

S-100 – Часть 10с

Модель данных и формат файла HDF5

Contents

10с-1 Цель	5
10с-2 Введение	5
10с-3 Соответствие	5
10с-4 Ссылки	5
10с-4.1 Нормативные ссылки	5
10с-4.2 Информативные ссылки	6
10с-5 Спецификация HDF5.....	6
10с-5.1 Абстрактная модель данных	7
10с-5.1.1 Файл	8
10с-5.1.2 Группа	9
10с-5.1.3 Набор данных	10
10с-5.1.4 Пространство данных	11
10с-5.1.5 Тип данных	12
10с-5.1.6 Атрибут	13
10с-5.1.7 Список свойств	15
10с-5.2 Библиотека HDF5 и модель программирования	15
10с-5.3 Запрещенные конструкции HDF5	15
10с-6 Профиль S-100 HDF5	15
10с-7 Типы данных	16
10с-8 Правила присвоения имен	17
10с-9 Структура продукта данных	18
10с-9.1 Общая структура.....	18
10с-9.1.1 Индексация элементов массива и сетки	20
10с-9.2 Метаданные	21
10с-9.2.1 Метаданные обнаружения	21
10с-9.2.2 Метаданные носителя (встроенные)	21
10с-9.2.3 Расширенные метаданные	22
10с-9.3 Обобщенные размеры и хранение координат и данных	22
10с-9.4 Корневая группа	27
10с-9.5 Группа информации о фичерах	32

10с-9.6 Группа контейнера фичеров	35
10с-9.6.1 Положение точки данных в ячейке	42
10с-9.6.2 Таблица атрибутов фичеров	43
10с-9.7 Группа реализации фичера	44
10с-9.7.1 Переопределяющие атрибуты	49
10с-9.7.2 Пример структуры контейнера и реализации	49
10с-9.8 Группа информации о тайлинге	50
10с-9.9 Группа индексов	51
10с-9.10 Группа позиционирования	52
10с-9.10.1 Стратегия пространственного представления	52
10с-9.10.2 Структуры данных для хранения информации о местоположении точек сетки	54
10с-9.11 Группы значений данных	55
10с-9.11.1 Регулярная сетка, ориентированная на фичер	61
10с-10 Общие перечни и словари	64
10с-10.1 Формат кодирования данных	64
10с-10.2 Тип горизонтальной CRS	65
10с-10.3 База вертикальных координат	65
10с-10.4 Ссылка на вертикальный датум	65
10с-10.5 Виды проекций	66
10с-10.6 Нули высот и глубин	67
10с-11 Файлы поддержки	69
10с-12 Файлы каталогов и метаданных	69
10с-13 Векторные пространственные объекты, фичеры и типы информации	69
10с-14 Ограничения и проверка	70
10с-14.1 Проверочные тесты	70
10с-15 Обновления	70
10с-16 Краткое описание модели	71
10с-17 Правила для разработчиков спецификаций	72
10с-17.1 Определение формата спецификации из этого профиля	72
10с-17.2 Различные правила	73

10с-17.3 Расширения этого профиля	73
10с-17.4 Расширения, добавляющие метаданные	74
10с-18 Практическое руководство	74
10с-19 Реализации прикладных схем из части 8 (информативный)	74

10с-1 Цель

Иерархический формат данных (HDF5) был разработан HDFgroup в качестве формата файлов для передачи данных, которые используются для получения изображений и данных, привязанных к сетке. Эта часть представляет собой профиль HDF5 и определяет формат обмена для облегчения перемещения файлов, содержащих записи данных, между компьютерными системами. Он определяет конкретную структуру, которая может использоваться для передачи файлов, содержащих типы данных и структуры данных, соответствующие общей модели фичеров S-100 (GML).

Эта часть определяет ограничения и соглашения, которые в совокупности определяют правила для форматов данных S-100 HDF5. Фичеры HDF5, не требуемые данными S-100 HDF5, исключены. Настоящая часть ограничивается описанием формата данных и не включает прикладную схему, а также не содержит руководящих принципов в отношении того, как разрабатывать спецификации на производство продуктов или правила наименования фичеров и атрибутов.

10с-2 Введение

HDF5 использует формат с открытым исходным кодом. Он позволяет таким пользователям, как ИО, сотрудничать с Группой HDF в отношении требований к функциональности и, в соответствующих случаях, позволяет учитывать опыт и знания пользователей для включения в продукты HDF.

HDF5 особенно хорошо справляется с данными, где важны комплексность и масштабируемость. Данные практически любого типа или размера могут храниться в HDF5, включая комплексные структуры данных и типы данных. HDF5 портативен, работает на большинстве операционных систем и машин. HDF5 масштабируется - он хорошо работает в вычислительных средах высокого класса, и может вместить объекты данных практически любого размера или множественности. Он также может эффективно хранить большие объемы данных - он имеет встроенное сжатие. HDF5 широко используется в государственных, научных кругах и промышленности.

10с-3 Соответствие

Формат данных S-100 HDF5 соответствует HDF5 издание 1.8.8.

10с-4 Ссылки

10с-4.1 Нормативные ссылки

The HDF Group, November 2011, *HDF5 User's Guide Release 1.8.8*

The HDF Group, November 2011, *HDF5 Reference Manual 1.8.8*

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

ISO 19123, *Geographic information — Schema for coverage geometry and functions*

10с-4.2 Информативные ссылки

Gilbert, W., *A Cube-filling Hilbert Curve*, *Mathematical Intelligencer* 6(3), p.78, 1984

Goodchild, M. F. and Grandfield, A. W., *Optimizing Raster Storage: An Examination of Four Alternatives*, *Proceedings Auto-Carto* 6(1), pp. 400-407), Ottawa, 1983

Kidner, D.B., *Higher-order interpolation of regular grid digital elevation models*, *International Journal of Remote Sensing*, 24(14), July 2003, pp. 2981-2987. DOI: 10.1080/0143116031000086835

Kidner, D., Mark Dorey, M., & Smith, D., *What's the point? Interpolation and extrapolation with a regular grid DEM*, *Proceedings of the 4th International Conference on GeoComputation*, Fredericksburg, Virginia. URL: http://www.geocomputation.org/1999/082/gc_082.htm (retrieved 26 April 2018)

Laurini, R. and Thompson, D., *Fundamentals of Spatial Information Systems*, Academic Press, 1992

10с-5 Спецификация HDF5

HDF5 реализует модель управления и хранения данных. Модель включает абстрактную модель данных и абстрактную модель хранения данных (формат данных), а также библиотеки для реализации абстрактной модели и отображения модели хранения на различных средствах хранения. Библиотека HDF5 обеспечивает интерфейс программирования для конкретной реализации абстрактных моделей. Библиотека также реализует модель передачи данных, то есть эффективное перемещение данных из одного сохраненного представления в другое. На рисунке ниже показаны взаимосвязи между моделями и реализациями.



Рисунок 10с-1 – Абстрактная модель данных

Абстрактная модель данных (*Abstract Data Model*) – это концептуальная модель данных, типов данных и организации данных. Абстрактная модель данных не зависит от среды хранения или среды программирования. *Модель хранения* является стандартным представлением объектов абстрактной модели данных. Спецификация формата файла HDF5 определяет модель хранения.

Программная модель (*Programming model*) - это модель вычислительной среды, включающая платформы от небольших систем до крупных мультипроцессоров и кластеров. Модель программирования манипулирует (реализует, заполняет и извлекает) объектами из абстрактной модели данных

Библиотека (*Library*) – это конкретные результаты применения модели программирования. Библиотека экспортирует API HDF5 в качестве интерфейса. Помимо реализации объектов абстрактной модели данных, Библиотека управляет передачей данных из одной хранимой формы в другую. Примеры передачи данных включают чтение с диска в память и запись из памяти на диск.

Сохраненные данные (*Stored Data*) – это конкретная реализация модели хранения данных. Модель хранилища сопоставляется с несколькими механизмами хранения, включая файлы на одном диске, несколько файлов (семейство файлов) и представления памяти.

Библиотека HDF5 - это модуль C, реализующий модель программирования и абстрактную модель данных. Библиотека HDF5 вызывает операционную систему или другое программное обеспечение для управления хранилищем (например, библиотеку MPI/IO) для хранения и извлечения постоянных данных. Библиотека HDF5 может также связываться с другими программами, такими как фильтры для сжатия. Библиотека HDF5 связана с прикладной программой, которая может быть написана на C, C++, Fortran или Java. Прикладная программа реализует алгоритмы и структуры данных для конкретных проблем и вызывает библиотеку HDF5 для хранения и извлечения данных.

Библиотека HDF5 реализует объекты абстрактной модели данных HDF5. Некоторые из этих объектов включают группы, наборы данных и атрибуты. Спецификация продукта S-100 сопоставляет структуры данных S-100 с иерархией объектов HDF5. Каждая спецификация продукта S-100m создаст отображение, наиболее подходящее для его целей.

Объекты абстрактной модели данных HDF5 сопоставляются с объектами модели хранения HDF5 и хранятся на носителе. Хранимые объекты включают в себя блоки заголовков, свободные списки, блоки данных, B-деревья и другие объекты. Каждая группа или набор данных хранится в виде одного или нескольких блоков заголовков и данных.

10с-5.1 Абстрактная модель данных

Абстрактная модель данных (ADM) определяет понятия для определения и описания комплексных данных, хранящихся в файлах. ADM представляет собой весьма общую модель, которая концептуально охватывает многие конкретные модели.

Множество различных видов данных могут быть сопоставлены с объектами ADM, и поэтому хранятся и извлекаются с помощью HDF5. Однако ADM не является образцом какой-либо конкретной проблемы или области применения. Пользователям необходимо сопоставлять свои данные с концепциями ADM.

К ключевым концепциям относятся:

- *File (файл)* - непрерывная строка байтов в компьютерном хранилище (памяти, диске и т.д.), а байты представляют ноль или более объектов модели;
- *Group (группа)* - коллекция объектов (включая группы);
- *Dataset (набор данных)* - многомерный массив элементов данных с атрибутами и другими метаданными;
- *Dataspace (Пространство данных)* - описание размеров многомерного массива;
- *Datatype (Тип данных)* - описание конкретного класса элемента данных, включая его компоновку в виде набора битов;
- *Attribute (Атрибут)* - именованное значение данных, связанное с группой, набором данных или именованным типом данных;
- *Property List (Список свойств)* - набор параметров (некоторые постоянные и некоторые переходные) управляющих параметрами в библиотеке;
- *Link (Связь)* - способ соединения объектов.

Эти ключевые концепции более подробно описываются ниже.

10с-5.1.1 Файл

Абстрактно, файл HDF5 является контейнером для организованной коллекции объектов. Объекты - это группы, наборы данных и другие объекты, как определено ниже. Объекты организованы как корневой ориентированный граф.

Каждый файл HDF5 имеет по крайней мере один объект, корневую группу. Смотрите рисунок ниже. Все объекты являются членами корневой группы или потомками корневой группы.

Объекты HDF5 имеют уникальный идентификатор внутри одного файла HDF5 и могут быть доступны только по его именам в иерархии файла. Объекты HDF5 в различных файлах не обязательно имеют уникальные идентификаторы, и доступ к постоянному объекту HDF5 возможен только через файл.

Когда файл создан, в свойствах создания файла задаются параметры файла. Свойства создания файлов включают информацию о версии и параметры глобальных структур данных. Когда файл открыт, в свойствах доступа к файлу (*file access properties*) задаются параметры текущего доступа к файлу. Свойства доступа к файлам включают параметры драйверов хранения и параметры для кэширования и сбора мусора. Свойства создания файла устанавливаются постоянно на срок службы файла, а свойства доступа к файлу могут быть изменены путем закрытия и повторного открытия файла.

Файл HDF5 может быть "смонтирован" как часть другого файла HDF5. Это аналогично установке файловой системы Unix. Корень смонтированного файла присоединяется к группе в файле монтирования, и все содержимое может быть доступно, как если бы смонтированный файл был частью файла монтирования.

10с-5.1.2 Группа

Группа HDF5 аналогична директории файловой системы. Абстрактно, группа содержит ноль или более объектов, и каждый объект должен быть членом по крайней мере одной группы. Корневая группа является особым случаем; она не может быть членом какой-либо группы.

Членство в группе фактически реализуется через объекты ссылки (link). Смотрите рисунок ниже. Объект связи принадлежит группе и указывает на именованный объект (named object). Каждая ссылка имеет имя (name), и каждая ссылка указывает на ровно один объект. Каждый именованный объект имеет по крайней мере одну и, возможно, множество ссылок на него.

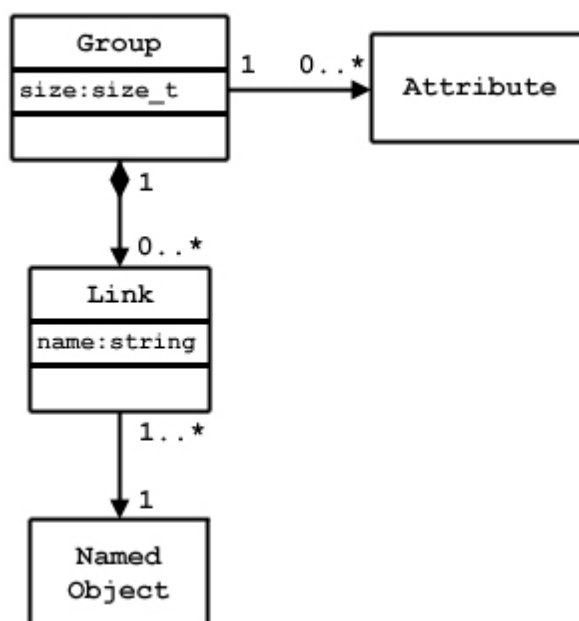


Рисунок 10с-2 - Членство группы через объекты ссылки

Существует три класса именованных объектов: группа, набор данных и именованный тип данных. Смотрите рисунок ниже. Каждый из этих объектов является членом по крайней мере одной группы, и это означает, что существует по крайней мере одна ссылка на него.

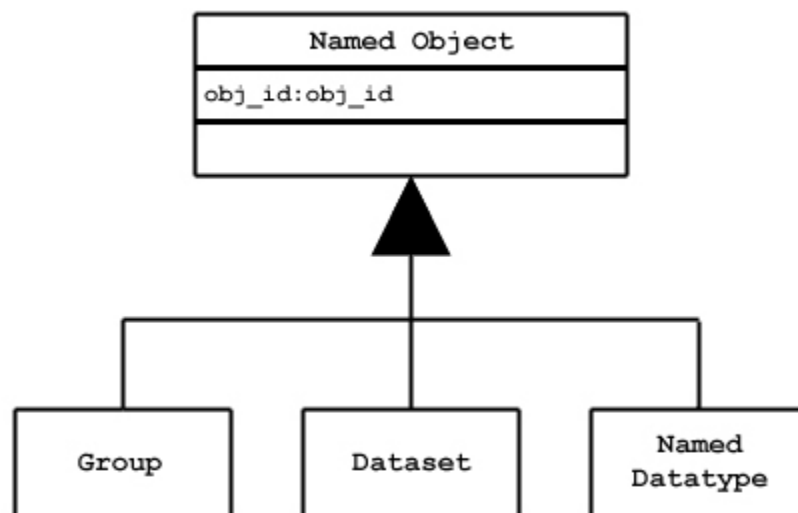


Рисунок 10с-3 - Классы именованных объектов

10с-5.1.3 Набор данных

Набор данных HDF5 - это многомерный массив элементов данных. Смотрите рисунок ниже. Форма массива (число измерений, размер каждого измерения) описывается объектом `dataspace`.

Элемент данных - это одиночная единица данных, которая может быть числом, символом, массивом чисел или символов или записью разнородных элементов данных. Элемент данных является набором битов. Разметка битов описывается типом данных (`datatype`).

Пространство данных и тип данных задаются при создании набора данных (`dataset`), и они не могут быть изменены в течение срока службы набора данных. Свойства создания набора данных задаются при создании набора данных. Свойства создания набора данных включают в себя значение заполнения и свойства хранения, такие как фрагментирование и сжатие. Эти свойства не могут быть изменены после создания набора данных.

Объект набора данных управляет хранением данных и доступом к ним. Хотя концептуально данные представляют собой смежный прямоугольный массив, они физически хранятся и передаются различными способами в зависимости от свойств хранилища и используемого механизма хранения. Фактическое хранение может быть набором сжатых фрагментов., и доступ может быть через различные механизмы хранения и кэши. Набор данных находится между концептуальным массивом элементов и фактическими хранящимися данными.

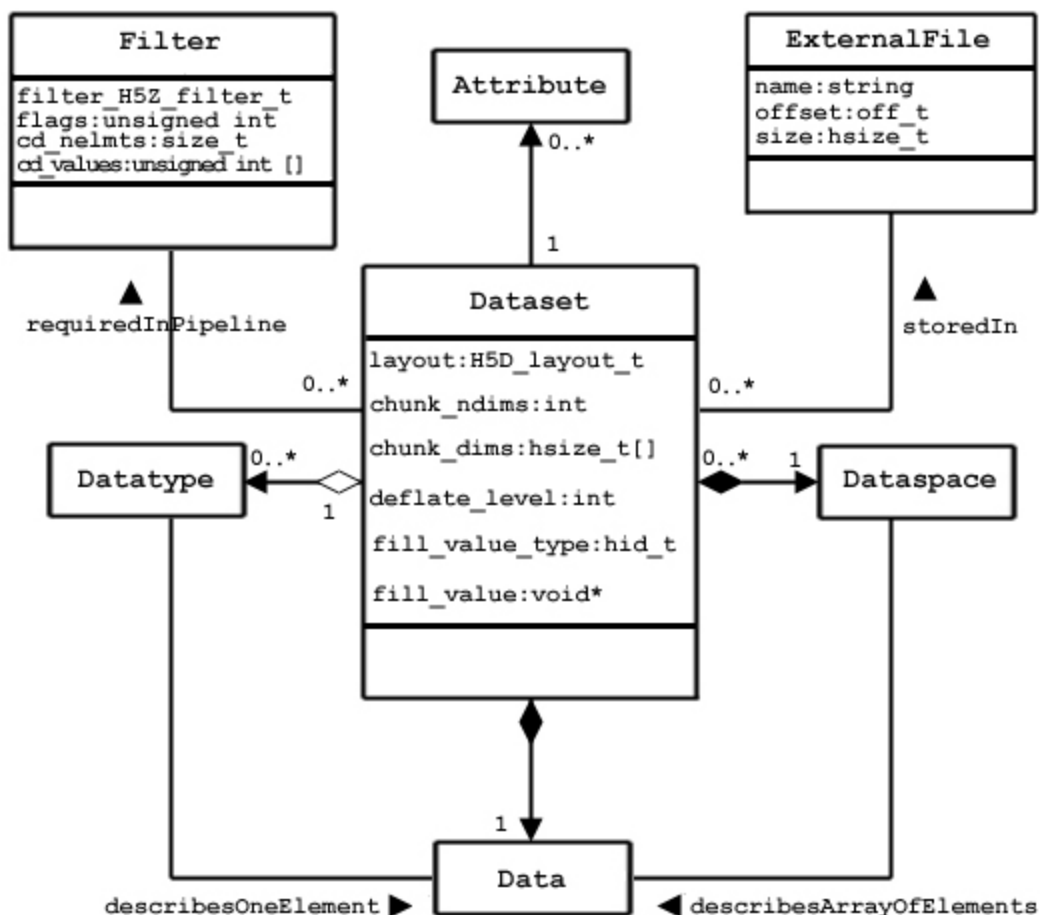


Рисунок 10с-4 – Набор данных

Концепция набора данных HDF5 означает массив, в то время как концепция S-100 определяется как "идентифицируемый сбор данных" (S-100 Приложение А - Термины и определения), который обычно интерпретируется как набор реализаций фичеров и/или типов информации.

В этой части термин "файл данных" часто используется для обозначения набора данных в смысле S-100, а "набор данных HDF5" - в смысле HDF. Если эти термины не используются, то смысл должен быть очевиден из контекста.

10с-5.1.4 Пространство данных (Dataspace)

Dataspace HDF5 описывает расположение элементов многомерного массива. Концептуально массив представляет собой гиперпрямоугольник от одного до 32 измерений. Dataspace HDF5 может быть расширяемым. Поэтому каждое измерение имеет текущий размер и максимальный размер, и максимум может быть неограниченным. Dataspace описывает этот гиперпрямоугольник. Это список размеров с текущими и максимальными (или неограниченными) значениями.

10с-5.1.5 Тип данных (DataType)

Объект типа данных HDF5 описывает макет одного элемента данных. Элемент данных является одним элементом массива; он может быть одним числом, символом, массивом чисел или носителей, или другими данными. Объект datatype описывает конфигурацию хранилища этих данных.

Типы данных делятся на 11 классов datatype. Каждый класс интерпретируется в соответствии с набором правил и имеет определенный набор свойств для описания его хранения. Например, числа с плавающей запятой имеют положение и размеры экспоненты, которые интерпретируются согласно соответствующим стандартам для представления чисел. Таким образом, класс datatype определяет значение элемента, а тип данных описывает, как он хранится.

На рисунке ниже показана классификация типов данных. Атомарные типы данных неразделимы. Каждый из них может быть одним объектом; числом, строкой или другими объектами. Композитные типы данных состоят из нескольких элементов атомных типов данных. В дополнение к стандартным типам, пользователи могут определять дополнительные типы данных, такие как 24-битное целое число или 16-битное десятичное.

Набор данных или атрибут имеют один связанный с ним объект типа данных. Смотрите Рисунок набора данных выше. Объект datatype может использоваться в определении нескольких объектов, но по умолчанию копия объекта datatype будет приватной для данного набора данных.

