ecCodes: Advanced Topics Part II

Shahram Najm
Development Section
Forecast Department

Overview

- Parameter database
- GRIB1 to GRIB2 conversion
- Local configuration

GRIB 1

“10 metre U component of wind”

indicatorOfParameter = 33 [u-component of wind (m/s)]
table2Version = 3
indicatorOfTypeOfLevel = 105 [Specified height level above ground (m)]
level = 10

GRIB 2

“10 metre U component of wind”

discipline = 0 [Meteorological products]

parameterCategory = 2 [Momentum]

parameterNumber = 0 [u component of wind (m s-1)]

typeOfFirstFixedSurface = 103 [Specified height level above ground (m)]

scaleFactorOfFirstFixedSurface = 0

scaledValueOfFirstFixedSurface = 10

typeOfSecondFixedSurface = 255 [Missing]

scaleFactorOfSecondFixedSurface = MISSING

scaledValueOfSecondFixedSurface = MISSING

GRIB 1 ECMWF local coding

“10 metre U component of wind”

indicatorOfParameter = 165 [10 metre u-component of wind (m/s)]
table2Version = 128

ecCodes solution

- Use a VIRTUAL (computed) key to decouple user level from coding level
- User code gets/sets a virtual key and ecCodes gets/sets the appropriate coded keys
- Local configuration is available to deal with local codes

ecCodes parameter

- ecCodes provides some edition independent keys to identify a parameter :
  - paramId
  - shortName
  - name
  - units
  - centre

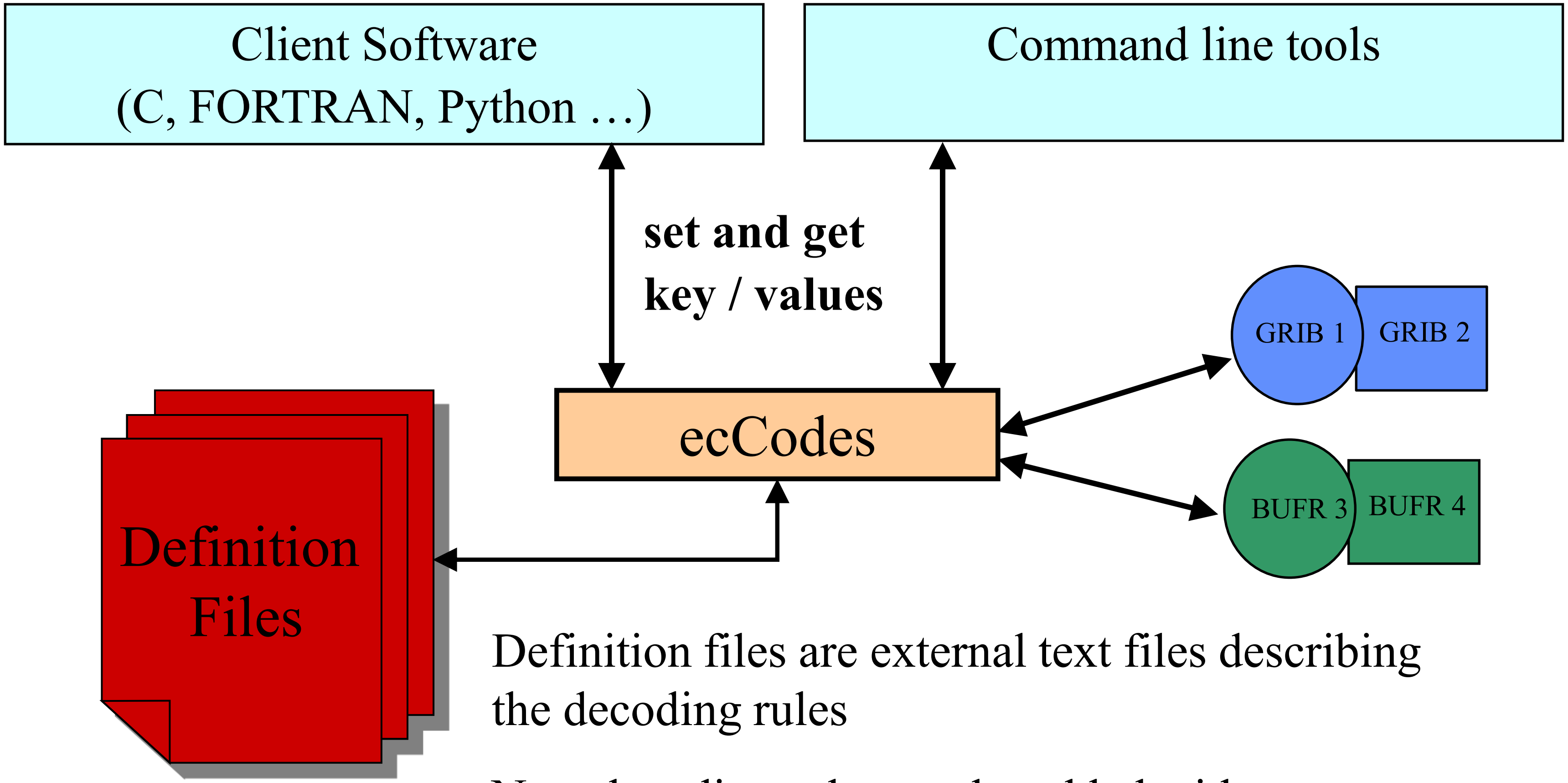
ecCodes parameters (GRIB 1)

