Validation in the ESS – A Member State Perspective

Q2016 - Madrid

31.05. – 03.06.2016

Background & Definition

The ESS.VIP Validation and the ESSnet ValiDat Foundation

- 2012: Strategic paper by Eurostat
- 2013: Start of ESS.VIP Validation
- 2014: Task force founded to integrate Member State interests
- 2015: ESSnet ValiDat Foundation (IT. NL, LT, DE and Eurostat)
- 2016: Validation: The Next Generation

Definition:

- Data validation is an activity aimed at verifying whether the value of a data item comes from the given (finite or infinite) set of acceptable values (UNECE 2013

Babylon

if employment status == “old-age pensioner” and

age < 35 then error “Too young!”

0.5 < turnover(curMonth)/turnover(prevMonth) < 2

WENN ANZAHL VON Familie[ALLE].Person[MIT Alter < 18] > 0 DANN … ENDE

```
IF maritalstate=married THEN
    Age>15 “Too young to be married”
ENDIF
```

profit* <= *0.6*revenue

Validation as a Problem

Is there a business case?

- When we did a survey on data validation in the ESS we were not completely aware of the scale of the „problem“:
  - Effort: The amount of effort put into data validation (and editing) in five sample domains was estimated by the member states to make up 40 to 60 % of the total effort
  - Relevance: The impact of data validation on data quality (non-sampling errors) is generally assumed of paramount importance

ValiDat – Foundation I

Business case - implications:

- If validation has such a high impact on data quality and consumes so many resources, then it should be
  - well understood,
  - fairly wide standardized
  - and as far as possible automated

- Sequence: Understanding is the
  - a) methodological foundation of
  - b) standardization which in turn will be the base for
  - c) technical innovation (and process enhancements)

ValiDat – Foundation II

The Base Line: Methodology

- A central part of the methodological work of the ESSnet project is writing a „handbook“ i.e. compiling from the work of others and make it available (pragmatically) for a general audience of statisticians
- Why are we doing validation (remember the business case!)?
  - Enhance data quality dimensions:
    - Directly (like accuracy, coherence and compatability)
    - Indirectly (timeliness) as restrictions

ValiDat - Foundation

The Base Line:

