OGC® DOCUMENT: 16-114R3

External identifier of this OGC® document: http://www.opengis.net/doc/BP/moving-features-ext-netCDF/1.0

Making location count.

MOVING FEATURES ENCODING EXTENSION: NETCDF

BEST PRACTICE
General

PUBLISHED

Version: 1.0
Submission Date: 2016-11-08
Approval Date: 2018-03-13
Publication Date: 2018-12-18
Editor: Martin Desruisseaux

Notice: This document defines an OGC Best Practice on a particular technology or approach related to an OGC standard. This document is *not* an OGC Standard and may not be referred to as an OGC Standard. It is subject to change without notice. However, this document is an *official* position of the OGC membership on this particular technology topic.
Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

License Agreement

Copyright notice

Note

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

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

CONTENTS

LIST OF FIGURES

I ABSTRACT

The netCDF Moving Features encoding extension is a summary of conventions that supports efficient exchange of simple moving features as binary files. This Best Practice is a complement to the *Moving Features Encoding Part I: XML Core* and an alternative to the *Simple Comma Separated Values (CSV)* extension. Compared to the CSV encoding, this netCDF encoding offers more compact storage and better performance at the cost of additional restrictions on the kinds of features that can be stored.

II KEYWORDS

The following are keywords to be used by search engines and document catalogues.

ogcdoc, OGC document, Best Practice, moving features, netCDF, CF convention, trajectory

III PREFACE

This document does not define any new Moving Features elements, rather it provides a method to encode discrete sampling geometries as netCDF files. This Best Practice combines two existing OGC specifications:

- The *Network Common Data Form (netCDF)* standard [OGC 10-092r3], which provides the binary encoding of array-oriented scientific data; and
- A subset of the *Climate and Forecast (CF)* conventions [OGC 11-165r2], which provides standard names and layout for the arrays to encode in a netCDF file.

This encoding is intended to provide a faster and more compact alternative to the Moving Features CSV encoding using a format readable by existing software.

IV SECURITY CONSIDERATIONS

No security considerations have been made for this document.

V SUBMITTING ORGANIZATIONS

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

- Central Research Laboratory, Hitachi Ltd.
- Geomatys

VI SUBMITTERS

All questions regarding this submission should be directed to the editor or the submitters:

Name	Affiliation
Martin Desruisseaux	Geomatys

VII REFERENCE NOTES

The OGC 11-165r2 CF-netCDF standard is based on Climate and Forecast (CF) conventions version 1.6. The Attribute Convention for Data Discovery (ACDD) adds additional attributes that provide useful metadata for Catalog Service on the Web (CSW). The National Centers for Environmental Information (NCEI) document is used as a guideline when an ambiguity needs to be resolved.

This document references OGC 11-165r2 when possible, but the on-line CF, ACDD, or NCEI documents may be referenced when additional information is considered relevant to moving feature binary encoding.

1

SCOPE

1 SCOPE

This OGC® Best Practice gives guidelines for encoding in netCDF files the geometries of features that move as rigid bodies. This is an alternative to the encoding defined by the *OGC Moving Features Encoding Extension: Simple Comma Separated Values (CSV)* standard. netCDF encoding can store moving features with the following characteristics:

- Each moving feature can be described with the Schema for Moving Features (ISO 19141:2008);
- The number of features simultaneously encoded with this format can be massive (many thousands of features);
- All features can be described using common space-time coordinates;
- Trajectories are described by a single point (not a line string) at each time step;
- Foliation order is sequential, i.e., trajectory elements that are parts of one track are ordered by time.

Deforming features such as floodwater and geometric complexes are out of scope.

This OGC document describes relevant attributes from the Climate and Forecast (CF) conventions in sufficient detail that client applications do not need to deal with the entire scope of CF conventions. Applications only need to understand a restricted subset of CF conventions in order to be able to interpret moving features in netCDF files.

2

CONFORMANCE

2 CONFORMANCE

Conformance with this specification shall be checked using all the relevant tests specified in Annex A (normative).

3

NORMATIVE REFERENCES

3 NORMATIVE REFERENCES

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Akinori Asahara, Ryosuke Shibasaki, Nobuhiro Ishimaru, David Burggraf: OGC 14-083r2, *OGC® Moving Features Encoding Part I: XML Core*. Open Geospatial Consortium (2015). http://www.opengis.net/doc/IS/movingfeatures/xmlcore/1.0.0.

Akinori Asahara, Ryosuke Shibasaki, Nobuhiro Ishimaru, David Burggraf: OGC 14-084r2, *OGC® Moving Features Encoding Extension: Simple Comma Separated Values (CSV)*. Open Geospatial Consortium (2015). http://www.opengis.net/doc/IS/movingfeatures/csv-extension/1.0.0.