Опционально, объект datatype может храниться в файле HDF5. Тип данных связан в группу, и поэтому ему присваивается имя. Именованный datatype может быть открыт и использован любым способом, чтобы использовать datatype.

Не все типы данных HDF5 имеют точные эквиваленты в базовых и производных типах данных S-100, определенных в пункте 1-4.5.2 части 1 (таблица 1-2). Соответствия между HDF5 и S-100 приведены в таблице 10с-2 ранее.

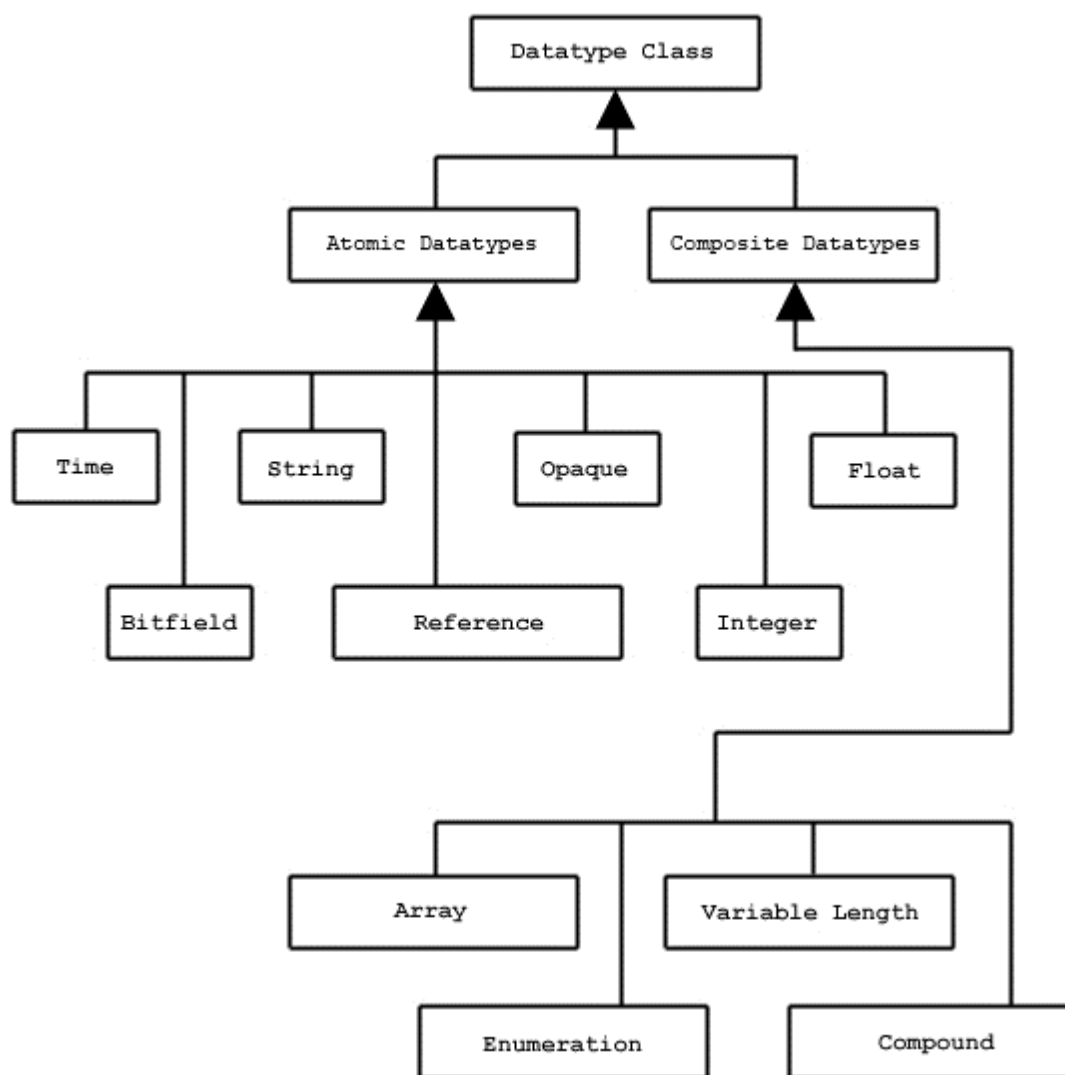


Рисунок 10с-5 – Классификация типа данных

10с-5.1.6 Атрибут

Любой именованный объект данных HDF5 (группа, набор данных или именованный тип данных) может иметь ноль или более определяемых пользователем атрибутов. Атрибуты используются для документирования объекта. Атрибуты объекта сохраняются вместе с объектом.

Атрибут HDF5 имеет имя и данные. Часть данных похожа по структуре на набор данных: `dataspace` определяет макет массива элементов данных, а `datatype` определяет макет хранилища и интерпретацию элементов. Смотрите рисунок ниже.

Атрибуты объектов данных в принципе эквивалентны тематическим атрибутам, но в этом издании профиля HDF5 не предусмотрены данные о векторных фичерах или типах информации в файлах HDF5 и поэтому не используются векторные атрибуты объектов. Атрибуты групп HDF5, наборов данных или именованных типов данных играют роль метаданных.

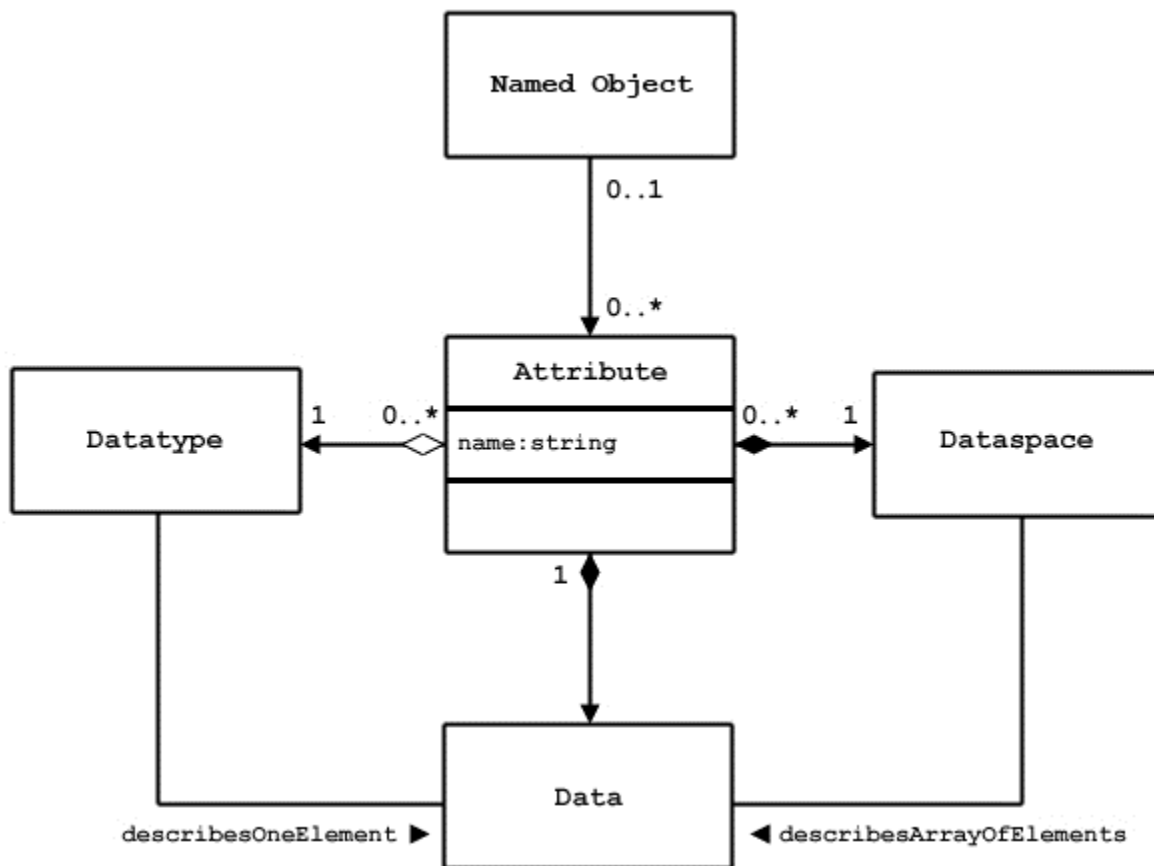


Рисунок 10с-6 – Элементы атрибутов данных

На самом деле, атрибут очень похож на набор данных со следующими ограничениями:

- Доступ к атрибуту возможен только через объект;
- Имена атрибутов важны только для объекта;
- Атрибут должен быть небольшим объектом;
- Данные атрибута должны читаться или записываться в одном доступе (частичное чтение или запись не допускается);
- Атрибуты не имеют атрибутов.

Обратите внимание, что значение атрибута может быть ссылкой на объект. Общий атрибут или атрибут большого массива может быть реализован как ссылка на набор данных.

Имя, дата и тип данных атрибута указываются при его создании и не могут быть изменены в течение срока службы атрибута. Атрибут может быть открыт по имени, по индексу или по итерации всех атрибутов объекта.

10с-5.1.7 Список свойств

HDF5 имеет общий объект списка свойств. Каждый список представляет собой набор пар имен-значений. Каждый класс списка свойств имеет определенный набор свойств. Каждое свойство имеет неявное имя, тип данных и значение. Объект списка свойств создается и используется аналогично другим объектам библиотеки HDF5.

Списки свойств прикреплены к объекту в библиотеке, они могут быть использованы любой частью библиотеки. Некоторые свойства являются постоянными (например, стратегия разбивки на части набора данных), другие - временными (например, размер буфера передачи данных). Общее использование списка свойств – это передача параметров вызывающей программы драйверу VFL или модулю конвейера.

Списки свойств концептуально похожи на атрибуты. Списки свойств – это информация, относящаяся к поведению библиотеки, а атрибуты – относятся к данным и приложению пользователя. Поскольку список свойств совпадает, спецификация данных для реализации списков свойств HDF5 в спецификациях продуктов S-100 не рекомендуется.

10с-5.2 Библиотека HDF5 и модель программирования

Библиотека HDF5 реализует абстрактную модель данных HDF5 и модель хранения. Две основные цели продуктов HDF5 заключаются в предоставлении инструментов, которые могут быть использованы на как можно большем числе вычислительных платформ (портативность) и обеспечении разумной объектно-ориентированной модели данных и интерфейса программирования.

Более подробная информация о реализации модели HDF5 содержится в Руководстве пользователя HDF5 Версия 1.8.8 и в Справочном руководстве HDF5 1.8.8. Спецификации на производство продуктов S-100 должны указывать группы HDF5, наборы данных и атрибуты в контексте общей модели фичеров S-100.

10с-5.3 Запрещенные конструкции HDF5

Не должны использоваться конструкции, которые не могут быть обработаны с использованием стандартных библиотек HDF5, указанных в этой части. Это означает, в частности, что конструкции HDF5, требующие использования библиотеки более позднего выпуска, чем тот, который указан в этой части, не должны использоваться.

10с-6 Профиль HDF5 S-100

Профиль HDF5 S-100 ограничивает типы данных и конструкции HDF5, которые могут использоваться в наборах данных HDF5 S-100; описывает соответствие

между типами данных S-100 и HDF5 и другими конструкциями; и определяет правила структурирования наборов данных HDF5 S-100.

Профиль HDF5 S-100 должен применяться к видам информации, перечисленным ниже – отмечая, что эти типы не являются взаимоисключающими, хотя в большинстве спецификаций отдельных продуктов будет использоваться только набор возможных комбинаций:

- данные по одной или нескольким отдельным стационарным станциям;
- данные с привязкой к регулярной сетке;
- данные с привязкой к нерегулярной сетке;
- сетки с переменными размерами ячеек;
- данные с привязкой к негеисправленной сетке (пункт 8-8.1.2 части 8);
- данные TIN;
- данные о движущейся платформе (например, о поверхностном дрейфе);
- статические данные или данные временных рядов (для любого другого вида данных) с фиксированными или переменными интервалами;
- тайловые и не тайловые покрытия;
- несколько классов фичеров в одном файле данных;
- несколько типов покрытий в одном файле данных.

Ограничения, соответствия и правила описаны в следующих разделах.

10с-7 Типы данных

Предопределенные типы данных HDF5 включают Integer, Float, String и Enumeration, но нет эквивалентов HDF5 для типов данных S-100 Boolean, S100_Codelist или S100_TruncatedDate. Последние типы сопоставляются с конструкциями HDF5, указанными в таблице ниже. Типы данных S-100 Date, DateTime и Time сопоставляются со строками HDF5 из-за потенциальных проблем с переносимостью на различные архитектуры процессоров HDF5 Time. В продуктах данных S-100 HDF5 типы данных S-100, определенные в части 3, сопоставляются с эквивалентными типами данных HDF5. Эти эквивалентные показатели кратко изложены в таблице 10с-1 ниже. Классы типов данных HDF5, не упомянутые в настоящей таблице, не используются.

Таблица 10с-1 – Эквивалентность типов данных S-100 и HDF5

Типы значений атрибутов S-100	Класс типа данных HDF5	Ограничение типа данных HDF5
real	Float (десятичный)	32 или 64-бит с плавающей точкой
integer	Integer (целый)	1, 2 или 4-байт целые со знаком и без знака
text (CharacterString in S-100 metadata)	String (строка)	строка переменной длины
enumeration	Enumeration (перечень)	Числовые коды должны быть 1 или 2-байтовыми беззнаковыми целыми числами, диапазон $[1, 2^8 - 1]$ или $[1, 2^{16} - 1]$
date	(Character)(символ) String, длина=8	Формат даты согласно таблице 1-2 (часть 1), то есть, полное представление, базовый формат, как указано в ISO 8601

time	(Character) Variable-length String (строка переменной длины)	Формат времени в соответствии с таблицей 1-2 (часть 1), то есть полное представление, базовый формат, как указано в ISO 8601. UTC обозначается суффиксом "Z"; местное время - отсутствием суффикса. Также допускается формат смещения для зоны); например, 123000+0100
dateTime	(Character) (строка) (строка переменной длины)	Формат даты и времени, указанный в ISO 8601. ПРИМЕРЫ: 19850412T101530Z 19850412T101530-0500
boolean	(Integer)	1 байт без знака, Значения: 1 (TRUE); 0 (FALSE)
S100_Codelist	Compound (комплексный) (Перечень, строка переменной длины)	Допускается использование только одного компонента; другим должно быть числовое значение 0 или пустая строка (0-длина) в соответствии с ее типом данных
URI, URL, URN	String (variablelength)	Формат указанный в RFC 3986 (URI, URL) или RFC 2141 (URN)
S100_TruncatedDate	String, длина=8	Формат в соответствии с частью 1 таблица 1-2
Значение записи (Часть 8)	Compound	Типы данных компонентов должны соответствовать значению типов атрибутов в прикладной схеме. Значение "value record" соответствует значению записи в части 8 Рис. 8-21, 8-22, 8-23, 8-28, 8-29
Ссылка на внешний объект	String	Формат: extObjRef:<fileName>:<recordIdentifier> где <fileName> - это базовое имя ISO 8211 или файла GML, а <recordIdentifier> - это идентификатор записи векторного объекта в этом файле. Часть расширения имени файла не используется. Идентификатором записи является идентификатор gml:id для наборов данных GML или идентификационный номер записи (RCID) для наборов данных ISO 8211. Файл должен присутствовать в том же наборе обмена.

10с-8 Правила присвоения имен

Имена элементов HDF5 (наборы данных, объекты и т.д.), которые кодируют элементы данных в прикладной схеме (например, классы фичеров, атрибуты, роли, перечни, списки кодов и т.д.), должны соответствовать именам в прикладной схеме (так как существует 1/1 сопоставление прикладной схемы с каталогом фичеров, это также означает, что требуется такое же соответствие каталогу фичеров). Имена должны указываться в виде camel case. В других разделах этой части указано, где используются названия из прикладной схемы (или, эквивалентно, из каталога фичеров).

Элементы встроенных ("носителей") метаданных и информации о местоположении, которые соответствуют атрибутам частей 4а-4с, должны также соответствовать именам в частях 4а-4с и 8.

Элементы, которые не имеют прямого соответствия, могут иметь имена, уникальные для формата HDF5 (различия предназначены для упрощения абстракций в ISO 19123 и S-100 части 4, 4b и 8, и сокращения полей, которые глубоко вложены в схему XML).

Названия 'latitude' и 'longitude' должны использоваться для географических координатных осей, когда они уместны, вместо 'X' и 'Y', которые должны

использоваться только в тех случаях, когда понятие широта/долгота не используется.

Соответствие между элементами метаданных носителя в этом профиле и частями 4-4с и частью 8 указывается ниже в настоящем документе.

Названия в не встроенных метаданных и каталогах наборов обмена рассматриваются как спецификации векторных продуктов - то есть они должны соответствовать стандартным метаданным и схемам каталогов обмена.

Группе HDF5, которая соответствует элементу схемы, уже названному в S-100 или в спецификации продукта, должно быть дано то же имя, что и этому элементу, используя код camelcase, если он указан. Например, если продукт временных рядов указывает имена для коллекций данных в определенные моменты времени, эти имена следует использовать в качестве имен групп, если коллекция закодирована как группа. (Разработчики спецификации продукта должны позаботиться о том, чтобы указать имена коллекций, которые соответствуют разрешенному синтаксису HDF5.)

Числовые суффиксы, которым предшествует символ подчеркивания (то есть суффикс 'NNN'), могут быть добавлены для различия групп, которые в противном случае имели бы одинаковые имена (например, группы одинаковых типов данных в разных точках времени).

Следующие имена групп зарезервированы для указанных видов использования:

Таблица 10с-2 – Зарезервированные имена групп

Positioning	Дискретная информация о местоположении всех видов и размеров. Тип данных позиционирования указывается атрибутом или атрибутами группы. Включает сжатые или компактные кодировки. Не включает позиционирование, которое может быть полностью определено по параметрам сетки или покрытия (такие параметры кодируются в атрибутах, прикрепленных к корневой группе). Спецификации, требующие неоднородного позиционирования (например, алгебраические формулы второго порядка), должны рассматриваться как не геопространственные сетки.
Group_F	Информация о фичерах. Например, имена фичеров и атрибутов, коды, типы, множественности, роли и т.д. Также включает в себя форматные метаданные, специфичные для формата HDF5.
Group_IDX	Индексы, если они закодированы в группе HDF5. Включает индексы разрозненных массивов.
Group_TL	Информация о тайлинге, если закодирован в группе.
Group_nnn	Данные по одному члену ряда; например, в определенный момент времени во временном ряду; или различные станции. "n" означает любую цифру от 0 до 9. Нумерация должна использовать 3 цифры, 001-999.

10с-9 Структура продукта данных

10с-9.1 Общая структура

Файл S-100 HDF5 состоит из групп, каждая из которых может содержать другие группы, атрибуты и наборы данных (HDF). Группы - это контейнеры для различных типов информации (означающие значения данных, информацию о местоположении, метаданные или дополнительную информацию). Наборы данных HDF предназначены для хранения больших объемов цифровых данных и могут

использоваться для хранения данных покрытия. Атрибуты предназначены для хранения однозначной информации, относящейся к группам или наборам данных, и могут использоваться для хранения определенных типов метаданных.

Следующие группы содержатся в корневой группе. (Уровни вложенности в списке ниже соответствуют уровням вложенности в файле HDF5.)

- 1) Группа информации о фичере.
- 2) Группы контейнеров фичеров - каждая из них действует как контейнер для отдельных реализаций класса фичера. Его атрибуты кодируют любые метаданные уровня класса фичера.
 - a) Группы реализаций фичеров - каждая из них действует как контейнер для позиционирования, тайлов, индексов и групп данных, относящихся к одной реализации фичера. Их атрибуты кодируют любые метаданные уровня реализации.
 - i) Информационная группа тайлинга (условная, только если значения хранятся в виде тайлов).
 - ii) Группа индексов (условная, только если требуются индексы данных).
 - iii) Группа позиционирования (условная, только если положения не вычислимы из метаданных).
 - iv) Группа(ы) значений данных. Только данные временных рядов будут иметь более одной группы значений.

Обратите внимание, что порядок хранения групп и наборов данных в файле может отличаться от порядка их создания.

Основная структура файла S-100 HDF5 показана на рисунке ниже. 'F' - число классов фичеров, определенных в спецификации продукта. Не обязательно, чтобы каждый файл данных содержал реализации всех классов фичеров. Существует одна группа значений для каждой временной точки во временных сериях¹ (наборы данных, не являющиеся временными рядами, будут иметь только одну группу значений в каждой группе реализации фичеров).

Группы FeatureContainer и Positioning являются абстрактными классами, поскольку их атрибуты и содержание зависят от типа покрытия.

Более подробный рисунок показан ниже в этой части.

¹ За исключением данных о движущихся и фиксированных станциях. Ниже в настоящей части описывается использование групп значений для каждого типа покрытия.