Methodology

- **Content of handbook:**
  - **What**
  - **Why**
  - **How**
  - **When**

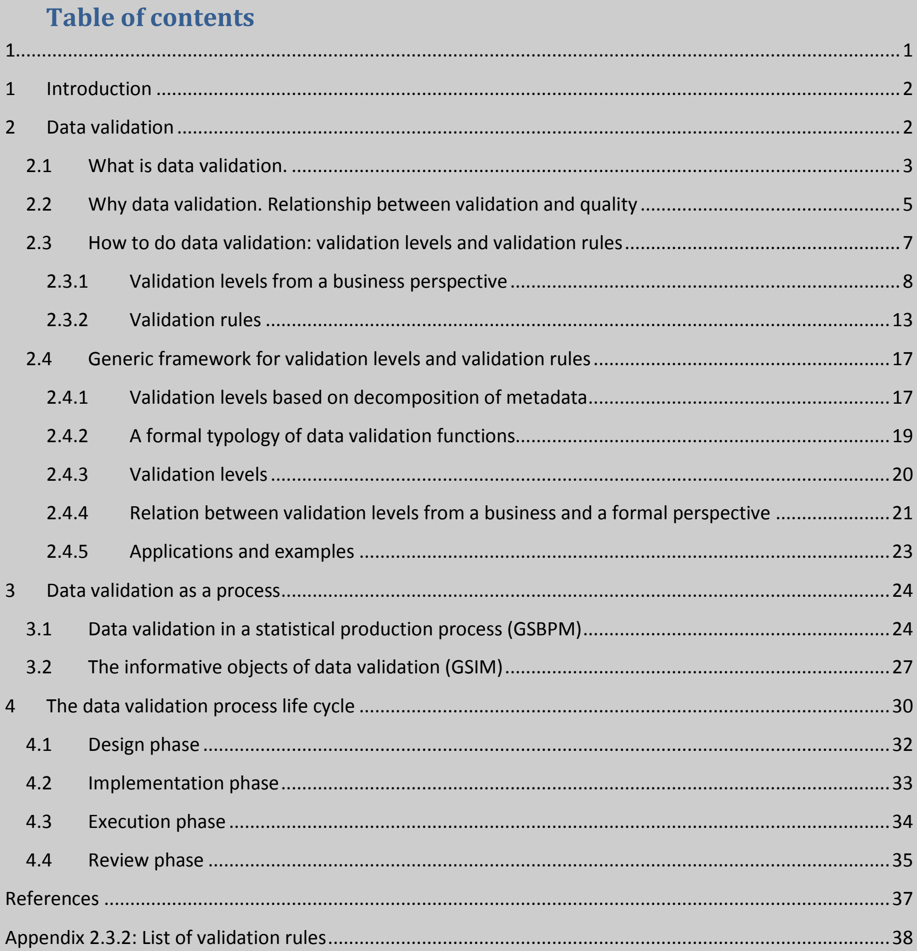
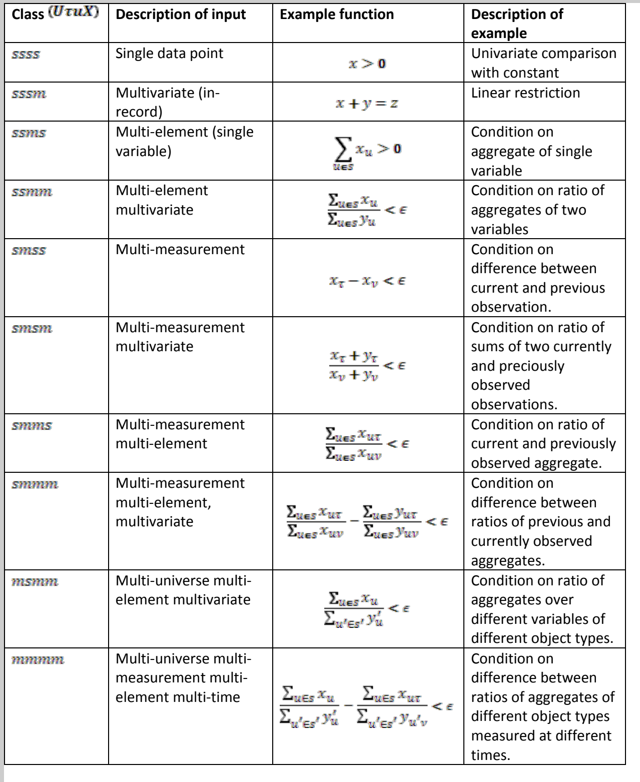


Table of contents

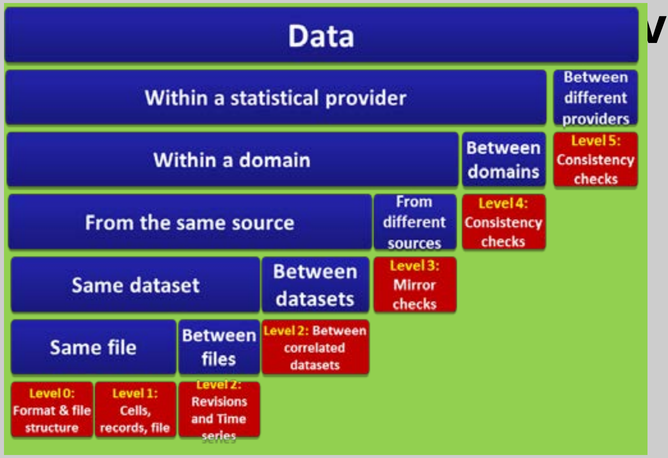
ValiDat - Foundation

The Base Line: Methodology – What?

