

S-100 – Part 3

Общая модель фичеров и Правила прикладной схемы

Содержание

3-1 Цель	5
3-2 Соответствие	5
3-3 Ссылки	6
3-4 Контекст	7
3-4.1 Объекты	7
3-4.2 Происхождение Общей модели фичеров	7
3-5 Принципы определения типов фичеров и информации	7
3-5.1 Идентифицируемые объекты.....	7
3-5.1.1 Фичеры	7
3-5.1.2 Типы информации	8
3-5.2 Общая модель фичеров	8
3-5.2.1 Введение	8
3-5.2.2 Назначение GFM	9
3-5.2.3 Основная структура GFM	9
3-5.2.4 S100_GF_NamedType	9
3-5.2.5 S100_GF_ObjectType	10
3-5.2.6 S100_GF_FeatureType	10
3-5.2.7 S100_GF_PropertyType	11
3-5.2.8 S100_GF_AttributeType	12
3-5.2.9 S100_GF_AssociationRole	12
3-5.2.10 GF_Operation	12
3-5.2.11 100_GF_AssociationType.....	13
3-5.2.12 S100_GF_InformationType	13
3-5.2.13 S100_GF_FeatureAssociationType	14
3-5.2.14 S100_GF_InformationAssociationType	15
3-5.2.15 S100_GF_Constraint	16
3-5.3 Атрибуты типов фичеров	17
3-5.3.1 Введение	17

3-5.3.2 S100_GF_ThematicAttributeType	17
3-5.3.3 S100_GF_ComplexAttributeType	18
3-5.3.4 S100_GF_SimpleAttributeType	18
3-5.3.5 S100_GF_SpatialAttributeType	18
3-5.3.6 GF_TemporalAttributeType	20
3-5.3.7 GF_MetadataAttributeType	20
3-5.3.8 GF_QualityAttributeType	20
3-5.3.9 GF_LocationAttributeType	20
3-5.3.10 S100_TruncatedDateAttributeType.....	21
3-5.3.11 S100_GF_CodelistAttributeType	21
3-5.3.12 S100_GF_EnumerationType	21
3-5.4 Взаимосвязи между поименованными типами.....	21
3-5.4.1 Введение	21
3-5.4.2 GF_InheritanceRelation	22
3-5.4.3 S100_GF_AssociationType	22
3-5.4.4 Ассоциации с типами информации	23
3-5.4.5 Имена по умолчанию для концов ассоциаций	23
3-5.5 Поведение типов фичеров	24
3-5.6 Ограничения	24
3-6 Правила для прикладных схем (ISO 19109 Пункт 8)	24
3-6.1 Процесс моделирования прикладных схем (ISO 19109 Пункт 8.1)	24
3-6.2 Прикладная схема (ISO 19109 Пункт 8.2)	25
3-6.2.1 Язык концептуальной схемы для прикладных схем	25
3-6.2.2 Основные правила	25
3-6.2.3 Идентификация прикладных схем	25
3-6.2.4 Документация прикладной схемы	25
3-6.3 Правила для прикладной схемы в UML (ISO 19109 Пункт 8.3)	26
3-6.3.1 Основные правила (ISO 19109 Пункт 8.3.1)	26
3-6.4 Профили доменов стандартных схем в UML (ISO 19109 Пункт 8.4)	27
3-6.4.1 Правила добавления информации в стандартную схему	27
3-6.4.2 Ограниченное использование стандартных схем	27

3-6.4.3	Правила использования схемы метаданных (ISO 19109 Пункт 8.5)	27
3-6.4.4	Правила времени (ISO 19109 Пункт 8.6)	27
3-6.5	Пространственные правила (ISO 19109 Пункт 8.7)	28
3-6.5.1	Общие пространственные правила (ISO 19109 Пункт 8.7.1)	28
3-6.5.2	Пространственные атрибуты	28
3-6.5.3	Пространственное качество	28
3-6.5.4	Геометрические агрегаты и комплексы для представления пространственных атрибутов фичеров	29
3-6.6	Правила каталогизации (ISO 19109 Пункт 8.8)	30
3-6.6.1	Введение (ISO 19109 Пункт 8.8.1)	30
3-6.6.2	Прикладная схема на основе каталога фичеров (ISO 19109 Пункт 8.8.2)	30
3-6.6.3	Кодирование символов	30
3-6.7	Списки кодов (Codelists)	30
3-7	Прикладная схема для покрытий (информативная)	31
3-7.1	Введение	31
3-7.2	Данные, привязанные к сетке	32
3-7.3	Сетка с переменным размером ячеек	33
3-7.4	Изображение, ориентированное на фичер	34
3-8	Временные рамки S-100	36
3-8.1	Временные определения	36
3-8.2	Временные взаимосвязи	36
3-8.3	Интерпретация моделей интервалов времени и периода	37
3-9	Использование типов специфических форматов для усеченных дат	38
3-10	Идентификаторы реализаций	38
3-10.1	Использование идентификаторов реализаций	39

3-1 Цель

Эта часть описывает Общую модель фичеров (General Feature Model - GFM), которая является концептуальной моделью фичеров, их характеристик и ассоциаций. Она также описывает правила разработки прикладной схемы, являющейся базовой частью любой спецификации продукта, созданной в соответствии с S-100.

Целями этой части являются:

- 1) Концептуальное моделирование фичеров и их свойств, полученных из реального мира;
- 2) Концептуальное моделирование типов информации и их атрибутов;
- 3) Определение прикладной схемы;
- 4) Правила прикладной схемы;

Следующие пункты остаются вне целей:

- 1) Представление типов фичеров, их свойств и типов информации и их свойств в каталоге;
- 2) Представление метаданных;
- 3) Правила меппинга одной прикладной схемы в другую;
- 4) Реализация прикладной схемы на компьютере;
- 5) Дизайн компьютерной системы и программного обеспечения прикладной схемы;
- 6) Программирование.

Компьютерные системы, дизайн программного обеспечения и программирование не относятся к теме данного документа.

3-2 Соответствие

Этот профиль соответствует классу 2 ISO 19106:2004. Далее дается краткая специализация и генерализация того, где и в чем Общая модель фичеров S-100 отличается от ISO 19109.

- 1) Введен новый S100_GF_NamedType.
- 2) Введен новый S100_GF_ObjectType, как специализация S100_GF_NamedType.
- 3) Введен новый S100_GF_InformationType, как специализация S100_GF_ObjectType, он ограничен до ассоциаций с S100_GF_ThematicAttributeType.
- 4) S100_GF_FeatureType является специализацией S100_GF_ObjectType.

- 5) S100_GF_AttributeType является специализацией GF_AttributeType, это абстрактный тип в S-100.
 - 6) Введен новый абстрактный S100_GF_SimpleAttributeType, являющийся специализацией S100_GF_ThematicAttributeType.
 - 7) GF_Operation не используется.
 - 8) GF_InheritanceRelation не используется; наследственность фичеров представляется наследственностью ассоциации.
 - 9) Ассоциация attributeOfAttribute не используется. Для выполнения аналогичных функций в S-100 используется комплексный атрибут.
 - 10) S100_GF_AssociationType не использует ассоциацию генерализации между GF_AssociationType и GF_FeatureType. Вместо нее используется специализация S100_GF_NamedType.
 - 11) S100_GF_AssociationType ассоциируется с S100_GF_ThematicAttributeType посредством агрегативной связи UML. Это означает, что ассоциация может иметь описательные характеристики.
 - 12) Новые мета классы S100_GF_FeatureAssociationType и S100_GF_InformationAssociationType введены как специализации S100_GF_AssociationType.
 - 13) Роль ассоциации linkBetween связи GF_FeatureType/GF_AssociationType в ISO 19109 реализуется следующим образом:
 - a) Роль linkBetween связи S100_FeatureType/S100_GF_FeatureAssociationType;
 - b) Роль linkBetween связи S100_InformationType / S100_GF_InformationAssociationType;
 - c) Роль informationLink связи S100_ObjectType / S100_InformationAssociationType.
- Это означает, что ассоциации, включающие только типы фичеров, имеют ограничения по семантике и множественности, что отличается от ассоциаций, включающих, как минимум, один тип информации.
- 14) GF_LocationAttributeType, GF_TemporalAttributeType, GF_MetaDataAttributeType и GF_QualityAttributeType не используются.

Дополнительные ссылки или пояснения к вышеуказанным изменениям можно найти в соответствующем тексте ниже.

3-3 Ссылки

ISO 8601:2004, *Data elements and interchange formats – Information interchange – Representation of dates and times*

ISO 19106:2003, *Geographic information - Geographic Information – Profiles*

ISO 19108:2002, *Geographical Information – Temporal Schema* (as corrected by Technical Corrigendum 1 – 2006)

ISO 19107:2003, *Geographic information - Spatial schema*

ISO 19109:2005, *Geographic information - Rules for application schema*

ISO 19110:2005, *Geographic information - Methodology for feature cataloguing*

ISO 19115-1:2018, *Geographic information – Metadata – Part 1 – Fundamentals* (as updated by Amendment 1, 2018)

ISO/CD 19115-2, *Geographic information - Metadata - Part 2 – Extensions for imagery and gridded data*

3-4 Контекст

3-4.1 Объекты

Содержание данных географических приложений определяется в соответствии с видом фичеров объектов реального мира и в контексте требований конкретного приложения. Содержимое структурируется в терминах объектов. Данный документ рассматривает два типа объектов:

- 1) Фичеры – фичеры определяются вместе с их свойствами.
- 2) Типы информации – типы информации используются для совместного применения фичерами и другими типами информации. Типы информации имеют только свойства тематических атрибутов.

GFM обеспечивает концептуальную модель для этих объектов. Определения типов фичеров содержатся в Каталоге фичеров. GFM также действует как концептуальная модель для Каталога фичеров.

3-4.2 Происхождение Общей модели фичеров

Концептуальная модель типов, которая будет использоваться в продуктах S-100 представлена в данном документе. Она известна как GFM и происходит из стандарта ISO 19109 Общая модель фичеров (General Feature Model) посредством реализации его классов (Рисунок 3-1).

3-5 Принципы определения типов фичеров и информации

3-5.1 Идентифицируемые объекты

3-5.1.1 Фичеры

Фичер – это абстрактное представление явлений реального мира. Фичеры имеют два аспекта – тип фичера и реализация фичера. Тип фичера является классом и

определяется в Каталоге фичеров. Реализация фичера является одиночным событием типа фичера и представляется как конкретный объект в наборе данных.

3-5.1.2 Типы информации

Тип информации – это класс объекта, определенный Каталогом фичеров. Реализацией типа информации является идентифицированный блок информации в наборе данных. Типы информации имеют только свойства тематических атрибутов. Реализации типа информации могут быть ассоциированы с одной или более реализацией фичеров или другими реализациями типов информации.

ПРИМЕР Примечание на карте может моделироваться как тип информации.