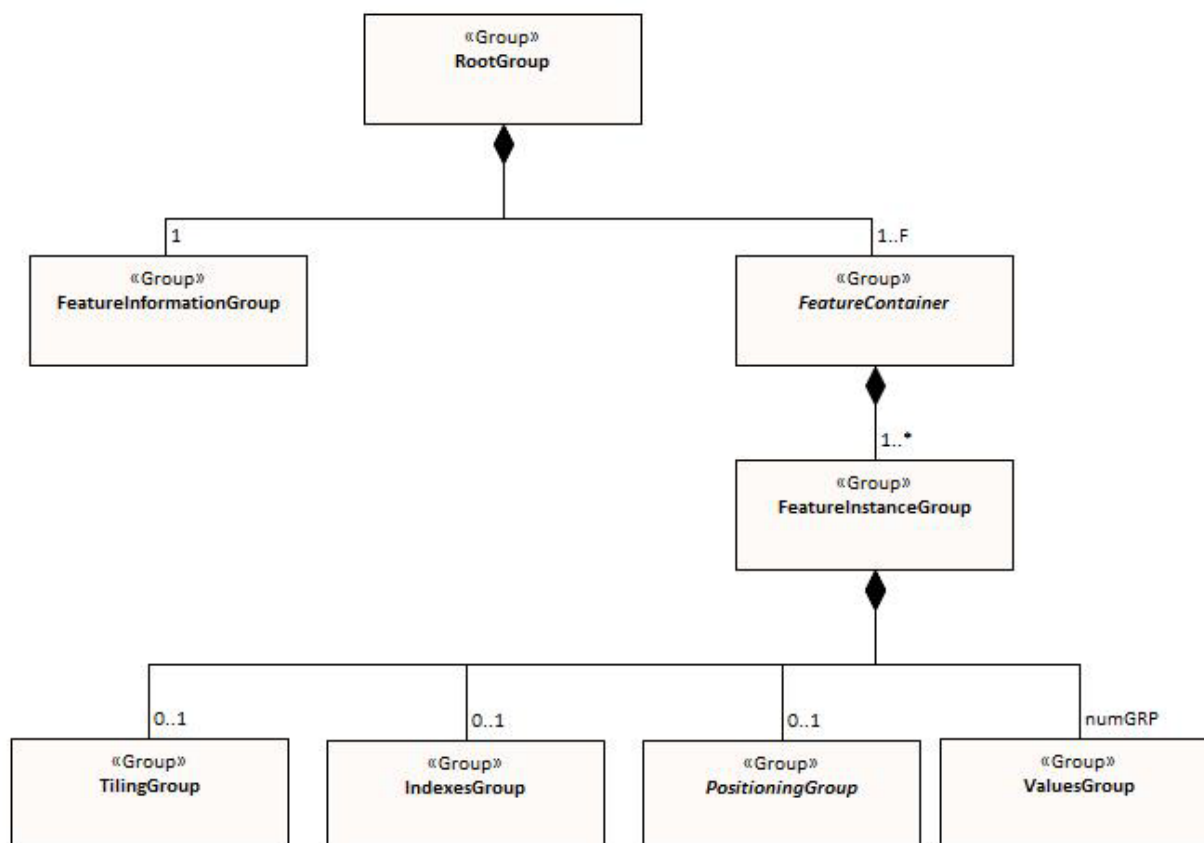


Рисунок 10с-7 – Базовая структура файла HDF5 S-100

10с-9.1.1 Индексация элементов массива и сетки

Массивы: При использовании индекса, ссылающегося на элемент массива, явно или неявно, этот массив должен рассматриваться как использующий ноль-индексацию.

ПРИМЕР: Член массива треугольников TIN определяет конкретный треугольник путем кодирования индексов первой, второй и третьей пары координат записей в массиве `geomeyValues`. Значения, закодированные в этом члене массива треугольников, являются 0, 1 и 2 соответственно. (Это явное использование индексных значений массива `geomeyValues`.)

Точки и ячейки сетки: При использовании индексного значения, ссылающегося на конкретные точки или ячейки сетки, либо явно, либо косвенно, к точкам или ячейкам сетки следует относиться так же, как и при ноль-индексации.

ПРИМЕР: При генерации кодов Мортон для нерегулярных сеток или переменных размеров ячеек используется ноль-индексация для точек сетки.

ПРИМЕЧАНИЕ (информативное): Разработчикам приложений может потребоваться учесть библиотечные или языковые соглашения. Например, язык, который использует индексацию массивов на базе 1-индексации, должен будет добавить 1 к значению индекса, закодированному в массиве треугольников, чтобы сослаться на предназначенный элемент в массиве `geomeyValues`.

10с-9.2 Метаданные

Метаданные определяются на различных уровнях логической структуры, так что метаданные в корневой группе применяются ко всем фичерам файла, метаданные на уровне контейнера фичеров применяются ко всем реализациям этого класса фичеров, и метаданные на уровне реализации применяются только к конкретной реализации фичера.

10с-9.2.1 Метаданные обнаружения

Полные метаданные обнаружения кодируются во внешнем файле метаданных обнаружения, как указано в частях 4a (метаданные) и 4b (метаданные изображений и сеточные данные). Смотрите пункт 10с-12 для правил присвоения имен.

10с-9.2.2 Метаданные носителя (встроенные)

Метаданные носителя - это метаданные, которые кодируются в файле HDF5. Они делятся на общие, типы и метаданные реализаций в зависимости от того, относятся ли они к файлу HDF5 в целом, описывают структуру и атрибуты классов объектов данных или предоставляют параметры, необходимые для чтения реализаций классов объектов данных. Метаданные кодируются в следующих местах:

- Общие метаданные, определяемые как общие параметры, применимые к файлу в целом. Общие метаданные состоят из параметров, применимых ко всей информации в файле данных, таких, как дата выпуска, данные и общая пространственная протяженность (границы покрытия данными). Это включает в себя основные общие элементы для обработки и расположение ячейки (остальная важная информация закодирована в реализации фичера). Эти метаданные кодируются как атрибуты корневой группы;
- Метаданные типа, определяемые как конкретные характеристики, которые описывают классы объектов данных в файле (например, относятся к конкретным фичерам и атрибутам) и которые, таким образом, будут различными для каждого класса фичеров. Эти метаданные используются для информации о фичерах и атрибутах (соответствующей записям в каталоге фичеров). Информация этого типа аналогична каталогу фичеров, описанному в Части 5, но может содержать только выдержки из Каталога фичеров, а также добавлять атрибуты формата, относящиеся только к кодировкам HDF5. Метаданные типа кодируются как содержимое (наборы данных HDF5) группы информации о фичерах. Группа информации о фичерах (Group_F) также является будущим контейнером для информации из каталога набора обмена или о файлах поддержки, если необходимо включить это в файл HDF5 и это не применимо к файлу в целом;
- Метаданные реализаций, определяемые как параметры для каждого класса фичеров в прикладной схеме. Сюда входят параметры, необходимые для чтения информации, содержащейся в продукте данных, даже в случае отсутствия внешних файлов метаданных, включая пространственные

параметры покрытия (масштабы, параметры сетки). Эти метаданные могут включать параметры, которые имеют значение только в контексте конкретного типа(ов) пространственного покрытия, разрешенного для класса фичеров в прикладной схеме. Эти метаданные кодируются как атрибуты реализаций в рамках каждой группы контейнеров фичеров.

Дополнительная информация, описывающая данные, содержится в группе значений в качестве атрибутов, применимых к набору данных в каждой группе значений. Данные могут представлять собой информацию о моменте времени или о станции, например название станции; и характеристики временных рядов, такие как временной интервал, количество значений и время начала и окончания.

10с-9.2.3 Расширенные метаданные

Расширенные элементы метаданных, определенные в спецификации продукта, кодируются либо как указано ниже, либо как оба из них:

- Дополнительные атрибуты корня контейнерной группы фичеров в зависимости от того, считаются ли они необходимыми для обработки и относятся ли они к данным в целом или к реализациям фичеров. Пример приводится ниже в этой части (таблица 10с-7). (Обратите внимание, что любые расширенные метаданные, необходимые для обработки, подразумевают использование модулей по конкретным продуктам.);
- Расширенные метаданные во внешних XML-файлах, кодирующие метаданные обнаружения или каталог обмена, если они считаются метаданными обнаружения.

Продукты данных могут также определять метаданные векторных фичеров; например, мета фичеры качества для векторной геометрии. Векторные фичеры кодируются не в файле HDF5, а в отдельном файле, соответствующем части 10a или части 10b. Если присутствуют векторные мета фичеры, ссылка на отдельный файл должна быть включена в метаданные носителя, с его именем в атрибуте `metaFeatures` (смотрите пункт 10с-9.4).

10с-9.3 Обобщенные размеры и хранение координат и данных

В этом разделе представлен обзор общего подхода к представлению информации о местоположении и хранению данных в наборах данных S-100 HDF5. Основной подход заключается в минимизации разнообразия структур данных, используемых для хранения записей данных. Этот профиль хранит данные одним из двух способов:

- 1) Многомерный массив данных ранга и размеров, точно соответствующий форме сетки. Используется только для регулярных сеток. В целях сокращения потребностей в пространстве координаты точек сетки в явной форме не хранятся, поскольку они могут быть рассчитаны на основе параметров сетки;

2) Одномерные массивы данных и координаты сетки, сопровождаемые метаинформацией, описывающей форму сетки. Это также используется для многоточечных данных (где нет фактической сетки).

Ключевая идея, лежащая в основе структуры, заключается в следующем: организация данных логически одинакова для каждого из различных типов данных, но сама информация будет интерпретироваться по-разному в зависимости от типа пространственного представления, который указывается атрибутом метаданных `dataCodingFormat` (определяется в таблице 10с-10 и пункте 10с-4).

В случае данных с привязкой к регулярной сетке информация о местоположении не хранится в виде явных координат, поскольку метаданные сетки (объем и интервалы между ячейками сетки) достаточны для указания координат каждой точки сетки. Например, для 2-D-сетки массивы значений являются двумерными, с размерами, заданными атрибутами `numPointsLongitudinal` и `numPointsLatitudinal`. Зная происхождение сетки и интервалы между точками сетки, положение каждой точки сетки можно рассчитать по простым формулам.

Для данных только с нерегулярной сеткой имеется дополнительная информация о местоположении. Характер информации о местоположении зависит от типа данных:

- Для стационарных станций и данных подвижных платформ информация о местоположении хранится в виде явных координат в одномерных массивах размером составных элементов `numPOS`. Компоненты составного элемента соответствуют координатным осям, например, широте, долготе, z-координате, времени и т.д. Последовательность точек соответствует либо положению стационарных станций, либо, в соответствующих случаях, последовательному положению подвижных платформ.
- Для не геопространственных сеток информация о местоположении также хранится в виде явных координат в одномерных массивах размера составных элементов `numPOS`, которые содержат координаты (как определено выше).
- Для нерегулярных сеток информация о местоположении хранится в виде одномерных массивов размером составных элементов `numPOS`, содержащих информацию о местоположении заполненных ячеек. Значения координат для каждой точки сетки явно не хранятся. Кроме того, группа тайлинга может быть заполнена тайлами, пространственное соединение которых точно покрывает сетку. Последовательность массивов расположения ячеек должна соответствовать атрибуту метаданных `sequencingRule` в контейнерной группе фичеров (раздел 10с-9.6). Необязательный компонент индекса тайлов (индекс в массиве тайлов - см пункт 10с-9.7) может быть добавлен спецификацией продукта для ускорения поиска. Если используется, компонент индекса тайла должен быть назван `tileIndex` и иметь целочисленный тип данных. Этот формат предназначен для сетки нерегулярной формы на основе единообразных прямоугольных ячеек.
- Для сетки с переменными размерами ячеек информация о местоположении хранится в виде двух одномерных массивов размером составных элементов `numPOS`, один массив, содержащий информацию о местоположении ячеек (как для нерегулярных ячеек), а другой о размерах ячеек. Значения

координат для каждой точки сетки явно не хранятся. Фактический размер ячейки описывается в терминах агрегаций значений размера ячейки. Формат предполагает, что различные ячейки приведены в соответствие с сеткой и что размеры ячеек кратны размеру ячейки в каждом измерении.

- Для данных TIN информация о местоположении хранится в виде одномерных массивов размера numPOS, кодирующих расположение вершин (с использованием того же типа составных элементов, что и для не геопространственных решеток выше) плюс массив треугольников, кодирующий ссылки на вершины треугольника и ссылки на смежные треугольники.

Для нерегулярных ячеек и ячеек переменных размеров вспомогательные массивы, описывающие местоположение и размеры ячеек, хранятся в группе значений, а не в группе позиционирования (это позволяет по-разному агрегировать ячейки в разных точках времени в формате переменных ячеек). Информация о хранении данных и значений координат обобщается в таблице ниже. (D' - это число измерений покрытия.)

Наборы данных HDF, хранящие координаты и значения, разработаны таким образом, чтобы использовать единообразные структуры хранения данных по различным типам покрытия, а также уменьшить общий объем данных. Эти критерии привели к тому, что дополнительная информация, необходимая для некоторых типов покрытия, хранится отдельно (например, информация о местоположении и размере ячеек для нерегулярных и переменных ячеек).

Таблица 10с-3 – Общие стратегии хранения координат и значений данных

Тип покрытия	Значения координат	Значения данных
Регулярная сетка	Не хранится явно Вычисляется по метаданным	D-мерный массив кортежей значений
Нерегулярная сетка	Не хранится явно Вычисляется по метаданным	1-мерный массив кортежей значений + информация о расположении ячеек
Сетка с переменным размером ячеек	Не хранится явно Вычисляется по метаданным	1-мерный массив кортежей значений + информация о размерах и расположении ячеек
Стационарные станции, Не геопространственная сетка, подвижная платформа	1-мерный массив кортежей координат	1-мерный массив кортежей значений
TIN	1-мерный массив кортежей координат + Информация о треугольниках	1-мерный массив кортежей значений

Группы данных - это отдельные группы, содержащие значения данных, которые хранятся в массивах, соответствующих информации о местоположении. Для типов покрытия, где информация о местоположении явно не хранится (N-мерные регулярные сетки), данные хранятся в N-мерных массивах ранга, соответствующего размерности сетки (например, для 2-D данных, 2-D массивы размера numROWS по numCOLS).

Для данных временных рядов присутствуют несколько групп данных. Общее число групп данных numGRP. Значение numGRP для каждого типа пространственного представления указано в таблице 10с-4. Формат позволяет получать данные временных рядов для всех представлений.

Положение в системах координат с более чем 2 осями координат кодируются с использованием соответствующих размерностей. Например, для 3-мерных данных вертикальное измерение используется в качестве третьего измерения.

Для обеспечения эффективности обработки в этом профиле рекомендуется ограничить число измерений не более чем четыремя (пространство и время), однако для продукта данных может использоваться более высокая размерность.

Переменные, определяющие размер массива (numROWS, numCOLS, numPOS и numGRP), различаются в зависимости от используемого формата кодирования. Они указаны в таблице 10с-4.

Таблица 10с-4 – Размерность массивов для различных типов покрытий

Формат	Тип данных	Положение	Значения данных			Время
		numPOS	numCOLS	numROWS	numZ (только 3-d)	numGRP
1	Постоянные станции	numberOfStations	1	numberOfStations	1	numberOfTimes
2	Регулярная сетка	(not used)	numPointsLongitudinal	numPointsLatitudinal	numPointsVertical	numberOfTimes
3	Не геопространственная сетка	numberOfNodes	1	numberOfNodes	1	numberOfTimes
4	Подвижная платформа	numberOfTimes	1	numberOfTimes	1	1
5	Нерегулярная сетка	numberOfNodes	1	numberOfNodes	1	numberOfTimes
6	Ячейка переменного размера	numberOfNodes	1	numberOfNodes	1	numberOfTimes
7	TIN	numberOfNodes	1	numberOfNodes	1	numberOfTimes
8	Фиксированные станции	numberOfStations	1	numberOfTimes	1	numberOfStations
9	Регулярная сетка, ориентированная на фичеры	Не используется	numPointsLongitudinal	numPointsLatitudinal	1	numberOfTimes

Обратите внимание, что numROWS, numCOLS, numZ и numPOS не кодируются явно в файле HDF5. Эта спецификация использует их только для указания размеров массива для целей применения. Это число станций, узлов, точек и т.д. которые кодируются как атрибуты реализаций фичеров (пункт 10с-9.7).

Имя каждой группы данных начинается с символов 'Group_nnn', где n нумеруется от 1 до numGRP. Допускается максимум 999 групп данных. Длина имени группы данных составляет 9.

Для всех типов данных логическая структура продукта в HDF5 состоит из (а) блока метаданных, за которым следует (b) группа информации о фичерах, затем (с) одна или несколько групп контейнеров данных, каждая из которых содержит одну или несколько групп реализаций фичеров, которые в свою очередь содержат тайлинг, индексирование, позиционирование и группы данных, описанные в пункте 10с-9.1. Группы тайлинга, индексации и позиционирования условно требуются в зависимости от типа данных, указанных атрибутом HDF5, который определяет формат кодирования.

Физическая компоновка файла может не совпадать с его логической структурой данных, однако API HDF5 позволяет пользователям получать доступ к информации, используя логическую структуру данных.

В следующих разделах описываются содержание и атрибуты каждой группы.

10с-9.4 Корневая группа

Корневая группа действует как контейнер для других групп. Метаданные носителя (Таблица 10с-6) содержатся как атрибуты в корневой группе. Метаданные носителя состоят из данных и параметров (а), необходимых для чтения и интерпретации информации в продукте, даже если файлы внешних метаданных отсутствуют, и, главным образом, (b) не включены в другие метаданные.

Таблица 10с-5 – Корневая группа

Группа	Категория HDF5	Имя	Тип данных	Пространство данных / Примечания	
/ (root)	Атрибуты	(Атрибуты метаданных носителя)	Integer, Float, Enumeration, или String	(нет) Описаны в таблице 10с-6	
	Группа	Group_F		Группа информации о фичерах (см. пункт 10с-9.6)	
	Группа(ы)	(featureCode)			Группа контейнеров фичеров - одна группа для каждого типа фичеров в продукте данных. Название - код фичера, который дается в Group_F. См. пункт 10с-9.6 по структуре и атрибутам
		Категория HDF5	Имя		
			(featureCode).N		Группа(ы) реализаций фичеров – одна для каждой реализации фичера. См. пункт 10с-9.7 по структуре и атрибутам
		Категория HDF5	Имя		
	Группа (опционно)	Group_TL		Информация о тайлинге, только если продукт использует тайлы. См. пункт 10с-9.8	

			Группа (опционно)	Group_IDX		Информация о пространственном индексе, только если продукт использует пространственные индексы См. пункт 10с-9.9
			Group	Positioning		Информация о местоположении – 2D или 3D. Не требуется для dataEncodingFormat = 2 (Регулярная сетка). См. пункт 10с-9.10
			Group(s)	Group_NNN		Статические данные - только 1 группа значений Данные временных рядов – от 000 до 999 групп См. пункт 10с-9.11

Общие (основные) элементы метаданных указываются как атрибуты корневой группы, перечисленные в таблице 10с-6. Корневая группа содержит только подмножество элементов минимальных метаданных, указанных в частях 4а и 4б. Внешний файл метаданных XML должен содержать все обязательные элементы метаданных.

Таблица 10с-6 – Встроенные в корневую группу метаданные (метаданные носителей)

No	Name	Camel Case	Mult	Data Type	Remarks and/or Units
1	Номер и версия спецификации продукта	productSpecification	1	String	Например ² , 'INT.IHO.S-NNN.X.X', с Xs, представляющим номер версии. "NNN" и "X" не подразумевают ограничения по длине. Соответствуют комбинации полей S100_ProductSpecification productIdentifier и version.
2	Время выпуска продукта данных	issueTime	0..1	String (Time format)	Должно быть совместимо с issueTime в метаданных обнаружения
3	Дата выпуска	issueDate	1	String (Date format)	Должно быть совместимо с issueDate в метаданных обнаружения

² Подлежит замене общим форматом, используемым во всех продуктах на основе S-100, после завершения работы над ним.

4	Горизонтальная CRS	horizontalCRS	1	Integer	Код EPSG или -1, если определена пользователем
5	Имя горизонтальной CRS	nameOfHorizontalCRS	0..1	String	Обязательно, если horizontalCRS = -1
6	Тип горизонтальной CRS	typeOfHorizontalCRS	0..1	Enumeration	Обязательно, если horizontalCRS = -1 См. таблицу 10с-23
7	Горизонтальная система координат	horizontalCS	0..1	Integer	Обязательно, если horizontalCRS = -1 Допустимые значения если typeOfHorizontalCRS = 1 (Геодезическая CRS 2D): <ul style="list-style-type: none"> • 6422 (Шир, Долг – градусы) Допустимые значения если typeOfHorizontalCRS = 2 (Проекционная CRS): <ul style="list-style-type: none"> • 4400 (Easting, Northing – metres) • 4500 (Northing, Easting – metres)
8	Горизонтальный датум	horizontalDatum	0..1	Integer	Обязательно, если horizontalCRS = -1 Код EPSG или -1, если определена пользователем
9	Имя горизонтального датума	nameOfHorizontalDatum	0..1	String	Обязательно, если horizontalCRS = -1
10	Начальный меридиан	primeMeridian	0..1	Integer	Обязательно, если horizontalCRS = -1, код EPSG
11	Сфероид	spheroid	0..1	Integer	Обязательно, если horizontalCRS = -1, код EPSG
12	Метод проекции	projectionMethod	0..1	Float	Обязательно, если horizontalCRS = 2, код EPSG, см. тб.10с-26
13	Параметр проекции 1	projectionParameter1	0..1	Float	Если только используется projectionMethod. См. тб. 10с-26
14	Параметр проекции 2	projectionParameter2	0..1	Float	Если только используется projectionMethod. См. тб. 10с-26
15	Параметр проекции 3	projectionParameter3	0..1	Float	Если только используется projectionMethod. См. тб. 10с-26
16	Параметр проекции 4	projectionParameter4	0..1	Float	Если только используется projectionMethod. См. тб. 10с-26
17	Параметр проекции 5	projectionParameter5	0..1	Float	Если только используется projectionMethod. См. тб. 10с-26
18	Ложный север	falseNorthing	0..1	Float	Если только используется projectionMethod. Применяется к координатам по оси север. [m]
19	Ложный восток	falseEasting	0..1	Float	Если только используется projectionMethod. Применяется к координатам по оси восток. [m]
20	Эпоха реализации	epoch	0..1	String	Код, обозначающий период времени геодезических данных, используемых CRS. Например, G1762 для реализации 2013-10-16 геодезических данных WGS84
21a	Границы района	westBoundLongitude	1	Float	См. dataCoverage.boundingBox > EX_GeographicBoundingBox
21b		eastboundLongitude	1	Float	

21c		southBoundLatitude	1	Float	Каждый компонент, ограничивающий район, кодируется как отдельный атрибут. Минимальная/максимальная широта и долгота данных базируется на географической CRS, которая использует тот же горизонтальный датум что и CRS, определенная для набора данных, который может быть защищен. Единицы измерения метры.
21d		northBoundLatitude	1	Float	
22	Географическое расположение ресурса (по описанию)	geographicIdentifier	0..1	String	EX_Extent > EX_GeographicDescription.geographicIdentifier > MD_Identifier.code
23	Метаданные	metadata	0..1	String	MD_Metadata.fileIdentifier Имя файла метаданных XML (пункт 10с-12). См. Часть 8 Должен присутствовать и заполняться, если файл метаданных ISO XML, описывающий этот набор данных, включен в набор обмена; в противном случае его включать не следует.
24	Вертикальная система координат	verticalCS	0..1	Integer	Код EPSG; Допустимые значения: <ul style="list-style-type: none"> • 6498 (Глубина –метры– ориентация вниз) • 6499 (Высота –метры– ориентация вверх)
25	База вертикальных координат	verticalCoordinateBase	0..1	Enumeration	См. таблицу 10с-24
26	Ссылка на вертикальный датум	verticalDatumReference	0..1	Enumeration	Только если verticalCoordinateBase = 2 См. таблицу 10с-25
27	Вертикальный датум	verticalDatum	0..1	Integer	Только если verticalCoordinateBase = 2 Если verticalDatumReference = 1, это значение из S100_VerticalAndSoundingDatum Если verticalDatumReference = 2, это код EPSG вертикального датума
28	Мета фичеры	metaFeatures	0..1	String	Имя по 8211 или файл GML, содержащий мета фичеры. Файлы GML должны иметь расширение .GML или .gml; файлы ISO 8211 должны иметь расширение .NNN, где N любая цифра.