- The handbook provides classification schemes for validation rules:
  - Levels
  - Pragmatic typology
  - Formal typology
- All have their merits and help
  validation



Class $(U\tau uX)$	Description of input	Example function	Description of example
$ssss$	Single data point	$x > 0$	Univariate comparison with constant
$sssm$	Multivariate (in-record)	$x + y = z$	Linear restriction
$ssms$	Multi-element (single variable)	$\sum_{u\in S} x_u > 0$	Condition on aggregate of single variable
$ssmm$	Multi-element multivariate	$\frac{\sum_{u\in S} x_u}{\sum_{u\in S} y_u} < \epsilon$	Condition on ratio of aggregates of two variables
$smss$	Multi-measurement	$x_\tau - x_v < \epsilon$	Condition on difference between current and previous observation.
$smsm$	Multi-measurement multivariate	$\frac{x_\tau + y_\tau}{x_v + y_v} < \epsilon$	Condition on ratio of sums of two currently and preciously observed observations.
$smms$	Multi-measurement multi-element	$\frac{\sum_{u\in S} x_{u\tau}}{\sum_{u\in S} x_{uv}} < \epsilon$	Condition on ratio of current and previously observed aggregate.
$smmm$	Multi-measurement multi-element, multivariate	$\frac{\sum_{u\in S} x_{u\tau}}{\sum_{u\in S} x_{uv}} - \frac{\sum_{u\in S} y_{u\tau}}{\sum_{u\in S} y_{uv}} < \epsilon$	Condition on difference between ratios of previous and currently observed aggregates.
$msmm$	Multi-universe multi-element multivariate	$\frac{\sum_{u\in S} x_u}{\sum_{u'\in S'} y'_u} < \epsilon$	Condition on ratio of aggregates over different variables of different object types.
$mmmm$	Multi-universe multi-measurement multi-element multi-time	$\frac{\sum_{u\in S} x_u}{\sum_{u'\in S'} y'_u} - \frac{\sum_{u\in S} x_{u\tau}}{\sum_{u'\in S'} y_{u'v}} < \epsilon$	Condition on difference between ratios of aggregates of different object types measured at different times.



Data

Within a statistical provider

Between different providers

Within a domain

Between domains

Level 5: Consistency checks

From the same source

From different sources

Level 4: Consistency checks

Same dataset

Between datasets

Level 3: Mirror checks

Same file

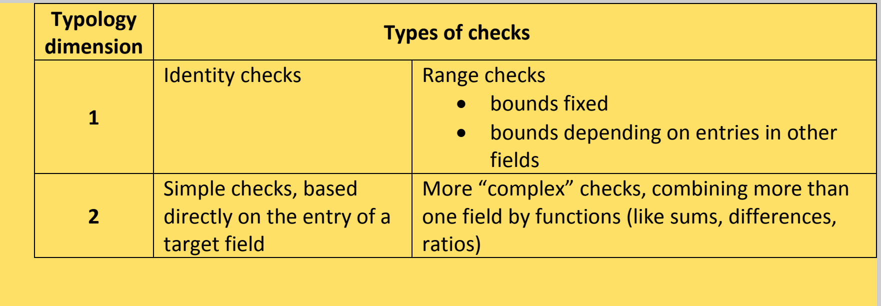
Between files

Level 2: Between correlated datasets

Level 0: Format & file structure

Level 1: Cells, records, file

Level 2: Revisions and Time series



Typology dimension	Types of checks	
1	Identity checks	Range checks • bounds fixed • bounds depending on entries in other fields
2	Simple checks, based directly on the entry of a target field	More "complex" checks, combining more than one field by functions (like sums, differences, ratios)