3-5.2 Общая модель фичеров (The General Feature Model (GFM))

3-5.2.1 Введение

Этот подраздел идентифицирует и описывает концепции, используемые для определения типов фичеров и информации и их взаимосвязей. Эти концепции выражаются в концептуальной модели, называемой GFM.

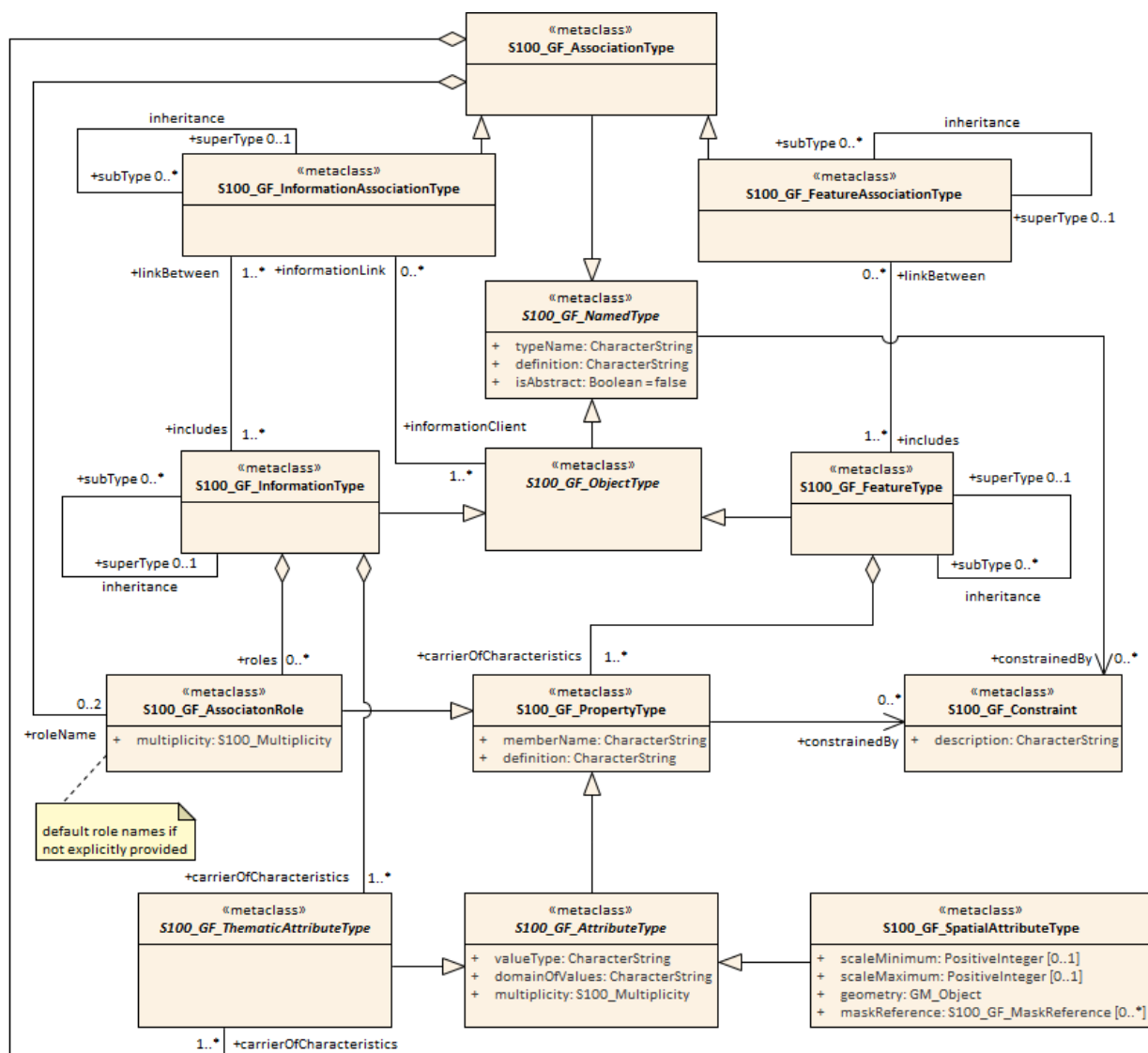


Рисунок 3-1 – Общая модель фичеров (GFM)

3-5.2.2 Назначение GFM

GFM является базисом для классификации типов фичеров и информации, и их свойств. GFM также действует как базис для структуры каталогов фичеров.

3-5.2.3 Основная структура GFM

Рисунок 3-1 показывает UML модель GFM S-100.

Следующие пункты определяют элементы GFM.

3-5.2.4 S100_GF_NamedType

Класс S100_GF_NamedType не реализуется из ISO 19109, но введен специально для GFM S-100. Это абстрактный суперкласс классов S100_GF_ObjectType и S100_GF_AssociationType. Цель введения этого класса - показать общность между

типами фичеров и типами ассоциаций S-100. Оба типа являются ключевыми идентифицируемыми объектами схем данных S-100.

Таблица 3-1— S100_GF_NamedType (поименованные типы)

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_NamedType	Абстрактный базовый класс для типов объектов и типов ассоциаций в GFM	-	-
Атрибут	typeName	Имя поименованного типа. Имя должно быть уникальным в пространстве имен.	1	CharacterString
Атрибут	definition	Определение, описывающее поименованный тип.	1	CharacterString
Атрибут	isAbstract	Если true, поименованный тип действует как абстрактный супертип. Невозможно создать реализацию абстрактного типа.	1	Boolean
Роль	constrainedBy	Роль указывает, что для поименованного типа создается ограничение	0..*	S100_GF_Constraint

3-5.2.5 S100_GF_ObjectType (типы объектов)

Класс S100_GF_ObjectType не реализован из ISO 19109, но введен специально для GFM S-100. Это абстрактный суперкласс классов S100_GF_FeatureType и S100_GF_InformationType. Цель введения этого класса - показать общность между типами фичеров и типами информации, в частности способность этих типов связываться с типами информации посредством ассоциации информации.

Таблица 3-2— S100_GF_ObjectType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_ObjectType	Абстрактный базовый класс для типов объектов в GFM	-	-
Роль	informationLink	Ссылка на ассоциацию информации, описывающее отношение к реализации типа информации	0..*	S100_GF_Information AssociationType

3-5.2.6 S100_GF_FeatureType (тип фичера)

Класс S100_GF_FeatureType является реализацией класса ISO 19109 - GF_FeatureType. Он отличается от класса ISO следующим:

- Это подтип класса S100_GF_NamedType;
- Он не реализует ассоциации Generalization и Specialization с классом GF_InheritanceRelation. Вместо этого класс имеет ассоциацию с самим собой с ролями subType и superType. GF_InheritanceRelation не реализована в GFM S-100;

- Множественность superType - 0..1, представляющая концепцию того, что фичер может иметь максимум один superType. Это сделано для того, чтобы избежать наследственность множественности в S-100;
- Множественность роли carrierOfCharacteristics с S100_GF_PropertyType (реализация S-100 GF_PropertyType) изменяется с 0..* на 1..*. Фичер S-100 должен иметь свойства.

Таблица 3-3— S100_GF_FeatureType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_FeatureType	Тип для абстрактного представления сущности реального мира	-	-
Роль	superType	Более общий тип фичера, из которого производится этот тип фичера	0..1	S100_GF_FeatureType
Роль	subType	Более специфические типы фичеров, производные от этого типа фичера	0..*	S100_GF_FeatureType
Роль	linkBetween	Ссылка на ассоциацию фичеров, описывающая взаимоотношение между одним типом фичера и таким же или другим типом фичера	0..*	S100_GF_FeatureAssociationType
Роль	carrierOfCharacteristics	Атрибуты и роли, описывающие характеристики типа фичера	1..*	S100_GF_PropertyType

3-5.2.7 S100_GF_PropertyType (тип свойства)

Класс S100_GF_PropertyType является реализацией класса GF_PropertyType согласно ISO 19109.

Он отличается от класса ISO по следующим позициям:

- 1) Множественность ассоциации с S100_GF_FeatureType изменяется от 1 до 1..*. Это изменение представляет тот факт, что фичеры и их свойства описываются в Каталоге фичеров S-100. Определение типа свойств может быть использовано в одном или более определениях типа фичеров;
- 2) Тип ассоциации с S100_GF_FeatureType изменяется от композиции до агрегации как результат изменения множественности, описанной выше.

Таблица 3-4 — S100_GF_PropertyType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_PropertyType	Абстрактный базовый класс для всех свойств типа фичера. Это атрибуты и роли.	-	-
Атрибут	memberName	Имя атрибута или роли.	1	CharacterString
Атрибут	definition	Описание атрибута или роли типа фичера.	1	CharacterString

Роль	constrainedBy	Роль указывает, что ограничение делается по свойствам	0..*	S100_GF_Constraint
------	---------------	---	------	--------------------

3-5.2.8 S100_GF_AttributeType (тип атрибута)

Класс S100_GF_AttributeType является реализацией S-100 - GF_AttributeType. Он практически идентичен классу ISO 19109, но отличается следующим образом:

- 1) Ассоциация attributeOfAttribute не реализуется в GFM S-100. Вместо этого S-100 вводит концепцию комплексных атрибутов. Комплексные атрибуты описаны в стандарте ISO 19109 подраздел 7.4.

Таблица 3-5— S100_GF_AttributeType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_AttributeType	Абстрактный базовый класс для всех атрибутов типа фичера. В этой модели два подкласса: тематические атрибуты и пространственные атрибуты.	-	-
Атрибут	valueType	Тип данных значения атрибута	1	CharacterString
Атрибут	domainOfValues	Описание набора значений. Для типов списка кодов это может быть URI, идентифицирующий словарь или "vocabulary".	1	CharacterString
Атрибут	multiplicity	Набор реализаций атрибута, который может быть ассоциирована с одиночной реализацией типа фичера.	1	S100_Multiplicity

3-5.2.9 S100_GF_AssociationRole (роль ассоциации)

Класс S100_GF_AssociationRole стандарта S-100 является реализацией класса ISO 19109 - GF_AssociationRole.

Table 3-6 — S100_GF_AssociationRole

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_AssociationRole	Роль, используемая в ассоциации.	-	-
Атрибут	multiplicity	Количество объектов, которые могут быть ассоциированы в ассоциацию.	1	S100_Multiplicity

3-5.2.10 GF_Operation (операция)

Класс GF_Operation не реализован в S-100 GFM, потому что S-100 поддерживает только модель передачи данных. Наборы данных не могут содержать операции.

