feature operations are either defined or reference a definition from another source,
 - 5) if any feature type, feature attribute, or feature association is identified by an alphanumeric code, that alphanumeric code is unique within the feature catalogue,
 - 6) if any feature attribute listed value is identified by an alphanumeric code, that alphanumeric code is unique within the feature attribute of which it is a listed value;
- c) reference: 6.2;
- d) test type: capability.

A.4 Test case for the feature catalogue class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class feature catalogue;
- b) test method: test each attribute and role listed in Table B.1 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.1;
- d) test type: capability.

A.5 Test case for the feature type class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class feature type;
- b) test method: test each attribute and role listed in Table B.2 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.2;
- d) test type: capability.

A.6 Test case for the inheritance relation class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class inheritance relation;
- b) test method: test each attribute and role listed in Table B.3 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.3;
- d) test type: capability.

A.7 Test case for the feature operation class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class feature operation;
- b) test method: test each attribute and role listed in Table B.5 (and Table B.4) by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied. In addition:
 - 1) If feature attributes are unique to a feature type (test modules A.19 and A.25), the type of the following roles shall be FC_FeatureAttribute rather than FC_BoundFeatureAttribute:
 - i) FC_FeatureOperation::triggeredByValuesOf (Table B.5, line 5.3),
 - ii) FC_FeatureOperation::observesValuesOf (Table B.5, line 5.4),
 - iii) FC_FeatureOperation::affectsValuesOf (Table B.5, line 5.5);
 - 2) If the role FC_FeatureOperation::featureType (Table B.4, line 4.4) exists more than once, the association class FC_Binding for each shall satisfy test case A.8. This condition occurs in test modules A.22 and A.28;
- c) reference: Annex B, Tables B.4 and B.5;
- d) test type: capability.

A.8 Test case for the binding class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class binding;
- b) test method: test each attribute listed in Table B.6 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.6;
- d) test type: capability.

A.9 Test case for the constraint class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class constraint;
- b) test method: test each attribute listed in Table B.7 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.7;
- d) test type: capability.

A.10 Test case for the feature attribute class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class feature attribute;
- b) test method: test each attribute and role listed in Table B.8 (and Table B.4) by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied. In addition:

If the role FC_FeatureAttribute::featureType (Table B.4, line 4.4) exists more than once, the association class FC_Binding for each shall satisfy test case A.8. This condition occurs in test modules A.20, A.21, A.22, A.26, A.27 and A.28;
- c) reference: Annex B, Tables B.4 and B.8;
- d) test type: capability.

A.11 Test case for the association role class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class association role;
- b) test method: test each attribute and role listed in Table B.9 (and Table B.4) by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied. In addition:
 - 1) Verify that the value of attribute type (Table B.9, line 9.2) is a domain code from Table B.10,

- 2) If the role `FC_AssociationRole::featureType` (Table B.4, line 4.4) exists more than once, the association class `FC_Binding` for each shall satisfy test case A.8. This condition occurs in test modules A.21, A.22, A.27 and A.28;

- c) reference: Annex B, Tables B.4, B.9 and B.10;
d) test type: capability.

A.12 Test case for the listed value class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class listed value;
b) test method: test each attribute and role listed in Table B.11 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
c) reference: Annex B, Table B.11;
d) test type: capability.

A.13 Test case for the feature association class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class feature association;
b) test method: test each attribute and role listed in Table B.12 (and Table B.2) by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
c) reference: Annex B, Tables B.2 and B.12;
d) test type: capability.

A.14 Test case for the definition source class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class definition source;
b) test method: test each attribute listed in Table B.13 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
c) reference: Annex B, Table B.13;
d) test type: capability.

A.15 Test case for the definition reference class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class definition reference;

- b) test method: test each attribute and role listed in Table B.14 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.14;
- d) test type: capability.

A.16 Test case for the bound feature attribute class

Information for the test case is as follows:

- a) test purpose: verify that the required information is included in objects of class bound feature attribute;
- b) test method: test each role listed in Table B.15 by verifying that for each the specified description, obligation/condition, maximum occurrence, type, and constraint are satisfied;
- c) reference: Annex B, Table B.15;
- d) test type: capability.