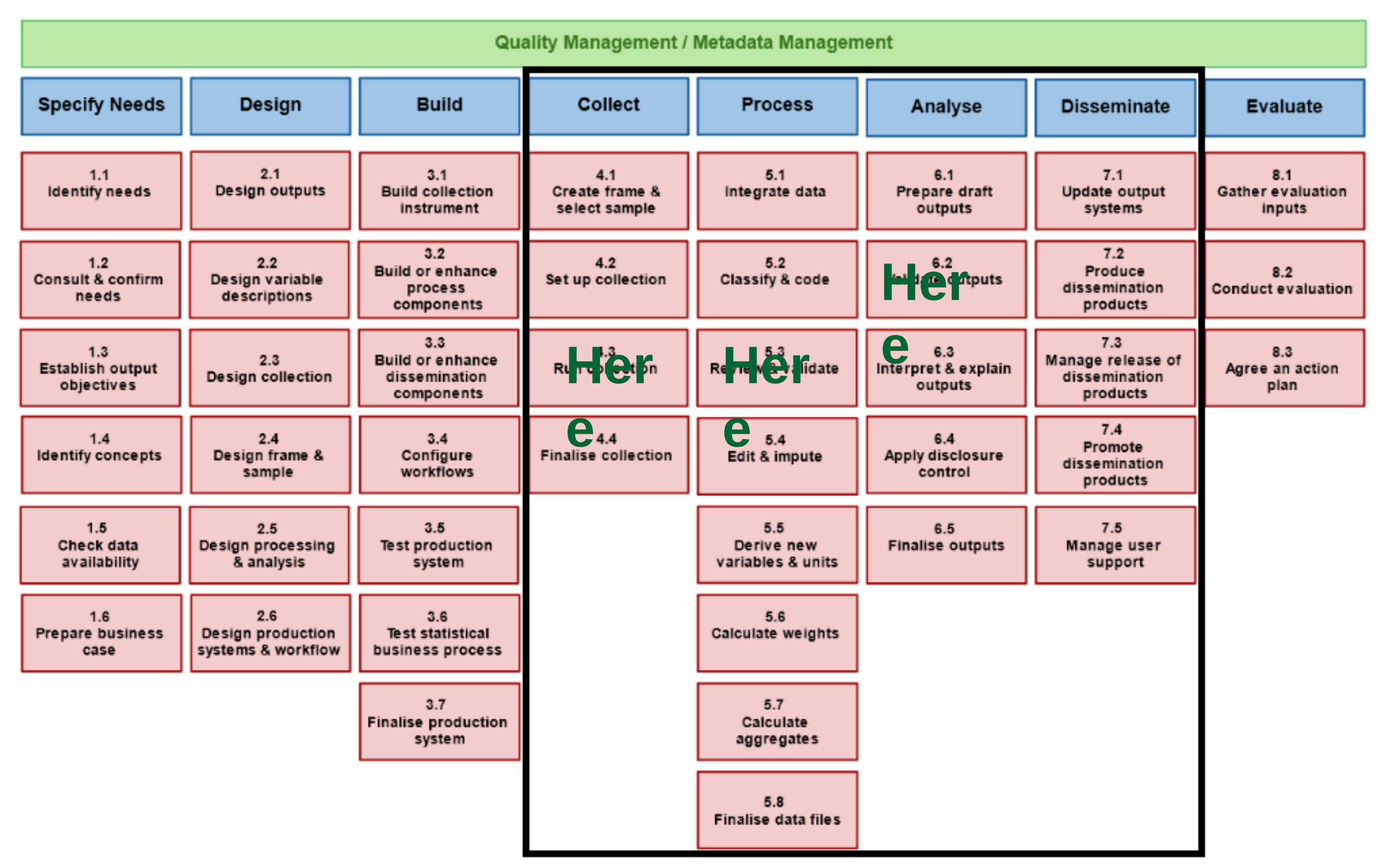
ValiDat - Foundation

The Base Line: Methodology – What?

- Levels and rule types are building blocks to discuss other important concepts like:
  - Structural vs. content based validation
  - Simple vs. complex rule types
  - Soft vs. hard checks
  - Micro data vs. macro data validation
- They can be used as a framework for metrics, languages and technologies

ValiDat - Foundation

The Base Line: Methodology – When?