3-5.2.11 S100_GF_AssociationType (тип ассоциации)

Класс S100_GF_AssociationType стандарта S-100 является реализацией класса ISO 19109 - GF_AssociationType. Он отличается от класса ISO 19109 следующим образом:

- 1) GFM стандарта ISO 19109 моделирует GF_AssociationType как подтип класса GF_FeatureType. Это сделано по причинам, которые указаны в Примечании 1 пункта 7.3.9 стандарта ISO 19109. Модель S-100 не моделирует класс как подтип S100_GF_FeatureType. В рамках ассоциаций S-100 между типами фичеров абстракции реального мира не рассматриваются. Результатом такого подхода к моделированию GFM является то, что только свойства ассоциаций могут иметь тематические атрибуты.
- 2) Кратность roleName может быть 0..2 вместо 1..*. Наименьшее значение 0 означает, что роль соответствует одной из ролей по умолчанию "source" или "target", и это очевидно из семантики прикладной схемы из имени типа ассоциации и имен участвующих типов фичеров или информации. Наибольшее значение выражает ограничение, которое S-100 не допускает для ассоциаций с более, чем двумя участвующими типами.

Таблица 3-7— S100_GF_AssociationType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_AssociationType	Абстрактный базовый класс для ассоциаций фичеров и информации.	-	-
Роль	carrierOfCharacteristics	Тематические атрибуты, описывающие ассоциацию.	0..*	S100_GF_ThematicAttributeType
Роль	roleName	Роли, описывающие концы ассоциации	0..2	S100_GF_AssociationRole

3-5.2.12 S100_GF_InformationType (тип информации)

Класс S100_GF_InformationType является классом типов информации в рамках S-100. Тип информации является идентифицируемым объектом, который может быть ассоциирован с фичерами для того, чтобы передавать конкретную информацию ассоциированным фичерам. Примером типа информации может быть Примечание на карте. Типы информации также могут быть ассоциированы друг с другом. Это может быть сделано там, где имеется дополнительная информация, относящаяся к типу информации или там, где нужно дать перевод информации. Например, информационный объект, имеющий Примечание на карте, может содержать текст на английском, а ассоциированный дополнительный информационный объект может содержать этот же текст на немецком.

Характеристики типов информации должны передаваться только типами тематических атрибутов.

Поэтому, S100_GF_InformationType ассоциируется только с S100_GF_ThematicAttributeType, а не с более общим классом S100_GF_PropertyType. Ассоциации информационных типов моделируются посредством S100_InformationAssociationType.

Таблица 3-8 — S100_GF_InformationType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_InformationType	Тип идентифицируемого объекта с дополнительной информацией для других объектов .	-	-
Роль	superType	Более общий тип информации, из которого получен этот тип информации .	0..1	S100_GF_InformationType
Роль	subType	Более конкретные типы информации, полученные на основе этого типа информации	0..*	S100_GF_InformationType
Роль	linkBetween	Ссылка на ассоциацию информации, указывающая связь между одним типом объекта и этим типом информации	0..*	S100_GF_InformationAssociationType
Роль	carrierOfCharacteristics	Тематические атрибуты, описывающие характеристики типа информации	1..*	S100_GF_ThematicAttributeType
Роль	roles	Роль ассоциаций по отношению к другим типам информации, которые предоставляют дополнительную информацию	0..*	S100_GF_AssociationRole

3-5.2.13 S100_GF_FeatureAssociationType (тип ассоциации фичеров)

Класс S100_GF_FeatureAssociationType по стандарту ISO 19109 не реализуется, но введен специально для GFM S-100. Причиной этому было то, что в S-100 имеется два типа ассоциаций: ассоциации фичеров и ассоциации информации. Они отличаются семантически и различны в модели. Этот класс описывает ассоциацию фичеров. Ассоциация фичеров является описанием взаимосвязи между двумя реализациями типов фичеров. Взаимосвязь может характеризоваться тематическими атрибутами и обычно имеет две роли. Роли описывают концы связей поскольку такая связь обычно не симметричная.

Table 3-9— S100_GF_FeatureAssociationType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_Feature AssociationType	Класс для описания взаимосвязи между двумя типами фичеров .	-	-
Роль	superType	Более общая ассоциация, из которого получен этот тип ассоциации фичеров .	0..1	S100_GF_FeatureAssociationType
Роль	subType	Более конкретные ассоциации фичеров, полученные из этой ассоциации фичеров	0..*	S100_GF_FeatureAssociationType
Роль	includes	Типы фичеров, включенные в эту взаимосвязь	1..*	S100_GF_FeatureType

3-5.2.14 S100_GF_InformationAssociationType (тип ассоциации информации)

Класс S100_GF_InformationAssociationType не получен из ISO 19109, а введен специально для GFM S-100. Причиной этому было то, что в S-100 имеется два типа ассоциаций: ассоциации фичеров и ассоциации информации. Они отличаются семантически и различны в модели. Этот класс описывает ассоциацию информации. Ассоциация информации является описанием взаимосвязи между произвольными фичерами и типом информации, дающим дополнительную информацию об этих фичерах. Взаимосвязь может характеризоваться тематическими атрибутами и ролью.

Таблица 3-10— S100_GF_InformationAssociationType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_Information AssociationType	Класс для описания связи между объектом и типом информации	-	-
Роль	superType	Более общий тип ассоциации информации, из которого получена эта ассоциация информации .	0..1	S100_GF_InformationAssociationType
Роль	subType	Более конкретные ассоциации фичеров, полученные из этой ассоциации фичеров	0..*	S100_GF_InformationAssociationType
Роль	includes	Тип информации, включенный во взаимосвязь	1..*	100_GF_InformationType
Роль	informationClient	Типы объектов, действующие как клиент в ассоциации информации.	1..*	S100_GF_ObjectType

3-5.2.15 S100_GF_Constraint (ограничение)

Класс S100_GF_Constraint является реализацией класса ISO 19109 - GF_Constraint с ассоциацией S100_GF_NamedType вместо ассоциации ISO 19109 GF_Feature_Type.

Таблица 3-11— S100_GF_Constraint

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_Constraint	Класс ограничений, которые могут быть связаны с поименованными типами или их свойствами	-	-
Атрибут	description	Ограничение, описанное на естественном языке и/или в официальном примечании .	1	CharacterString

3-5.3 Атрибуты типов фичеров

3-5.3.1 Введение

Этот раздел более подробно описывает роль атрибутов фичеров и типов информации.

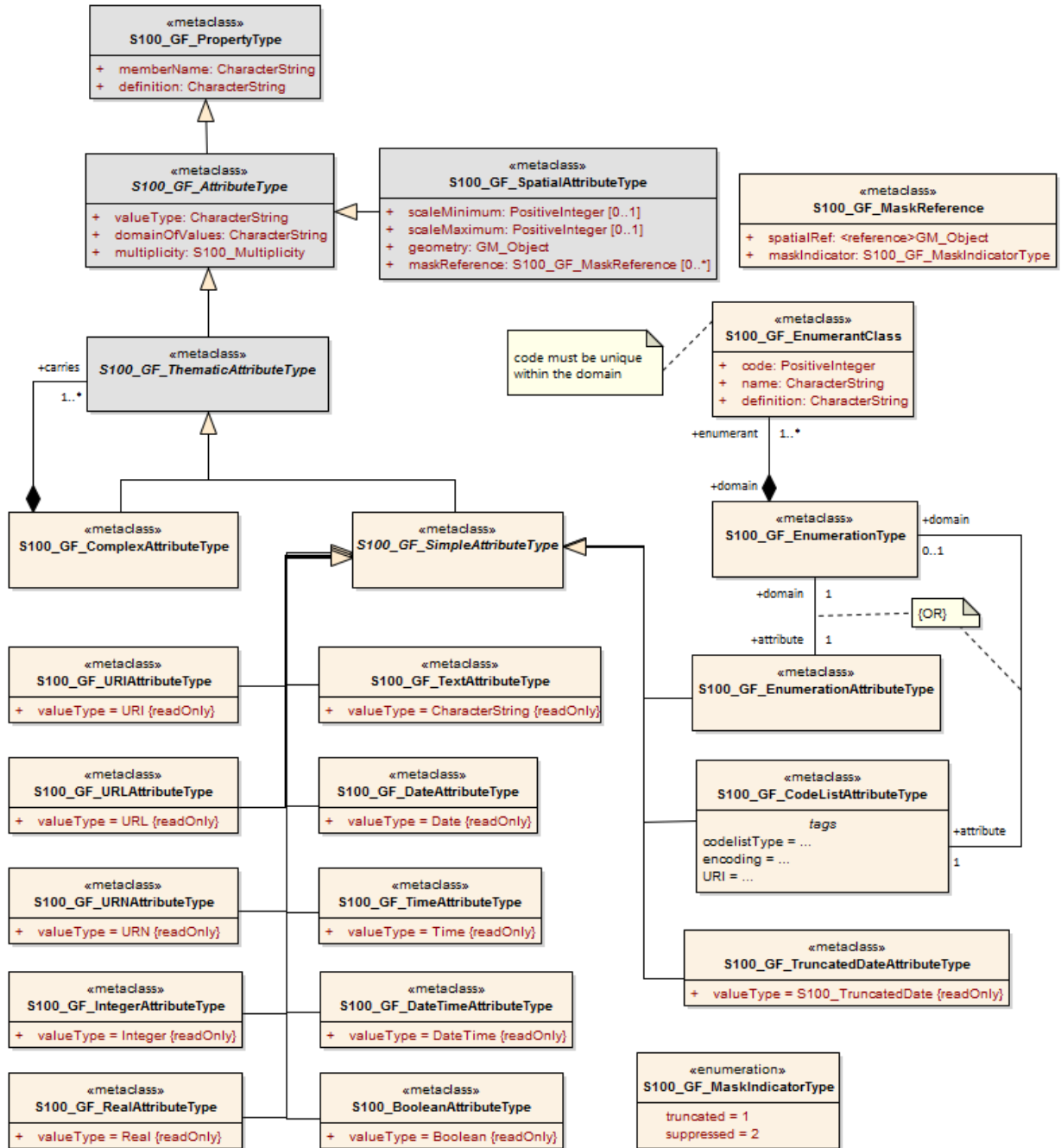


Рисунок 3-2 — Атрибуты

3-5.3.2 S100_GF_ThematicAttributeType (тип тематического атрибута)

Класс S100_GF_ThematicAttributeType является реализацией класса ISO 19109 - GF_ThematicAttributeType. Типы тематических атрибутов содержат описательные

характеристики объектов, отличающиеся от тех, что указаны в пунктах 7.4.3 – 7.4.7 стандарта ISO 19109. Этот класс отличается от класса ISO 19109 следующим:

- 1) GF_ThematicAttributeType определен в ISO 19109 как конкретный класс. Реализация GFM S-100 является абстрактным классом с двумя конкретными подклассами – S100_GF_SimpleAttributeType и S100_GF_ComplexAttributeType.
- 2) Временная информация должна иметь тип значения, определенный типами Date, Time, DateTime, S100_TruncatedDate или комплексными структурами, использующими комбинации типов временных примитивов.