Ben Domenico: OGC 10-090r3, *OGC Network Common Data Form (NetCDF) Core Encoding Standard version 1.0*. Open Geospatial Consortium (2011).

Ben Domenico: OGC 10-092r3, *NetCDF Binary Encoding Extension Standard: NetCDF Classic and 64-bit Offset Format*. Open Geospatial Consortium (2011).

Ben Domenico and Stefano Nativi: OGC 11-165r2, *CF-netCDF3 Data Model Extension standard*. Open Geospatial Consortium (2013). http://www.opengis.net/doc/is/netcdf-data-model-extension/1.0.0.

ISO: ISO 19101-1:2014, *Geographic information — Reference model — Part 1: Fundamentals*. International Organization for Standardization, Geneva (2014). https://www.iso.org/standard/59164.html.

ISO: ISO 19141:2008, *Geographic information — Schema for moving features*. International Organization for Standardization, Geneva (2008). https://www.iso.org/standard/41445.html.

ISO: ISO 19107:2003, *Geographic information — Spatial schema*. International Organization for Standardization, Geneva (2003). https://www.iso.org/standard/26012.html.

ISO: ISO 19123:2005, *Geographic information — Schema for coverage geometry and functions*. International Organization for Standardization, Geneva (2005). https://www.iso.org/standard/40121.html.

Lawrence Livermore National Laboratory: NetCDF CF Metadata Conventions. http://cfconventions.org/

ESIP: Attribute Convention for Data Discovery (ACDD). http://wiki.esipfed.org/index.php/Attribute_Convention_for_Data_Discovery

NOAA: NCEI netCDF Templates. http://www.nodc.noaa.gov/data/formats/netcdf/

4

TERMS AND DEFINITIONS

4 TERMS AND DEFINITIONS

This document uses the terms defined in OGC Policy Directive 49, which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this document and OGC documents do not use the equivalent phrases in the ISO/IEC Directives, Part 2.

This document also uses terms defined in the OGC Standard for Modular specifications (OGC 08-131r3), also known as the ‘ModSpec’. The definitions of terms such as standard, specification, requirement, and conformance test are provided in the ModSpec.

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

This document uses the terms defined in Sub-clause 5.3 of OGC 06-121r8, which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

4.1. Moving feature

Representation, using a local origin and local ordinate **vectors**, of a **geometric object** at a given reference time.

[**SOURCE:** ISO 19141:2008]

4.2. Vector

Quantity having direction as well as magnitude.

[**SOURCE:** ISO 19123:2005]

4.3. Geometric object

Spatial object representing a geometric set.

[**SOURCE:** ISO 19107:2003]

4.4. Geometric primitive

Geometric object representing a single, connected, homogeneous element of space.

[**SOURCE:** ISO 19107:2003]

4.5. Period

1-dimensional **geometric primitive** representing extent in time.

[**SOURCE:** ISO 19141:2008]

4.6. Instant

0-dimensional **geometric primitive** representing a position in time.

[**SOURCE:** ISO 19141:2008]

4.7. Point

0-dimensional **geometric primitive** representing a position.

[**SOURCE:** ISO 19107:2003]

4.8. Trajectory

Path of a moving **point** described by a one parameter set of points.

[**SOURCE:** ISO 19141:2008]

4.9. Foliation

One parameter set of geometries such that each point in the **prism** of the set is in one and only one **trajectory** and in one and only one **leaf**.

[**SOURCE:** ISO 19141:2008]

4.10. One parameter set of geometries

Function f from an interval $t \in [a, b\ \ ]$ such that $f(t)$ is a geometry and for each point $P \in f(a)$ there is a one parameter set of points (called the trajectory of P) $P(t):[a, b] \rightarrow P(t)$ such that $P(t) \in f(t)$.

[**SOURCE:** ISO 19141:2008]

4.11. Prism (one parameter set of geometries)

Set of points in the union of the geometries (or the union of the trajectories) of a one parameter set of geometries.

[**SOURCE:** ISO 19141:2008]

4.12. Leaf (one parameter set of geometries)

Geometry at a particular value of the parameter.

[**SOURCE:** ISO 19141:2008]

4.13. Dataset

An identifiable collection of data.

4.14. Attribute

NetCDF attributes hold metadata. An attribute contains information about properties of a variable or an entire **dataset**.

4.15. Global attribute

Global **attributes** apply to a whole **dataset** and may be used to record properties of all the data in a file, such as processing history or conventions used.