ПРИМЕЧАНИЯ:

1) Если CRS определяется пользователем, поддерживаются только следующие системы координат:

Геодезическая CS (Широта, Долгота) – Градусы; и

Декартова CS (Север, Восток или Восток, Север) – метры.

2) Для горизонтального датума допускаются все предопределенные датумы EPSG или любое сочетание предопределенных Главных Меридианов или предопределенных Сфероидов.

3) Методы проектирования ограничиваются методами, приведенными в таблице 10с-24.

4) Если горизонтальные CRS определены кодом EPSG, то определенные CRS не должны использовать никакие другие элементы, кроме тех, которые разрешены для определенных пользователем CRS; (например, нет метода проекции, которого нет в таблице).

5) Границы района - это ячейка с границами; реализации фичеров покрытия могут покрывать, а могут и не покрывать всю ограничивающую ячейку. Если существует только один фичер покрытия, то его размер может быть таким же, как и размер ячейки.

6) Основные атрибуты соответствуют атрибутам метаданных S100_DatasetDiscoveryMetadata (часть 4а) или атрибутам данных изображения/сетки/покрытия в части 8. Соответствия приведены в столбце Примечания.

7) Вертикальный датум является необязательным, поскольку он не применим к некоторым типам ссылок на глубину, используемым в некоторых продуктах данных; например, поверхностные течения.

Спецификации продуктов, требующие дополнительных атрибутов метаданных, могут включать их в качестве дополнительных атрибутов, определенных в спецификации продукта. Дополнительные атрибуты должны быть определены так же, как и в таблице 10с-6 - в частности, они должны иметь имя, начинающееся с буквы в нижнем регистре, кратностью 0. 1 (факультативно) или 1 (в обязательном порядке) и быть одним из разрешенных типов, перечисленных в таблице 10с-1. Кроме того, могут быть добавлены ограничения или дополнительные условия для основных атрибутов метаданных носителей. Типы данных общих атрибутов метаданных носителя не могут быть изменены, однако диапазон допустимых значений может быть ограниченным, или необязательные атрибуты могут стать обязательными или условно обязательными.

ПРИМЕР: В таблице ниже показано, как спецификация продукта может определять дополнительный атрибут (вертикальная ссылка), вводить условную проверку основного атрибута метаданных (вертикальная ссылка на датум) и делать дополнительный атрибут метаданных обязательным (время выпуска продукта данных).

Таблица 10с-7 – Пример расширенных атрибутов метаданных и дополнительные условия по корневым атрибутам метаданных

No	Name	Camel Case	Mult	Data Type	
<i>Дополнительные метаданные носителя</i>					
11	вертикальная ссылка	depthTypeIndex	1	Enumeration	1: Средний уровень 2: Поверхность моря 3: Вертикальный датум (см. verticalDatum) 4: Дно моря
<i>Дополнительные ограничения или условия в отношении метаданных основного носителя</i>					
2	время выпуска продукта данных	issueTime	1	String (Time format)	Обязательный в S-111
9	вертикальная ссылка на датум	verticalDatum	0..1	Enumeration	Требуется, если и только если depthTypeIndex=3

Вопрос о том, как в спецификации продукта описываются основные и расширенные атрибуты метаданных, оставлен на усмотрение составителей спецификаций, однако спецификации должны отличать основные атрибуты от расширенных атрибутов, а также четко указывать любые дополнительные ограничения или условия по основным атрибутам. Может использоваться формат ISO для указания расширений метаданных (пункт 4а-5.6.5 части 4а).

10с-9.5 Группа информации о фичерах

Группа информации о фичерах содержит спецификации классов фичеров и их атрибуты. Компоненты группы информации о фичерах описаны в таблице ниже.

Таблица 10с-8 – Компоненты группы информации о фичерах

Группа	Категория HDF5	Имя	Тип данных или категория HDF	Пространство данных
/Group_F	Набор данных	featureCode	String (переменной длины)	Массив (1-d): i=0, F-1 Значения = коды классов фичеров (F это номер классов фичеров в прикладной схеме)
	Набор(ы) данных (наборы данных информации о фичерах – по одному на каждый фичер в массиве featureCode)	<featureCode> Например: SurfaceCurrent, WaterLevel	Array of Compound (String X 8)	Массив (1-d): i=0, NAF-1 (NAF = число атрибутов фичеров, указанных в <featureCode>). Компоненты составного типа: код: код атрибута из каталога фичеров имя: длинное имя из каталога фичеров uom.name: единицы измерения (uom>name из каталога фичеров S-100) fillValue: значение заливки (значение integer или float, строчное представление) datatype: тип данных HDF5 data type, от функции H5Tget_class() lower: нижняя граница значения атрибута upper: верхняя граница значения атрибута closure: тип замыкания Компоненты “code” и “datatype” кодируют атрибут rangeType фичера покрытия в Части 8 “lower”, “upper” и “closure” кодируют ограничения значений атрибутов как указано в каталоге фичеров (см. “S100_FC_SimpleAttribute>constraints” в Части 5 и S100_NumericRange в Части 1)

Примечание:

1) Маскированные или неизвестные значения суши представлены в атрибуте значением *fillValue*.

Все числовые значения в наборе данных описания фичеров являются строковыми представлениями числовых значений; например, "-9999.0" не является десятичным значением -9999.0. Ожидается, что приложения будут анализировать строки для получения числового значения. Неприменимые записи представлены нулевыми значениями или пустой строкой (длиной 0).

Запись в Group_F требуется для каждого типа фичера, которая используется в файле данных HDF5. Это означает, что:

- Массив **featureCode** должен включать каждый тип фичера, для которого существует реализация фичера где-то в текущем физическом файле.
- Для каждого типа фичеров в массиве featureCode должен быть набор данных описания фичеров.
- Каждый набор данных описания фичеров должен содержать список всех атрибутов типа фичера (как прямых, так и унаследованных), как указано в Каталоге фичеров.

Обратите внимание, что вышеприведенные требования не требуют внесения записей в Group_F для типов фичеров, которые определены в XML каталоге фичеров, но для которых нет реализаций в текущем файле данных.

Число атрибутов для каждого типа фичеров (NAF в таблице 10с-8) не указано явно, но может быть определено с помощью HDF5 API для определения числа строк в каждом наборе данных описания фичеров.

Рисунок ниже изображает Group_F для гипотетического продукта с двумя типами фичеров, SurfaceCurrent и WaterLevel. Эти два фичера названы (используя коды из каталога фичеров) в наборе данных **featureCode**. Наборы данных описания фичеров **SurfaceCurrent** и **WaterLevel** описывают атрибуты каждого типа фичеров. Наборам данных описания фичеров даются те же имена, что и значениям в наборе данных **featureCode**, которые являются кодами фичеров из XML каталога фичеров. Каждый набор данных описания фичеров представляет собой массив составных элементов типа, компонентами которых являются 8 компонентов, указанных в таблице 10с-8.

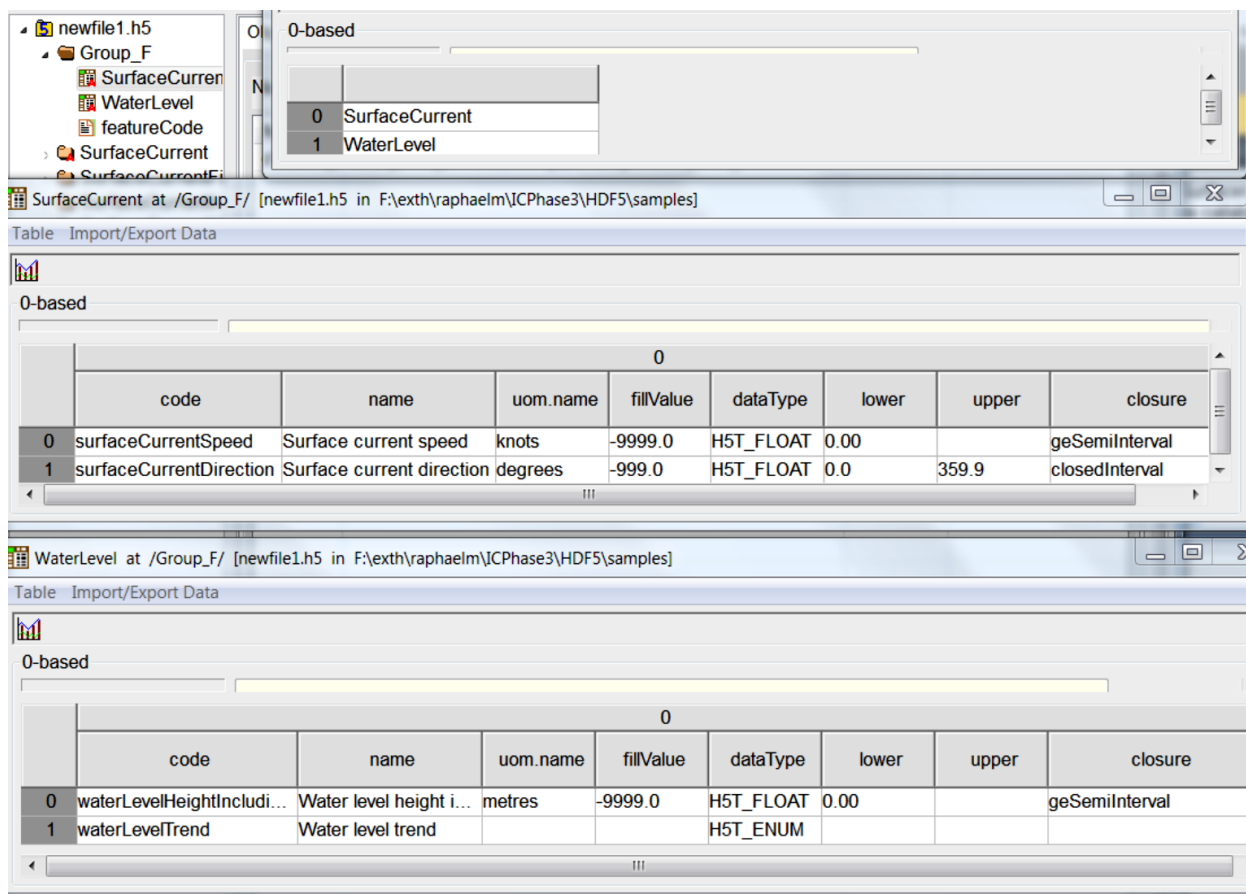


Рисунок 10с-8 – Пример Group_F

10с-9.6 Группа контейнера фичеров

Группы контейнеров фичеров содержат координаты и значения для всех реализаций одного класса фичеров. Каждой реализации фичера выделяется собственная группа в группе контейнеров фичеров. Эта организация позволяет привязывать общеклассовые атрибуты к классу в целом и специфические атрибуты к соответствующей реализации фичера.

ПРИМЕЧАНИЕ: Решение о создании отдельной группы для каждой реализации фичера основано на том факте, что в некоторых обстоятельствах будет несколько наборов данных для одной реализации (например, индекс, TIN и т.п.), и помещение всех наборов данных непосредственно в группу контейнеров, вероятно, внесет путаницу в организацию данных, по крайней мере с человеческой точки зрения (хотя суффиксов может быть достаточно для того, чтобы различать различные случаи для целей программирования).

Структура группы контейнеров фичеров показана в таблице 10с-9 ниже. В этой таблице также показана группа(ы) реализаций фичеров. Имена осей задаются в наборе данных на уровне контейнера фичеров. Метаданные, общие для всех реализаций класса фичеров (например, размерность), кодируются на уровне контейнера фичеров, и эти элементы метаданных перечислены в таблице 10с-10. Метаданные, специфичные для реализаций фичеров (например, параметры сетки), кодируются на уровне реализаций, и эти элементы перечислены в таблице 10с-12.

Спецификации на производство могут добавлять атрибуты метаданных по конкретным продуктам. Руководящие принципы в отношении дополнительных элементов метаданных являются такими же, как и в отношении дополнительных элементов метаданных в корневой группе (пункт 10с-9.4).

Таблица 10с-9 – Структура групп контейнеров фичеров

Группа	Категория HDF5	Имя	Тип данных	Примечания / Пространство данных
//feature code)	attribute	См. таблицу 10с-10	(См. таблицу)	Однозначные атрибуты, описанные в таблице 10с-10
	Набор данных	axisNames	String	Массив (1-D): 0..D-1, где D - значение атрибута <i>dimension</i> Оси должны быть в порядке большая-малая; то есть, если хранилище будет в порядке row-major, то ось X/долгота должна быть первой.
	Набор данных (опция)	coordinateSize	Integer	Массив (1-D): 0..D-1, где D - значение атрибута <i>dimension</i> Размер координатной кодировки в байтах. Допустимые значения - 1, 2, 4 или 8. Если этот набор данных отсутствует, координаты должны быть закодированы с использованием 64 бит (8 байт) для координат с плавающей точкой и 32 бита (4 байта) для целочисленных координат
	Набор данных (опция)	interpolationParameters	Float	Массив (1-D) параметров интерполяции Требуется тогда и только тогда, когда значение атрибута <i>interpolationType</i> является 'biquadratic' или 'bicubic'
	Набор данных (опция)	eatureAttributeTable	Compound	Массив (1-D) индексируемых атрибутов фичеров (раздел 10с-9.6.2). Требуется тогда и только тогда, когда значение атрибута <i>dataCodingFormat</i> равно 9
	Группа	//(feature code).N		

ПРИМЕЧАНИЯ:

- 1) "uncertainty" это погрешность значений данных, погрешность местоположения (как горизонтального, так и вертикального) кодируется отдельно.
- 2) Длина набора данных *interpolationParameters* и последовательность параметров должна указываться в Спецификации на производство продукта.

Таблица 10с-10 – Атрибуты групп контейнеров фичеров

№	Имя	Обозначение	Множ	Тип данных	Примечание и/или единицы измерения
	Индекс организации данных	dataCodingFormat	1	Enumeration	Указание типа покрытия в реализации этого фичера Используется для чтения данных (см. таблицу 10с-4) 1: Временные ряды на постоянных станциях 2: Массивы с регулярной сеткой 3: Массивы не геопространственной сетки 4: Движущаяся платформа 5. Нерегулярная сетка 6. Переменный размер ячеек 7. TIN 8: Временные серии на фиксированной станции 9: Регулярная сетка на базе фичеров
	Размерность	dimension	1	Integer	Размер реализаций фичеров Это число осей координат, а не ранг массивов HDF5, содержащих координаты или значения. Например, набор данных о стационарных станциях с позициями по широте и долготе будет иметь размерность=2
	Правило общей точки	commonPointRule	1	Enumeration	Процедура, используемая для оценки покрытия в месте, которое находится на границе или в зоне перекрыша между геометрическими объектами. Значения из CV_CommonPointRule (Таблица 10с-20)
	Погрешность горизонтального местоположения	horizontalPositionUncertainty	1	Float	Погрешность в горизонтальных координатах. Например, - 1,0 (неизвестно/неприменимо) или положительное значение (м)
	Погрешность вертикального местоположения	verticalUncertainty	1	Float	Погрешность в вертикальных координатах. Например, -1.0 (неизвестно/неприменимо) или положительное значение (м)
	Погрешность времени	timeUncertainty	0..1	Float	Погрешность значения времени. Например, -1.0 (неизвестно/неприменимо) или положительное значение (с) Только для данных временных рядов
	Количество реализаций фичеров	numInstances	1	Integer	Количество реализаций фичеров (Записи в одном временном ряду или в последовательности движущейся платформы учитываются как одна реализация, а не как отдельные реализации)

	(дополнительные общие атрибуты)				(Как указано в спецификации продукта)
dataCodingFormat = 1					
	(нет)				
dataCodingFormat = 2					
	Правило последовательности	sequencingRule.type	1	Enumeration	Метод, используемый для присвоения значений координатам сетки из последовательности значений. Тип и направление сканирования кодируются как отдельные атрибуты type: Enumeration CV_SequenceType (таблица 10c-21) scanDirection: Строка <axisNames entry> (разделенная запятыми). Например, "latitude, longitude". Направление обратного сканирования вдоль оси указано префиксом '-' у имени оси
		sequencingRule.scanDirection	1	String	
	Тип интерполяции	interpolationType	1	Enumeration	Метод интерполяции, рекомендуемый для оценки S100_GridCoverage Values: S100_CV_InterpolationMethod (Таблица 10c-22)
	Смещение точки данных в ячейке	dataOffsetCode	0..1	Enumeration	См. раздел 10c-9.6.1 1: XMin, YMin ("Lower left") угол является ("Cell origin") 2: XMax, YMax ("Upper right") угол 3: XMax, YMin ("Lower right") угол 4: XMin, YMax ("Upper left") угол 5: Barycenter (центроид) ячейки
	Смещение точки данных в ячейке как вектор	dataOffsetVector	0..1	Float	Массив (1-D) 0..D-1 where D is the value of the dimension attribute Values must be real numbers in the range [0,1] See clause 10c-9.6.1
dataCodingFormat = 3					
	Правило интерполяции	interpolationType	1	Enumeration	Метод интерполяции, рекомендуемый для оценки S100_GridCoverage Values: S100_CV_InterpolationMethod (Таблица 10c-22)
dataCodingFormat = 4					
	(нет)				
dataCodingFormat = 5					
	Правило последовательности	sequencingRule.type	1	Enumeration	Метод, используемый для присвоения координатам сетки значений из последовательности значений.
		sequencingRule.scanDirection	1	String	

					Тип и направление сканирования кодируются как отдельные атрибуты. type: Перечень CV_SequenceType (Таблица 10с-21). scanDirection: Строка <axisNames entry> (разделенная запятой). Например, "latitude, longitude". Направление обратного сканирования вдоль оси указано префиксом '-' у имени оси.
	Тип интерполяции	interpolationType	1	Enumeration	Метод интерполяции, рекомендуемый для оценки S100_GridCoverage Values: S100_CV_InterpolationMethod (Таблица 10с-21)
	Смещение точки данных в ячейке	dataOffsetCode	0..1	Enumeration	См. раздел 10с-9.6.1 1: XMin, YMin ("Lower left") угол является ("Cell origin") 2: XMax, YMax ("Upper right") угол 3: XMax, YMin ("Lower right") угол 4: XMin, YMax ("Upper left") угол 5: Barycenter (центроид) ячейки
	Смещение точки данных в ячейке как вектор	dataOffsetVector	0..1	Float	Array (1-D) 0..D-1 where D is the value of the dimension attribute Values must be real numbers in the range [0,1] See clause 10с-9.6.1
dataCodingFormat = 6					
	Правило последовательности	sequencingRule.type	1	Enumeration	Метод, используемый для присвоения значений координатам сетки из последовательности значений. Тип и направление сканирования кодируются как отдельные атрибуты. type: Enumeration CV_SequenceType (Таблица 10с-21). scanDirection: Строка <axisNames entry> (разделенные запятой). Например, "latitude, longitude". Направление обратного сканирования вдоль оси указано префиксом '-' у имени оси.
		sequencingRule.scanDirection	1	String	
	Тип интерполяции	interpolationType	1	Enumeration	Метод интерполяции, рекомендуемый для оценки S100_GridCoverage Values: S100_CV_InterpolationMethod (Таблица 10с-21)
	Смещение точки данных в ячейке	dataOffsetCode	0..1	Enumeration	См. раздел 10с-9.6.1 1: XMin, YMin ("Lower left") угол является ("Cell origin") 2: XMax, YMax ("Upper right") угол 3: XMax, YMin ("Lower right") угол 4: XMin, YMax ("Upper left") угол 5: Barycenter (центроид) ячейки

	Смещение точки данных в ячейке как вектор	dataOffsetVector	0..1	Float	Array (1-D) 0..D-1 where D is the value of the dimension attribute Values must be real numbers in the range [0,1] See clause 10c-9.6.1
dataCodingFormat = 7					
	Тип интерполяции	interpolationType	1	Enumeration	Метод интерполяции, рекомендуемый для оценки S100_GridCoverage Values: S100_CV_InterpolationMethod (Таблица 10c-22)
dataCodingFormat = 8					
	(нет)				
dataCodingFormat = 9					
	Смещение точки данных в ячейке	dataOffsetCode	0..1	Enumeration	См. раздел 10c-9.6.1 1: XMin, YMin ("Lower left") угол является ("Cell origin") 2: XMax, YMax ("Upper right") угол 3: XMax, YMin ("Lower right") угол 4: XMin, YMax ("Upper left") угол 5: Barycenter (центроид) ячейки
	Смещение точки данных в ячейке как вектор	dataOffsetVector	0..1	Float	Array (1-D) 0..D-1 where D is the value of the dimension attribute Values must be real numbers in the range [0,1] See clause 10c-9.6.1
	Правило последовательности	sequencingRule.type	1	Enumeration	Метод, используемый для присвоения значений координатам сетки из последовательности значений. Тип и направление сканирования кодируются как отдельные атрибуты. type: Enumeration CV_SequenceType (Таблица 10c-21). scanDirection: Строка <axisNames entry> (разделенные запятой). Например, "latitude, longitude". Направление обратного сканирования вдоль оси указано префиксом '-' у имени оси.
		sequencingRule.scanDirection	1	String	
(любое другое значение dataCodingFormat)					
	(дополнительные атрибуты)				(Как указано в спецификации продукта)

10с-9.6.1 Положение точки данных в ячейке

Спецификации продукта могут требовать, чтобы их продукты данных указывали относительное местоположение точки данных в ячейке сетки по отношению к углам ячейки. Положение может указываться с помощью или атрибута *dataOffsetCode* или *dataOffsetVector*. Эти атрибуты могут использоваться только в сеточных покрытиях, а не с временными рядами, TIN или данными движущейся платформы. Спецификации продукта могут использовать или *dataOffsetCode*, или *dataOffsetVector*, но не оба.