Таблица 3-12— S100_GF_ThematicAttributeType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_ThematicAttributeType	Абстрактный базовый класс для всех атрибутов, кроме пространственных	-	-

3-5.3.3 S100_GF_ComplexAttributeType (тип комплексного атрибута)

В S-100 GFM введен класс S100_GF_ComplexAttributeType. Комплексные атрибуты составлены из других простых или комплексных атрибутов

3-5.3.4 S100_GF_SimpleAttributeType (тип простого атрибута)

В S-100 GFM введен класс S100_GF_SimpleAttributeType. Простой атрибут содержит описательную характеристику поименованного типа.

3-5.3.5 S100_GF_SpatialAttributeType (тип пространственного атрибута)

Класс S100_GF_SpatialAttributeType является реализацией класса ISO 19109 - GF_SpatialAttributeType. Тип пространственного атрибута должен иметь GM_Object в качестве типа значения. GM_Object и его подтипы определены в Пространственной схеме (Spatial Schema, S-100 Часть 7).

ПРИМЕЧАНИЕ: Класс S100_GF_SpatialAttributeType применяется только частично в Каталоге фичеров, описанном в S-100 часть 5 и при кодировании, описанном в части 10.

Таблица 3-13— S100_GF_SpatialAttributeType

Имя роли	Имя	Описание	Множ	Тип данных
Класс	S100_GF_SpatialAttributeType	Класс, представляющий пространственный атрибут, который используется для выражения пространственных характеристики типа фичера	-	-
Атрибут	scaleMinimum	Наименьший знаменатель масштаба для использования типа фичера (например, для изображения).	0..1	PositiveInteger

Атрибут	scaleMaximum	Наибольший знаменатель масштаба для использования типа фичера (например, для изображения).	0..1	PositiveInteger
Атрибут	geometry	Объект, описывающий геометрию реализации типа фичеров	1	GM_Object
Атрибут	maskReference	Ссылка, указывающая на скрытые или усеченные пространственные примитивы или объекты	0..*	S100_MaskReference

Маскирование или усечение должны указываться идентификаторами маскированных или усеченных примитивов и индикатором того, что маскируется или отсекается указанный примитив в атрибутах *maskReference*. Структура атрибута *maskReference* определяется типом S100_GF_MaskReference, показанным в таблице 3-14 ниже.

Применение ссылок маскирования в различных форматах S-100 указывается в соответствующих спецификациях форматов данных (часть 10a для формата данных ISO 8211 и часть 10b для формата GML) и может использовать конструкции, встроенные в основную спецификацию. Например, Формат GML S-100 использует GML тип ReferenceType с ограничениями на допустимые значения атрибутов xlink:href и xlink:role. Формат ISO 8211 использует без знаковые целые числа, содержащие идентификатор записи пространственного объекта и цифровой код значения индикатора маскирования.

Пространственные объекты, на которые делается ссылка в атрибуте маскирования, должны быть среди компонентов GM_Object, представляющих собой пространственный объект, на который ссылается та же самая реализация пространственного атрибута. Они могут быть компонентами на любом уровне, например, компоненты компонентов и т.д. (Другими словами, геометрия маскирования или усечения должна быть частью геометрии данной конкретной реализации пространственного атрибута.)

Спецификации продукта должны ограничивать использование маскирования конкретными пространственными типами, если это необходимо, например, кривыми.

Таблица 3-14— S100_GF_MaskReference

Имя роли	Имя	Описание	Мнж	Тип данных
Класс	S100_GF_MaskReference	Ссылка на маскированный или усеченный пространственный примитив. Модель основана на gml:Reference, но ограничивает допустимые атрибуты и делает идентификатор и роль обязательными	-	-

Атрибут	spatialRef (или xlink:href)	Идентификатор пространственного примитива	1	<(reference>GM_Object
Атрибут	maskIndicator (или xlink:role)	Указывает, является ли пространственный примитив маскированным или усеченным границами набора данных	1	Перечень <i>S100_GF_MaskIndicatorType</i>

Таблица 3-15— S100_GF_MaskIndicatorType

Имя роли	Имя	Описание	Код	Примечания
Перечень	S100_GF_MaskIndicatorType	Указывает маскирование или усечение границами набора данных	-	-
Буквенный		Пространственный примитив усечен границами набора данных	1	
Буквенный		Отображение пространственного примитива подавляется	2	

3-5.3.6 GF_TemporalAttributeType (тип атрибута времени)

Класс ISO 19109 - GF_TemporalAttributeType в S-100 GFM явно не реализован. Информация о времени должна моделироваться с помощью типа тематического атрибута S100_GF_ThematicAttributeType (подробности см. в разделе 3-6.4.4).

3-5.3.7 GF_MetadataAttributeType (тип атрибута метаданных)

Класс ISO 19109 - GF_MetadataAttributeType в GFM S-100 явно не реализован. Типы метаданных должны моделироваться с помощью комплексных тематических атрибутов, реализующих типы из компонента метаданных S-100 Часть 4а. Комплексные тематические атрибуты должны быть определены в Каталоге фичеров.

3-5.3.8 GF_QualityAttributeType (тип атрибута качества)

Класс ISO 19109 - GF_QualityAttributeType в S-100 GFM явно не реализован. Типы метаданных качества должны моделироваться с помощью комплексных тематических атрибутов, реализующих типы из S-100 Часть 4с Приложение 4с-А Качество данных (Data Quality). Комплексные тематические атрибуты должны быть определены в Каталоге фичеров.

3-5.3.9 GF_LocationAttributeType (тип атрибута положения)

Класс ISO 19109 GF_LocationAttributeType в S-100 GFM не реализован.

3-5.3.10 S100_TruncatedDateAttributeType (тип атрибута усеченная дата)

Класс S100_TruncatedDateAttributeType предназначен для моделирования значений даты с одним или более пропущенным значащим компонентом. Это дает возможность использовать часть даты, например, периодических, повторяющихся периодов.

3-5.3.11 S100_GF_CodelistAttributeType (тип атрибута список кодов)

Класс S100_GF_CodelistAttributeType введен в S-100 GFM для моделирования списков кодов S-100. Атрибуты списков кодов должны быть ассоциированы или с перечнем (для открытых списков кодов перечней) или со словарем (для открытых и закрытых списков кодов словарей), но не для обоих сразу. Структура словаря определяется внешней спецификацией.

Table 3-16— S100_GF_CodelistAttributeType

Имя роли	Имя	Описание	Множ	Тип данных	Примечания
Класс	S100_GF_CodelistAttributeType	Абстрактный базовый класс для атрибутов S100_Codelist	-	-	-
Тэг	codelistType	Тип списка кодов	1	CharacterString	Должно быть одним из: Открытый перечень Открытый словарь Закрытый словарь
Тэг	URI	Определяет словарь для открытых или закрытых списков кодов	0..1	CharacterString	Только для открытых или закрытых списков кодов словарей
Тэг	encoding	Подсказка кодирования для дополнительных значений	0..1	CharacterString	Только для открытых списков кодов перечней или словарей

3-5.3.12 S100_GF_EnumerationType (тип перечня)

S100_GF_EnumerationType и S100_GF_EnumerantClass вместе моделируют перечни, определяющие допустимые значения перечисляемых атрибутов перечней и их семантики. Пример перечисляемого типа может определять набор допустимых значений перечисляемых атрибутов или списков кодов, или оба вместе.

3-5.4 Взаимосвязи между поименованными типами

3-5.4.1 Введение

Этот подраздел описывает взаимосвязи между типами фичеров. Взаимосвязи классифицируются следующим образом:

- 1) Генерализация / Специализация типов фичеров и типов информации.
- 2) Ассоциации между типами фичеров и типами информации.

3-5.4.2 GF_InheritanceRelation (наследственная связь)

Класс GF_InheritanceRelation в S-100 GFM не реализован, но наследственность объектов допускается путем использования идентичной ассоциации класса S100_GF_FeatureType и класса S100_GF_InformationType (см. рисунок 3-3). Множественность конца ассоциации superType такова, что подтип может иметь только один супертип. Это сделано для того, чтобы не допускать моделирования множественных наследований. Ассоциации наследственных связей моделируются на уровне конкретного класса, а не абстрактного класса S100_GF_NamedType. Это предотвращает наследование типа фичера из типа информации и наоборот.

Ассоциации наследования существуют только между поименованными типами фичеров, а не между реализациями поименованных типов (это сущности, содержащиеся в наборе данных).

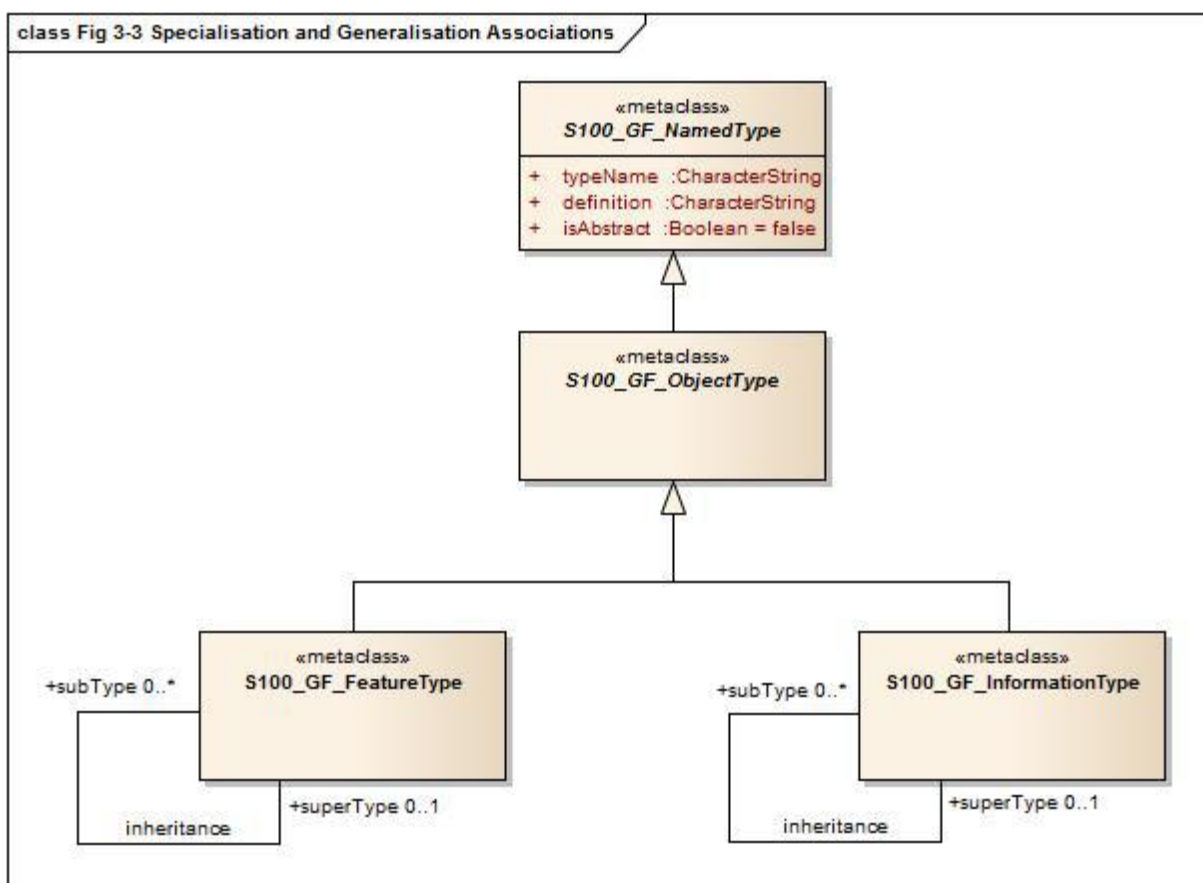


Рисунок 3-3 — Ассоциации специализации и генерализации

3-5.4.3 S100_GF_AssociationType (тип ассоциации)

Ассоциации определяются классом S100_GF_AssociationType с двумя ролями и определением. Классы ISO 19109 - GF_AggregationType, GF_SpatialAssociationType и GF_TemporalAssociationType в GFM S-100 явно не реализованы. Эти классы могут использоваться только если ассоциации разрешается иметь свойства. ISO 19109 GFM допускает это, потому что GF_AssociationType является подтипом GF_FeatureType. Однако, S100_GF_AssociationType не является подтипом S100_GF_FeatureType.