[**SOURCE:** OGC 10-090r3]

4.16. Variable attribute

Variable **attributes** record the properties of one **variable**.

[**SOURCE:** OGC 10-090r3]

4.17. Variable

A netCDF variable has a name, type, **shape**, **attributes**, and values. In face, much of the netCDF specification consists of defining the specific characteristics of variables. Variables hold data values. In the netCDF model, a variable can hold a multidimensional array of values of the same type.

[**SOURCE:** OGC 10-090r3]

4.18. Shape

The shape of a **variable** is specified with a list of zero or more **dimensions**.

[**SOURCE:** OGC 10-090r3]

4.19. Dimension

NetCDF dimensions are used to specify variable **shapes**, common grids, and coordinate systems.

[**SOURCE:** OGC 10-090r3]

6

CONVENTIONS

6 CONVENTIONS

This section provides details and examples for conventions used in the document.

6.1. Abbreviated terms

The following symbols and abbreviated terms are used in this Best Practice:

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
ACDD	Attribute Convention for Data Discovery
CF conventions	Climate and Forecast conventions
CRS	Coordinate Reference System
CSV	Comma Separated Values
EPSG	IOGP’s EPSG geodetic dataset
IOGP	International Association of Oil & Gas producers
ISO	International Organization for Standardization
NCEI	NOAA National Centers for Environmental Information
NetCDF	Network Common Data Form
OGC	Open Geospatial Consortium
WKT	Well Known Text

7

DATA MODEL

7 DATA MODEL

The data model used by the binary encoding is equivalent to that used by the CSV encoding. Figure 1 shows an example for trajectories of three moving points A , B and C . Each trajectory has the same start time and the same end time.

- At $t=0$ (start of all data), all points start moving.
- At $t=1$, the movement of A is changed.
- At $t=2$ (end of all data), the movement of A , B and C are changed.

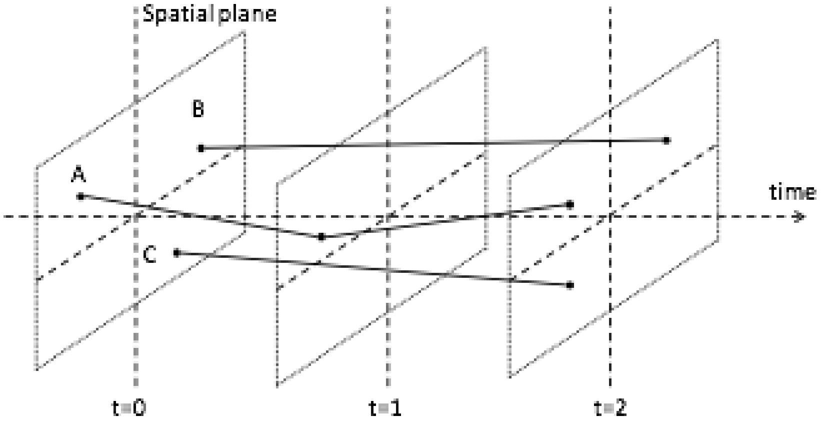


Figure 1 — example for three trajectories

In the case, the trajectory of A from $t=0$ to $t=1$ and from $t=1$ to $t=2$ is encoded, together with the trajectories of B and C from $t=0$ to $t=2$.

8

DATA STRUCTURE AND FILE FORMAT

8 DATA STRUCTURE AND FILE FORMAT

The binary encoding of arrays and attributes is defined by OGC 10-092r3 – *NetCDF Binary Encoding Extension Standard: netCDF Classic and 64-bit Offset Format*. That netCDF standard defines only a “container” format – it does not define the meaning of attributes stored in the file (the “schema” in XML terminology).

The full content of OGC 10-092r3 standard is needed for Moving Feature Simple Binary; this Best Practice does not propose any addition or subset of OGC 10-092r3.

REQUIREMENT 1
http://www.opengis.net/spec/mf_binary/1.0/req/netcdf_valid
All Moving Features binary files *shall* be compliant with the netCDF binary encoding as specified by the OGC 10-092r3 standard.

Meanings of attributes in a netCDF file are defined by a separated standard, OGC 11-165r2 – *CF-netCDF3 Data Model Extension*. All remaining requirements and recommendations in this Moving Feature Simple Binary document apply to a subset of those CF conventions.

8.1. NetCDF global attributes

Global attributes apply to the netCDF file as a whole. They specify a succinct description of what is in the dataset, the kind of sampling geometry, the geographic bounding box, the vertical extent, the time period, the coordinate reference systems used and many other metadata (see CF convention). Some global attributes are merely a convenience for information that may otherwise be tedious to compute; those attributes are presented in Clause 8.1.2.