Спецификации продукта, в которых точка данных находится в точке сетки (XMin, YMin) не должны использовать данные *dataOffsetCode* или *dataOffsetVector*.

Атрибут *dataOffsetCode* может использоваться только с двумерными сетками. Он указывает, является ли точка данных одним из четырех углов клетки или центром ячейки.

Обратите внимание, что определения кодов, обозначающих углы, приведены в координатах сетки X и Y относительно начала отсчета сетки. (Это означает, что в сетке с осью X, направленной с востока на запад, и осью Y с севера на юг, "нижний левый" угол отличается от "нижнего левого" угла в сетке с осью X, направленной с запада на восток и с юга на север.)

Атрибут *dataOffsetVector* предназначен для использования с сетками более высокой размерности или в случаях, когда расположение точки данных находится не в одном из углов или в центре ячейки. Значения в этом массиве показывают относительное смещение по каждой оси точки данных от точки сетки, координаты сетки которой наиболее близки к координатам начала отсчета сетки. В двумерной сетке это будет точка с наименьшими координатами X и Y. Опять же следует отметить, что направление осей и расположение начала отсчета сетки определяют, какой угол является началом отсчета ячейки. Каждое смещение относительно размерности ячейки вдоль соответствующей оси. Порядок значений в *dataOffsetVector* должен соответствовать порядку осей в массиве *axisNames* (таблица 10с-9).

Для покрытий TIN точки данных расположены в точках вершин TIN. Смотрите S-100 часть 8, раздел 8-7.4.

10с-9.6.2 Таблица атрибутов фичеров

Этот элемент является опциональной записью с именем "featureAttributeTable". Элемент используется только в группе контейнеров фичеров, если dataCodingFormat = 9 (Регулярная сетка на базе фичеров). Во всех других случаях элемент не должен использоваться. Элемент "featureAttributeTable" может использоваться только для записей "values" в "data values group(s)".

Элемент "featureAttributeTable" соответствует типу данных HDF5 для CompoundDataset в случаях 1D массива. Он всегда состоит из фиксированного столбца с именем "id" и какого-то количества дополнительных колонок.

Колонка id

Колонка "id" всегда содержит значения один к одному беззнакового целого типа данных. Значения всегда больше нуля (>0), так как значение zero (0) указывается как значение no data (нет данных). Значения не должны соответствовать определенному порядку. Они могут быть выбраны случайным образом. Это позволяет использовать идентификационное значение внешнего источника данных; например, базы данных. В колонке "id" всегда содержится значение "values-record" из "data values group(s)" из "feature instance group(s)". Значение служит связующим звеном между растровой геометрией объекта и его описательными данными в "featureAttributeTable". Значения 1 к 1 столбца "id" могут быть в пикселях растрового изображения, но всегда должны описывать точно одну геометрию фичеров. Эффект избыточности растрового изображения может быть значительно уменьшен путем сжатия в наборе данных HDF5.

Дополнительные столбцы

Количество дополнительных колонок зависит от информации, подлежащей транспортировке. Число всегда должно быть больше нуля (>0). Таблица 10с-8 определяет, что имя "feature container group" всегда должно соответствовать имени camelCase объекта типа "Feature Type" Реестра геопространственной информации ИНО, Регистра словаря данных. Имена других столбцов основаны на атрибутах выбранного типа фичеров Регистра словаря данных. В соответствии с формой отображения атрибутов в формате HDF5 могут использоваться только простые типы данных (см пункт 10с-7). Для отображения сложных типов данных необходима соответствующая запись с помощью точки. Соответствующие уровни комплексного типа данных разделяются точкой до простого типа данных; например, "surveyDateRange.dateEnd" & "surveyDateRange.dateStart".

10с-9.7 Группа реализации фичера

Группы реализаций фичеров содержатся в группах контейнеров фичеров. Структура группы реализации фичеров определена в таблице 10с-11. Атрибуты, специфичные для каждой реализации фичера, определены в следующей таблице (Таблица 10с-12) и состоят из информации, которая может изменяться для различных реализаций в одном наборе данных, таких как размер, местоположение, время и размер сетки.

Таблица 10с-11 – Структура групп реализаций фичеров

Группа	Категория HDF5	Имя	Тип данных	Примечания / Пространство данных
//(feature code).N Например: SurfaceCurrent.01	атрибуты	См. таблицу 10с-12	(См. таблицу)	Однозначные атрибуты, описанные в таблице 10с-12
	Набор данных (опция)	domainExtent.polygon	Compound (Float, Float)	Пространственная протяженность домена покрытия Массив (1-d): i=0, P Компоненты: <longitude, latitude> или <X, Y> (координаты ограничивающих вершин многоугольника как замкнутого кольца, то есть первый и последний элементы будут содержать те же самые значения) Необходимо заполнить этот или ограничивающий атрибут. Для нерегулярных массивов в этом наборе данных должен быть указан многоугольник, покрывающий область, для которой предоставляются данные
	Набор данных (опция)	domainExtent.verticalElement	Compound (Integer X 2, Float X 2)	Массив (1-d) составных элементов каждый из них обеспечивает расположение сетки и максимальная, минимальная вертикальная протяженность в месте расположения Компоненты составного типа следующие: gridX, gridY: Integer (количество точек осей сетки X/longitude и Y/latitude) minimumValue, maximumValue (Float): минимальные и максимальные значения Z в точках сетки, указанных в gridX и gridY Применимо только для 3-D сеток. Или этот набор данных или атрибут verticalExtent (Таблица 10с-12) должен быть заполнен для 3-D сеток
Набор данных (опция)	extent		Compound (Integer X D)	1-D массив, составных элементов, 2 строки. Строка 0 дает "low" значения, строка 1 "high" значения. Область сетки, по которой представляются данные. (Часть 8 Рис. 8-23)

			Компоненты составного типа именуются по названиям осей в наборе данных axisNames
Набор данных (опция)	uncertainty	Compound (String, Float)	Массив (1-d): i = 0, (до) NA _F Код и погрешность значений данных Например, ("surfaceCurrentSpeed", 0.1) Количество атрибутов для этого класса фичеров (NA _F) может быть определено из Group_F
Набор данных (опция)	cellGeometry	Compound (String, Float X 2, Integer X 1)	Геометрия ячейки. Массив (1-d) такой же длины, как и массив axisNames, определенный выше (это означает, что если данный набор данных существует, то он кодирует все оси, включая широту, долготу и т.д.) Условные, требуемые только для регулярных сетей (dataEncodingFormat=2), использующие системы координат с осями, отличными от (широта, долгота, вертикальная) или более чем 3 измерений Этот массив служит для расширения информации, закодированной в атрибутах параметров сетки (происхождение, шаг, число точек), определенных в таблице 10с-12 (атрибуты группы реализаций фичеров) для продуктов данных, которые используют сетки более высокой размерности или нестандартные координатные оси Компоненты: axisName: string (запись в массиве axisNames, определенном выше) gridOrigin: Float (начало оси, названной в компоненте axisName) gridSpacing: Float (Расстояние между ячейками для именованной оси) numPoints: Integer (число линий сетки по указанной оси)
Группа (опция)	/Group_TL		Информация о тайлах. Условный, требуется, если спецификация продукта указывает тайлинг.
Группа (опция)	/Group_IDX		Метод пространственной индексации. Условный, требуется, если спецификация продукта указывает пространственную индексацию.
Группа (опция)	/Positioning		Информация о местоположении. Координаты значений данных. Условный, требуется, если dataCodingFormat не 2 (Регулярная сетка)
Группа	/Group_nnn		Группа(ы) значений данных.

Таблица 10с-12 – Атрибуты групп реализаций фичеров

№	Имя	Обозначение	Множ	Тип данных	Примечания и/или единицы измерения
	Ограниченная область	westBoundLongitude	0..1	Float	Географическая протяженность сетки, как ограниченной области См. domainExtent: EX_GeographicExtent > EX_GeographicBoundingBox Либо эта, либо набор данных domainExtent должны быть заполнены. Все границы должны быть заполнены или пропущены. Единицы измерения должны соответствовать CRS, используемой для набора данных (например, градусы для географической 2D CRS EPSG 4326; и метры для зон UTM в проекции CRS EPSG 32710)
		eastboundLongitude	0..1	Float	
		southBoundLatitude	0..1	Float	
		northBoundLatitude	0..1	Float	
	Количество записей времени	numberOfTimes	0..1	Integer	Общее количество записей времени Только для данных временных серий. Для dataCodingFormat = 8, эта переменная может быть переписана соответствующей переменной из значений атрибутов группы (Таблица 10с-19)
	Интервал времени	timeRecordInterval	0..1	Integer	Интервал между записями времени. Единицы измерения: секунды Только для временных серий. Для dataCodingFormat = 8, эта переменная может быть заменена соответствующей переменной в атрибутах группы значений (таблица 10с-19)
	Значение самого раннего времени	dateTimeOfFirstRecord	0..1	Character	Валидность самой ранней записи времени. Единицы измерения: DateTime. Только для данных временных серий
	Значение самого позднего времени	dateTimeOfLastRecord	0..1	Character	Валидность самой поздней записи времени. Единицы измерения: DateTime. Только для данных временных серий
	Вертикальная протяженность	verticalExtent.minimumZ	0..1	Float	Вертикальная протяженность 3-D сеток minimumZ, maximumZ: Минимальные и максимальные значения пространственной протяженности сетки в вертикальном направлении. Они кодируются как отдельные атрибуты
		verticalExtent.maximumZ	0..1	Float	
	Количество групп	numGRP	1	Integer	Количество групп значений данных, содержащихся в этой группе реализаций.
	(дополнительные атрибуты, относящиеся к продукту данных)	(как определено в Спецификации продукта)			
dataCodingFormat = 1					
	Число фиксированных станций	numberOfStations	1	Integer	Число фиксированных станций
dataCodingFormat = 2 или 9					

	Долгота начального отсчета сетки	gridOriginLongitude	1	Float	Долгота начального отсчета сетки. Unit: градусы дуги
	Широта начального отсчета сетки	gridOriginLatitude	1	Float	Широта начального отсчета сетки. Градусы дуги
	Вертикальное начало отсчета сетки	gridOriginVertical	0..1	Float	Начало отсчета сетки в вертикальном измерении. Только для 3-D сеток. Единицы измерения, указаны в спецификации продукта
	Шаг сетки, долг.	gridSpacingLongitudinal	1	Float	Размер ячейки по X/долготе. Это X/долготный компонент смещения вектора (S-100 часть 8, пункт 8-7.5). Единицы: Градусы дуги
	Шаг сетки, шир..	gridSpacingLatitudinal	1	Float	Размер ячейки по У/широте. Это У/широтный компонент вектора смещения (S-100 часть 8, пункт 8-7.5). Единицы: Градусы дуги
	Шаг сетки, Z	gridSpacingVertical	0..1	Float	Размер ячейки по вертикали. Только для 3D-сеток. Единицы указаны в спецификации продукта.
	Кол-во точек, долг.	numPointsLongitudinal	1	Integer	Количество точек сетки по X/долготе. (iMax)
	Кол-во точек, шир.	numPointsLatitudinal	1	Integer	Количество точек сетки по У/широте. (jMax)
	Кол-во точек, вертикально	numPointsVertical	0..1	Integer	Количество точек сетки по-вертикали. (kMax)
	Начало отсчета	startSequence	1	String	Координаты точки сетки, которой присваивается первое в последовательности значение. Выбор точки для начала отсчета определяется порядком обхода. Формат: n, n... (список разделенных запятой точек сетки, одна на размерение – например, 0,0)
dataCodingFormat = 3					
	Узлы сетки	numberOfNodes	1	Integer	Общее количество точек сетки
dataCodingFormat = 4					
	Количество станций	numberOfStations	1	Integer	Значение всегда 1
dataCodingFormat = 5 or 6					
	Долгота начала отсчета сетки	gridOriginLongitude	1	Float	Долгота начала отсчета сетки. Единицы: Градусы дуги
	Широта начала отсчета сетки	gridOriginLatitude	1	Float	Широта начала отсчета сетки. Единицы: Градусы дуги
	Начало отсчета сетки по вертикали	gridOriginVertical	0..1	Float	Начало отсчета по-вертикали. Только для 3-D сеток. Единицы указаны в спецификации продукта
	Шаг сетки, долг.	gridSpacingLongitudinal	1	Float	Размер ячейки по X/долготе. Это X/долготный компонент вектора сдвига (S-100 часть 8, пункт 8-7.5). Единицы: Градусы дуги Для сетки с переменным размером ячеек это размер самой маленькой ячейки в этом измерении
	Шаг сетки, шир.	gridSpacingLatitudinal	1	Float	Размер ячейки по У/широте. Это У/широтный компонент вектора сдвига (S-100 часть 8, пункт 8-7.5). Единицы: Градусы дуги Для сетки переменного размера ячейки это размер единичной ячейки

	Шаг сетки, Z	gridSpacingVertical	0..1	Float	Размер ячейки по вертикали. Только для 3-D сеток. Единицы указаны в спецификации продукта Для сетки переменного размера ячейки это размер единичной ячейки
	Узлы сетки	numberOfNodes	1	Integer	Общее количество точек сетки
	Начало отсчета	startSequence	1	String	Сеточные координаты точки сетки, которые присваиваются первой в последовательности значений. Выбор точки для начала отсчета определяется порядком обхода. Формат: n, n... (список разделенных запятыми точек сетки, одна на размерение, например, 0,0)
dataCodingFormat = 7					
	Узлов в сетке	numberOfNodes	1	Integer	Общее количество точек сетки
	Треугольников в сетке	numberOfTriangles	1	Integer	Общее количество треугольников в TIN
dataCodingFormat = 8					
	Число фиксированных станций	numberOfStations	1	Integer	Число фиксированных станций
(другое значение dataCodingFormat)					

ПРИМЕЧАНИЯ:

- 1) Специфичные по типу атрибуты для обычных и переменных ячеек сетки одинаковы, за исключением того, что параметры, дающие число точек в каждом измерении, заменяются общим числом узлов в сетке.
- 2) Атрибуты " Значение самого раннего времени " и " Значение последнего значения времени" обеспечивают компонент времени атрибута domainExtent в модели сетки (S-100 часть 8, рисунки 8-22, 8-23 и 8-24).
- 3) Спецификации продукта могут требовать использования одного или другого из атрибутов domainExtent или boundingBox, в зависимости от того, насколько точно известно, что пространственные пределы реализаций фичеров прямоугольны в системе координат или определенно имеют неправильную форму.

10с-9.7.1 Переопределяющие атрибуты

Группа реализаций фичеров может также нести любой из следующих атрибутов, определенных в группах более высокого уровня. Значение атрибута, присвоенное в группе реализаций фичеров, переопределяет значение в верхней группе.

- Атрибут “Vertical datum reference” (verticalDatum) из корневой группы;
- Любой атрибут из группы контейнеров фичеров, **кроме** “Number of feature instances” (numInstances).

Спецификации продукта могут запретить переопределение атрибута, если это не требуется для их продуктов.

ПРИМЕЧАНИЯ:

1) Переопределение атрибутов позволяет определенным продуктам кодировать вариации типов фичеров в одном и том же файле данных, например, если прикладная схема определяет функцию, которая может иметь либо обычную сетку, либо информацию о фиксированной станции, и поэтому может потребовать различные атрибуты метаданных. Авторы спецификации продуктов должны, однако, иметь в виду, что этот вопрос может быть решен в прикладных схемах путем определения соответствующей специализации класса фичеров, которые были бы различными типами фичеров и, следовательно, кодироваться в различные контейнеры фичеров.

2) Переопределение атрибутов также позволяет устанавливать различия между производительностью и временем, такие как разные вертикальные даты для разных фичеров. Хотя на практике по возможности этого следует избегать, с тем чтобы уменьшить вероятность человеческой ошибки при разработке приложений, а также со стороны конечного пользователя.

10с-9.7.2 Пример структуры контейнера и реализации

На рисунке ниже показана структура гипотетического файла данных, содержащего 3 реализации **SurfaceCurrent**.

- Вертикальная панель слева показывает общую структуру. Продукт данных состоит из 2 фичеров (**SurfaceCurrent** и **WaterLevel**). Каждый из них представлен группой, расположенной чуть ниже корневой группы. Также показана группа Feature Information, описанная ранее (пункт 10с-9.5).
- Группа Feature Container названная как **SurfaceCurrent** содержит 3 реализации типа фичеров **SurfaceCurrent** (гипотетически, данные для 3 отдельных мест, каждое с локальной сеткой покрытия). Каждая реализация содержит подгруппы (Group_001 и т.д.) для данных временных рядов.
- Местоположения закодированы в наборе данных **geometryValues** в группе **Positioning** (панель вверху справа). Слева в панели **axisNames** указываются имена компонентов **geometryValues** (то есть, оси координат).
- Панель **SurfaceCurrent** посередине показывает атрибуты метаданных, общих для всех реализаций, которые прикреплены к группе контейнеров фичеров **SurfaceCurrent**.

- Две панели внизу показывают метаданные по отдельным реализациям для реализаций фичеров **SurfaceCurrent.01** и **SurfaceCurrent.02**.

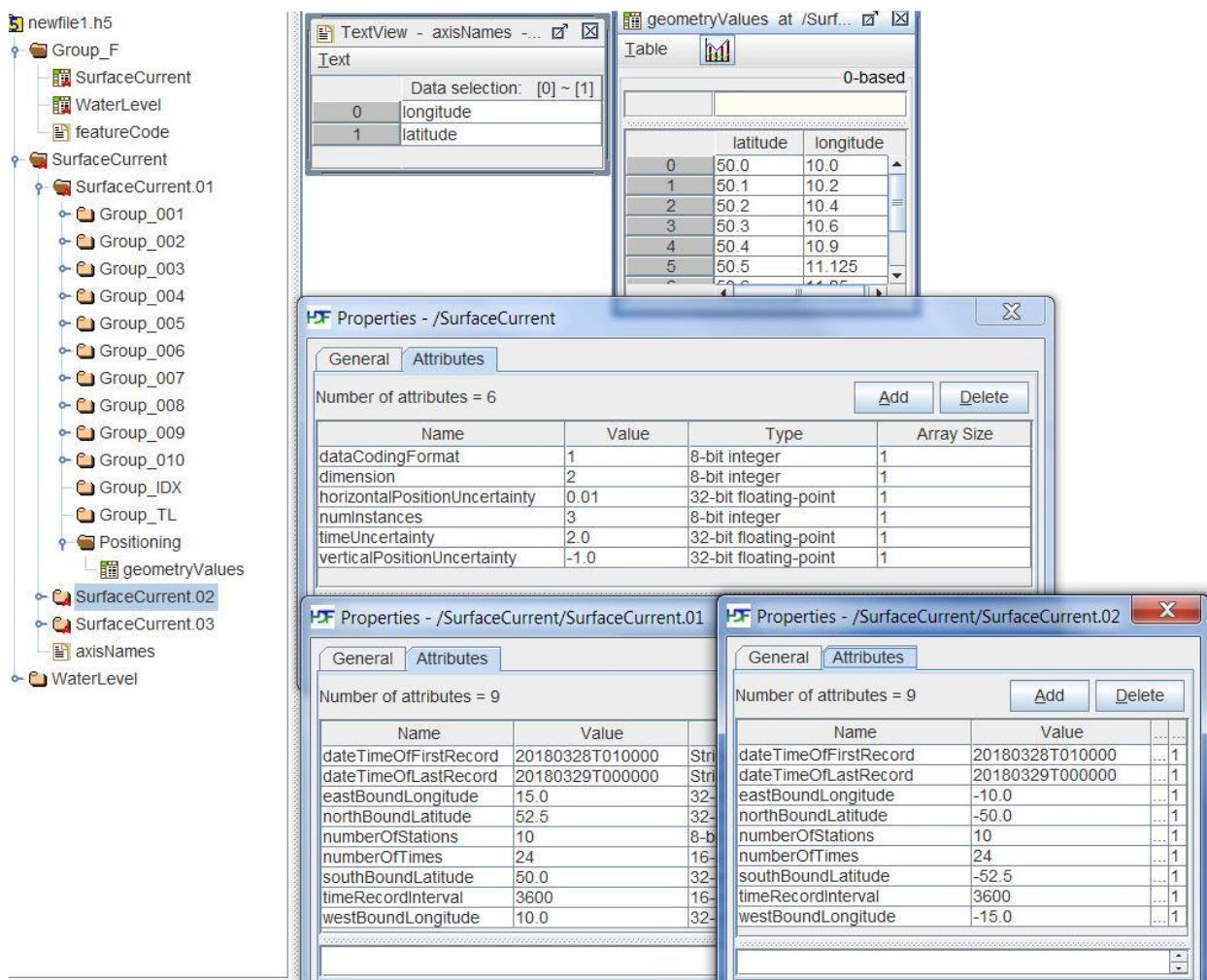


Рисунок 10с-9 – Иллюстративный пример структуры набора данных.

