Implementing main types of international validation rules in national validation processes

Olav ten Bosch, Mark van der Loo — *Statistics Netherlands*
Sónia Quaresma — *Statistics Portugal*

UNECE Workshop on Statistical Data Editing (SDE), Sep. 2020

Contents

- International data validation
- Eurostat main types of rules
- Pilot NL: Implementation in R
- Pilot PT: Implementation in SQL
- Wrap up
- ValidatFOSS2

International data validation (1)

- ***Invalid*** data may lead to ***costly*** retransmissions or reprocessing (data ping pong)
- To guarantee overall data ***quality*** and ***efficiency***, the European Statistical System (ESS) is moving towards more harmonised validation activities
- International validation rules are agreed in domain specific ***statistical working groups***
- Data producer (NSIs) and data consumers (international organisations) ***validate*** data against the same rules

International data validation (2)

ESSnet Validat Foundation 2015-2

ESSnet Validat Integration, 2017 (

- Handbook on validation
- A study on VTL 1.0
- PoC with 3 national validation la
- Validation principles
- Business architecture scenario's
- Generic validation report → Paper SDE 2019
- Generic / main types of validation rules

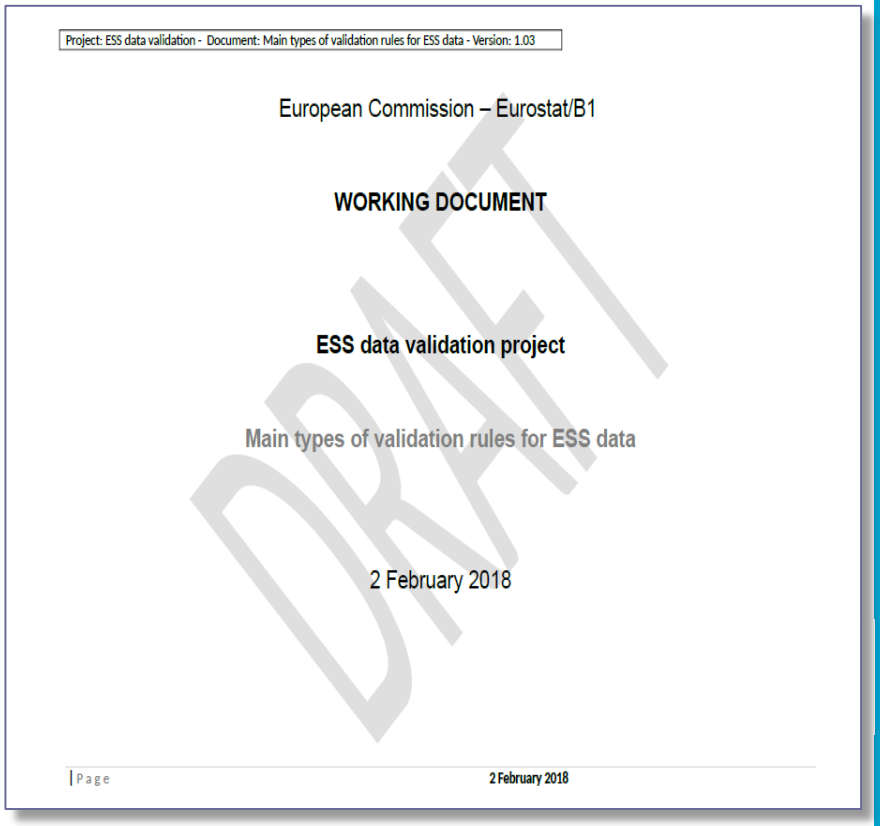
Validation principles:

1. *The sooner, the better*
2. *Trust but verify*
3. *Well-documented and appropriately communicated validation rules*
4. *Well-documented and appropriately communicated validation errors*
5. *Comply or explain*
6. *Good enough is the new perfect*

https://ec.europa.eu/eurostat/cros/content/data-validation-overview_en

Eurostat main types of rules (1)

- 2018: Eurostat identified 21 '***main types of validation rules***' for ESS data
- They reflect the ***majority*** of checks needed in today's International data validation
- Specified in ***natural language*** and ***VTL***
- Can we implement them in national systems?