8.1.1. Non-redundant global attributes

This sub-section presents some global attributes that do not duplicate information provided by other attributes.

8.1.1.1. Identification of conventions

To identify that the file uses the CF convention, OGC 11-162r2 §7.3.2.1 requires that the `Conventions` global attribute is set to the `"CF-1.6"` string value. This Best Practice uses also some attribute convention for data discovery, which is notified by the `"ACDD-1.3"` string value.

REQUIREMENT 2 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/conventions

The file *shall* define the global attribute `Conventions` to a string of comma-separated values containing at least the `"CF-1.6"` element. The list of values should contain also the `"ACDD-1.3"` element. A complete string value containing the two elements is `"CF-1.6, ACDD-1.3"`.

8.1.1.2. Feature type

The CF conventions allow different kinds of sampling geometry such as point, time series, profile, or trajectory. This binary encoding Best Practice focuses on trajectories. This choice constraint the data layout to a series of data points along a path through space with monotonically increasing times (OGC 11-165r2 §7.5.2.3).

REQUIREMENT 3 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/featureType

The file *shall* define the global attribute `featureType` to the string value `"trajectory"`.

8.1.1.3. Title

Both CF convention and ACDD recommend the inclusion of a title in the global attributes.

RECOMMENDATION 1 (FROM CF AND ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/title

The file should define the global attribute `title` to a succinct description of what is in the dataset.

8.1.2. Convenience global attributes for data discovery

This sub-section presents global attributes that duplicate information available in other attributes, but in a more convenient manner. For example, the *geographic bounding box* attributes (Clause 8.1.2.1) duplicate information provided by the *geospatial bounds* attributes (Clause 8.1.2.2 to Clause 8.1.2.4), but may avoid the need to apply an inverse map projection. The geospatial bounds attributes themselves duplicate attributes associated to coordinate variables. But the attributes described in this sub-section make it easier to discover the data. Indeed, all attributes in this sub-section come from the Attribute Convention for Data Discovery

(ACDD). By contrast, most attributes in other sections come from the Climate and Forecast (CF) convention.

8.1.2.1. Geographic bounding box

The geographic bounding box is given in longitude and latitude regardless of the Coordinate Reference System (CRS) used by the trajectories. Since those metadata are used for discovery purpose only, the geodetic datum is unspecified. Coordinate precision may be of only two decimal places of a degree.

RECOMMENDATION 2 (FROM ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/geographicBoundingBox

The file should define the following global attributes:

- `geospatial_lat_min`
- `geospatial_lat_max`
- `geospatial_lon_min`
- `geospatial_lon_max`

Respective values are the southernmost and northernmost latitudes, followed by the easternmost and westernmost longitudes covered by the dataset. Units of measurement should be degrees with positive latitudes in the North hemisphere and longitude values increasing toward east. The “minimal” longitude may be greater than the “maximal” longitude if the bounding box crosses the discontinuity meridian. The Prime Meridian should be Greenwich. Values type should be floating point.

This *geographic* bounding box is redundant with the *geospatial* bounds defined below. The geographic bounding box is optional metadata for easier discovery of a dataset without the need to parse Well Known Text, lookup EPSG codes, and reproject envelopes.

8.1.2.2. Geospatial (not necessarily geographic) bounds

The horizontal part of the spatial boundary of moving features is specified as geometry in Well Known Text (WKT) format. The Coordinate Reference System is specified in Clause 8.1.2.5.

This metadata is equivalent to the `mf:STBoundedBy` element defined by Moving Features Core and to the `@stboundedby` element defined by the CSV extension, except that the geometry is not an envelope (but may be an equivalent polygon) and does not contain vertical or temporal components.

RECOMMENDATION 3 (FROM ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/spatialBounds

The file should define the global attribute `geospatial_bounds` to a Well Known Text (WKT) that specifies the horizontal part of a geometry encompassing all moving features in the dataset. The geometry may be a polygon. Meaning and order of values for each point's coordinates depend on the Coordinate Reference System (Clause 8.1.2.5).

8.1.2.3. Vertical bounds

If the feature trajectories are three-dimensional, then the vertical part of the spatial boundary of moving features is specified as a range of floating-point values. The Coordinate Reference System is specified in Clause 8.1.2.5.

This metadata is equivalent to the vertical component of `mf:STBoundedBy` (Moving Features Core) or `@stboundedby` (CSV extension).