3-5.4.4 Ассоциации с типами информации

В GFM S-100 внесена ассоциация между S100_GF_ObjectType и S100_GF_InformationType. Роль *additionalInformation* является ролью по умолчанию для этой ассоциации в GFM S-100 и означает, что для поименованного типа имеется дополнительная информация.

3-5.4.5 Имена по умолчанию для концов ассоциации

Прикладные схемы могут присваивать имена концам ассоциации (имена ролей). Если имена явно не даны, следует использовать следующее по умолчанию.

- 1) Если явное имя “<rolename>” дано только одному концу ассоциации, другой конец должен иметь имя по умолчанию “inv_<rolename>”.
- 2) Если явное имя не дано ни одному концу ассоциации, имя роли по умолчанию будет “the<target class name>”, в котором класс цели указывается из класса источника.
- 3) Вышеуказанные правила могут не выполняться для явного имени каждого конца ассоциации в прикладной схеме, поэтому спецификации на производство могут определять различные или дополнительные правила, если это необходимо.
- 4) Если требуются стандартные имена, то могут использоваться следующие по умолчанию вместо тех, что перечислены выше.
 - a. Роль “additionalInformation” является именем роли по умолчанию для ассоциаций от типов фичеров до типов информации.
 - b. Ассоциации фичер/фичер или информация/информация, действующие только в одном направлении, могут использовать по умолчанию имена концов “source” и “target”. Имя “associatedWith” может использоваться на обоих концах двух-направленной ассоциации.

Спецификации на производство могут смешивать индивидуальные и стандартные умолчания, но должны быть недвусмысленными (однозначными) в части того, какое имя применяется к каждому конкретному концу ассоциации

3-5.5 Поведение типов фичеров

Поведение типов фичеров описывается операциями, которые могут выполняться над или с самими реализациями типа фичеров. Операции используются только с функционально совместимой моделью и не используются в модели обмена данными.

3-5.6 Ограничения

Для обеспечения целостности данных могут вводиться ограничения. Ограничения уменьшают свободу действий в приложении для предотвращения возникновения ошибочных данных путем уточнения комбинаций данных, которые или допустимы или не допустимы. Прикладная схема должна идентифицировать ограничения недвусмысленным способом.

Ограничения могут иметь только поименованные типы и свойства.

3-6 Правила для прикладных схем (ISO 19109 Раздел 8)

3-6.1 Процесс моделирования прикладных схем (ISO 19109 Раздел 8.1)

Прикладная схема служит двум целям:

- 1) Дает общее и корректное понимание содержания и структуры данных в рамках конкретного прикладного поля.
- 2) Обеспечивает машиночитаемую схему для применения механизмов управления данными.

Для создания прикладной схемы используется пошаговый процесс. Шаги могут быть кратко описаны как:

- 1) Служащие требованиям предназначенного прикладного поля
- 2) Создающие концептуальную модель применения с концепциями, определенными в GFM. Эта задача состоит из идентифицирующих типов фичеров, их свойств и ограничений.
- 3) Описывающие элементы прикладной схемы формальным языком моделирования там, где это необходимо. Прикладные схемы S-100 должны описываться с помощью UML в соответствии с правилами, определенными в этой части S-100.
- 4) Интегрирующие формальные прикладные схемы с другими стандартизированными схемами (пространственная схема, схема качества и т.п.) в единую прикладную схему.

3-6.2 Прикладная схема (ISO 19109 Раздел 8.2)

3-6.2.1 Язык концептуальной схемы для прикладных схем

Если для создания прикладной схемы S-100 используется концептуальный язык, тогда это должен быть UML.

3-6.2.2 Основные правила

Структура данных прикладной схемы должна моделироваться в прикладной схеме.

Все классы, используемые в прикладной схеме для передачи данных должны быть устанавливаемые. Это подразумевает то, что интегрированный класс не должен быть шаблонным <<interface>>.

3-6.2.3 Идентификация прикладных схем

1) Идентификация каждой прикладной схемы должна включать имя и версию. Включение версии гарантирует то, что поставщик и пользователь согласовали в какой версии прикладная схема описывает содержание конкретного набора данных. Система определения уникальных имен и версий прикладных схем S-100 должна быть определена.

2) В UML прикладная схема должна быть описана в PACKAGE, который несет в себе имя прикладной схемы и версию, указанную в документации PACKAGE.

3-6.2.4 Документация прикладной схемы

1) Прикладная схема должна быть задокументирована. Средства документирования прикладных схем S-100 должны быть определены для того, чтобы обеспечить согласованность между спецификациями продуктов S-100.

2) Документация прикладной схемы на UML может использовать средства документирования программного инструмента, используемого для создания прикладной схемы, если эта информация может быть экспортирована.

3) Если CLASS или другой компонент UML соответствует информации каталога фичеров, ссылка на каталог должна быть задокументирована.

4) Документация типов фичеров в прикладной схеме должна быть в каталоге со структурой, вытекающей из GFM так, как в каталоге в соответствии с S-100 Часть 5. Это может быть в текстовом формате или в XML, сопровождаемом таблицей стиля (XSLT), используемой для создания текстовой версии.

3-6.3 Правила для прикладной схемы в UML (ISO 19109 Раздел 8.3)

3-6.3.1 Основные правила (ISO 19109 Раздел 8.3.1)

Основные правила для прикладных схем в UML следующие:

- 1) Реализация S100_GF_NamedType должна представляться как CLASS.
- 2) Реализация S100_GF_ObjectType должна представляться как CLASS.
- 3) Реализация S100_GF_FeatureType должна представляться как CLASS.
- 4) Реализация S100_GF_InformationType должна представляться как CLASS.
- 5) Реализация S100_GF_FeatureAssociationType имеет роль linkBetween в ассоциации с реализациями S100_GF_FeatureType, представленными как CLASSы. Она должна представляться как один из следующих классов:
 - a) Случай 1: Каждая реализация S100_FeatureAssociationType, не ассоциированная с какой-либо реализацией S100_GF_ThematicAttributeType, должна представляться как ASSOCIATION между этими CLASSами.
 - b) Случай 2: Каждая реализация S100_FeatureAssociationType, ассоциированная с какой-либо одной или несколькими реализациями S100_GF_ThematicAttributeType, должна представляться как ASSOCIATION CLASS; ассоциированные реализации S100_GF_ThematicAttributeType должны представляться как ATTRIBUTES ассоциированного класса ASSOCIATION CLASS.
- 6) Реализация S100_GF_InformationAssociationType имеет роль informationLink в ассоциации с реализацией S100_GF_FeatureType или S100_GF_InformationType, представленной как CLASSES. Она должна представляться одним из следующих случаев:
 - a) Случай 1: Реализация S100_InformationAssociationType, не ассоциированная с какой-либо реализацией S100_GF_ThematicAttributeType, должна представляться как ASSOCIATION между этими CLASSами.
 - b) Случай 2: Реализация S100_InformationAssociationType, ассоциированная с какой-либо одной или более реализациями S100_GF_ThematicAttributeType, должна представляться как ASSOCIATION CLASS; ассоциированные реализации S100_GF_ThematicAttributeType должны представляться как ATTRIBUTES ассоциированного класса ASSOCIATION CLASS.
- 7) Реализация S100_GF_AttributeType должна представляться как ATTRIBUTE.
- 8) Реализация S100_GF_SimpleAttributeType должна представляться как ATTRIBUTE.

9) Реализация S100_GF_ComplexAttributeType должна представляться как CLASS. Представленный CLASS должен иметь одну или несколько реализаций S100_GF_SimpleAttributeType и/или S-100_GF_ComplexAttributeType в качестве ее ATTRIBUTE(ов).

10) Реализация ассоциации inheritanceRelation должна представляться связью UML GENERALISATION.

3-6.4 Профили доменов стандартных схем в UML (ISO 19109 Раздел 8.4)

3-6.4.1 Правила добавления информации в стандартную схему

Стандартные схемы не должны расширяться в рамках прикладных схем. Стандартные схемы – это те схемы, которые задокументированы в S-100, например, пространственная схема, Схема Каталога фичеров и т.п.

3-6.4.2 Ограниченное использование стандартных схем

Для некоторых стандартных схем, например, для S-100 Часть 7 (Пространственная схема), можно переопределять схему таким образом, чтобы использовались только выбранные части схемы и только некоторые определения классов и связей.

1) Спецификация ограниченного профиля стандартной схемы должна быть описана в новом пакете UML путем копирования фактических определений (классов и связей) из стандартной схемы. Атрибуты и операции в рамках классов могут пропускаться.

2) Сокращение стандартной схемы должно выполняться в соответствии с разделом соответствия действующего стандарта.

3-6.4.3 Правила использования схемы метаданных (ISO 19109 Раздел 8.5)

Схема метаданных, определенная в S100 Часть 4, является прикладной схемой для наборов метаданных. Метаданные являются данными описывающими и документирующими данные. Метаданные географических данных обычно обеспечивают информацией об их идентификации, протяженности, качеству, пространственных и временных аспектах, пространственных связях и распространении.

Типы метаданных должны представляться как комплексные атрибуты, реализующие элементы из S100 Часть 4. Таким образом, атрибуты метаданных должны быть типами тематических атрибутов.