- centre
- table2Version
- indicatorOfParameter
- levelType
- level
- …

ecCodes parameters (GRIB 2)

- discipline
- parameterCategory
- parameterNumber
- typeOfFirstFixedSurface
- scaleFactorOfFirstFixedSurface
- scaledValueOfFirstFixedSurface
- typeOfSecondFixedSurface
- scaleFactorOfSecondFixedSurface
- scaledValueOfSecondFixedSurface
- productDefinitionTemplateNumber
- …

ecCodes - Design



Definition files are external text files describing the decoding rules

New decoding rules can be added without rebuilding the ecCodes

Concept: shortName

GRIB 2

‘2t’ = { discipline = 0 ; productDefinitionTemplateNumber = 8 ; parameterCategory = 0 ; parameterNumber = 0 ; typeOfFirstFixedSurface = 103 ; scaleFactorOfFirstFixedSurface = 0; scaledValueOfFirstFixedSurface = 2; typeOfSecondFixedSurface = 255 ; scaleFactorOfSecondFixedSurface = missing(); scaledValueOfSecondFixedSurface = missing(); }

GRIB1

‘2t’ = { indicatorOfParameter=11; table2Version=3; levelType=103; level=2;}

GRIB1 ECMWF local coding

‘2t’ = { indicatorOfParameter=167; table2Version=128;}

Concept: paramId

GRIB 2

```
167 = { discipline = 0 ;  productDefinitionTemplateNumber = 8 ;
    parameterCategory = 0 ; parameterNumber = 0 ;   typeOfFirstFixedSurface =
    103 ; scaleFactorOfFirstFixedSurface = 0;  scaledValueOfFirstFixedSurface =
    2;  typeOfSecondFixedSurface = 255 ;  scaleFactorOfSecondFixedSurface =
    missing();  scaledValueOfSecondFixedSurface = missing(); }
```

GRIB1

```
167 = { indicatorOfParameter=11; table2Version=3;
          levelType=103; level=2;}
```

GRIB1 ECMWF local coding

```
167 = { indicatorOfParameter=167; table2Version=128;}
```

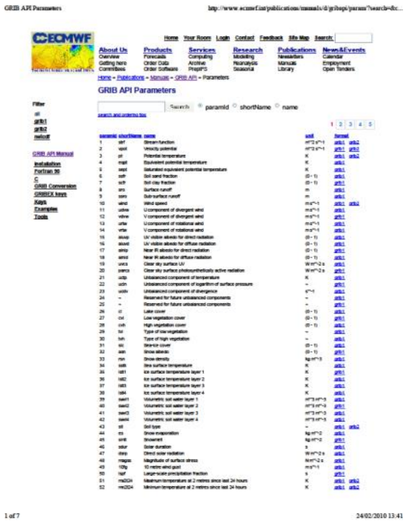
GRIB Parameters in ecCodes

- The Parameters database is accessible here:

  http://apps.ecmwf.int/codes/grib/param-db/

Parameters database

WEB



GRIB API Parameters http://www.ecmwf.int/publications/manuals/d/gribapi/param/?search=&c...