A.17 Test module for a catalogue with single-use feature attributes

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types and feature attributes that are unique to a feature type;
- b) test method: perform the following nine test cases: A.2 (existence), A.3 (general), A.4 (feature catalogue), A.5 (feature type), A.9 (constraint), A.10 (feature attribute), A.12 (listed value), A.14 (definition source) and A.15 (definition reference);
- c) reference: A.2, A.3, A.4, A.5, A.9, A.10, A.12, A.14 and A.15;
- d) test type: capability.

A.18 Test module for a catalogue with single-use feature attributes and association roles

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes and association roles that are unique to a feature type;
- b) test method: perform test module A.17 and the two test cases A.11 (association role) and A.13 (feature association);
- c) reference: A.11, A.13 and A.17;
- d) test type: capability.

A.19 Test module for a catalogue with single-use feature attributes, association roles and operations

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes, association roles and feature operations that are unique to a feature type;
- b) test method: perform test module A.18 and test case A.7 (feature operation), including verification that the type of the following FC_FeatureOperation roles shall be FC_FeatureAttribute rather than FC_BoundFeatureAttribute:
 - 1) FC_FeatureOperation::triggeredByValuesOf (Table B.5, line 5.3),
 - 2) FC_FeatureOperation::observesValuesOf (Table B.5, line 5.4), and
 - 3) FC_FeatureOperation::affectsValuesOf (Table B.5, line 5.5);
- c) reference: A.7; A.18 and Annex B, Table B.5;
- d) test type: capability.

A.20 Test module for a catalogue with multiple-use feature attributes

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types and feature attributes that may be members of multiple feature types;
- b) test method: perform test module A.17 and in addition:

If the role FC_FeatureAttribute::featureType (Table B.4, line 4.4) exists more than once for the same FC_FeatureAttribute, the association class FC_Binding for each shall satisfy test case A.8 (binding);
- c) reference: A.8; A.17 and Annex B, Table B.4;
- d) test type: capability.

A.21 Test module for a catalogue with multiple-use feature attributes and association roles

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes and association roles that may be members of multiple feature types;
- b) test method: perform test module A.18 and in addition:
 - 1) If the role FC_FeatureAttribute::featureType (Table B.4, line 4.4) exists more than once for the same FC_FeatureAttribute the association class FC_Binding for each shall satisfy test case A.8 (binding),
 - 2) If the role FC_AssociationRole::featureType (Table B.4, line 4.4) exists more than once for the same FC_AssociationRole the association class FC_Binding for each shall satisfy test case A.8 (binding);

- c) reference: A.8, A.18 and Annex B, Table B.4;
- d) test type: capability.

A.22 Test module for a catalogue with multiple-use feature attributes, associations and operations

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes, association roles and feature operations that may be members of multiple feature types;
- b) test method: perform test module A.19 and the test case A.16 (bound feature attribute). In addition:
 - 1) If the role FC_FeatureAttribute::featureType (Table B.4, line 4.4) exists more than once for the same FC_FeatureAttribute, the association class FC_Binding for each shall satisfy test case A.8 (binding),
 - 2) If the role FC_AssociationRole::featureType (Table B.4, line 4.4) exists more than once for the same FC_AssociationRole, the association class FC_Binding for each shall satisfy test case A.8 (binding),
 - 3) If the role FC_FeatureOperation::featureType (Table B.4, line 4.4) exists more than once for the same FC_FeatureOperation, the association class FC_Binding for each shall satisfy test case A.8 (binding);
- c) reference: A.8, A.16; A.19 and Annex B, Table B.4;
- d) test type: capability.

A.23 Test module for a catalogue with single-use feature attributes and inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types and feature attributes that are unique to a feature type, plus inheritance;
- b) test method: perform test module A.17 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.17;
- d) test type: capability.

A.24 Test module for a catalogue with single-use feature attributes and association roles with inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes and association roles that are unique to a feature type, plus inheritance;
- b) test method: perform test module A.18 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.18;
- d) test type: capability.