3-6.4.4 Правила времени (ISO 19109 Раздел 8.6)

S-100 не включает в себя профиль ISO 19108. Атрибуты времени должны моделироваться с помощью типов Date, Time или DateTime, S100_TruncatedDate или комплексных атрибутов, использующих комбинации этих типов времени. Использование этих типов делает атрибуты реализацией S100_GF_SimpleAttributeType или S100_GF_ComplexAttributeType.

3-6.5 Пространственные правила (ISO 19109 Раздел 8.7)

3-6.5.1 Общие пространственные правила (ISO 19109 Раздел 8.7.1)

Значение домена типов пространственных атрибутов должно соответствовать спецификации, данной в S-100 Часть 7, которая обеспечивает концептуальными схемами описания пространственных характеристик фичеров и набора пространственных операторов, соответствующих этим схемам.

S-100 Часть 7 явно исключает топологические примитивы и какие-либо правила топологии, указанные в пункте 8.7 стандарта ISO 19109, которые не соответствуют этому профилю.

3-6.5.2 Пространственные атрибуты

- 1) Пространственные характеристики фичера должны быть описаны одним или более пространственными атрибутами. В прикладной схеме пространственный атрибут является подтипом атрибута фичера (см. 5.3), и систематика его значений определяется в S-100 Часть 7.
- 2) Пространственный атрибут должен представляться в прикладной схеме одним из следующих способов:
 - a) Случай 1: как ATTRIBUTE класса UML CLASS, представляющий фичер, в котором ATTRIBUTE будет принимать в качестве типа данных значение одного из пространственных объектов, определенных в пространственной схеме, ISO 19107, или
 - b) Случай 2: как UML ASSOCIATION между классом, который представляет фичер и одним из пространственных объектов, определенных в пространственной схеме, ISO 19107.
- 3) Пространственный атрибут должен принимать пространственный объект как его значение. Пространственные объекты классифицируются как геометрические объекты, которые являются подклассом как примитивы, комплексы или агрегаты (геометрических объектов). Типы значений пространственных атрибутов должны быть типами, описанными в Части 7, или их подтипами.

3-6.5.3 Пространственное качество

Качество местоположения пространственного объекта должно быть описано односторонней ассоциацией с S100_GF_InformationType, которая в свою очередь ассоциирована с S100_GF_SimpleAttribute, определяющим точность местоположения.

3-6.5.4 Геометрические агрегаты и комплексы для представления пространственных атрибутов фичеров

3-6.5.4.1 Введение

Пространственная конфигурация многих фичеров не может представляться одиночным геометрическим примитивом. Типы GM_Aggregate и GM_Complex поддерживают представление таких фичеров как набор объектов геометрии.

3-6.5.4.2 Геометрические агрегаты

Пространственный профиль S-100 поддерживает только тип геометрического агрегата GM_Multipoint. GM_Multipoint должен использоваться как значение пространственного атрибута, представляющего фичер как набор точек.

3-6.5.4.3 Геометрические комплексы

Геометрические комплексы используются для представления пространственных характеристик фичеров как набор связанных геометрических примитивов. Дополнительно, реализации GM_Complex позволяют геометрическим примитивам принадлежать пространственным атрибутам разных фичеров. Явных связей между GM_Primitives в GM_Complex нет; связь между GM_Primitives может быть получена из данных координат.

1) GM_Complex должен использоваться как значение пространственного атрибута, представляющее фичер набором связанных GM_Objects, которые не пересекаются за исключением на их границах. Подклассы GM_Complex могут использоваться для ограничения структуры GM_Complex, используемого для представления конкретной пространственной конфигурации.

2) Фичеры, которые имеют общую геометрию, должны представляться как GM_Complexes, являющиеся подкомплексами более крупного GM_Complex.

3-6.5.4.4 Геометрические композиции

Геометрическая композиция является геометрическим комплексом, имеющим все свойства геометрического примитива, кроме того, что он составлен из меньших геометрических примитивов того же типа. Геометрические композиции используются для представления комплексных элементов, которые составлены из меньших геометрических объектов одного типа геометрии. GM_Composite должен

использоваться для представления комплексного фичера, имеющего геометрические свойства геометрического примитива.

3-6.5.4.5 Фичеры с общей геометрией

Разные фичеры могут иметь, частично или полностью, общую геометрию, когда они находятся в одном местоположении. Для того чтобы иметь общую геометрию, атрибуты пространственного фичера должны иметь один или более GM_Objects.

Имеется два способа показать общую геометрию. Полная общая геометрия возникает тогда, когда два фичера имеют один и тот же GM_Object в качестве значения пространственного атрибута. Это можно сделать путем указания ограничения в прикладной схеме. При отсутствии такого ограничения, его можно ввести по необходимости.

- 1) В прикладной схеме может потребоваться указать общую геометрию для двух или более типов фичеров посредством введения ограничения - что GM_Objects, представляющие эти фичеры, должны быть равными.
- 2) В прикладной схеме можно исключить два или более типа элементов, имеющих полную общую геометрию посредством введения ограничения - что GM_Objects представляющие эти фичеры, не равны.

3-6.6 Правила каталогизации (ISO 19109 Раздел 8.8)

3-6.6.1 Введение (ISO 19109 Раздел 8.8.1)

Каталог фичеров является хранилищем описаний сущностей реального мира значимого конкретного домена. Методология каталогизации фичеров дает подробное описание организации данных, представляющих эти сущности в таких категориях, чтобы результирующая информация была однозначной, исчерпывающей и полезной, насколько это возможно.

3-6.6.2 Прикладная схема на основе каталога фичеров (ISO 19109 Раздел 8.8.2)

Прикладная схема S-100 должна быть полностью сконструирована на определениях Каталога фичеров, реализующего профиль каталога фичеров S-100.

3-6.6.3 Кодирование символов

Кодирование символов, используемое в наборе данных, должно быть определено в прикладной схеме. Там, где используется кодирование более одного символа, прикладная схема должна указывать как они используются в наборе данных.

3-6.7 Списки кодов (Codelists)

Прикладные схемы, использующие атрибут типа codelist должны иметь CLASS с тэгами как это указано ниже. Типы codelist описаны в Части 1.

Таблица 3-17 — Тэги для типов codelist

Тип codelist	Тэги и значения
open enumeration открытый перечень	codelistType=open enumeration encoding=other: [something]
closed dictionary закрытый словарь	codelistType=closed dictionary URI=<dictionary URL>
open dictionary открытый словарь	codelistType=open dictionary URI=<dictionary URL> encoding=other: [something]

Нормативная форма кодирования “other: [something]” должна быть символьной строкой в формате, указанном ниже:

Слово ‘other’ и далее двоеточие и пробел (это ‘other: ’ без кавычек), затем продолжается одной или несколькими алфавитно-цифровыми строками, разделяемыми одиночными пробелами.

Нормативный шаблон части строки, следующей за ‘other: ’ (используя шаблоны XML Schema 1.0/1.1) должен быть:

[a-zA-Z0-9]+([a-zA-Z0-9]+)*

Заметим, что за левой круглой скобкой следует одиночный пробел, а шаблон завершается звездочкой.

Примеры:

Таблица 3-18 — Примеры “extra” значений атрибутов codelist

other: loxodromic	разрешается
other: Seeschifffahrtsstraßen Ordnung	не разрешается (содержит символ ß, которого нет в наборе допустимых)
other: German Shipping Regulations	разрешается
other: German Shipping Regulations	не разрешается (2 последовательных пробела)
German Shipping Regulations	не разрешается (не начинается с “other:“)
other: 287	разрешается
other: 1,3,5-Trinitroperhydro- 1,3,5-triazine	не разрешается (дефис и символы запятой не входят в разрешенный набор)

3-7 Прикладная схема для покрытий (информативная)

3-7.1 Введение

Этот набор правил прикладных схем предназначен для прикладных схем данных, ориентированных на фичеры. Однако, прикладные схемы также предназначаются для покрытий (coverages).

Этот раздел включает примеры того, как прикладные схемы могут определяться для данных изображений и сеток. Компоненты прикладных схем определены в ISO 19123, а не в ISO 19109. Тем не менее, покрытие может базироваться на геометрии типа фичеров и, в этом случае, концептуально аналогично набору фичеров. Такие ориентированные на фичеры покрытия описаны далее.

3-7.2 Данные, привязанные к сетке

Эта прикладная схема определяет четырехугольное сетевое покрытие с ассоциированными мета данными. Метаданные в общем плане соответствуют ISO 19115 и 19115-2. Конкретный выбор мета данных в этой схеме не делается. Эта схема может использоваться как для матричных, так и для растровых данных в зависимости от выбранных метаданных.

Сеточные данные состоят из одиночного элемента: изображение ("image") или матрица ("matrix"), вместе с ассоциированными мета данными, взятыми из MD_Metadata (или MI_Metadata). CV_Coverage служит в качестве пространственного атрибута сеточного набора данных. Он определяет область, которая «покрыта» функцией покрытия. Для непрерывного покрытия, определенного в этой прикладной схеме, покрытие дает значение в каждой точке области покрытия путем функции интерполяции. Регулярная сетевая матрица является набором значений для получения интерполированных значений. В этом случае матрица является сеткой значений в точках пересечения для линейной интерполяции между ними по осям (x, y). Пространственная связь определяется системой координат.

Эта шаблонная прикладная схема обеспечивает большинство приложений, работающих с изображениями и сеточными данными.