RECOMMENDATION 4 (FROM ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/verticalBounds

If the moving features feature trajectories are three-dimensional, then the file should define the global attribute `geospatial_vertical_min` and `geospatial_vertical_max` to the numerically smaller and larger vertical limit, respectively. Values type should be floating point. Meaning of values depends on the Coordinate Reference System (Clause 8.1.2.5).

8.1.2.4. Temporal bounds

The times of the first and last data point in the file are specified as a range of dates. Dates are formatted as specified by ISO 8601:2004, preferably the extended date-time format (*YYYY-MM-DD* T *hh:mm:ss* optionally followed by the time zone).

RECOMMENDATION 5 (FROM ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/temporalBounds

The file should define the global attribute `time_coverage_start` to the time of the first data point, and `time_coverage_end` to the time of the last data point. Dates are encoded as strings in ISO 8701:2004 format.

8.1.2.5. Spatial Reference System

The Coordinate Reference System (CRS) of the geospatial and vertical bounds (Clause 8.1.2.2 and Clause 8.1.2.3) is specified by an authority code, preferably from the EPSG geodetic dataset. For two-dimensional trajectories, only one CRS should be specified. For three-dimensional trajectories, a single three-dimensional CRS or two distinct CRS (one horizontal and one vertical) may be specified depending on the nature of the vertical heights.

While ACDD uses simple strings of the form `"EPSG:4326"`, this document recommends URNs of the form `"urn:ogc:def:crs:EPSG::4326"` instead. However, software should be prepared to read both forms. Note that axis order is the same with both forms, namely (*latitude*, *longitude*) in EPSG:4326 case.

This metadata is equivalent to the `srsName` attribute in Moving Feature core and to the `srid` attribute in CSV extension, except for the possible separation between a horizontal and a vertical CRS.

RECOMMENDATION 6 (FROM ACDD)

http://www.opengis.net/spec/mf_binary/1.0/req/boundsCRS

The file should define the global attribute `geospatial_bounds_crs` and *may* define the global attribute `geospatial_bounds_vertical_crs` to authority codes (preferable from the EPSG geodetic dataset) determined according the following choices:

- If the trajectories are three-dimensional **and** horizontal coordinates are geodetic (geographic or geocentric) latitudes and longitudes **and** the vertical coordinates are *heights above the ellipsoid*, then:
  - `geospatial_bounds_vertical_crs` attribute is not present.
  - `geospatial_bounds_crs` is the authority code of a single three-dimensional geodetic CRS used by both geospatial_bounds (for the horizontal part) and by `geospatial_vertical_min` and `geospatial_vertical_max` (for the vertical part) attribute values. For example EPSG:4979 is the three-dimensional variant of EPSG:4326.
- Otherwise:
  - `geospatial_bounds_crs` is the authority code of the two-dimensional CRS used by the `geospatial_bounds` attribute value.
  - `geospatial_bounds_vertical_crs` is the authority code of the CRS used by `geospatial_vertical_min` and `geospatial_vertical_max`, if present.

8.2. Trajectory lines (data body)

The CF convention proposes four different ways to organize the feature coordinates and attributes in a netCDF file. This best-practice paper chooses the *Contiguous Ragged Array* representation, which has the following characteristics:

- Can mix short and large features without waste of space;
- Number of points of each feature must be known in advance;
- File can be updated with new features or new points in the last feature (but not new points in previous features); and
- Foliation order is restricted to “sequential”.

The file is organized by first defining netCDF dimensions (not to be confused with spatio-temporal dimensions), then variables. The dimension and variable names shall comply with the restrictions documented in OGC 11-162r2 §7.3.2.1:

REQUIREMENT 4 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/names

NetCDF dimension names and netCDF variable names *shall* comply with the following restrictions:

- Names shall begin with a letter;
- Names shall be composed of letters, digits or underscores; and
- No variable has the same name than a dimension, except the variable holding feature identifiers (Clause 8.2.2.1).

8.2.1. NetCDF dimensions

The ragged array representation needs three netCDF dimensions, described below.

8.2.1.1. Maximal length of feature identifiers

If each feature is identified by a string value, then the netCDF file needs to declare the maximal number of characters allowed in those identifiers. Identifiers shorter than the maximal length will be padded by spaces of null characters (readers must be prepared to handle both).

REQUIREMENT 5 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/identifierLength

If features are identified by string values, then the file *shall* define a netCDF dimension for identifier characters (the *identifier char dimension*). The name of this dimension can be any name compliant with the requirements in Clause 8.2. The length of this dimension is the maximum number of characters than can be stored in a feature identifier.