A.25 Test module for a catalogue with single-use feature attributes, association roles and operations with inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes, association roles and feature operations that are unique to a feature type, plus inheritance;
- b) test method: perform test module A.19 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.19;
- d) test type: capability.

A.26 Test module for a catalogue with multiple-use feature attributes and inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types and feature attributes that may be members of multiple feature types, plus inheritance;
- b) test method: perform test module A.20 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.20;
- d) test type: capability.

A.27 Test module for a catalogue with multiple-use feature attributes and association roles with inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes and association roles that may be members of multiple feature types, plus inheritance;
- b) test method: perform test module A.21 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.21;
- d) test type: capability.

A.28 Test module for a catalogue with multiple-use feature attributes, association roles and operations with inheritance

Information for the test module is as follows:

- a) test purpose: verify that the feature catalogue supports core representation functionality, comprised of feature types, and feature attributes, association roles and feature operations that may be members of multiple feature types, plus inheritance;
- b) test method: perform test module A.22 and the test case A.6 (inheritance relation);
- c) reference: A.6, A.22;
- d) test type: capability.

Annex B (normative)

Feature catalogue template

This normative annex presents the template for the organization of feature catalogue information according to this International Standard. Tables B.1 through B.15 present templates for the representation of feature catalogue contents. Feature catalogue information elements and the relationships among them are identified using the following notation for obligation and conditions:

- M – The element is mandatory; it shall be included in the feature catalogue.
- C – The element is conditional; the condition is stated as a question. If the answer to the question is yes, the element shall be included in the feature catalogue.
- O – The element is optional; if an element is included in the feature catalogue, mandatory sub-elements of the element shall also be included.

Table B.15 and Figure B.1 illustrate the feature catalogue template in the form of a Unified Modeling Language (UML) package (ISO/TS 19103). Figures B.2 through B.5 illustrate how the structure of a standard feature catalogue conforms to the General Feature Model (ISO 19109 —¹, 7.3).

Table B.1 — Feature catalogue

No.	Name/Role Name	Description	Obligation/ Condition ^a	Maximum Occurrence ^b	Type	Constraint
1	Class FC_FeatureCatalogue	A feature catalogue contains its identification and contact information, and definition of some number of feature types with other information necessary for those definitions.	—	—	—	—
1.1	Attribute name	name for this feature catalogue	M	1	CharacterString	—
1.2	Attribute scope	subject domain(s) of feature types defined in this feature catalogue	M	N	CharacterString	—
1.3	Attribute fieldOfApplication	description of kind(s) of use to which this feature catalogue may be put	O	N	CharacterString	—
1.4	Attribute versionNumber	version number of this feature catalogue, which may include both a major version number or letter and a sequence of minor release numbers or letters, such as “3.2.4a.” The format of this attribute may differ between cataloguing authorities.	M	1	CharacterString	—
1.5	Attribute versionDate	effective date of this feature catalogue	M	1	Date	—
1.6	Attribute producer	name, address, country, and telecommunications address of person or organization having primary responsibility for the intellectual content of this feature catalogue	M	1	ISO 19115 Metadata:: CI_ResponsibleParty	—
1.7	Attribute functionalLanguage	formal functional language in which the feature operation formal definition occurs in this feature catalogue	C/Mandatory if feature operation formal definition occurs in feature catalogue.	1	CharacterString	—
1.8	Role featureType	role that links this feature catalogue to the feature types that it contains	M	N	FC_FeatureType	Aggregation
1.9	Role definitionSource	role that links this feature catalogue to the sources of definitions of feature types, property types, and listed values that it contains	O	N	FC_DefinitionSource	Aggregation
^a M = mandatory; O = optional; C = conditional. ^b N = repeating occurrences.						