Eurostat main types of rules (2)

Examples:

- Range check:

4.3.5 (VIR) Values are In a Range

Check that the observation value is higher (or equal) to a minimum value and/or is lower (or equal) to a maximum value.

- Aggregation check:

4.3.7 (VAD) Values for Aggregates are consistent with Details

Check that values for aggregates are consistent with the sum of values for detailed data.

- *A tolerance (acceptable margin) expressed in % or absolute number is possible.*

- Completeness of time series:

4.3.2 (RTS) Records are all present for Time Series

Check that time series provided in one file are complete (between the oldest and the most recent time period expected in the file, no period is missing).

Pilot NL: Implementation in R (1)

ValidatFOSS: validation with Free and Open-Source Software

- Short Term Statistics (STS):
  - All rules could be implemented in one line of R-validate code
  - Some of the textual rules descriptions lacked preciseness
- National Accounts (NA):
  - Chain linking formula implemented
  - Majority of code is about selecting the right slice of data from the database, the actual implementation of the rule was only one line of R-validate code
- Eurostat main types of rules:
  - Implemented in R-package
  - Documentation in R-style providing context-sensitive help in R and/or RStudio
  - Example datasets from specification document included
  - Automatic tests defined based on the examples in the specification document

Eurostat main types of rules

Implemented:

- FDT: FielD Type
- FDL: FielD Length
- FDM: FielD is Manatory or empty
- COV: COdes are Valid
- RWD: Records are Without Duplicate id-keys
- REP: Records Expected are Provided
- RTS: Records are all present for Time Series
- RNR: Records' Number is in a Range
- COC: COdes are Consistent
- VIR: Values are In a Range
- VCO: Values are COnsistent
- VAD: Valueas for Aggregates are consistent with Details
- VSA: Values for Seasonally Adjusted data are plausible

VIR	*Check that values are within a range*

Description

Check that values are within a range

Usage

```
VIR(d, Min = NULL, Max = NULL)
```

Arguments

`d`	When used in a validation rule, a bare (unquoted) name of a variable. Otherwise a vector of class `character`. Coerced to character as necessary.
`Min`	smallest allowed value
`Max`	largest allowed value

Value

A `logical` with the length of `d`.

R-package GenericValidationRules:

https://github.com/SNStatComp/GenericValidationRules

Eurostat main types of rules

Implemented:

- FDT: FielD Type
- FDL: FielD Length
- FDM: FielD is Manatory or empty
- COV: COdes are Valid
- RWD: Records are Without Duplicate id-keys
- REP: Records Expected are Provided
- RTS: Records are all present for Time Series
- RNR: Records' Number is in a Range
- COC: COdes are Consistent
- VIR: Values are In a Range
- VCO: Values are COnsistent
- VAD: Valueas for Aggregates are consistent with Details
- VSA: Values for Seasonally Adjusted data are plausible

RTS — *Check that records are present for time series*

Description

A record set is split by a set of *dimensions*. For each split, the variable indicating the time period must be both gapless and within bounds.

Usage

```
RTS(timevar, ftp, ltp, ...)
```

Arguments

timevar	
ftp	
ltp	
...	

```
# RTS examples
data(RTSdat)

# Example using RTS with 'validate'
library(validate)
rules <- validator(
 RTS(TIME_PERIOD, ftp = "2008", ltp = "2010"
   , TABLE, FREQ, REPORTING, PARTNER, DIRECTION, AGE, ADJUST) == TRUE
)
cf <- confront(RTSdat, rules)
summary(cf)
out <- as.data.frame(cf)
```

R-package GenericValidationRules:

https://github.com/SNStatComp/GenericValidationRules

Domain specific validation rules

Implemented rules

- Short term statistics rules:
  - STS01: "Correct series"
  - STS02: "No gaps"
  - STS03: "Prices positive"
  - STS04: "No negative observations"
  - STS05: "unique observations"
  - STS06: "all series types"
  - STS10: "base index is 100"
- National Accounts rules:
  - NA_MAIN_VCO_Consistency_between_Prices: "Chain linked formula"