8.2.1.2. Feature instance dimension

The netCDF file shall declare a dimension for information about each feature as a whole (i.e., information that does not depend on the time). The length of this dimension is the maximal number of features that can be stored in the file. It is acceptable to declare a length larger than needed in order to reserve room for future feature additions, provided that values in the count variable (Clause 8.2.2.2) are set to zero for all missing features.

REQUIREMENT 6 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/instanceDimension

The file *shall* define a netCDF dimension for feature instances (the *instance dimension*). The name of this dimension can be any name compliant with the requirements in Clause 8.2. The length of this dimension is the maximum number of features that can be stored in the file.

8.2.1.3. Sample dimension

The netCF file shall declare a dimension for the actual data (time, geospatial coordinates and feature attributes). This dimension could have a fixed length, but it is more convenient to declare this dimension length as unlimited if new data need to be appended. Note that a netCDF file can have only one dimension of unlimited length.

REQUIREMENT 7 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/sampleDimension

The file shall define a netCDF dimension for time-dependent data (the *sample dimension*). The name of this dimension can be any name compliant with the requirements in Clause 8.2. The length of this dimension should be unlimited, unless the total number of points is known in advance.

8.2.2. NetCDF variables

In the continuous ragged array representation, the netCDF file shall contain the following variables:

- One variable listing feature identifiers;
- One variable counting the number of points in each feature;
- Three or four *auxiliary* coordinate variables (not to be confused with “simple” coordinate variables): examples: x , y , (z) and t ; and
- An arbitrary number of variables for *feature* attributes (not to be confused with *global* attributes or *variable* attributes): examples: *state*, *temperature*.

8.2.2.1. Feature identifiers

Each trajectory should have identifying text that specifies the moving feature. The text can be a person ID, a vehicle ID, etc. While the identifier is usually a string, integer types are also allowed.

The data stored in this variable are equivalent to the `mfIdRef` attribute value in Moving Feature XML file, or to the values in the `mfidref` column in a Moving Feature CSV file.

REQUIREMENT 8 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/identifiers

The file *shall* define a variable holding the identifier of each trajectory.

- The variable name shall be identical to the name of the *instance dimension* defined in Clause 8.2.1.2.
- The variable type should be the character type. Integer types are also valid.
- The variable dimension shall be the *instance dimension* (Clause 8.2.1.2) and the *identifier char dimension* (Clause 8.2.1.1), in that order. The char dimension shall be omitted if the variable type is not a character type.
- The collection of variable attributes shall contain:
  - A `cf_role` attribute set to the `"trajectory_id"` string value.

8.2.2.2. Count variable

The netCDF file shall contain a variable holding the number of points in each trajectory. The length of this variable is the maximum number of features that the netCDF file can contain.

REQUIREMENT 9 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/count

The file *shall* define a variable holding the count of the number of points in each trajectory (the *count variable*).

- The variable name can be any name compliant with the requirements in Clause 8.2.
- The variable type shall be an integer.
- The variable dimension shall be the *instance dimension* defined in Clause 8.2.1.2.
- The collection of variable attributes shall contain:
  - A `sample_dimension` attribute set to the name of the *sample dimension* defined in Clause 8.2.1.3.

8.2.2.3. Auxiliary coordinate variables

Trajectory coordinates are specified in one variable for each spatiotemporal dimension. Those “*auxiliary* coordinate variables” are not subject to the usual restrictions of netCDF “coordinate variables”. In particular:

- The variable name does not match the dimension name;
- The values do not need to be ordered monotonically;
- The variable does not have `axis` attribute; and
- The variable may have missing values.

Ordinate values of the first feature – i.e., the feature at index 0 in the *feature instance dimension* (Clause 8.2.1.2) – are written first. The number of values to write for that first feature is given by the value of the *count variable* (Clause 8.2.2.2) at index 0. Then the ordinate values of the second feature – i.e., the feature at index 1 in the *feature instance dimension* – follow. The number of values to write for that second feature is given by the value of the *count variable* at index 1, and so on.

For example, if a file contains three features identified as “A”, “B” and “C” and if their trajectories are described by the following points:

- A: start at (11, 2) then move to (12, 3) and finally (10, 3);
- B: start at (10, 2) then move to (11, 3); and
- C: start at (12, 1) then move to (10, 2) and finally (11, 3).