Home Your Room Login Contact Feedback Site Map Search:

About Us Overview, Getting here, Committees

Products Forecasts, Order Data, Order Software

Services Computing, Archive, PrepIFS

Research Modelling, Reanalysis, Seasonal

Publications Newsletters, Manuals, Library

News&Events Calendar, Employment, Open Tenders

Home » Publications » Manuals » GRIB API » Parameters

GRIB API Parameters

Filter
- all
- grib1
- grib2
- netcdf

GRIB API Manual
- Installation
- Fortran 90
- C
- GRIB Conversion
- GRIBEX keys
- Keys
- Examples
- Tools

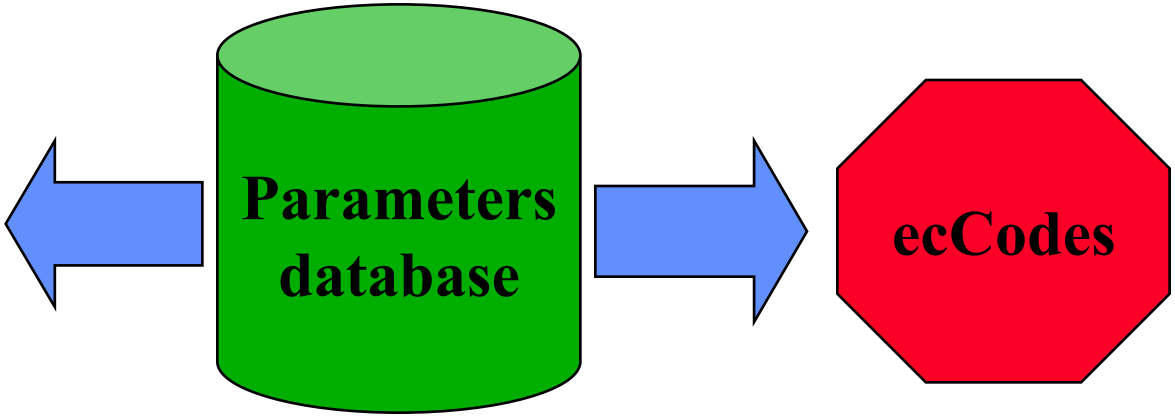
Search ● paramId ○ shortName ○ name

search and ordering tips

1 2 3 4 5

paramId	shortName	name	unit	format
1	strf	Stream function	m^2 s^-1	grib1 grib2
2	vpot	Velocity potential	m^2 s^-1	grib1 grib2
3	pt	Potential temperature	K	grib1 grib2
4	eqpt	Equivalent potential temperature	K	grib1
5	sept	Saturated equivalent potential temperature	K	grib1
6	ssfr	Soil sand fraction	(0 - 1)	grib1
7	scfr	Soil clay fraction	(0 - 1)	grib1
8	sro	Surface runoff	m	grib1
9	ssro	Sub-surface runoff	m	grib1
10	wind	Wind speed	m s^-1	grib1 grib2
11	udvw	U component of divergent wind	m s^-1	grib1
12	vdvw	V component of divergent wind	m s^-1	grib1
13	urtw	U component of rotational wind	m s^-1	grib1
14	vrtw	V component of rotational wind	m s^-1	grib1
15	aluvp	UV visible albedo for direct radiation	(0 - 1)	grib1
16	aluvd	UV visible albedo for diffuse radiation	(0 - 1)	grib1
17	alnip	Near IR albedo for direct radiation	(0 - 1)	grib1
18	alnid	Near IR albedo for diffuse radiation	(0 - 1)	grib1
19	uvcs	Clear sky surface UV	W m^-2 s	grib1
20	parcs	Clear sky surface photosynthetically active radiation	W m^-2 s	grib1
21	uctp	Unbalanced component of temperature	K	grib1
22	ucln	Unbalanced component of logarithm of surface pressure	~	grib1
23	ucdv	Unbalanced component of divergence	s^-1	grib1
24	~	Reserved for future unbalanced components	~	grib1
25	~	Reserved for future unbalanced components	~	grib1
26	cl	Lake cover	(0 - 1)	grib1
27	cvl	Low vegetation cover	(0 - 1)	grib1
28	cvh	High vegetation cover	(0 - 1)	grib1
29	tvl	Type of low vegetation	~	grib1
30	tvh	Type of high vegetation	~	grib1
31	ci	Sea-ice cover	(0 - 1)	grib1
32	asn	Snow albedo	(0 - 1)	grib1
33	rsn	Snow density	kg m^-3	grib1
34	sstk	Sea surface temperature	K	grib1
35	istl1	Ice surface temperature layer 1	K	grib1
36	istl2	Ice surface temperature layer 2	K	grib1
37	istl3	Ice surface temperature layer 3	K	grib1
38	istl4	Ice surface temperature layer 4	K	grib1
39	swvl1	Volumetric soil water layer 1	m^3 m^-3	grib1
40	swvl2	Volumetric soil water layer 2	m^3 m^-3	grib1