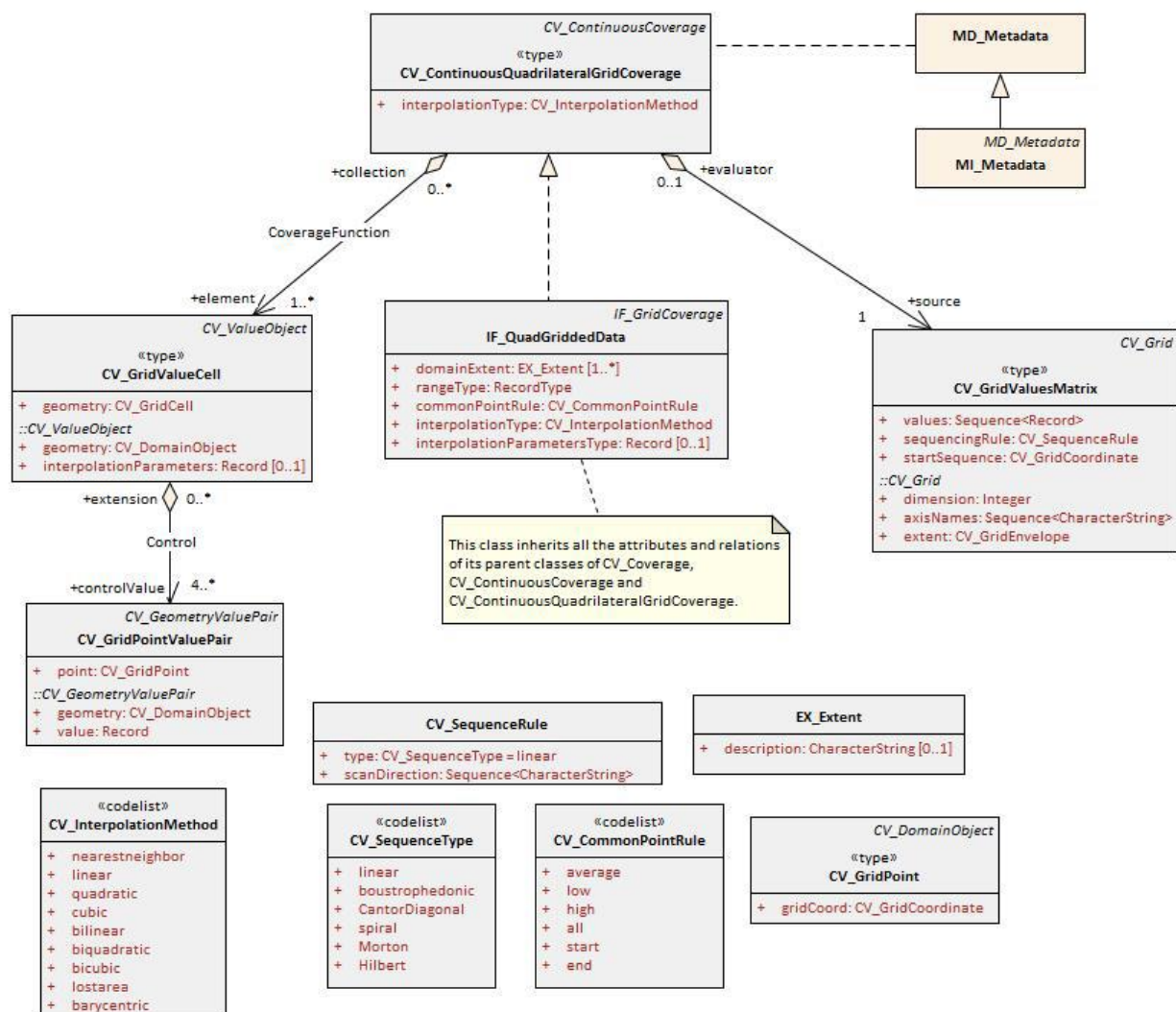


Рисунок 3-4 – Шаблонная прикладная схема для четырехстороннего сетевого покрытия

3-7.3 Сетка с переменным размером ячеек (Variable Cell Size Grid)

Эта прикладная схема описывает сетку с переменным размером ячеек (ISO 19123). Порядок пересечений является порядком Мортонa, чтобы обеспечить поддержку в трех (или более) измерениях. Это нужно для конкретного использования для представления гидрографических данных с большим объемом измерений, выполненных сонаром в 3D сетке, но где ячейки одинаковых глубин могут быть легко агрегированы.

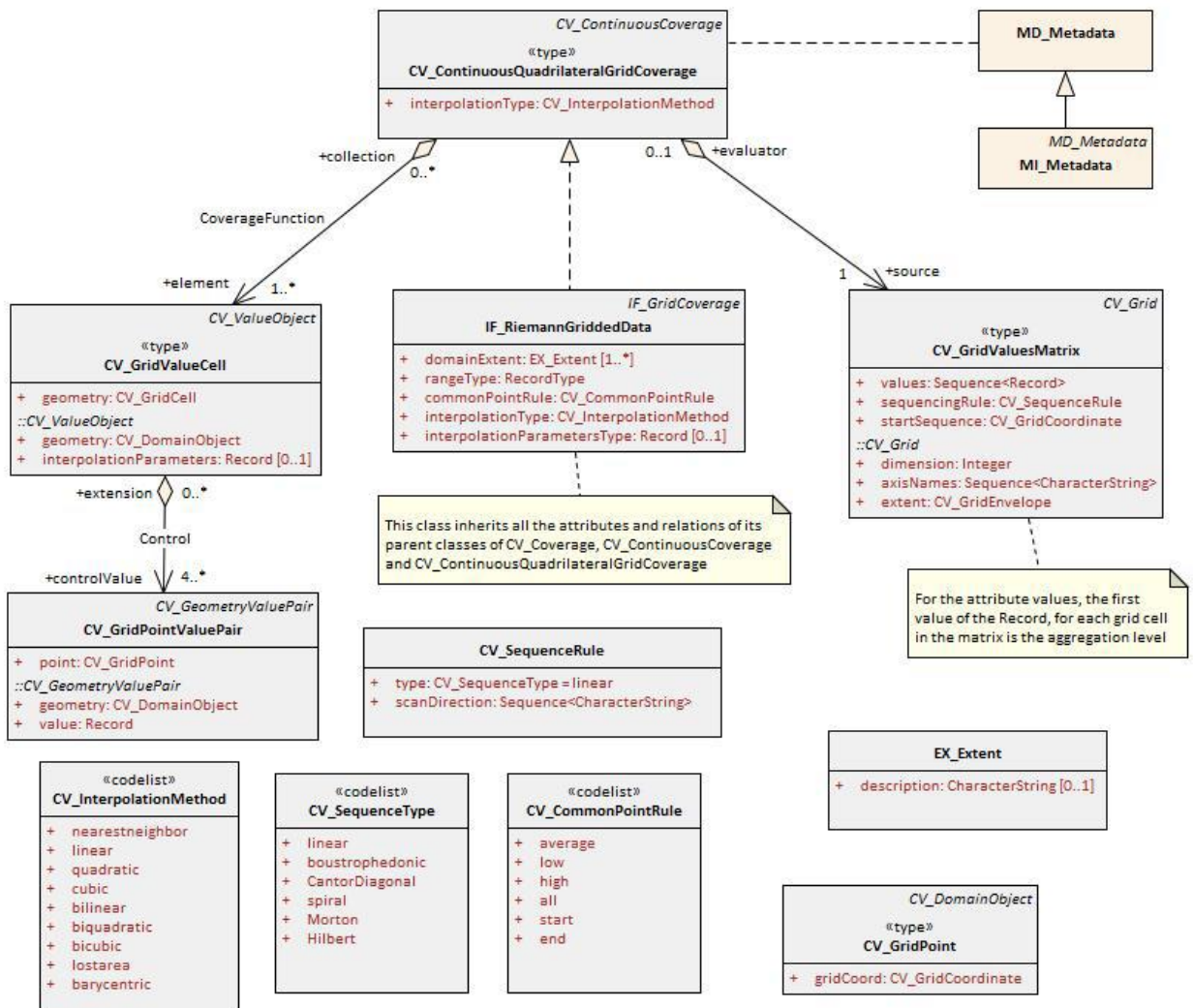


Рисунок 3-5 – Шаблон прикладной схемы для сеточного покрытия Reimann

3-7.4 Изображение, ориентированное на фичер

Все сеточные наборы данных являются фичеро-ориентированными, в этом смысле покрытие является подтипом фичера. Это означает, что весь набор сеточных данных может рассматриваться как одиночный фичер. Структура фичера может применяться к сеточным данным двумя различными путями. Первый, дискретное покрытие может иметь атрибут, несущий код фичера. Например, покрытие, соответствующее системе почтовых кодов будет иметь дискретные значения для каждого почтового кода, все вместе покрывающие всю страну. Единственная разница в прикладной схеме – это взаимосвязи между дискретным покрытием и фичером.

Типовая прикладная схема на рисунке 3-6 отражает как классы покрытия "дискретной точки", так и классы покрытия "дискретной точки сетки". Типичная спецификация на производство выбирает ту или иную (или обе) в зависимости от типа необходимого покрытия.

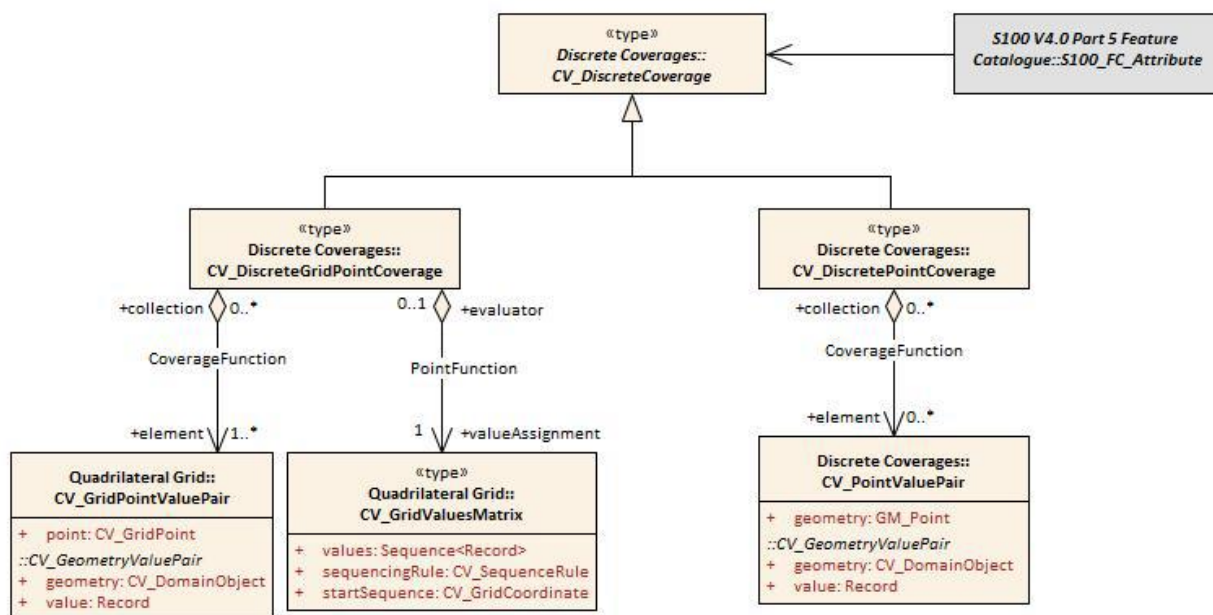


Рисунок 3-6 – Дискретное покрытие, ориентированное на фичер

Второй метод установления структуры фичера – это разработка композитного набора данных, содержащего множество отдельных, но примыкающих друг к другу покрытий. Покрытия могут быть непрерывными или дискретными. Это очень похоже на то как компонуется векторный набор данных, где каждый фичер имеет свою собственную геометрию и атрибуты. Фактически векторные данные могут смешиваться с данными покрытия в одном наборе данных. Прикладная схема просто позволяет множественность фичера.

