

Extra examples

Roberto Ribas

Forecast Department

Copernicus Production Section

How to change the attributes of a key

So far, we have changed the BUFR structure, and changed some key values, but, can we change the attributes of a key?

We will start from the **synop.bufr**, and create a very simple message where we can play with the attributes.

Create a rules filter that changes the width of the key **airTemperature** so its new value is **26**. Does it work? Why?

solution

- This filter tries to change the width of the key airTemperature

```
set unpack=1;  
set airTemperature->width=26;  
set pack=1;  
write;
```

- However, it does not succeed as the attribute is read-only

```
ECCODES ERROR : Error while setting key airTemperature->width (Value is read  
only)  
ERROR: Value is read only
```

Change attributes of a key

We will get a different approach, we will start from a very simple message containing some geographic information (latitude/longitude/heightOfStation) and we will find the way to change the width of the airTemperature key.

1. Find in table B the descriptors for

airTemperature

temperature

In table D common sequences, find the expansion for **301022**

2. Create a simple message by setting the unexpanded descriptors to the keys found in part 1. Start with the 301022 sequence.

Hint:

```
unexpandedDescriptors={301022,12101,12023};
```

solution

- This rules file produces a simple message

```
set unexpandedDescriptors={301022,12101,12023};  
write;
```

- The keys have their original width

```
"key" : "airTemperature",  
"value" : null,  
"index" : 4,  
"code" : "012101",  
"units" : "K",  
"scale" : 2,  
"reference" : 0,  
"width" : 16
```

```
"key" : "temperature",  
"value" : null,  
"index" : 5,  
"code" : "012023",  
"units" : "C",  
"scale" : 0,  
"reference" : -99,  
"width" : 8
```

Change attributes of a key

Check with **bufr_dump** that you have the expected keys. Add some values to the latitude, longitude, heightOfStation, temperature and airTemperature.

What happens if you try to set the key temperature to 301?

What happens if you try to set the key airTemperature to 1000?

Look at table B common sequences for the two keys.

Solution

If you try to set these values you get

```
ECCODES ERROR : encoding airTemperature=1000  
ECCODES ERROR : Error while setting key pack  
(Value out of coding range)  
ERROR: Value out of coding range
```

The airTemperature is out of the range. The same happens to temperature. To change it we have to apply an operator to both keys.

```
set unexpandedDescriptors={301022,201138,12101,12023,201000};  
set unpack=1;  
set temperature=301;  
set airTemperature=1000;  
set pack=1;  
write;
```

solution

- The sequence **2 01 138** increases the width of the following key 12101 in 10 bits as we have to subtract 128 from 138 (see table rules).

F X			
2 01	YYY	Change data width	Add (YYY-128) bits to the data width given for each data element in Table B, other than CCITT IA5 (character) data, code or flag tables.

- Then, we have to end this operator sequence, after the temperature key, so both are affected by the change of width operator. That is why the sequence ends with **201000**.

Change attributes of a key

We may use BUFR operators to change the width attribute of a key, that way, we can store bigger values. To do so, we have to refer to table C, that contains the Data Descriptor operators.

We can use the operator to change the data width of airTemperature and see if we can set a higher value for this key.

Bear in mind that the operator affects all subsequent descriptors, so we must disable the operator at the end of the sequence change.

Change attributes of a key

Now set your unexpandedDescriptors to the following sequence

```
{301022, 12023, 201138,12101,201000, 12023}
```

Try now to set airTemperature to 1000. What is the width of this key now? Compare it with the width of this key before we activated the operator.

Does the temperature key width attribute change? Why?

Solution

```
ECCODES ERROR : encoding temperature=301  
ECCODES ERROR : Error while setting key pack  
(Value out of coding range)  
ERROR: Value out of coding range
```

Now we get an error as the temperature key width is not modified and retains its original value.

Change attributes of a key

- Now move the data operator 201000 after the 12023 descriptor, set the temperature to 301. Run the filter. Does it work?

What happens if you don't disable to change of width operator 2 01 138?

Try to add another data descriptor to the unexpandedDescriptors for example (dewpointTemperature 012024). What is the width of this key now.

So, it is important to disable the change of width operator, so it modifies only the intended keys.

solution

The keys after 201000 are not affected by the change of width operator and then retain their original width.

```
"key" : "temperature",  
  "value" : 31,  
  "index" : 5,  
  "code" : "012023",  
  "units" : "C",  
  "scale" : 0,  
  "reference" : -99,  
  "width" : 8  
},  
{  
  "key" : "dewpointTemperature",  
  "value" : 32,  
  "index" : 6,  
  "code" : "012024",  
  "units" : "C",  
  "scale" : 0,  
  "reference" : -99,  
  "width" : 8
```

```
"key" : "airTemperature",  
  "value" : 1000,  
  "index" : 4,  
  "code" : "012101",  
  "units" : "K",  
  "scale" : 2,  
  "reference" : 0,  
  "width" : 26
```