41	swvl3	Volumetric soil water layer 3	m^3 m^-3	grib1
42	swvl4	Volumetric soil water layer 4	m^3 m^-3	grib1
43	slt	Soil type	~	grib1 grib2
44	es	Snow evaporation	kg m^-2	grib1
45	smlt	Snowmelt	kg m^-2	grib1
46	sdur	Solar duration	s	grib1
47	dsrp	Direct solar radiation	W m^-2 s	grib1
48	magss	Magnitude of surface stress	N m^-2 s	grib1
49	10fg	10 metre wind gust	m s^-1	grib1
50	lspf	Large-scale precipitation fraction	s	grib1
51	mx2t24	Maximum temperature at 2 metres since last 24 hours	K	grib1 grib2
52	mn2t24	Minimum temperature at 2 metres since last 24 hours	K	grib1 grib2

1 of 7 24/02/2010 13:41



Parameters: Practicals

To get the practicals:

```
tar xvf  ~trx/ecCodes/grib_parameters.tar
```

1. You have two grib messages start.grib1 and start.grib2
2. Create the file 10u.grib1 setting shortName=10u in start.grib1
3. Create the file 10u.grib2 setting shortName=10u in start.grib2
4. Do `grib_ls -n parameter 10u.grib1 10u.grib2`. Do you see any difference
5. Compare the grib_dump –O of the two files and of the two messages in each file

GRIB1 to GRIB2 conversion

```
grib_set -s edition=2 in.grib1 out.grib2
```

conversion of

- time
- geography
- vertical
- parameter
- local
- data

GRIB1 to GRIB2 conversion

- Parameter conversion is particularly complex due to the difference between the two coding standards and the local tables used by some meteorological centres
- The conversion is based on the parameter’s unique identifier “paramId”

paramId based conversion

- How to produce a GRIB for a "2 metre temperature"

```
grib_set -s paramId=165 in.grib1 out.grib1
grib_set -s paramId=165 in.grib2 out.grib2
```

- How to convert a GRIB1 to GRIB2

```
grib_set -s edition=2 in.grib1 out.grib2
```

- During the conversion to edition 2 ecCodes copies the paramId value from the GRIB1 to the GRIB2:
  1. get paramId(=165) from GRIB1
  2. change edition to 2 producing a GRIB2
  3. set paramId(=165) in GRIB2

paramId based conversion

- The conversion is possible only if a paramId is defined for both editions

- Check on the parameters database website if a conversion is possible

Parameters: Practicals

We refer to the same files produced in the previous practical

1. **Convert 10u.grib1 to its GRIB2 version 10u_converted.grib2.**
2. **Do** `grib_ls -n parameter 10u.grib2`
3. **Do** `grib_ls –n parameter 10u_converted.grib2`
4. **Take the first message from start.grib1 and save it to ecmf.grib1**
5. **Set the paramId of ecmf.grib1 to 162089. Save it as ecmf.162089.grib1**
6. **Convert ecmf.162089.grib1 to GRIB edition 2. Why does it fail?**