Then the feature instance dimension has a length of 3 and the sample dimension (if not unlimited) has a length of 3 + 2 + 3 = 8. In such cases the variables presented in previous sub-sections have the following values:

NetCDF variables having the *feature instance dimension* (Clause 8.2.1.2):

identifier (Clause 8.2.2.1)	*count* (Clause 8.2.2.2)
A	3
B	2
C	3

The auxiliary coordinate variables can be as below:

NetCDF variables having the *sample dimension* (Clause 8.2.1.3):

	TIME	*x*	*y*
Feature A time 1	8:00	11	2
Feature A time 2	8:10	12	3
Feature A time 3	8:20	10	3
Feature B time 1	8:05	10	2
Feature B time 2	8:15	11	3

Feature C time 1	7:50	12	1
Feature C time 2	8:00	10	2
Feature C time 3	8:10	11	3

REQUIREMENT 10 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/coordinates

The file *shall* contain three or four auxiliary coordinate variables.

- The variable name can be any name compliant with the requirements in Clause 8.2.
- The variable type shall be a numerical type (usually floating point).
- The variable dimension shall be the *sample dimension* defined in Clause 8.1.2.3.
- The collection of attributes of the time coordinates variable shall contain:
  - A `standard_name` attribute set to the `"time"` string value.
  - A `units` attribute set to the “*units* `since` *YYYY-MM-DD hh:mm:ss*” string value where:
    - “*units*” can be “days”, “hours”, “minutes” or “seconds”,
    - “*YYYY-MM-DD hh:mm:ss*” is the epoch in UTC.
- The collection of attributes of spatial coordinates variables shall contain:
  - A `standard_name` attribute if and only if a standard name exists in the CF Standard Name Table (http://cfconventions.org/standard-names.html) for that variable. Examples of standard names are `"longitude"`, `"latitude"` and `"altitude"`.
  - A `units` attribute. Examples are `"degrees_east"`, `"degrees_north"` and `"km"`.
- The collection of attributes of all coordinates variable shall contain:
  - An `axis` attribute with one of the following names: `"X"`, `"Y"`, `"Z"` or `"T"`.

In addition to the attributes listed in above requirement, some non-standards attributes may help parsers to identify the coordinate system. In particular, the `_CoordinateAxisType` attribute may be set to `"Lon"`, `"Lat"`, `"Height"` or `"Time"` values for a geographic CRS, or to the `"GeoX"`, `"GeoY"`, `"GeoZ"` or `"Time"` values for other kind of CRS.

8.2.2.4. Feature attribute variables

If the file contains additional attributes for the moving features, they can be declared in array having the same length than the auxiliary coordinate variables. Elements in those variables are mapped to features and time in the same way than the values in auxiliary coordinate variables (Clause 8.2.2.3).

REQUIREMENT 11 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/featureAttributes

If the file contains additional feature attributes, then those values *shall* be stored in one distinct variable for each feature attribute. The dimension of those variables is the *sample dimension*. Values at index i are associated to the same feature at the same time than the values of auxiliary coordinate variables at index i .

OGC 11-162r2 has additional requirements for the exchange of scientific data.

REQUIREMENT 12 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/standardName

If the variable represents one of the physical quantities for which a standard name exists in the CF Standard Name Table (http://cfconventions.org/standard-names.html), then the variable *shall* define a `standard_name` attribute to the string value given in that table. Otherwise the variable *shall* define a `long_name` attribute to a long descriptive name that may, for example, be used for labeling plots.

REQUIREMENT 13 (FROM OGC 11-162R2)

http://www.opengis.net/spec/mf_binary/1.0/req/units

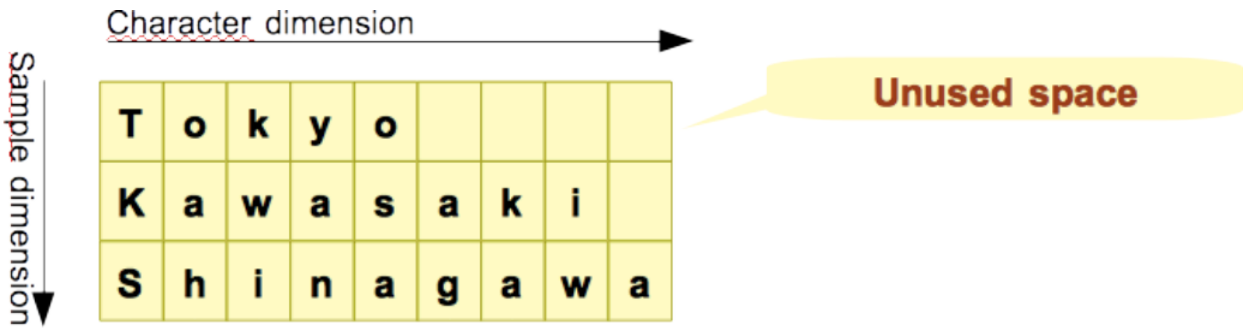
If the variable represents a dimensional quantity, then it *shall* define a `units` attribute to a string value that can be recognized by UNIDATA’s udunits package, with the addition of `"level"`, `"layer"` and `"sigma_level"` string values. That unit shall be consistent with the unit given in the CF Standard Name Table for the `standard_name` attribute value, if the later attribute exists for this variable.