Геометрические элементы, такие как сетки, могут совместно использоваться множеством фичеров и элементы могут связываться путем композиции или с помощью других связей как это допускается общей моделью фичеров ISO 19109. Комплексный фичер может включать как непрерывное сеточное покрытие, так и векторные данные, такие как границы полигонов. Фичеро-ориентированный набор данных может содержать как непрерывное покрытие океана, полученное с помощью сонара, так и точечные, и линейные фичеры, соответствующие средствам навигационного оборудования. Топологические примитивы могут связываться со всеми фичерами. Это позволяет создавать полезные конструкции. Растровая морская карта может включать дополнительные векторные данные, описывающие средства навигационного оборудования, опасности, районы ограниченного плавания, которые невидимы, потому что не отображаются, но активно используется растровой морской картой. Таким образом мореплаватель может определить находится ли его судно в опасной зоне или выполнять функции ЭКНИС.

3-8 S-100 Временные рамки

3-8.1 Временные определения

S-100 не включает полный профиль ISO 19108. Временные параметры моделируются с использованием примитивных типов Date; Time или DateTime; S100_TruncatedDate; S100_IndeterminateDate; или комплексных атрибутов, использующих комбинацию этих временных типов. Использование этих типов делает атрибут реализацией S100_GF_SimpleAttributeType или компонентом реализации S100_GF_ComplexAttributeType, в зависимости от обстоятельств.

Временные отметки обычно определяются как UTC или местное время со стандартным смещением от UTC или часовым поясом, используя григорианский календарь. Поэтому в стандарте S-100 не применяются никакие положения раздела 5.3 ISO 19108 Temporal Reference Systems.

- Отметка времени S-100 “instant” в правилах ISO 19108 (ISO 19108 5.2.3.2) применяется как одиночные S-100 Date, Time или DateTime (см. S-100 Часть 1, пункт 1-4.5.3.10 S100_TM_Instant).
- Период времени S-100 “period” (TM_Period ISO 19108 5.2.3.3) применяется как непрерывный интервал между двумя S100_TM_Instants (см. S-100 Часть 1, пункт 1-4.5.3.11), выраженный соответствующим комплексным атрибутом, например, *fixedDateRange*.
- S100_IndeterminateDate является отметкой времени внутри определенного периода.
- S100_TM_Instant указанный с помощью TruncatedDate может представлять бесконечное множество значений ISO 19108.

3-8.2 Временные взаимосвязи

S-100 не реализует в полной мере положения временной топологии ISO 19108 и временные относительные положения; Однако взаимосвязи между двумя моментальными и/или периодическими значениями A и B должны определяться и описываться в S-100 с использованием значений ISO 19108 TM_RelativePosition в соответствии с таблицей 3-19 ниже. Только записи, отмеченные (†), могут быть значимым образом определены, когда S100_TM_Instant является S100_IndeterminateDate, выраженным в неопределенном формате с ограниченным началом или концом.

Таблица 3-19 — Взаимосвязи между временными значениями (из ISO 19108 5.2.3.5 TM_Relative Position)

A~B	A (Single S100_TM_Instant)	A (Single S100_TM_Period)
B (Single S100_TM_Instant)	Before ($A < B$) Equals ($A = B$) After ($A > B$)	Before ($A_{end} < B$) [†] EndedBy ($A_{end} = B$) Contains ($A_{begin} < B$ and $A_{end} > B$) BegunBy ($A_{begin} = B$) After ($A_{begin} > B$) [†]
B (Single S100_TM_Period)	Before ($A < B_{start}$) [†] Ends ($A = B_{end}$) Within ($A > B_{begin}$ and $A < B_{end}$) Begins ($B_{begin} = A$) After ($A > B_{end}$) [†]	Before ($A_{end} < B_{begin}$) [†] Meets ($A_{end} = B_{begin}$) Overlaps ($A_{begin} < B_{begin}$ and $A_{end} > B_{begin}$ and $A_{end} < B_{end}$) Begins ($A_{begin} = B_{begin}$ and $A_{end} < B_{end}$) BegunBy ($A_{begin} = B_{begin}$ and $A > B_{end}$) During ($A_{begin} > B_{begin}$ and $A_{end} < B_{end}$) Contains ($A_{begin} < B_{begin}$ and $A_{end} > B_{end}$) Equals ($A_{begin} = B_{begin}$ and $A_{end} = B_{end}$) [†] OverlappedBy ($A_{begin} > B_{begin}$ and $A_{begin} < B_{end}$ and $A_{end} > B_{end}$) Ends ($A_{begin} > B_{begin}$ and $A_{end} = B_{end}$) EndedBy ($A_{begin} < B_{begin}$ and $A_{end} = B_{end}$) MetBy ($A_{begin} = B_{end}$) After ($A_{begin} > B_{end}$) [†]

TruncatedDate S-100 позволяет определять регулярные временные периоды или отметки. Они представляют собой объединение потенциально бесконечных серий периодов или периодов без пересечений. Например:

"2021---05" = { 2021-01-05, 2021-02-05, 2021-03-05, 2021-04-05, 2021-05-05, 2021-06-05, 2021-07-05, 2021-08-05, 2021-09-05, 2021-10-05, 2021-11-05, 2021-12-05 }

Усеченные даты позволяют определить ряд периодов. Отношения, определяемые между одним TM_Instant или TM_Period (A) с набором периодов, определенных датой в усеченном формате (B) (то есть, отношение A~B) определяются в таблице 3-20 ниже (указатель " means" for all" и относится ко всем TM_Periods в пределах B (обратите внимание, некоторые отношения (до/после) не могут быть определены, когда B является набором неограниченных регулярных периодов времени).

Таблица 3-20 — Взаимосвязь между временными значениями и усеченной датой

A~B	A (Single TM_Instant)	A (Single TM_Period)
B_n (Multiple TM_Periods)	Before ($A < B_{start}$) $\forall B$ Within ($A >= B_{begin}$ and $A <= B_{end}$) for some B _i After ($A > B_{end}$) $\forall B$	Before ($A_{end} < B_{begin}$) $\forall B$ Within ($A_{begin} >= B_{begin}$ and $A_{end} <= B_{end}$) for some B _i After ($A_{begin} > B_{end}$) $\forall B$

3-8 Интерпретация моделей интервалов времени и периода

Начало и конец периода (и интервалов) должны включаться в период (или интервал), если спецификация на производство не указывает иную интерпретацию. Это соответствует требованию ISO 8601:2004 § 2.1.3, который определяет интервалы времени как «часть оси времени, ограниченной двумя моментами», и

определяет, что «Интервал времени включает в себя все моменты между двумя ограничивающими его моментами, включая и сами ограничивающие его моменты времени.» Использование атрибутов «до» и «после» не допускается.

Моменты начала и конца определяются датой/временем наименьшей градуировки. Например, если месяц окончания является наименьшим компонентом, данным в каком-то периоде, моментом окончания будет последний день данного месяца.

Примеры применения этих принципов к кодируемым интервалам с использованием уменьшенной точности типа `truncatedDate` показаны в таблице 3-17. Таблица также показывает, как могут кодироваться високосные годы.

Таблица 3-17 — Примеры периодов

<S100_TruncatedDateAttributeType> periodStart	----01--	000000 в январе 1 через 240000 29 февраля високосного года и 28 февраля в не високосные годы
	год и день не кодируются	
<S100_TruncatedDateAttributeType> periodEnd	----02--	000000 в январе 1 через 240000 28 февраля каждого года
	год и день не кодируются	
<S100_TruncatedDateAttributeType> periodStart	----0101	000000 5 января, 2012 через 240000 18 июня, 2012
	год не кодируется	
<S100_TruncatedDateAttributeType> periodEnd	----0228	
	год не кодируется	
<S100_DateAttributeType> dateStart	20120105	
<S100_DateAttributeType> dateEnd	20120618	

3-9 Использование типов специфических форматов для усеченных дат

Форматы данных могут использовать специальные типы, поддерживаемые этим форматом, для того, чтобы включать усеченные значения. Там, где это случается, описание формата должно указывать мэппинг между значениями `S100_TruncatedDateAttributeType` и этими типами специфических форматов.

Пример: Кодирование на базе XML может использовать простой тип атрибута `gMonthDay` (который является встроенным типом XML Схемы) как эквивалентное представление даты «17 декабря каждого года»: `xs:gMonthDay:--12-17`

Это эквивалентно значению `----1217` в формате данных, который полностью соответствует стандарту S-100.

3-10 Идентификаторы реализаций

Идентификаторы реализаций должны использовать концепцию морского ресурсного имени (Maritime Resource Name (MRN)) и пространство имени (namespace). Пространство имени MRN администрируется IALA через веб сайт <http://mrnregistry.org>, который также содержит ссылки на полный набор правил,

применяемых к концепции MRN. Пространство имени высшего уровня urn:mrn остается фиксированным с последующими пространствами имени, разделенных колонками и доступными через процесс применения, объясняемым на вебсайте. Любая организация, желающая выдать идентификаторы, соответствующие требованиям MRN, должна подать заявку на пространство имен от IALA или от организации, которая уже имеет зарегистрированное пространство имен.

Например, ИО применяет к пространству имен, и впоследствии дает всем государствам-членам подпространство под пространством urn:mrn:iho; для NOAA это может быть urn:mrn:iho:us, а для CHS это может быть urn:mrn:iho:ca. NOAA и CHS будут администрировать свои соответствующие пространства имен по мере необходимости и в рамках правил MRN.

Следующие правила применяются к пространству имен MRN.

Специальная строка пространства имен (Namespace Specific String (NSS)) всех URN, которые используют "mrn" NID должны иметь следующую структуру:

```
<URN> ::= "urn:mrn:" <OID> ":" <OSS>
<OID> ::= 1*(ALPHA / DIGIT) ; Organizational ID
<OSS> ::= <OSNID> ":" <OSNS> ; Organizational specific string
<OSNID> ::= 1*(ALPHA / DIGIT / "-") ; Organizational specific namespace ID
<OSNS> ::= 1*<URN chars> ; Organizational specific namespace string
DIGIT ::= %x30-39 ; 0-9
ALPHA ::= %x61-7A ; a-z
```

Основы ABNF записи используют:

" " буквы (строки терминальных символов); термины не в кавычках не являются терминалами

/ альтернативы.

() означает группу последовательности, используемую в качестве единственного альтернативного варианта или в качестве единой повторяющейся группы

<a>* указывает, что следующие термины или группы могут повторяться как минимум <a> и не более раз; значения по умолчанию - 0 и бесконечность соответственно

; комментарий

Весь URN является чувствительным к заглавным буквам.

<URN chars> как определено в RFC2141

Процесс назначения уникальных идентификаторов организации управляется IALA. Подробности и процесс подачи заявок можно найти на <http://www.mrnregistry.org>.

3-10.1 Использование идентификаторов реализаций

Когда необходимо обеспечить обнаружение одних и тех же реализаций между продуктами, будь то одна и та же спецификация продукта или между разными спецификациями продуктов, требуется минимальное использование идентификатора реализаций. Этот идентификатор реализации должен быть атрибутом во всех классах реализаций типа фичера или типа информации, которые должны совместно идентифицироваться. Атрибут должен называться *interoperabilityIdentifier* и должен иметь тип данных URN.