Table B.2 — Feature type

No.	Name/Role Name	Description	Obligation/ Condition ^a	Maximum Occurrence ^b	Type	Constraint
2	Class FC_FeatureType	class of real world phenomena with common properties	—	—	—	name realizes GF_FeatureType ::typeName; isAbstract realizes GF_FeatureType ::isAbstract; constrainedBy realizes GF_FeatureType ::constrainedBy
2.1	Attribute typeName	text string that uniquely identifies this feature type within the feature catalogue that contains this feature type	M	1	LocalName	—
2.2	Attribute definition	definition of the feature type in a natural language. This attribute is required if the definition is not provided by FC_FeatureCatalogue::definitionSource. If not provided, the definitionReference should specify a citation where the definition may be found, and any additional information as to which definition is to be used.	C/Mandatory if definition not provided by definition source.	1	CharacterString	—
2.3	Attribute code	code that uniquely identifies this feature type within the feature catalogue that contains this feature type	O	1	CharacterString	—
2.4	Attribute isAbstract	indicates if the feature type is abstract or not	M	1	Boolean	Initial value = FALSE
2.5	Attribute aliases	equivalent name(s) of this feature type	O	N	LocalName	—
2.6	Role inheritsFrom	role that links this feature type to a set of superclasses from which it inherits operations, associations and properties	O	N	FC_InheritanceRelation	—
2.7	Role inheritsTo	role that links this feature type to a set of subclasses which inherit its operations, associations and properties	O	N	FC_InheritanceRelation	—
2.8	Role featureCatalogue	role that links this feature type to the feature catalogue that contains it	M	1	FC_FeatureCatalogue	—
2.9	Role carrierOfCharacteristics	role that links this feature type to the property types that it contains The association class FC_Binding describes particular information regarding the use of this property type within this feature type.	O	N	FC_PropertyType; FC_Binding	Association Class is FC_Binding; Aggregation

Table B.2 (continued)

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
2.10	Role constrainedBy	role that links this feature type to the constraints placed upon it	O	N	FC_Constraint	—
2.11	Role definitionReference	role that links this feature type to the source of its definition	O	1	FC_DefinitionReference	—
^a M = mandatory; O = optional; C = conditional.						
^b N = repeating occurrences.						

Table B.3 — Inheritance relation

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
3	Class FC_InheritanceRelation	FC_InheritanceRelation realizes GF_InheritanceRelation.	—	—	—	FC_InheritanceRelation always assumes that its GF_InheritanceRelation::uniqueInstance is TRUE.
3.1	Attribute name	text string that uniquely identifies this inheritance relation within the feature catalogue that contains this inheritance relation	M	1	CharacterString	—
3.2	Attribute description	natural language description of this inheritance relation	M	1	CharacterString	—
3.3	Attribute uniqueInstance	indicates if an instance of the supertype can be an instance of at most one of its subtypes	M	1	Boolean	—
3.4	Role subtype	identifies one feature type to which the associated superclass feature type supplies inherited properties, associations and operation	M	1	FC_FeatureType	—
3.5	Role supertype	identifies one feature type from which the associated subtype class inherits properties, associations and operations	M	1	FC_FeatureType	—
^a M = mandatory.						

Table B.4 — Property type

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
4	Class FC_PropertyType	abstract class for feature properties	—	—	—	—
4.1	Attribute memberName	member name that locates this member within a feature type	M	1	LocalName	—
4.2	Attribute definition	definition of the member in a natural language. This attribute is required if the definition is not provided by FC_FeatureCatalogue::definitionSource. If not provided, the definitionReference should specify a citation where the definition may be found, and any additional information as to which definition is to be used.	C/Mandatory if definition not provided by definition source.	1	CharacterString	—
4.3	Attribute cardinality	cardinality of the member in the feature class. If this is an attribute or operation, the default cardinality is 1. If this is an association role, then the default cardinality is 0..*. For operations, this is the number of return values possible. This is an elaboration of the GFM to allow for complete specifications for various programming and data definition languages.	M	1	Multiplicity	Initial value =1
4.4	Role featureType	role that links the operations, attributes and association roles with feature types that contain them. The association class FC_Binding describes particular information regarding the use of this property type within this feature type. This is a "strong aggregation" or composition in the General Feature Model (ISO 19109). Here it is realized by a "weak aggregation". This is valid since a weak aggregation can be converted to a strong aggregation by replicating the value of the target role for each use. In this case, the property type represents a value, and the realization of the GFM's property type would create a GF_PropertyType whose identity is the combination of the FC_PropertyType and the owning FC_FeatureType.	O	N	FC_FeatureType; FC_Binding	Association Class is FC_Binding; Aggregation
4.5	Role constrainedBy	role that links this property type to the constraints placed upon it	O	N	FC_Constraint	—
4.6	Role definitionReference	role that links this instance to the source of its definition	O	1	FC_DefinitionReference	—
^a M = mandatory; O = optional; C = conditional. ^b N = repeating occurrences.						