10с-9.8 Группа информации о тайлинге

Эта группа кодирует информацию о схеме тайлинга, используемой в наборе данных (S-100). Она присутствует тогда и только тогда, когда данные закодированы более чем в одном тайле. Некоторые схемы тайлинга описаны в части 8 (пункт 8-6). Это издание профиля HDF5 поддерживает только два тайлинга: простая сетка и простая сетка переменной плотности. В обоих случаях размеры тайлов указаны в терминах ограничивающих их областей (Таблица 10с-12).

Пространственное объединение поверхностей тайлов должно покрывать все фичеры в наборе данных (S-100), но обратное не является обязательным. (Неофициально это означает, что могут существовать части тайлов, которые не покрываются геометрией какого-либо фичера в наборе данных, но не наоборот - не

может быть частей геометрии фичеров, которые не покрыты по крайней мере одним тайлом.)

Обратите внимание, что тайлинг не совсем то же самое понятие, как "фрагментация", так как последняя определена в HDF5 и NetCDF - тайлах на основе координат географических участков, в то время как фрагменты HDF5 определяют нарезку данных для оптимизации производительности хранения и поиска.

Таблица 10с-13 – Группа информации о тайлинге

Группа	Категория HDF5	Имя	Тип данных или Категория HDF	Примечания / Пространство данных
/Group_TL	Атрибут	numTiles	Integer	Количество тайлов значение > 0
	Атрибут	tilingScheme	Enumeration	1: Простая сетка 2: Простая сетка переменной плотности (Спецификация продукта должна выбрать одно)
	Набор данных	tiles	Составной массив (Float X 4, Integer)	Ограниченная область тайлов. Компоненты: westBoundLongitude: Float eastboundLongitude: Float southBoundLatitude: Float northBoundLatitude: Float tileID: Integer (идентификатор тайла)

В этом издании S-100 детали методов тайлинга оставлены на усмотрение спецификаций продукта. Этот профиль не определяет порядок расположения тайлов, а также не контролирует использование или неиспользование иерархических схем тайлинга. Часть 8 (раздел 8-6) требует, чтобы любая используемая схема тайлинга была полностью описана как часть спецификации продукта для конкретного продукта данных. Это включает в себя размеры, расположение и плотность данных тайлов, а также механизм идентификации тайлов (tileID).

10с-9.9 Группа индексов

Группа индексов кодирует информацию о пространственной индексации, если она используется в спецификации продукта. Эта группа кодируется тогда и только тогда, когда спецификация продукта предписывает метод пространственной индексации и требует четкого кодирования пространственного индекса.

Таблица 10с-14 – Группа индексов

Группа	Категория HDF5	Имя	Тип данных или Категория HDF	Примечания / Пространство данных
/Group_IDX	Атрибут	indexingMethod	Enumeration	Метод пространственной индексации. (Описано в спецификациях продукта)
	Набор(ы) данных	spatialIndex	(Зависит от метода индексации)	Данные, кодирующие пространственный индекс. (Описано в спецификациях продукта)

Детали методов индексации и структура наборов данных индекса оставлены на усмотрение спецификаций продуктов в этом издании S-100.

10с-9.10 Группа позиционирования

В зависимости от формата кодирования данных может существовать группа позиционирования, Positioning. Эта группа не содержит атрибутов, она содержит набор данных координат, который представляет собой массив составного типа с компонентами, именуемыми так же, как набор данных axisNames в группе Feature Container. Эта группа используется для значений *dataCodingFormat* 1, 3, 4, 7 и 8 (пункт 10с-9.3). Она не используется для *dataCodingFormat* = 2 (регулярные сетки), 5 (нерегулярная сетка), 6 (сетка с переменным размером ячейки) или 9 (Регулярная сетка на базе фичеров).

Порядок обхода для сеток различных типов задаётся последовательностью атрибута метаданных *sequencingRule* в группе контейнеров. Порядок обхода не используется для фиксированной станции, движущейся платформы или данных TIN (*dataCodingFormat* = 1, 4, 7 или 8).

Размерность D данных определяется атрибутом метаданных размерности в группе контейнеров фичеров.

10с-9.10.1 Стратегия пространственного представления

В случае данных в регулярной сетке (*dataCodingFormat* = 2 или 9) число точек сетки в каждом измерении, интервалы между точками и начало отсчета сетки кодируются в атрибутах метаданных. (Например, для 2-D сетки, атрибуты метаданных *numPointsLongitudinal* и *numPointsLatitudinal* кодируют точки по долготе и широте.) Учитывая эти параметры и индексы точек в сетке, положение точки можно вычислить с помощью простых формул.

Для данных временных рядов по фиксированным станциям, негеопространственным данным с координатной привязкой к сетке, данных движущихся платформ и триангулированных нерегулярных сеток (то есть, когда *dataCodingFormat* равен 1, 3, 4, 7 или 8), местоположение каждой точки должно

указываться индивидуально. Это достигается в наборе данных HDF5 в группе "Positioning", которая дает индивидуальные координаты местоположения (например, долготу и широту) для каждого положения. Для данных временных рядов по постоянным станциям значения долготы и широты являются позициями станций; число станций - *numberOfStations*. Для не геопривязанных данных с привязкой к сетке значения являются позициями каждой точки сетки; число точек сетки равно *numberOfNodes*. Для данных движущейся платформы значения являются положением платформы в каждый момент времени; число платформ - *numberOfStations*.

Для покрытий нерегулярных сеток и сеток с переменным размером ячеек (dataCodingFormat 5 и 6) формат хранения использует те же метаданные, что и для обычных сеток, плюс наборы данных HDF5, указывающие, какие ячейки заполняются или агрегируются соответственно. Последние наборы данных кодируют местоположение ячеек относительно точки сетки или адреса ячейки в координатах сетки - то есть индексов в сетке или код Мортон - а не географические координаты (широта/долгота). Порядок последовательности и осей, необходимые для интерпретации координат сетки как географических координат, задаются атрибутами *sequencingRule* и *scanDirection* соответственно. Путем комбинирования этой информации с параметрами сетки, представленными в метаданных, положение заполненных ячеек/точек может быть рассчитано с помощью несколько более сложных формул, чем для регулярных данных с координатной привязкой к сетке.

В таблице ниже кратко излагаются стратегии хранения информации о координатах.

Таблица 10с-15 – Позиционирование типов и размеров наборов данных для различных типов покрытия

Тип покрытия	Формат dataCoding	Структура набора данных о координатах
Фиксированные станции	1	1-мерный массив, длина = numberOfStations
Регулярная сетка	2	Не используется
Не геопривязанная сетка	3	1-мерный массив, длина = numberOfNodes
Подвижная платформа	4	1-мерный массив, длина = numberOfTimes
Нерегулярная сетка	5	Не используется
Сетка с переменным размером ячеек	6	Не используется
TIN	7	1-мерный массив, длина = numberOfNodes
Фиксированные станции	8	1-мерный массив, длина - numberOfStations
Регулярная сетка на базе фичеров	9	Не используется

ПРИМЕЧАНИЕ: Несколько движущихся платформ могут быть закодированы как разные реализации фичеров.

10с-9.10.2 Структуры данных для хранения информации о местоположении точек сетки

Число местоположений рассчитывается в соответствии с таблицей 10с-4 в пункте 10с-9.3.

Таблица 10с-16 – Группа позиционирования

Группа	Категория HDF5	Имя	Тип данных	Пространство данных
/Positioning	Набор данных	geometryValues	Составной (Float X D)	Массив (1-мерный) размера, зависящего от dataEncodingFormat, см. таблицу 10с-15 Компоненты составного типа именованы по названиям осей (например, 'latitude', 'longitude', 'Z', и т.п.) Имена размерности D и компонентов указываются в атрибуте размера контейнера фичеров и в наборе данных axisNames соответственно (Таблицы 10с-10 и 10с-9)
	Набор данных	triangles (опционально)	Массив (Integer)	Массив (2-d): размерения numberOfTriangles X 3 Каждая строка кодирует треугольник как индексы 3 координат в наборе данных geometryValues Требуется только для dataEncodingFormat = 7 (TIN)
	Набор данных	adjacency (опционально)	Массив (Integer)	Массив (2-d): размерения numberOfTriangles X 3 Каждая строка кодирует треугольники, смежные с любым данным треугольником, указывая их индексы в наборе данных треугольников adjacency[i][0] = triangle adjacent to the edge specified by triangles[i][0] & triangles[i][1] adjacency[i][1] = triangle adjacent to edge triangles[i][1] & triangles[i][2] adjacency[i][2] = triangle adjacent to edge triangles[i][2] & triangles[i][0] Элементы для рёбер без смежных треугольников заполняются значением -1 Применимо только для dataEncodingFormat = 7 (TIN), Но опциональный для TIN.

10с-9.11 Группы значений данных

Структура содержания значений данных аналогична структуре содержания местоположений, за исключением того, что обычные значения данных сетки (`dataEncodingFormat = 2` или `9`) хранятся в виде D-размерного массива, соответствующего порядку осей в наборе данных `axisNames` в группе `Feature Container` (большой индекс предшествует меньшему). Размерность D кодируется в атрибуте размерности группы `Feature Container`.

ПРИМЕР: Для двумерных данных с регулярной сеткой массивы значений являются двумерными с размерами `numPointsLongitudinal` и `numPointsLatitudinal`.

Для данных временных рядов фиксированных станций, не геопривязанных данных с координатной привязкой к сетке, данных от движущейся платформы и триангулированных нерегулярных сеток (то есть, когда `dataCodingFormat` равен 1, 3, 4, 7 или 8), значения данных хранятся как 1-мерные наборы данных длины, заданной атрибутом метаданных `numberOfStations` или `numberOfStations` группы реализаций фичеров (таблица 10с-12) в зависимости от `dataEncodingFormat`.

Для нерегулярных сеточных покрытий (`dataCodingFormat=5`) хранение значений данных такое же, как для не геопривязанных сеток и т.д. (то есть, 1-мерный массив записей значений, длина = `numberOfNodes`) но группа значений включает набор данных, который указывает адрес точки сетки или ячейки, связанные с каждой записью в массиве значений. Этот второй набор данных использует координаты сетки - то есть индексы в сетке или код Мортонса - а не географические координаты (широта/долгота). Порядок последовательности и оси, необходимый для интерпретации координат сетки как географических координат задаются атрибутами `sequencingRule` и `scanDirection` соответственно.

Для покрытий с переменным размером ячеек (`dataCodingFormat=6`) хранение значений данных является таким же, как и для покрытий нерегулярных сеток, но группы значений содержат набор данных индекса сетки, используемый нерегулярными сетками, а также набор данных, указывающий, какие ячейки агрегированы в более крупные ячейки.

Различные наборы данных и их компоненты описаны в следующей таблице.

Таблица 10с-17 – Типы значений набора данных и размер различных форматов кодирования данных

Тип покрытия	Формат dataCoding	Структура значений и дополнительных наборов данных HDF5	HDF5 Dataset components
Фиксированная станция	1	значения: 1-мерный массив, длина = numberOfStations	<p>Составной, по одному компоненту для каждого атрибута, указанного в соответствующем наборе данных информации о фичерах в группе информации о фичерах (таблица 10с-8)</p> <p>Имя компонента: код атрибута, указанный в наборе данных информации о фичерах</p> <p>Тип компонента: Любой тип данных HDF5, соответствующий типу данных атрибута, указанному в наборе данных Feature Information</p>
Регулярная сетка	2	<p>значения: D-мерный массив, размеры указаны в:</p> <p>2-D: numPointsLatitudinal X numPointsLongitudinal</p> <p>3-D: numPointsLatitudinal X numPointsLongitudinal X numPointsVertical</p> <p>Если cellGeometry имеется в группе реализаций фичеров: продукт всех значений cellGeometry[i].numPoints</p>	Как для фиксированных станций
Не геопривязанная сетка	3	значения: 1-мерный массив, длина = numberOfNodes	Как для фиксированных станций
Подвижная платформа	4	значения: 1-мерный массив, длина = numberOfTimes	Как для фиксированных станций
Нерегулярная сетка	5	значения: 1-мерный массив, длина = numberOfNodes	<p>Как для фиксированных станций.</p> <p>Упорядочено в соответствии с правилом последовательности, указанным атрибутами sequencingRule и scanDirection группы контейнеров фичеров</p> <p>Группа контейнеров (таблица 10с-10)</p>
		<p>gridIndex: 1-мерный массив, длина = numberOfNodes (атрибут набора данных codeSize: Integer – обозначает длину бит-поля)</p>	<p>Тип элемента: bitfield (длина определяется размерами сетки)</p> <p>Порядок элемента соответствует массиву значений</p> <p>Каждый элемент содержит код ячейки (точки сетки) в соответствии с правилом последовательности, заданным атрибутами sequencingRule и scanDirection.</p>

			Например, код Мортоня ячейки
Переменный размер ячеек	6	значения: 1-мерный массив, длина = numberOfNodes	Как для фиксированных станций
		gridIndex: 1-мерный массив, длина = numberOfNodes (атрибут набора данных codeSize: Integer - обозначает длину бит-поля)	(как для массива <i>gridIndex</i> для нерегулярной сетки) Для ячеек, агрегированных в один блок, используйте первую ячейку (точку сетки), встречающуюся в порядке последовательности. Например, код Мортоня ячейки
		cellScale 1-мерный массив, длина = numberOfNodes	Тип элемента: Составной Порядок элементов соответствует массиву значений Компоненты составного типа называются в соответствии с именами осей в наборе данных axisNames в группе контейнеров фичеров Каждый компонент имеет тип Integer и дает число ячеек, агрегированных по названной оси
TIN	7	1-мерный массив, длина = numberOfNodes	Как для фиксированных станций
Фиксированная станция	8	1-мерный массив, длина = numberOfTimes	Как для фиксированных станций
Регулярная сетка на базе фичеров	9	значения: D-мерный массив, размерения определяются: 2-D: numPointsLatitudinal X numPointsLongitudinal Если cellGeometry представлена в группе реализаций фичеров: продуктом всех значений cellGeometry[j].numPoints	Как для регулярной сетки The name of the feature container group must correspond to a camelCase name of a feature type object from the IHO Geospatial Information (GI) Registry, Data Dictionary Register

ПРИМЕЧАНИЯ:

2) 64-разрядные беззнаковые целые числа для массивов gridIndex позволяют 4-мерные сетки с максимальным числом $2^{16} - 1$ (65535) точек/ячеек в каждом измерении.

3) Наборы данных *gridIndex* имеют целочисленный атрибут *codeSize*, который дает длину (в битах) битового поля, содержащего индекс. Это зависит от типа кода и числа измерений. Например, двухмерная сетка с 8 точками в каждой размерности нуждается в 6-битных кодах Мортоня.

4) Размер битового поля вычисляется путём умножения числа битов, необходимых для размещения наибольшей размерности на число размерностей (D). Для уменьшения сложности каждому измерению выделяется одинаковое число бит в битовом поле. Например, массиву 200 X 1000 задаётся 20-битное битовое поле, вычисляемое как:

$\text{codesize} = 2 \times \max([\log_2 200], [\log_2 1000])$.

Рисунок, который следует ниже, представляет собой массивы *gridIndex* и *cellScale* для нерегулярной сетки (слева) и массива ячеек переменного размера (справа). Оба используют коды Мортон и двумерные сетки (номинально) 4x4 ячеек в каждом измерении. Обратите внимание, что на рисунке коды присваиваются ячейкам, а не точкам сетки. Панели слева описывают нерегулярную сетку с 11 заполненными ячейками. Панели справа описывают сетку переменных размеров ячеек с двумя агрегированными ячейками, каждая из которых объединяет 2x2 единичных ячейки.

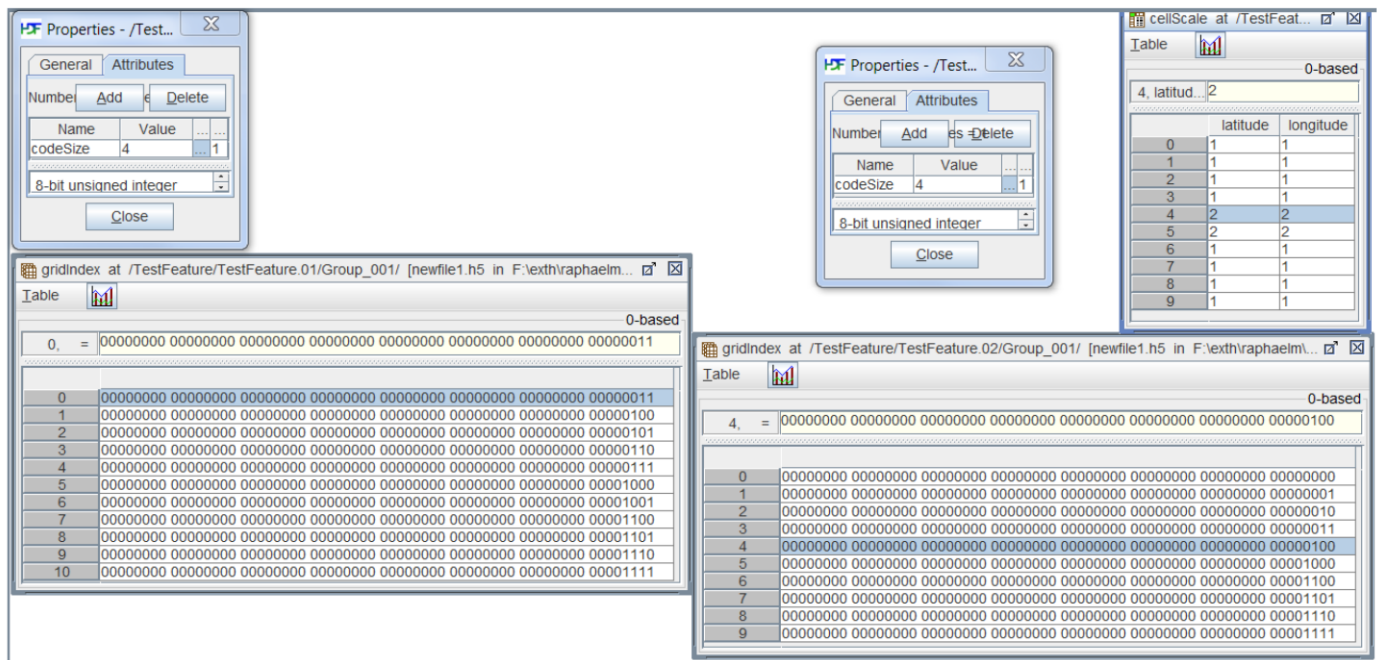
Сами сетки показаны ниже панелей, а коды Мортон показаны в соответствующих ячейках³. Пример справа также показывает масштабирование каждой ячейки в скобках (предполагается, что масштаб одинаков во всех измерениях; то есть ячейки 0100 и 1000 каждая составляют 2x2 области сетки).

Например, в случае нерегулярной сетки отсутствующие ячейки в сетке не отображаются. Например, для примера с переменным размером ячеек серые клетки агрегируются с ячейками 0100 или 1000.

Для сетки с переменным размером ячеек этот профиль определяет размер агрегированных ячеек с точки зрения числа единичных ячеек, которые они покрывают в каждом направлении, вместо применения того же коэффициента масштабирования в каждом измерении, как показано в примере в правом нижнем углу рисунка. Это для лучшего размещения прямоугольных агрегаций и агрегаций неправильной формы. Регионы неправильной формы должны быть разбиты на несколько прямоугольных агрегатов. (Использование прямоугольных агрегатов сопряжено с дополнительными затратами на хранение.)

Дальнейшие оптимизации могут быть рассмотрены в будущих изданиях этого профиля.

³ Два изображения сетки в нижней части рисунка взяты из "Elevation Surface Model Standardized Profile" (DGIWG 116-1) Ed. 1.0.1, Defense Geospatial Information Working Group (10 June 2014).



			1110	1111
1000	1001	1100	1101	
		0011	0110	0111
			0100	0101

1010	1011	1110 (1)	1111 (1)
1000 (2)	1001	1100 (1)	1101 (1)
0010 (1)	0011 (1)	0110	0111
0000 (1)	0001 (1)	0100 (2)	0101

Рисунок 10с-10 – Примеры массива индексов сетки для нерегулярных сеток (слева) и массивов индексов сетки и масштабов ячеек для сетки с переменным размером ячеек (справа).

Теперь можно описать структуру групп значений данных. Каждая группа структурирована так, как показано в таблице ниже.