Domain specific rule implemented in main type of rule RTS

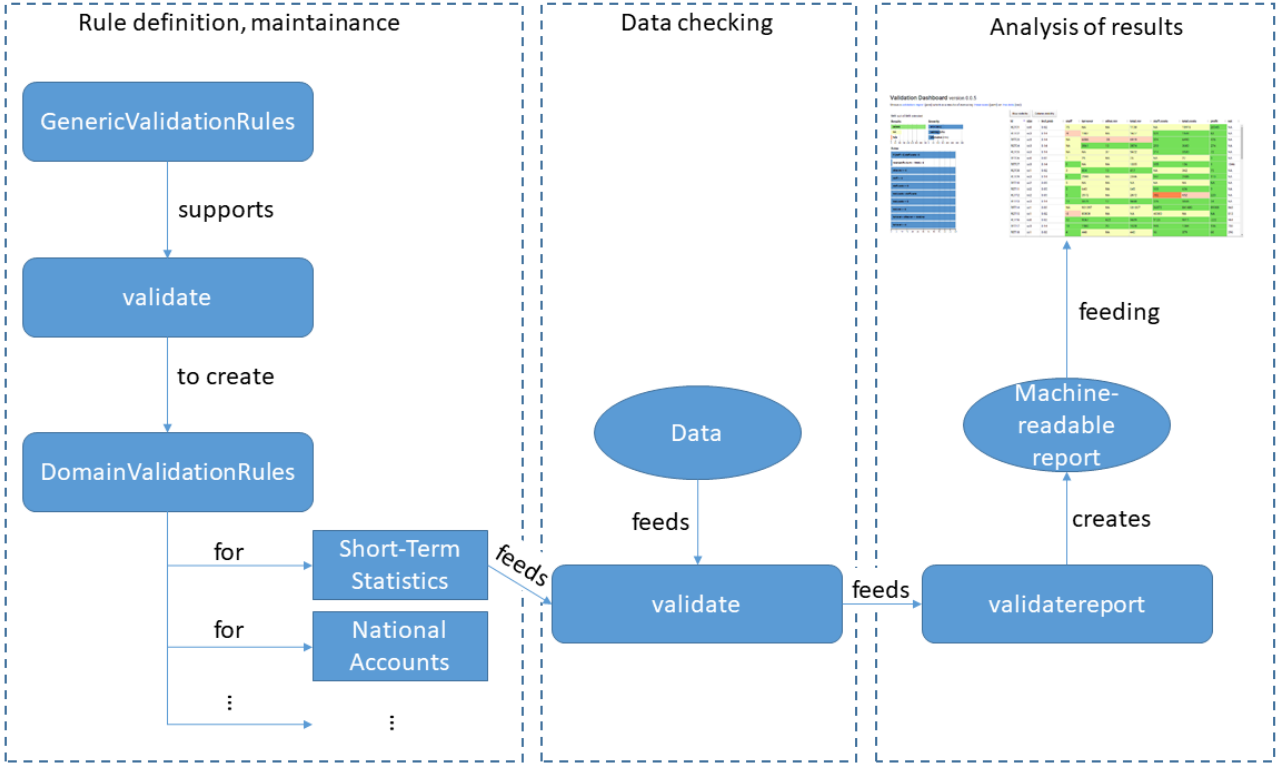
```
- expr: 'RTS(TIME_PERIOD, ftp="2017-Q1",ltp="2019-Q3",FREQ,
        REF_AREA, SEASONAL_ADJUST, INDICATOR, ACTIVITY) == TRUE'
  name: "STS02"
  label: "No gaps"
  description: |
    No missing observations (gaps) are accepted in time series,
    sent in one or several files - i.e. files should be sent in
    the chronological order based on the latest observation.
```

```
# Define validator:
v <- validator(A-((B/C)*D)<1)
```