Table B.5 — Feature operation

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
5	Class FC_FeatureOperation	operation that every instance of an associated feature type must implement	—	—	—	triggeredByValuesOf realizes GF_Operation::triggeredByValuesOf; observesValuesOf realizes GF_Operation::observesValuesOf; affectsValuesOf realizes GF_Operation::affectsValuesOf
	Subtype of FC_PropertyType	Table B.4 — Property type	—	—	—	—
5.1	Attribute signature	name and parameters for this operation. It may contain optional returned parameters. This signature is usually derived from the formalDefinition. The signature of an operation must be unique. This is the equivalent of the UML signature.	M	1	Character String	—
5.2	Attribute formalDefinition	formal description of the behaviour of the member, expressed in the symbol set defined by FC_FeatureCatalogue::functionalLanguage; involves operational parameters, and interactions with other members of the feature type.	O	1	Character String	—
5.3	Role triggeredByValuesOf	specifies attribute which may trigger an operation	O	N	FC_Bound FeatureAttribute	—
5.4	Role observesValuesOf	specifies attribute that may be used as input to perform an operation	O	N	FC_Bound FeatureAttribute	—
5.5	Role affectsValuesOf	specifies attribute that will be affected by an operation	O	N	FC_Bound FeatureAttribute	—
^a M = mandatory; O = optional. ^b N = repeating occurrences.						

Table B.6 — Binding

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence	Type	Constraint
6	Class FC_Binding	class that is used to describe the specifics of how a property type is bound to a particular feature type; used as an association class for the association MemberOf between feature type and property type	—	—	—	—
6.1	Attribute description	description of how a property type is bound to a particular feature type	O	1	CharacterString	—
^a O = optional.						

Table B.7 — Constraint

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence	Type	Constraint
7	Class FC_Constraint	class for defining constraints for types	—	—	—	—
7.1	Attribute description	description of the constraint that is being applied	M	1	CharacterString	—
^a M = mandatory.						

Table B.8 — Feature attribute

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
8	Class FC_FeatureAttribute	characteristic of a feature type	—	—	—	—
	Subtype of FC_PropertyType	Table B.4 — Property type	—	—	—	—
8.1	Attribute code	numeric or alphanumeric code that uniquely identifies the feature attribute within the feature catalogue	O	1	CharacterString	—
8.2	Attribute valueMeasurementUnit	unit of measure used for values of this feature attribute	O	1	UnitOfMeasure	—
8.3	Attribute listedValue	permissible values of this feature attribute. If present, then this feature attribute is enumerated (such as with a code list). If not present, then this feature attribute is not enumerated.	C/Mandatory if feature attribute valueType is not given.	N	FC_ListedValue	—
8.4	Attribute valueType	type of the value of this feature attribute; a name from some namespace. Implementations of this International Standard shall specify which namespace implementation is to be used. One possibility is the URI (RFC 2396) ^[6] .	C/Mandatory if feature attribute listedValue is empty.	1	TypeName	—
^a O = optional; C = conditional.						
^b N = repeating occurrences.						