8.2.2.5. Feature attribute variables as character strings

If the values of a feature attribute variable are character strings, then there is a choice:

1. If only a limited amount of distinct values are used (e.g., values are members of an enumeration), then those values can be encoded as flags as described in CF conventions §3.5 (details below);

2. Otherwise a new dimension need to be defined with a length equal to the maximal number of characters to be stored in the feature attribute variable. This dimension is added as the last dimensions of the feature attribute variable, as defined in CF-convention §2.2. This dimension is shown as the “character dimension” in the figure below.



T	o	k	y	o				
K	a	w	a	s	a	k	i	
S	h	i	n	a	g	a	w	a

The second approach may result in a netCDF file as big as or bigger than an equivalent CSV file since all character strings shorter than the maximal length are padded with spaces or zero values. For efficiency reasons, this paper recommends the use of flags.

RECOMMENDATION 7 (FROM CF-CONVENTION)

http://www.opengis.net/spec/mf_binary/1.0/req/strings

Feature attribute variables of type character strings should be encoded as flags if the amount of distinct values is reasonably small.

- Variable type should be `byte` or `short`.
- The collection of variable attributes should contain:
  - A `flag_values` attribute of the same type (`byte` or `short`) than the variable. The attribute value is the list of distinct numerical values that can appear in the variable.
  - A `flag_meanings` attribute of the type string. The attribute value is a space-separated list of tokens (a token is a word or a phrase in which space characters have been replaced by underscore characters). The first token is the string representation of the first numerical value listed in the `flag_values` array, the second token is the string representation of the second numerical value listed in the `flag_values` array, *etc.*

A

ANNEX A (NORMATIVE) CONFORMANCE TEST

ANNEX A (NORMATIVE) CONFORMANCE TEST

Moving Features encoded in a netCDF file may be tested with the UCAR library, for example using the Java code below (replace “myFile.nc” and “myAttribute” by character strings applicable to the data to be tested). As of version 4.6.4 of UCAR netCDF library, this test requires the non-standard `_CoordinateAxisType` attribute on the auxiliary coordinate variables (see the note after requirement 10). For that reason, this test is non-normative.

```
import java.util.Formatter;
import java.io.IOException;
import ucar.nc2.constants.FeatureType;
import ucar.nc2.ft.FeatureCollection;
import ucar.nc2.ft.FeatureDataset;
import ucar.nc2.ft.FeatureDatasetFactoryManager;
import ucar.nc2.ft.FeatureDatasetPoint;
import ucar.nc2.ft.PointFeature;
import ucar.nc2.ft.PointFeatureIterator;
import ucar.nc2.ft.TrajectoryFeature;
import ucar.nc2.ft.TrajectoryFeatureCollection;
import ucar.unidata.geoloc.EarthLocation;

public class TestMovingFeatureNetCDF {
    public static void main(String[] args) throws IOException {
        try (FeatureDatasetPoint fp = (FeatureDatasetPoint)
FeatureDatasetFactoryManager
            .open(FeatureType.TRAJECTORY, “myFile.nc”, null, new
Formatter(System.out)))
        {
            for (FeatureCollection fc : fp.getPointFeatureCollectionList()) {
                TrajectoryFeatureCollection tfc =
(TrajectoryFeatureCollection) fc;
                while (tfc.hasNext()) {
                    TrajectoryFeature tf = tfc.next();
                    PointFeatureIterator it = tf.getPointFeatureIterator(-1);
                    while (it.hasNext()) {
                        PointFeature f = it.next();
                        EarthLocation loc = f. getLocation();
                        double time = f. getObservationTime();
                        Object value =
f.getDataAll().getScalarObject(“myAttribute”);
                        System.out.println(loc + " at " + time + " = " +
value);
                    }
                    it.finish();
                }
            }
        }
    }
}
```

B

ANNEX B (INFORMATIVE) REVISION HISTORY

B

ANNEX B (INFORMATIVE) REVISION HISTORY

DATE	RELEASE	AUTHOR	PARAGRAPH MODIFIED	DESCRIPTION
2016-06-16	r1	Martin Desruisseaux	All	First draft
2016-09-19	r2	Martin Desruisseaux	§7.2.2, annex A	Clarification
2016-12-12	r2	Dorian Ginane	§i, Annex A	Discussion Paper
2018-09-05	R3	Scott Simmons	All	Best Practice