Таблица 10с-18 – Структура группы значений

Группа	Категория HDF5	Имя	Тип данных	Пространство данных
/Group_NNN	Атрибут	См. таблицу 10с-19	См. таблицу	Значение атрибута как описано в таблице 10с-19
	Набор данных	values	Compound	Массив составного типа с рангом массива в зависимости от dataCodingFormat и пространственного измерения, как описано в таблице 10с-17
	Набор данных	gridIndex	Bitfield	Требуется для dataEncodingFormat = 5 или 6 Описан в таблице 10с-17
	Набор данных	cellScale	Compound	Требуется для dataEncodingFormat = 6 Описан в таблице 10с-17

Данные временных рядов для всех, кроме формата движущихся платформ (dataEncodingFormat = 4, 8), закодированы в последовательные группы, содержащиеся в группе реализаций.

Каждая подгруппа содержит значение дата-время и массивы записей значений. Для dataCodingFormat = 2, 3, 5 или 6, дата-время даются для всей сетки. Массивы значений данных являются двумерными с числом столбцов (numCOLS) и строк (numROWS). Для временных рядов значения данных будут относиться к каждому времени в рядах. Для сетки значений скорости и направления будут для каждой точки сетки.

Группы пронумерованы 001, 002 и т.д., до максимального числа групп, numGRP. Для всех типов покрытия, за исключением движущихся платформ и временных серий фиксированных станций, количество Групп – это количество записей времени. Для данных движущейся платформы существует только одна группа, соответствующая одной платформе; дополнительные платформы могут быть размещены в дополнительных реализациях фичеров. Для данных фиксированных станций количество групп – это количество станций.

Число отдельных групп определяется по переменной метаданных, numGRP. Интервал времени между отдельными периодами определяется с помощью переменной метаданных timeRecordInterval.

Значения, представляющие разное время, сохраняются последовательно, от древнейшего до новейшего. Начальное значение дата-время содержится в атрибуте метаданных (Таблица 10с-12). Зная временной интервал между каждой записью, можно вычислить время, применимое к каждому значению.

Группы, если они представляют различные времена, нумеруются последовательно, от древнейшего до новейшего.

Атрибуты (таблица 10с-19) могут состоять из единичного значения (timePoint) для сеточных данных, или из списка переменных, описывающих временные серии фиксированных станций (dataCodingFormat = 8).

Таблица 10с-19 – Атрибуты для групп значений

№	Имя	Обозначение	Множ	Тип данных	Примечания и/или единицы измерения
dataCodingFormat = 1, 2, 3, 4, 5, 6 или 7					
1	Отметка времени	timePoint	1	String	DateTime
dataCodingFormat = 8					
1	Имя станции	stationName	0..1	String	Например, географическое описание или 'Не доступна'
2	Идентификация станции	stationIdentification	0..1	String	Например, алфавитно цифровой код станции или 'Не доступна'
3	Количество записей	numberOfTimes	0..1	Integer	Значение здесь превосходит соответствующее значение на уровне реализации
4	Индекс интервала времени	timeIntervalIndex	1	(Integer)	1 (TRUE) означает одинаковый временной интервал; интервал дается в <i>timeRecordInterval</i> 0 (FALSE) означает неравномерный временной интервал. Логический тип данных реализованный как описано в таблице 10с-1
5	Интервал времени	timeRecordInterval	0..1	Integer	Только если <i>timeIntervalIndex</i> = 1 Равномерный интервал между записями времени в секундах. Значение здесь превосходит соответствующее значение на уровне реализации
6	Самое раннее значение времени	startDateTime	0..1	String	Формат DateTime
7	Самое позднее значение времени	endDateTime	0..1	String	Формат DateTime
	(дополнительные атрибуты)				(Как указано в Спецификации продукта)

10с-9.11.1 Регулярная сетка, ориентированная на фичер

dataCodingFormat описывает возможность создания фичеров в форме растровой геометрии и предоставляет дополнительную информацию. **dataCodingFormat** может использоваться только вместе с таблицей атрибутов фичеров (пункт 10с-9.6.2, таблица 10с-9).

Название группы контейнеров фичеров должно соответствовать имени camelCase объекта типа фичеров из реестра геопространственной информации (GI) МГО, регистра словарей данных.

Каждая ячейка сетки определяет значение, которое позволяет идентифицировать соответствующий фичер. Тип данных идентификационного значения фичера идентичен идентификационному значению из таблицы атрибутов фичеров (пункт 10с-9.6.2). Код, а также имя идентификационного значения фичера определяются в группе информации фичеров (пункт 10с-9.5); например, индекс или идентификатор. Идентификационные значения фичера могут возникать несколько раз в сетке, поскольку несколько ячеек сетки вместе определяют геометрию сетки фичера. Сжатие набора данных HDF5 исключает эффект избыточности. Значение идентификации фичера является связью между геометрией сетки в сетке и информацией из таблицы атрибутов фичеров.

Использование `dataCodingFormat` может потребовать создание еще одной группы контейнеров в растровом формате. Это ссылка для фичеров, созданных здесь. Пространственная протяженность `dataCodingFormat`, описанного здесь, относится к ранее упомянутой группе контейнеров фичеров. Пространственная протяженность должна совпадать. Какая Группа контейнеров фичеров устанавливается в качестве ссылки должно быть определено в используемой Спецификации продукта.

Если в определенной области сетки не могут/не должны создаваться фичеры, соответствующим ячейкам сетки присваивается идентификационное значение фичера 0 (ноль). Таким образом, значение 0 (ноль) рассматривается как значение `nodata`. Рисунок 10с-11 ниже показывает упрощенную иллюстрацию, используя пример для S-102.

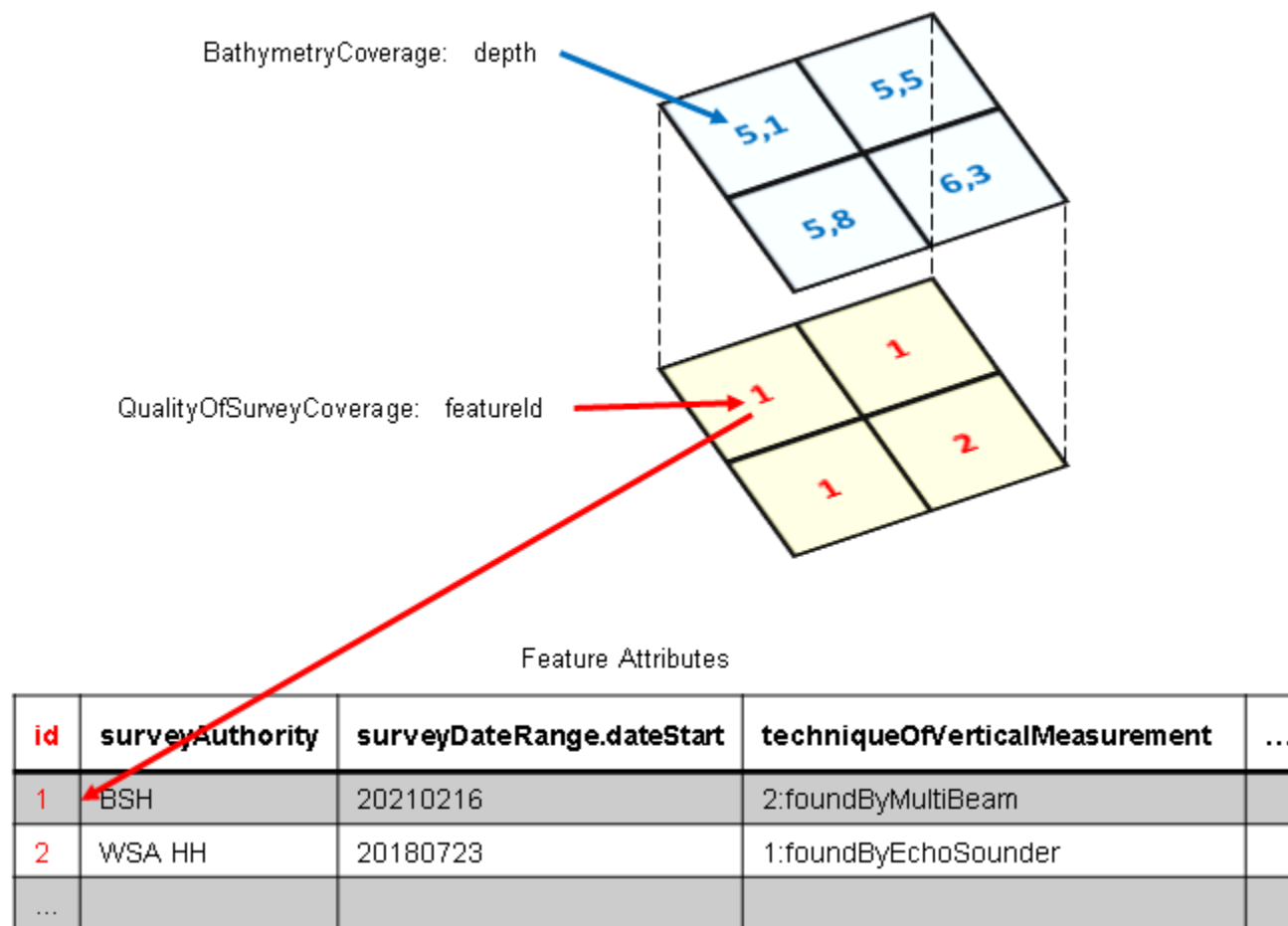


Рисунок 10с-11 – Иллюстративный пример dataCodingFormat

10с-10 Общие перечни и словари

10с-10.1 Формат кодирования данных

Таблица 10с-20 – Перечень формата кодирования данных

Пункт	Имя	Описание	Код	Примечание
Перечень	S100_HDF_DataCodingFormat	Форматы кодирования данных S-100 HDF5		
Буквенный	fixedStations	Данные в нескольких фиксированных точках	1	
Буквенный	regularGrid	Данные по точкам сетки, образующие сетку ячеек с постоянным интервалом	2	Регулярные сетки обычно состоят из перпендикулярно пересекающихся линий с равным интервалом по каждому измерению, создавая квадратные или прямоугольные ячейки
Буквенный	ungeorectifiedGrid	Данные, которые не включают информацию, которая может быть использована для определения географических координат ячеек или наличия переменного интервала ячеек, и нет predetermined связи между расположением одной ячейки и расположением другой ячейки	3	Например, цифровая перспективная аэрофотография без ортофотоинформации
Буквенный	movingPlatform	Данные в последовательных дискретных точках подвижной платформы датчиков	4	
Буквенный	irregularGrid	Данные распределены по сетке с равномерным расстоянием между ячейками, но нерегулярной общей формой	5	Неравномерность формы может состоять из непрямоугольности покрытия или относительно больших регионов, в которых данные отсутствуют
Буквенный	variableCellSize	Сетка с переменной плотностью, содержащая одну или несколько областей с шагом, кратным равным общему минимальному расстоянию между ячейками	6	Форма сетки в целом может быть непрямоугольной
Буквенный	TIN	Триангулированная нерегулярная сеть	7	TIN – это представление непрерывной поверхности, состоящей полностью из треугольников. Вершины углов каждого треугольника являются общими со смежными треугольниками. Эти вершины образуют контрольные точки функции покрытия.
Буквенный	stationwiseFixed	Данные в нескольких фиксированных точках организованные по станциям	8	

Буквенный	featureOrientedRegularGrid	Данные в точках сетки, образующие регулярную сетку с постоянным расстоянием между ячейками	9	Регулярные сетки обычно состоящие из перпендикулярно пересекающихся линий с равным интервалом по каждому измерению, создавая квадратные или прямоугольные ячейки
-----------	----------------------------	--	---	--

10с-10.2 Тип горизонтальной CRS

Таблица 10с-21 – Тип горизонтальной CRS

Пункт	Имя	Описание	Код	Примечание
Перечень	typeOfHorizontalCRS	Коды описания типа двумерной горизонтальной CRS		
Буквенный	geodeticCRS2D	Двухмерная геодезическая CRS	1	
Буквенный	projectedCRS	Проекционная CRS	2	

10с-10.3 База вертикальных координат

Таблица 10с-22 – База вертикальных координат

Пункт	Имя	Описание	Код	Примечание
Перечень	verticalCoordinateBase	Коды для описания базового уровня вертикальной системы координат (нуля)		
Буквенный	seaSurface	Базой (нулем) вертикальной системы координат является поверхность моря	1	
Буквенный	verticalDatum	Базой (нулем) вертикальной системы координат является определенный вертикальный датум	2	
Буквенный	seaBottom	Базой (нулем) вертикальной системы координат является дно моря	3	

10с-10.4 Ссылка на вертикальный датум

Таблица 10с-23 – Ссылка на вертикальный датум

Пункт	Имя	Описание	Код	Примечание
Перечень	verticalDatumReference			
Буквенный	s100VerticalDatum	Вертикальный датум – один из перечисленных в S100_VerticalAndSoundingDatum	1	
Буквенный	EPSG	Вертикальный датум – один из перечисленных в Реестре EPSG	2	

10с-10.5 Виды проекций

Таблица 10с-24 – Виды проекций и их параметры

Имя	Код EPSG	Параметр1	Параметр2	Параметр3	Параметр4	Параметр5
Меркатор	9805	Широта 1-й стандартной параллели	Начальный меридиан	-	-	-
Поперечный Меркатор	9807	Начальная широта	Начальный меридиан	Масштабный множитель начального меридиана	-	-
Косой Меркатор	9815	Широта центра проекции	Долгота центра проекции	Азимут начальной линии	Угол между прямой и косой сеткой	Масштабный множитель на начальной линии
Hotline Косой Меркатор	9812	Широта центра проекции	Долгота центра проекции	Азимут начальной линии	Угол между прямой и косой сеткой	Масштабный множитель на начальной линии
Коническая равноугольная проекция Ламберта (1SP)	9801	Начальная широта	Начальный меридиан	Масштабный множитель начального меридиана	-	-
Коническая равноугольная проекция Ламберта (2SP)	9802	Начальная ложная широта	Начальная ложная долгота	Широта 1-й стандартной параллели ⁵	Широта 2-й стандартной параллели ⁶	-
Косая стереографическая	9809	Начальная широта	Начальный меридиан	Масштабный множитель начального меридиана	-	-
Полярная стереографическая	9810	Начальная широта ⁷	Начальный меридиан	Масштабный множитель начального меридиана	-	-
Косая коническая равноугольная Кривока	9819	Широта центра проекции	Долгота центра проекции	Азимут начальной линии	Широта псевдо стандартной параллели	Масштабный множитель псевдо стандартной параллели
Американская поликоническая	9818	Начальная широта	Начальный меридиан	-	-	-
Равноплощадная Альберса	9822	Начальная ложная широта	Начальная ложная долгота	Широта 1-й стандартной параллели ⁵	Широта 2-й стандартной параллели ⁶	-
Азимутальная равноплощадная проекция Ламберта	9820	Начальная широта	Начальный меридиан	-	-	-

ПРИМЕЧАНИЕ: Все широты и долготы параметров проекций должны даваться в градусах (южные и западные - отрицательные). Азимуты даваться в градусах. Подробное описание способов проекций смотрите в документации EPSG.

⁵ Стандартная параллель, ближайшая к экватору.

⁶ Стандартная параллель, самая дальняя от экватора.

⁷ Должна быть 90 градусов или -90 градусов

10с-10.6 Нули высот и глубин

Таблица 10с-25 – Нули высот и глубин

Значение	Название	Описание	Код	Примечания
S100_Codelist	S100_VerticalAndSoundingDatum	Допустимые нули высот и глубин	-	Открытый перечень
Значение	meanLowWaterSprings		1	(MLWS)
Значение	meanLowerLowWaterSprings		2	-
Значение	meanSeaLevel		3	(MSL)
Значение	lowestLowWater		4	-
Значение	meanLowWater		5	(MLW)
Значение	lowestLowWaterSprings		6	-
Значение	approximateMeanLowWaterSprings		7	-
Значение	indianSpringLowWater		8	-
Значение	lowWaterSprings		9	-
Значение	approximateLowestAstronomicalTide		10	-
Значение	nearlyLowestLowWater		11	-
Значение	meanLowerLowWater		12	(MLLW)
Значение	lowWater		13	(LW)
Значение	approximateMeanLowWater		14	-
Значение	approximateMeanLowerLowWater		15	-
Значение	meanHighWater		16	(MHW)
Значение	meanHighWaterSprings		17	(MHWS)
Значение	highWater		18	(HW)
Значение	approximateMeanSeaLevel		19	-
Значение	highWaterSprings		20	-
Значение	meanHigherHighWater		21	(MHHW)
Значение	equinoctialSpringLowWater		22	-
Значение	lowestAstronomicalTide		23	(LAT)
Значение	localDatum		24	-
Значение	internationalGreatLakesDatum1985		25	-
Значение	meanWaterLevel		26	-
Значение	lowerLowWaterLargeTide		27	-
Значение	higherHighWaterLargeTide		28	-
Значение	nearlyHighestHighWater		29	-
Значение	highestAstronomicalTide		30	(HAT)
Значение	balticSeaChartDatum2000	Baltic Sea Chart Datum 2000	44	-
Значение	internationalGreatLakesDatum2020	Обновление 2020 Международного датума Великих озер (IGLD), официальная система отсчета для измерения высоты уровня воды в Великих озерах, соединяющих каналы и речную систему Св. Лаврентия	46	В отличие от двух предыдущих в рамках этого обновления данных будут использоваться вертикальные данные на основе геоидов, доступ к которым будет обеспечиваться с помощью глобальных навигационных спутниковых систем (ГНСС), таких, как GPS
Значение	seaFloor	Дно океанов и морей, где есть, как правило, плавный градиент.	47	-
Значение	seaSurface	Двумерное (в горизонтальной плоскости) поле, представляющее воздушно-морской интерфейс, с высокой частотой	48	-

		колебания, такие как ветровые волны и зыбь, но не астрономические приливы		
Значение	hydrographicZero	Вертикальный отсчет вблизи наинизшего астрономического прилива (LAT), ниже которого уровень моря падает лишь в исключительных случаях	49	Расхождение между LAT и гидрографическим нулем может быть связано с сильным антициклоническим атмосферным состоянием, увеличение веса водяного столба, который может вызывать наинизший уровень моря и падение ниже астрономического наинизшего уровня

ПРИМЕЧАНИЕ: Цифровыми кодами являются коды, указанные в GI реестре ИО для эквивалентных перечисленных значений атрибута Hydro Domain Vertical Datum, поскольку Реестр в настоящее время (20 июня 2018 года) не содержит записей для атрибутов метаданных наборов обмена и метаданных набора данных.

Датумы, не включенные в перечень S-100, должны кодироваться с использованием формы "other: ...". Если датум, о котором идет речь, указан в GI реестре ИО (в качестве одного из стандартных перечисленных значений атрибута Вертикальный датум в домене ИО Hydro), то "camel case code" в реестре должен использоваться в элементе "other: ...". Для датумов из реестра EPSG, не указанных в GI Реестре ИО, форма должна быть "other: EPSG_NNNN".

ПРИМЕР 1: "Local Low Water Reference Level" указан в GI Реестре ИО, но не указан в стандарте S-100. Он должен кодироваться кодом camel case в GI Реестре как: "other: localLowWaterReferenceLevel".

ПРИМЕР 2: "European Vertical Reference Frame 2019 mean tide" указан в Реестре EPSG в списке вертикальных датумов (EPSG 1287) но отсутствует в списке GI Реестра ИО. Он должен кодироваться как: "other: EPSG_1287".

Если датум отсутствует в списке и в таблице, данной выше, разработчики GI Реестра ИО или Реестра EPSG должны определить ему специальный код после консультации с рабочей группой ИО и администратором GI Реестра ИО.

Использование датумов, которые не фигурируют ни в перечне, приведенном выше, ни в GI Реестре ИО или Реестре EPSG, не рекомендуется. Производители, которым необходимо использовать данные, не указанные в перечне S-100, должны предложить включить их в GI Реестр ИО и/или в этот перечень посредством внесения предложения установленным S-100 порядком.

Обратите внимание, что к прикладному программному обеспечению не предъявляются требования к обработке информации, закодированной в форме "other: ...", это означает, что, например, программное обеспечение ECDIS, не обязано распознавать датумы, закодированные как "other: ...", и поэтому не смогут исправлять глубины ЭНК, привязанные к уровню как это

принято в наборах данных S-104, а будут удалять их как несовместимые с набором данных ЭНК S-101.

10с-11 Файлы поддержки

Формат HDF5 не кодирует информацию о файлах поддержки в качестве атрибутов фичера; то есть, тематические атрибуты прикладной схемы не могут быть ссылками на файлы поддержки. Это означает, что ссылки на фотографии или текстовые файлы и т.д., не допускаются в фичерах покрытия.

Кроме того, не допускаются ассоциации фичеров и информации из покрытия к векторным фичерам.

Атрибут корневой группы HDF5 "metadata" является ссылкой на внешний файл метаданных. Ссылка должна быть строкой формы:

fileRef:<fileName>

где <fileName> - базовое имя файла ISO 8211 или GML. Часть расширения имени файла не используется.

Продукты данных смешанного векторного покрытия могут продолжать использовать файлы поддержки в связи с классами векторных фичеров и определять классы векторных фичеров или информации с атрибутами, которые являются ссылками на файлы поддержки, как обычно.

10с-12 Файлы каталогов и метаданных

Каталоги наборов обмена и файлы метаданных должны соответствовать стандартным схемам XML для каталогов и метаданных, определенных для этого издания S-100 и соответствующих стандартов ISO. Файлы должны быть названы следующим образом:

CATALOG.XML (or .xml) Exchange catalogue XML file.