Table B.9 — Association role

No.	Name/Role Name	Description	Obligation/ Condition ^a	Maximum Occurrence	Type	Constraint
9	Class FC_AssociationRole	role of the feature association FC_AssociationRole::relation.	—	—	—	roleName = FC_PropertyType::memberName; FC_PropertyType::cardinality realizes GF_AssociationRole::cardinality
	Subtype of FC_PropertyType	Table B.4 — Property type	—	—	—	—
9.1	Attribute cardinality	number of instances of the feature type that can act in this role relative to a single instance of the target feature type of the association	M	1	Multiplicity	Initial value = 0..*
9.2	Attribute type	type of association role, indicating whether this role acts as a “is part of” or “is a member of” semantics.	M	1	FC_RoleType	Initial value = 1 (“ordinary”)
9.3	Attribute isOrdered	indicates if the instances of this association role within the containing feature instance are ordered or not, with FALSE = “not ordered” and TRUE = “ordered”. If TRUE, the FC_PropertyType::definition shall contain an explanation of the meaning of the order.	M	1	Boolean	Initial value = FALSE
9.4	Attribute isNavigable	indicates whether this role is navigable from the source feature to the target feature of the association	M	1	Boolean	Initial value = TRUE
9.5	Role relation	relation of which this association role is a part	M	1	FC_Feature Association	—
9.6	Role valueType	type of the target value of this association role	M	1	FC_Feature Type	—
^a M = mandatory.						

Table B.10 — Role type code list

No.	Name	Domain Code	Description
10	Class FC_RoleType	—	code list for the classification of roles
10.1	ordinary	1	indicates an ordinary association
10.2	aggregation	2	indicates a UML aggregation (part role)
10.3	composition	3	indicates a UML composition (member role)

Table B.11 — Listed value

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence	Type	Constraint
11	Class FC_ListedValue	value for an enumerated feature attribute domain, including its codes and interpretation	—	—	—	—
11.1	Attribute label	descriptive label that uniquely identifies one value of the feature attribute	M	1	CharacterString	—
11.2	Attribute code	numeric or alphanumeric code (such as a country code) that uniquely identifies this value of the feature attribute	O	1	CharacterString	—
11.3	Attribute definition	definition of the attribute value in a natural language. If not provided, the definitionReference may specify a citation where the definition may be found, and any additional information as to which definition is to be used.	O	1	CharacterString	—
11.4	Role definitionReference	role that links this instance to the source of its definition	O	1	FC_DefinitionReference	—
^a M = mandatory; O = optional.						

Table B.12 — Feature association

	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
12	Class FC_FeatureAssociation	relationship that links instances of this feature type with instances of the same or of a different feature type. The memberOf-linkBetween association in the General Feature Model is not directly implemented here since it can be easily derived from combining the Role and MemberOf associations.	—	—	—	—
	Subtype of FC_FeatureType	Table B.2 — Feature type	—	—	—	—
12.1	Role role	roles that are a part of this association	M	N	FC_AssociationRole	Aggregation
^a M = mandatory.						
^b N = repeating occurrences.						

Table B.13 — Definition source

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence ^b	Type	Constraint
13	Class FC_DefinitionSource	class that specifies the source of a definition	—	—	—	—
13.1	Attribute source	actual citation of the source, sufficient to identify the document and how to obtain it	M	1	ISO 19115 Metadata:: CI_Citation	—
^a M = mandatory.						

Table B.14 — Definition reference

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence	Type	Constraint
14	Class FC_DefinitionReference	class that links a data instance to the source of its definition	—	—	—	—
14.1	Attribute sourceIdentifier	additional information to help locate the definition in the source document. The format of this information is specific to the structure of the source document.	O	1	CharacterString	—
14.2	Role definitionSource	role that links this definition reference to the citation for the source document	M	1	FC_DefinitionSource	—
^a M = mandatory; O = optional.						

Table B.15 — Bound feature attribute

No.	Name/Role Name	Description	Obligation/Condition ^a	Maximum Occurrence	Type	Constraint
15	Class FC_BoundFeatureAttribute	class that represents an association between a particular feature type and a particular property type, in order that operational effect information may be supplied for feature operations	—	—	—	—
15.1	Role featureType	feature type involved in the binding	M	1	FC_FeatureType	—
15.2	Role attribute	property type involved in the binding	M	1	FC_FeatureAttribute	—
^a M = mandatory.						

