

ecCodes GRIB: Advanced Topics

Part II

Shahram Najm

Development Section

Forecast Department

Overview

- **Parameter database**
- **GRIB1 to GRIB2 conversion**
- **Local configuration**

GRIB 1 parameter (WMO coding)

“10 metre U wind component”

indicatorOfParameter = 33 [u-component of wind (m/s)]

table2Version = 3

indicatorOfTypeOfLevel = 105 [Specified height level above ground (m)]

level = 10

GRIB 2 parameter (WMO coding)

“**10 metre U wind component**”

discipline = 0 [Meteorological products]

parameterCategory = 2 [Momentum]

parameterNumber = 2 [u-component of wind (m s⁻¹)]

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

scaleFactorOfFirstFixedSurface = 0

scaledValueOfFirstFixedSurface = 10

GRIB 1 parameter (ECMWF local coding)

“10 metre U wind component”

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

table2Version = 128

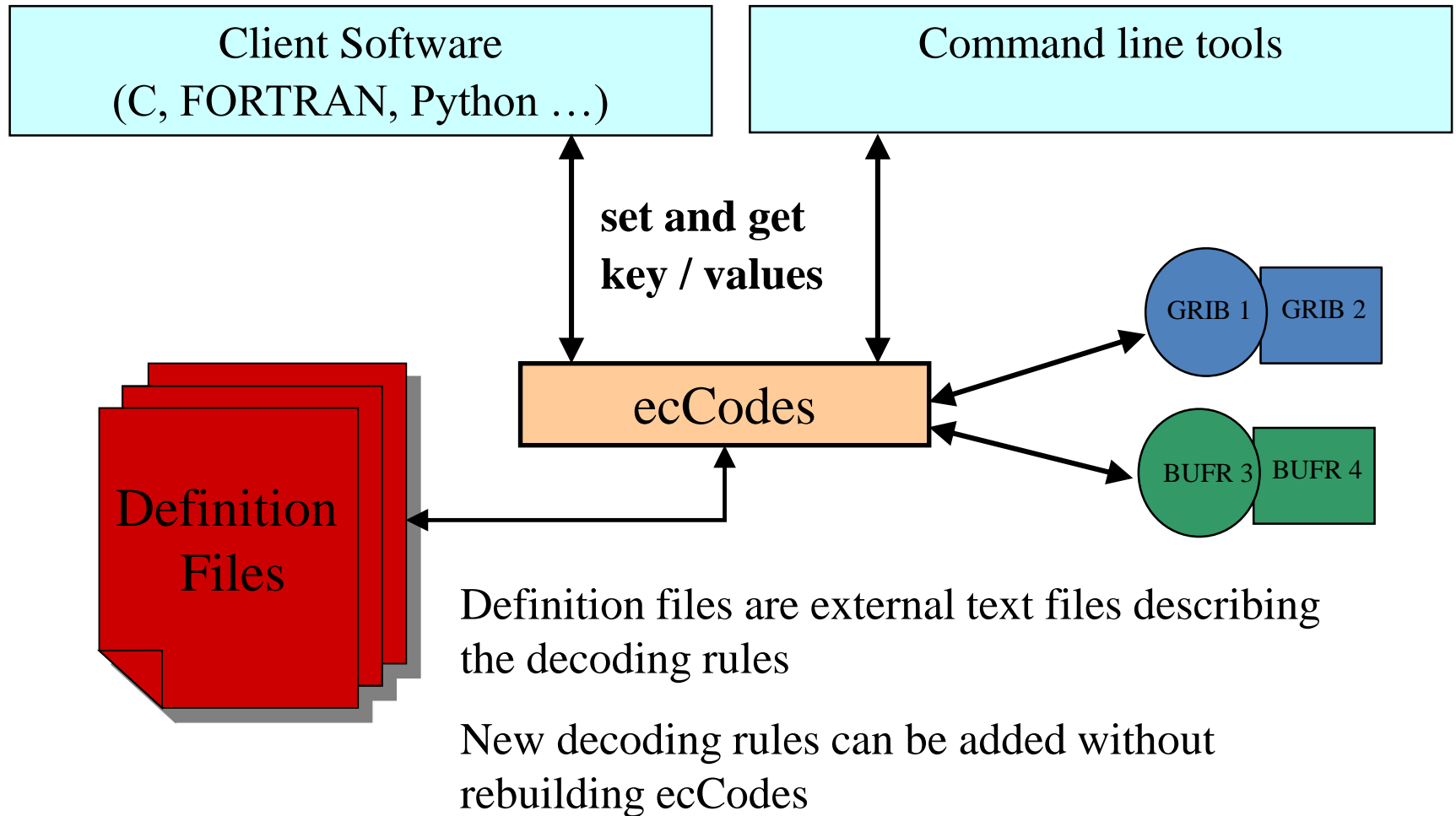
ecCodes solution

- There are different ways of coding a parameter!
- 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 - Design



Concept: shortName

GRIB 2

'2t' = { **discipline=0; parameterCategory=0; parameterNumber=0 ;**
typeOfFirstFixedSurface=103; scaleFactorOfFirstFixedSurface=0;
scaledValueOfFirstFixedSurface=2; }