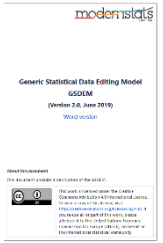
https://github.com/SNStatComp/DomainValidationRules

Aligns to ESS standards

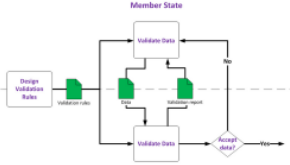
Data validation workflow



Handbook on data validation methodology (2015)



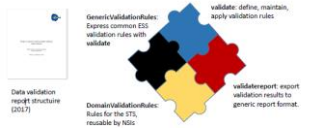
GSDEM (2019)



Business architectures (2016)

Data validation report structuire (2017)

R-based FOSS tooling (CBS)



For all software (and more): **awesomeofficialstatistics.org**

Pilot PT: Implementation in SQL (1)

- ***HyVImp***: Hybrid Validation Implementation Project
- Focus was on rules in domain ***ANIMAL***
- Manual translation of VTL -> parametrized ***SQL***
- Implemented in the central Statistical Data Warehouse (***SDW***)
- Advantages:
  - ***Centralized*** maintenance of main types of validation rules
  - Domain knowledge ***encapsulated*** in parameters; domain specialists do not need IT specialists for implementing rules
  - Solutions in one domain can be ***reused*** in other domains
  - Solution ***integrated*** into existing data reporting environment

Pilot PT: Implementation in SQL (2)

COC – Codes are Consistent

VTL Rule

```
ds:= ANI_gipcat_s_2016;
comb := count(ds group by freq, dim_cl_h_gipcat);
check (not exists_in (comb, matrix_freq_code,all)
errorcode "Combination of Freq, DIM_CL_H_GIPCAT not
possible"
errorlevel "Error");
```

SQL Rule with Parameters

```
Key_list := freq, dim_cl_h_gipcat;
tbl_dsd := ANI_gipcat_s_2016;
tbl_codes:= matrix_freq_code;
tbl_codes_fld:= freq, dim_cl_h_gipcat;

SELECT ' || num || ' as ID,'|| key_list || ',
        CASE
           WHEN ' || REPLACE(key_list,',','||') || ' NOT IN (Select   ' || REPLACE(',',',','||') || '
from ' || tbl_codes || ' b) THEN "false" END  AS BOOL_VAR,
        CASE
           WHEN ' || REPLACE(key_list,',','||') || ' NOT IN (Select   ' ||
REPLACE(tbl_codes_fld,',',',','||') || '  from ' || tbl_codes || ' b) THEN "Combination of Freq,
DIM_CL_H_GIPCAT not possible " END AS ERRORCODE,
        CASE
           WHEN ' || REPLACE(key_list,',',',','||') || ' NOT IN (Select   ' ||
REPLACE(tbl_codes_fld,',',',','||') || '  from ' ||tbl_codes || ' b) THEN "ERROR" END AS
ERRORLEVEL, sysdate as VAL_DATE
        FROM  '  ||tbl_dsd;
```

All rules: https://github.com/SoniaQuaresma/MainTypeValidRules

Wrap-up

- Pilots NL and PT show that implementing Eurostat main types of validation rules in national contexts is ***feasible*** and ***effective***
- If international rules are ***expressed in*** such main types of rules, this approach could be used to implementing validation in national systems
- These main types of rules were identified from current practices. Ideally, we need a ***minimum*** set of high level, parametrized, generic validation rules that cover ***most*** or ***all*** of the validation needs in the ESS.

Next: ValidatFOSS2 (2020/2021)

- Builds on previous results
- Further improve R-based validation toolset; alignment SDMX
- Develop a complete set of high level and easy applicable validation rules for official statistics to be used in all process stages and in all domains
- Build a community: use, share and improve generic and domain specific rule implementations
- Results expected 2021

Questions, ideas, suggestions

Olav ten Bosch	o.tenbosch@cbs.nl	@kobosch
Mark van der Loo	mpj.vanderloo@cbs.nl	@markvdloo
Sónia Quaresma	sonia.quaresma@ine.pt	