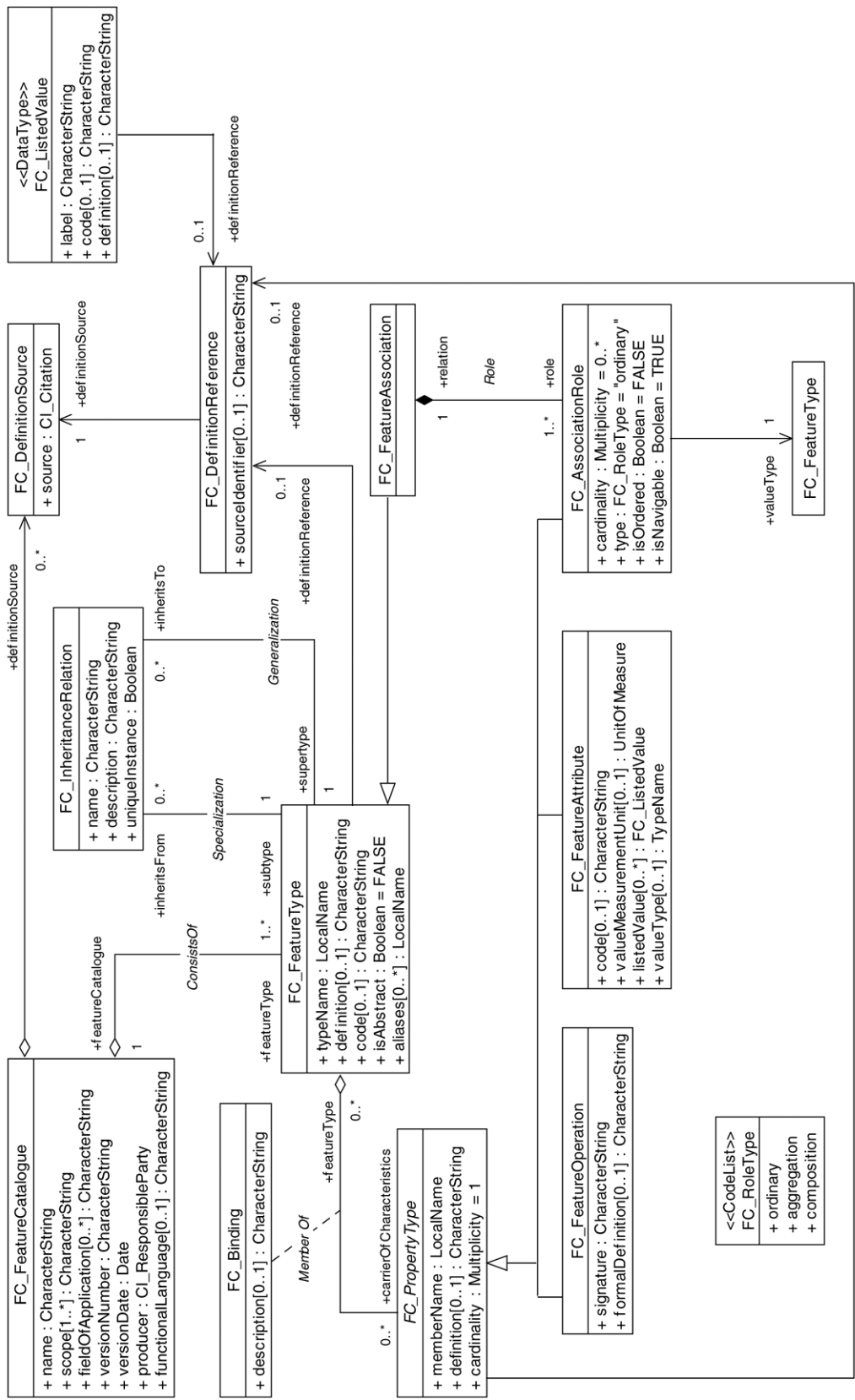


Figure B.1 — Conceptual model of a feature catalogue