MD_<HDF5 data file base name>.XML (or .xml) ISO metadata

10с-13 Векторные пространственные объекты, фичеры и типы информации

В некоторых случаях может потребоваться использование векторных пространственных объектов, таких как полигоны зон влияния. Это издание профиля не кодирует векторные пространственные объекты непосредственно в файле данных HDF5. Вместо этого пространственные объекты должны быть определены во внешнем файле (в формате GML или ISO 8211) и содержать ссылку на закодированный пространственный объект. Ссылка должна быть строкой формы:

extObjRef:<fileName>:<recordIdentifier>

где <fileName> - базовое имя файла ISO 8211 или GML, а <recordIdentifier> - идентификатор записи векторного объекта в этом файле. Часть расширения имени файла не используется. Идентификатором записи является идентификатор gml:id для наборов данных GML или идентификационный номер записи (RCID) для наборов данных ISO 8211. Файл должен присутствовать в том же наборе обмена.

Этот метод может быть использован для ссылки на полигоны и т.д., определенные во внешних файлах в формате GML или файлов данных 8211 в том же наборе обмена. Он также может быть использован для ссылок на реализации типов фичеров или информации в файле GML или ISO 8211.

ПРИМЕРЫ:

USSFC00001:S093546 ссылается на объект gml:id S093456 в файле данных GML - USSFC00001.GML (GML).

USSFC00001:93546 ссылается на объект с идентификатором записи 93456 в файле данных ISO 8211 - USSFC0000.000 (ISO 8211).

10с-14 Ограничения и проверка

10с-14.1 Проверочные тесты

Тестовые испытания должны быть определены в спецификации продукта и включать проверки на предмет того, что:

- Структура файла HDF5 соответствует этому профилю;
- Обязательные атрибуты в группах присутствуют в соответствии с закодированным значением dataCodingFormat;
- Имена групп, наборов данных и атрибутов соответствуют этому профилю;
- Длины массивов данных о позиционировании и значения записей последовательны;
- Компоненты составных типов называются в соответствии с требованиями спецификации.

10с-15 Обновления

Обновления к данным HDF5 рекомендуется использовать в той же структуре, что и базовый файл данных HDF5. Обновления могут включать только обновляемые наборы данных HDF5. Обновляемые конкретные наборы данных полностью включаются в обновленный файл данных.

Это положение предполагает, что наборы данных S-100 могут частично обновляться и полностью заменяться обновленными данными, однако от спецификации продуктов не требуются разрешения для обеспечения возможности частичного обновления. Они могут определять процессы создания и управления обновлениями, которые больше подходят для их конкретных областей и приложений. Однако если допускается обновление частей наборов данных S-100, то должно соблюдаться правило, содержащееся в предыдущем пункте.

10с-16 Краткое описание модели

Базовая структура профиля HDF5 (рисунок 10с-7) теперь может быть представлена в виде более подробной концептуальной модели с использованием спецификаций групп и набора данных в предыдущих разделах. Концептуальная модель содержимого файла HDF5 показана на следующем рисунке. Этот рисунок показывает структуру групп и наборов данных, которые содержат пространственные представления и значения данных. (Атрибуты метаданных и наборы данных, содержащие метаданные, не включены в целях упрощения.) Ассоциация *MatchingOrders* указывает на то, что последовательность элементов в ассоциированных наборах данных является взаимозависимой.

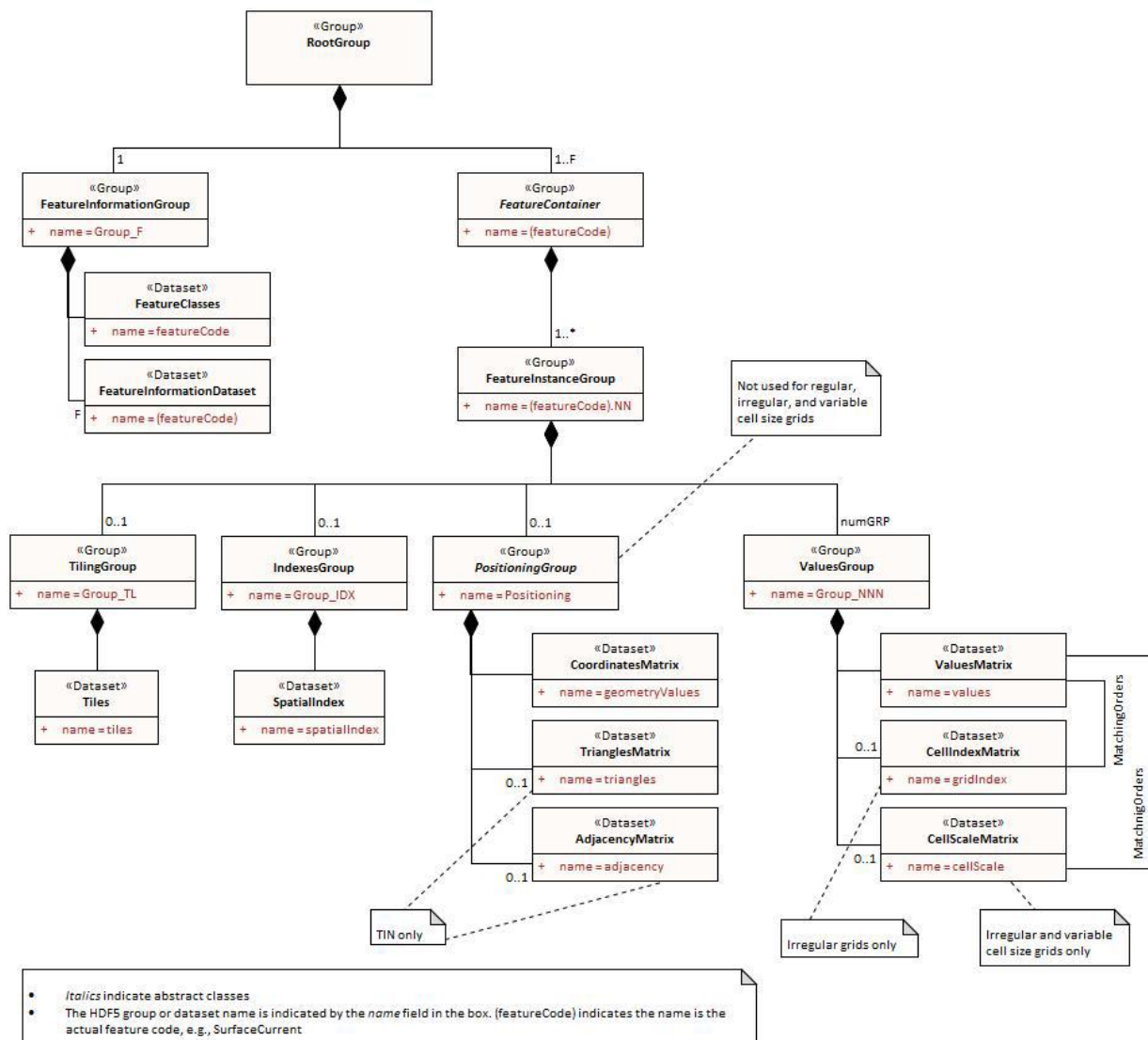


Рисунок 10с-11 - Концептуальная модель содержания

10с-17 Правила для разработчиков спецификаций

10с-17.1 Определение формата спецификации из этого профиля

Большинству спецификаций продукта потребуется только подмножество этого профиля. Однако все спецификации продуктов должны включать обязательные элементы этого профиля.

Логическая структура данных должна соответствовать логической структуре, изображенной на рисунке 10с-11 и указанной в предыдущих разделах.

Раздел 'Формат данных' спецификации продукта должен указывать, какая часть профиля используется (например, какие значения может принимать `dataCodingFormat`, какие группы и наборы данных используются, является ли пространственное представление 2-мерным, 3-мерным и т.д.).

Диаграммы UML, полученные на основе описания концептуальной структуры в настоящей части, рекомендованы, но не обязательны. Должны представляться таблицы документации, в которых указываются конкретные ограничения или ограничения в отношении метаданных и содержания продуктов, если соответствующая таблица в этом профиле не применяется без изменений.

Спецификации, требующие наличия нерегулярных сеток, должны рассматриваться как не геопривязанные сетки и иметь четко закодированные координаты каждого местоположения.

Этот профиль не препятствует тому, чтобы класс фичеров имел различные типы покрытия, однако повторение пространственных атрибутов в одной и той же реализации в этом профиле невозможно. Это означает, что реализация фичера не может иметь две сетки, независимо от того, являются ли они одним и тем же типом покрытия. Если для одной и той же реализации спецификации продукта необходимы несколько покрытий, рассмотрите возможность объединения их в один объект покрытия или использования двух реализаций фичеров.

Ассоциации фичеров и информации не полностью реализованы в этом профиле. Однако можно связать объекты покрытия с векторными фичерами или информационными объектами в сопроводительных наборах данных GML или ISO 8211 с использованием методов ссылки на объекты, описанных в пункте 10с-13. Ссылки на векторные объекты, такие как многоугольники влияния, должны кодироваться одним и тем же методом.

В спецификациях продуктов следует указывать точность цифровых элементов метаданных, которые закодированы в формате данных HDF5 либо по отдельности, либо в бланковых инструкциях. Например, спецификация продукта может потребовать, чтобы все атрибуты метаданных типа `Float` были закодированы с использованием 64-битных чисел с плавающей запятой.

Если погрешность в позициях или значениях данных варьируется в зависимости от пространственного охвата одного фичера, разработчикам спецификации продукта следует рассматривать решения как часть спецификации продукта; например, разбивка сетки на различные реализации фичеров, или решение этого вопроса на уровне прикладной схемы путем определения фичера наложения для кодирования

погрешностей или добавления атрибута погрешности в запись значений. Эта часть не требует какого-либо конкретного подхода к этой проблеме.

10с-17.2 Различные правила

Использование строк переменной длины в качестве компонентов составных типов не рекомендуется из-за сообщаемых проблем производительности.

В теории, использование тайлов может взаимодействовать с HDF5, чтобы повлиять на производительность. Спецификации продуктов, для которых эффективность является важным соображением, возможно, потребуется рассмотреть различные эффекты взаимодействия и изучить их масштабы и последствия.

10с-17.3 Расширения этого профиля

Спецификации продукта могут расширять формат данного профиля путем определения дополнительных структур данных или расширения структур данных, определенных в этом профиле; но все расширения должны сохранять основные спецификации этого профиля, так что **реализации должны быть в состоянии обрабатывать и изображать данные без обработки дополнительных структур данных. Спецификация продукта должна быть написана так, чтобы использование этих дополнительных структур данных для обработки или изображения было необязательным.**

Такие дополнения должны быть размещены в соответствующем месте в файле данных HDF5; например, пространственные индексы в группе Group_IDX.

Расширения не должны повторно использовать имена элементов, определенных в этом профиле. Элементы, определенные в этом профиле, не должны переименовываться в спецификации продукта.

Ниже приводятся некоторые примеры допустимых и недопустимых расширений.

- Допустимые расширения:
 - Индекс Quadtree, добавленный как набор данных HDF5 в группу индексов.
 - Расширение структуры записи значений, которое сохраняет основной формат, описанный в этом профиле (то есть, одномерную структуру массива и указанные компоненты).
 - Массивы линейного масштаба, указывающие точки сетки на каждой оси, где размер ячейки изменяется, в качестве дополнения к массивам переменных размеров ячейки.
 - Метаданные по конкретным продуктам как атрибуты любой из групп, указанных в настоящем профиле.
 - Метаданные по конкретным продуктам в качестве дополнительных наборов данных HDF5 в любой из групп, указанных в этом профиле.
 - Дополнительные группы, если они не используются в качестве замены одной из обязательных групп в данном профиле.
- Недопустимые расширения:

- Изменение ранга типа набора данных массива; например, использование двумерного массива вместо одномерного массива.
- Изменения в правилах именования компонента составного типа данных, определенного в этом профиле.

10с-17.4 Расширения, добавляющие метаданные

Хотя раздел 10с-17.3 допускает добавление метаданных, определение метаданных по конкретным продуктам означает, что при использовании дополнительных метаданных, помимо простого их отображения, расширение должно включать кодирование конкретных продуктов в прикладных целях. С учетом того, что экосистема S-100 включает в себя несколько продуктов данных, которые в идеале можно было бы обрабатывать (включая изображение) с помощью приложения S-100, в настоящей части рекомендуется не добавлять метаданные по конкретным продуктам, которые оказывают какое-либо воздействие на обработку или изображение. Если такие добавления будут сочтены необходимыми, то их следует предложить в качестве расширения самой системы S-100 с использованием механизма поддержания, описанного в S-100 и связанных с ней документах. Отображение только метаданных (то есть, если приложение только должно отображать содержимое добавленного атрибута) может быть добавлено, но не рекомендуется.

10с-18 Практическое руководство

HDF5 C API содержит интерфейсы для определения типов составных компонентов. Это позволяет предположить, что размер типа данных может быть проверен для уменьшения возможных проблем преобразования.

HDF5 C API также определяет итераторы для итерации атрибутов или элементов в группе. Эти итераторы могут использоваться для обнаружения наборов данных, групп или атрибутов профиля из наборов данных, групп и атрибутов, определенных только в спецификациях отдельных продуктов (наименования продуктов будут отличаться от названий элементов профиля).

Порядок получения объектов может не совпадать с порядком создания. Исполнители должны разрешить это или исследовать доступность функций сохранения порядка в HDF5 API.

Связь между каталогом фичеров XML и объектами в файле HDF5 сохраняется с помощью кодов (same case) для фичеров и атрибутов.

10с-19 Реализации прикладных схем из части 8 (информативный)

Рисунок 10с-13 ниже иллюстрирует реализацию различных видов покрытий, описанных в пункте 10с-10.1 и таблице 10с-4, из видов покрытия, определенных в части 8. Группы по разработке спецификаций продуктов, возможно, пожелают разработать аналогичные рисунки для своих спецификаций, с тем чтобы описать,

каким образом виды покрытий, закодированные в формате HDF5 для их данных, реализуются на основе типов покрытий, определенных в части 8.

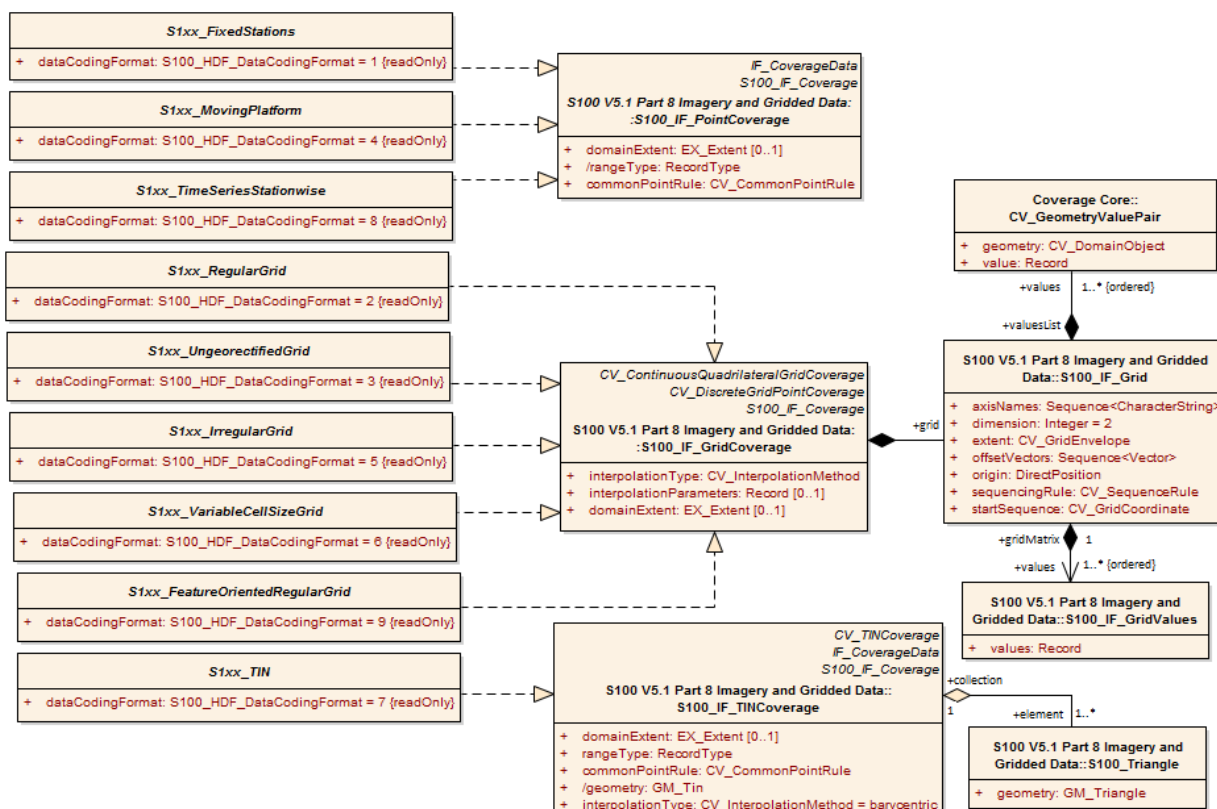


Рисунок 10с-13 - Реализации HDF5 кодирования типов покрытий из типов покрытий части 8

Рисунок 10с-13 можно расширить, включив в него производные верхнего уровня типов ISO за счет потенциального увеличения более сложного рисунка. На рисунке 10с-14 проиллюстрирована идея покрытия регулярной сеткой. Обратите внимание, что гипотетическая схема применения позволяет охватить как дискретные точки сетки, так и непрерывные типы сеток; если требуется только непрерывное покрытие сеткой, классы CV_DiscreteGridPointCoverage и CV_DiscreteCoverage могут быть опущены. Кроме того, поскольку гипотетическая прикладная схема использует только регулярные сетки, координаты точек сетки можно рассчитать от начала отсчета сетки и размеру ячеек, а класс CV_GeometryPair, связанный с S100_IF_Grid, не нужен и поэтому на рисунке 10с-14 был опущен.

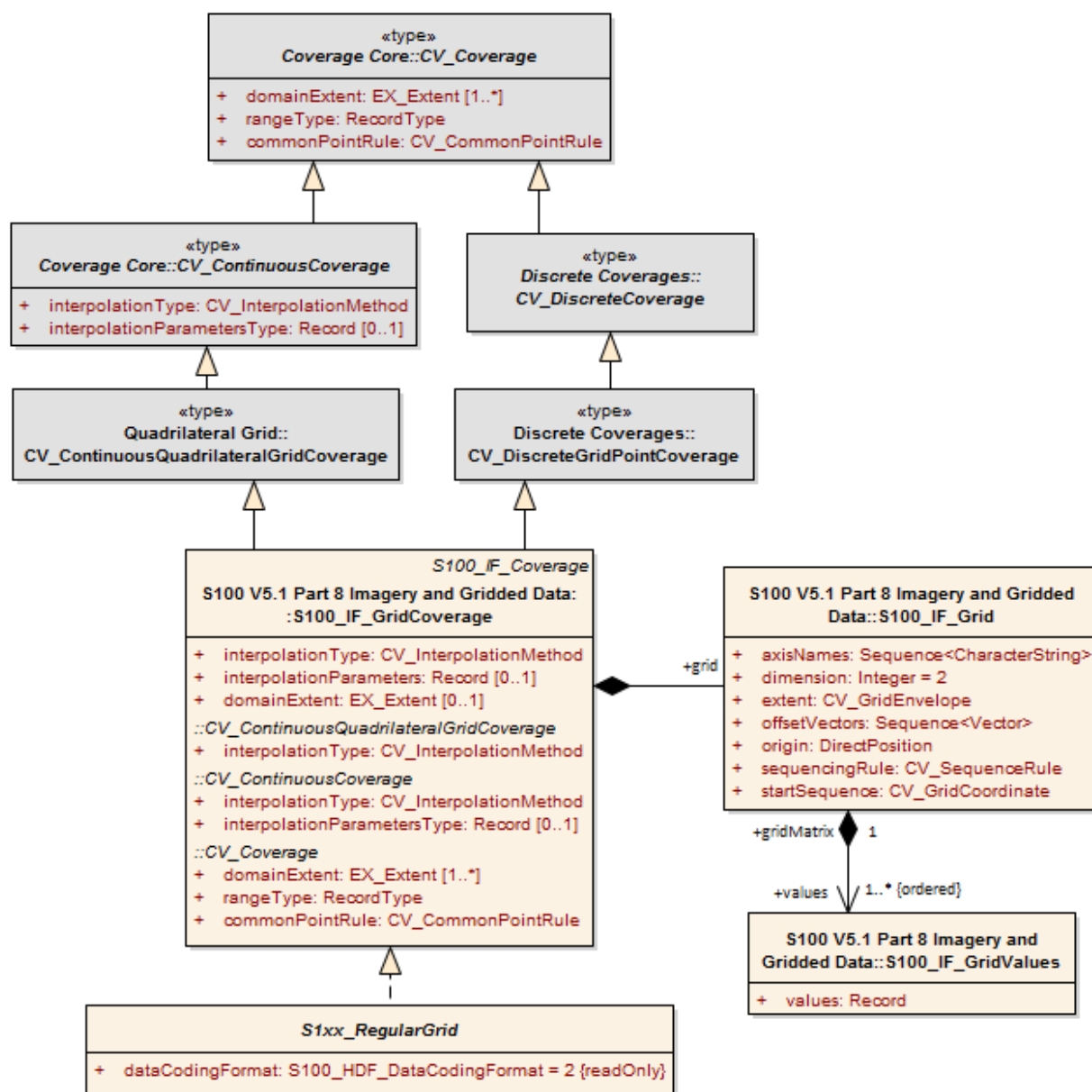


Рисунок 10с-14 – Фрагмент гипотетической прикладной схемы, включающей типы ISO верхнего уровня. Серые блоки являются классами ISO 19123.