GRIB1

'2t' = { **indicatorOfParameter=11; table2Version=3;**
levelType=105; level=2;}

GRIB1 ECMWF local coding

'2t' = { **indicatorOfParameter=167; table2Version=128;}**

Concept: paramId

GRIB 2

167 = { **discipline=0; parameterCategory=0; parameterNumber=0 ;**
typeOfFirstFixedSurface=103; scaleFactorOfFirstFixedSurface=0;
scaledValueOfFirstFixedSurface=2; }

GRIB1

167 = { **indicatorOfParameter=11; table2Version=3;**
levelType=105; 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/>

■ **Parameter ID:** 131

■ **Name:** U component of wind

■ **Short Name:** u

■ **Units:** m s⁻¹

GRIB Edition 1 **GRIB Edition 2**

WMO

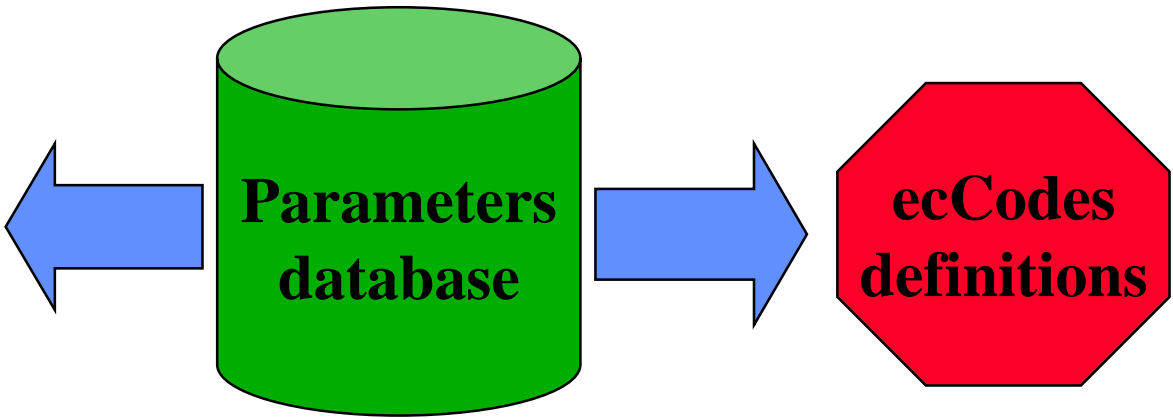
Key	Value
discipline	0
parameterCategory	2
parameterNumber	2

Parameters database

WEB

GDSD API Parameters <http://www.ecmwf.int/publications/technical/0/getapi/param/param-db...>

id	name	shortName	unit
1	Wind direction	dir	deg
2	Wind velocity	vel	m/s
3	Pressure	pres	hPa
4	Equivalent potential temperature	eqpot	K
5	Equivalent potential temperature	eqpot	K
6	Sea level pressure	slp	hPa
7	Sea level pressure	slp	hPa
8	Sea level pressure	slp	hPa
9	Sea level pressure	slp	hPa
10	Sea level pressure	slp	hPa
11	Component of wind speed	comp1	m/s
12	Component of wind speed	comp2	m/s
13	Component of wind speed	comp3	m/s
14	Component of wind speed	comp4	m/s
15	Component of wind speed	comp5	m/s
16	Component of wind speed	comp6	m/s
17	Component of wind speed	comp7	m/s
18	Component of wind speed	comp8	m/s
19	Component of wind speed	comp9	m/s
20	Component of wind speed	comp10	m/s
21	Component of wind speed	comp11	m/s
22	Component of wind speed	comp12	m/s
23	Component of wind speed	comp13	m/s
24	Component of wind speed	comp14	m/s
25	Component of wind speed	comp15	m/s
26	Component of wind speed	comp16	m/s
27	Component of wind speed	comp17	m/s
28	Component of wind speed	comp18	m/s
29	Component of wind speed	comp19	m/s
30	Component of wind speed	comp20	m/s
31	Component of wind speed	comp21	m/s
32	Component of wind speed	comp22	m/s
33	Component of wind speed	comp23	m/s
34	Component of wind speed	comp24	m/s
35	Component of wind speed	comp25	m/s
36	Component of wind speed	comp26	m/s
37	Component of wind speed	comp27	m/s
38	Component of wind speed	comp28	m/s
39	Component of wind speed	comp29	m/s
40	Component of wind speed	comp30	m/s
41	Component of wind speed	comp31	m/s
42	Component of wind speed	comp32	m/s
43	Component of wind speed	comp33	m/s
44	Component of wind speed	comp34	m/s
45	Component of wind speed	comp35	m/s
46	Component of wind speed	comp36	m/s
47	Component of wind speed	comp37	m/s
48	Component of wind speed	comp38	m/s
49	Component of wind speed	comp39	m/s
50	Component of wind speed	comp40	m/s
51	Component of wind speed	comp41	m/s
52	Component of wind speed	comp42	m/s



1 of 7 24/02/2010 13:41

Parameters: Practicals

```
cd $SCRATCH  
tar xf ~trx/ecCodes/eccodes_grib_parameters.tar  
cd grib_parameters
```

- 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

- According to WMO:
“...*the use of Local tables in messages intended for non-local or international exchange is strongly discouraged*”
- 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

- 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: define a 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: define 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: Practical

```
cd $SCRATCH
```

```
tar xf ~trx/ecCodes/eccodes_grib_localConfig.tar
```

1. What parameter is contained in the 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 files/directories of “mydefs”