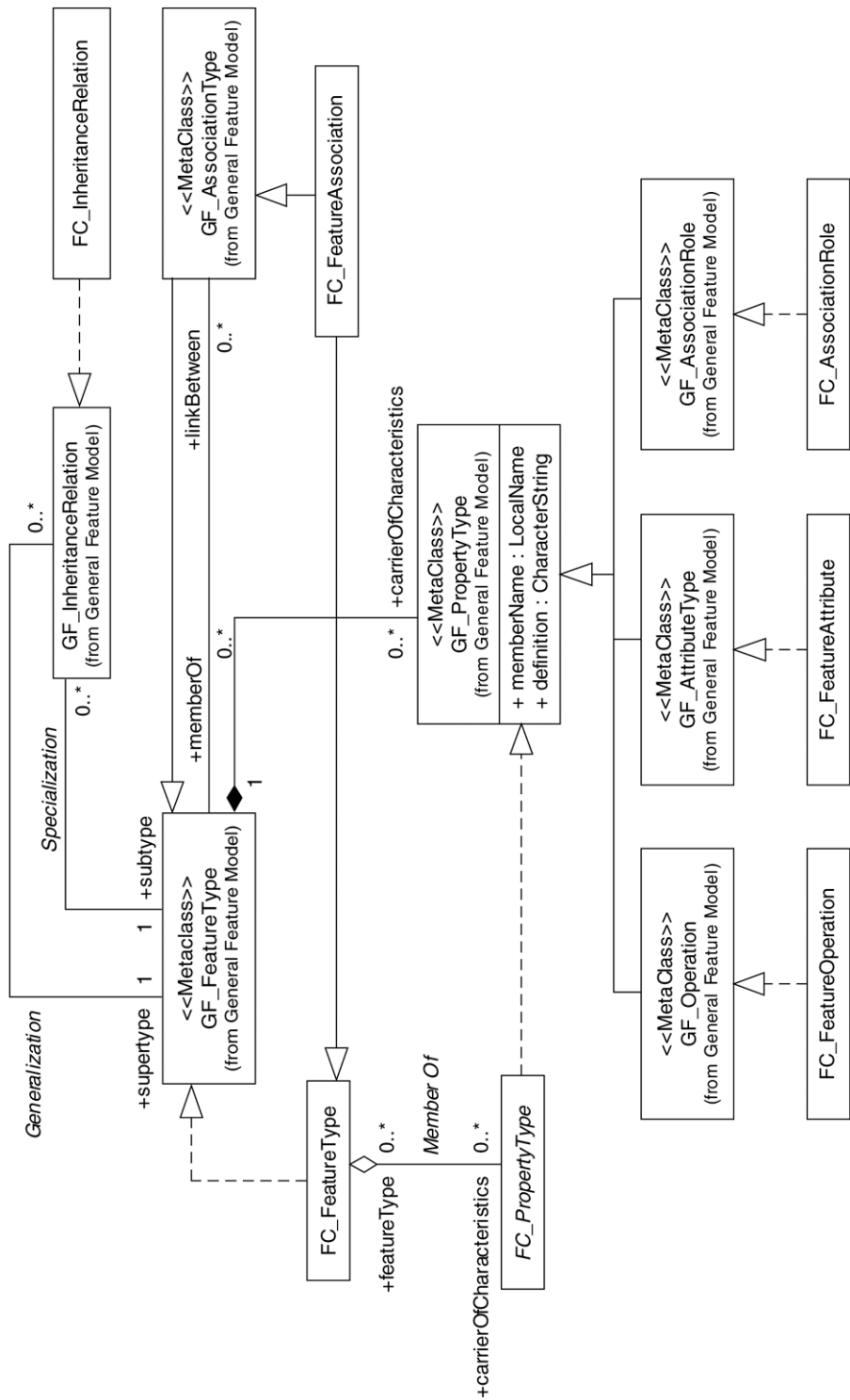


Figure B.2 — Feature cataloguing classes as realizations of General Feature Model metaclasses