Local configuration

- The external text files defining the decoding rules used by the decoding engine are called definition files
- For each installation there is a default set of definition files
- The ECCODES_DEFINITION_PATH environment variable can be set to use local definition files instead of the definition files provided within the distribution

Local configuration

- **For GRIB, the parameter descriptions for a given “centre” are contained in the files shortName.def, paramId.def, units.def, name.def in the directories**

  BASE_DIR/definitions/grib1/localConcepts/[centre:s]

  BASE_DIR/definitions/grib2/localConcepts/[centre:s]

Note: ‘centre:s’ means the centre as a *string* e.g. ecmf, kwbc, cnmc etc

- **The general parameter descriptions are contained in the files shortName.def, paramId.def, units.def, name.def in the directories**

  BASE_DIR/definitions/grib1

  BASE_DIR/definitions/grib2

Local configuration

ECCODES_DEFINITION_PATH=/my/definitions:/eccodes/definitions

- The library searches for each required definition file first in /my/definitions and then in /eccodes/definitions
- If the file is found in /my/definitions then it used by the decoding engine
- The user can override all the definition files with his/her own definition files
- We suggest you only override the definition files containing the parameter information

Local configuration: defining a GRIB parameter locally

- Get the directory of the definition files with the utility codes_info
- set the environment variable ECCODES_DEFINITION_PATH=local_dir:default_definition_dir
- Create the directories:

  local_dir/grib1/localConcepts/[centre:s]

  local_dir/grib2/localConcepts/[centre:s]

And add files shortName.def, paramId.def, name.def & units.def.

Local configuration: defining a parameter locally

- Example from paramId.def (for GRIB1)

```
#Direction of wind waves
'500072' = {
         table2Version = 112;
         indicatorOfParameter = 101;
}
```

- Example from shortName.def (for GRIB1)

```
#Total precipitation of at least 10 mm
'tpg10' = {
         table2Version = 131;
         indicatorOfParameter = 62;
}
```

Local configuration: GRIB Practical

Get the practicals: tar xvf ~trx/ecCodes/localConfig.tar

1. What parameter is contained within the two files x.grib1 and x.grib2?
2. Run codes_info to find the location of the default definitions
3. Now set ECCODES_DEFINITION_PATH to include the "mydefs" directory e.g.

   ```
   export
   ECCODES_DEFINITION_PATH=`pwd`/mydefs:/path/to/defaults
   ```
4. Now see if ecCodes recognizes the name, units etc
5. Test the GRIB1 to GRIB2 conversion. Compare the output with the provided x.grib2 file
6. Study the directory structure of "mydefs" and its local concept files

Local configuration: BUFR local tables

- Warning: The use of local tables in messages intended for non-local or international exchange is strongly discouraged
- The local BUFR tables are stored in the following definitions directories:

  definitions/bufr/tables/[masterTableNumber]/
  local/[localTablesVersionNumber]/
  [bufrHeaderCentre]/[bufrHeaderSubCentre]

  e.g. definitions/bufr/tables/0/local/101/98/0/   (98 is for ecmwf)

- The environment variable ECCODES_DEFINITION_PATH can be set to locate local BUFR tables. We follow the same convention as we saw previously:

```
ECCODES_DEFINITION_PATH=/path/to/local/defs:/path/to/default/defs
```

Local configuration: BUFR Practical

Get the practicals: tar xvf ~trx/ecCodes/localBufrTable.tar

1. Try decoding the BUFR file ikco_217.local.bufr using bufr_dump. This file uses a local table version (localTablesVersionNumber=66)
2. Run codes_info to find the location of the default definitions
3. Now set ECCODES_DEFINITION_PATH to include the provided “mydefs” directory e.g.

   ```
   export
   ECCODES_DEFINITION_PATH=`pwd`/mydefs:/path/to/defaults
   ```
4. Now see if ecCodes can decode the BUFR file
5. Search the output of bufr_dump for the overridden unit “MyOwnUnits”. This was added as an example of a centre defining its own parameter attributes

Questions ?