Quality Management / Metadata Management

Specify Needs	Design	Build	Collect	Process	Analyse	Disseminate	Evaluate
1.1 Identify needs	2.1 Design outputs	3.1 Build collection instrument	4.1 Create frame & select sample	5.1 Integrate data	6.1 Prepare draft outputs	7.1 Update output systems	8.1 Gather evaluation inputs
1.2 Consult & confirm needs	2.2 Design variable descriptions	3.2 Build or enhance process components	4.2 Set up collection	5.2 Classify & code	6.2 Validate outputs	7.2 Produce dissemination products	8.2 Conduct evaluation
1.3 Establish output objectives	2.3 Design collection	3.3 Build or enhance dissemination components	4.3 Run collection	5.3 Review & validate	6.3 Interpret & explain outputs	7.3 Manage release of dissemination products	8.3 Agree an action plan
1.4 Identify concepts	2.4 Design frame & sample	3.4 Configure workflows	4.4 Finalise collection	5.4 Edit & impute	6.4 Apply disclosure control	7.4 Promote dissemination products	
1.5 Check data availability	2.5 Design processing & analysis	3.5 Test production system		5.5 Derive new variables & units	6.5 Finalise outputs	7.5 Manage user support	
1.6 Prepare business case	2.6 Design production systems & workflow	3.6 Test statistical business process		5.6 Calculate weights			
		3.7 Finalise production system		5.7 Calculate aggregates			
				5.8 Finalise data files			

Here

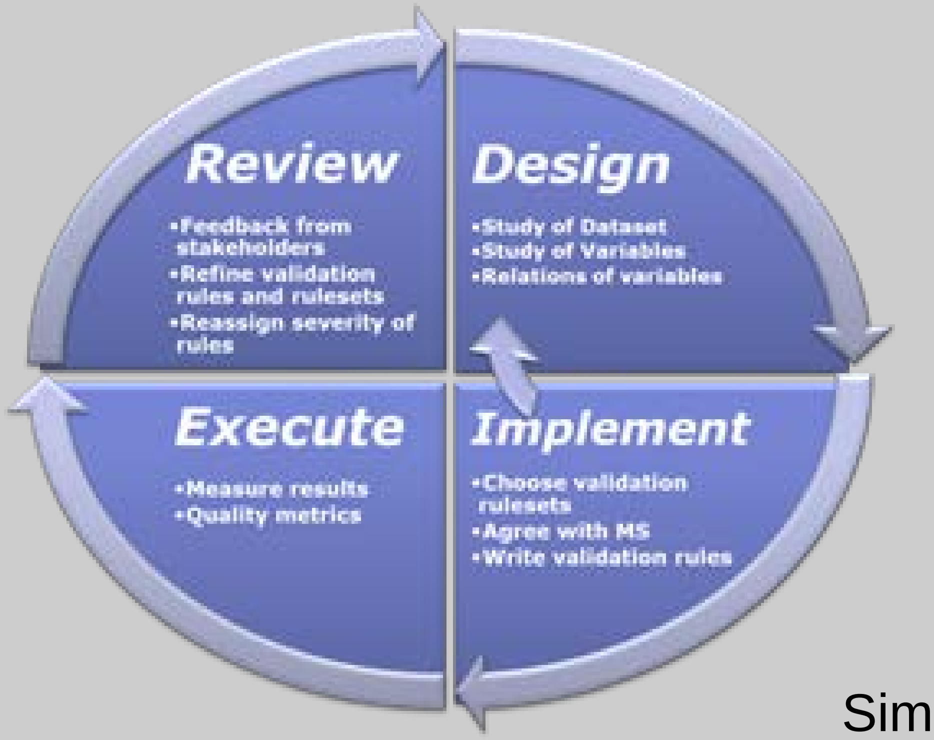
Here

Here

ValiDat - Foundation

The Base Line: Methodology – How?

- **Validation Life Cycle**



Review

- Feedback from stakeholders
- Refine validation rules and rulesets
- Reassign severity of rules

Design

- Study of Dataset
- Study of Variables
- Relations of variables

Execute

- Measure results
- Quality metrics

Implement

- Choose validation rulesets
- Agree with MS
- Write validation rules

Simon et al. 2015

ValiDat - Foundation

The Base Line: Methodology – How?

- How do we know that we have struck the right balance between
  - Improving data quality
  - At acceptable costs

- Our solution: use metrics!
  - Analyse the internal consistency of validation rule sets
  - Analyse the value of validation rules on observed data
  - Analyse validation rule sets in comparison to observed and expected data

ValiDat - Foundation

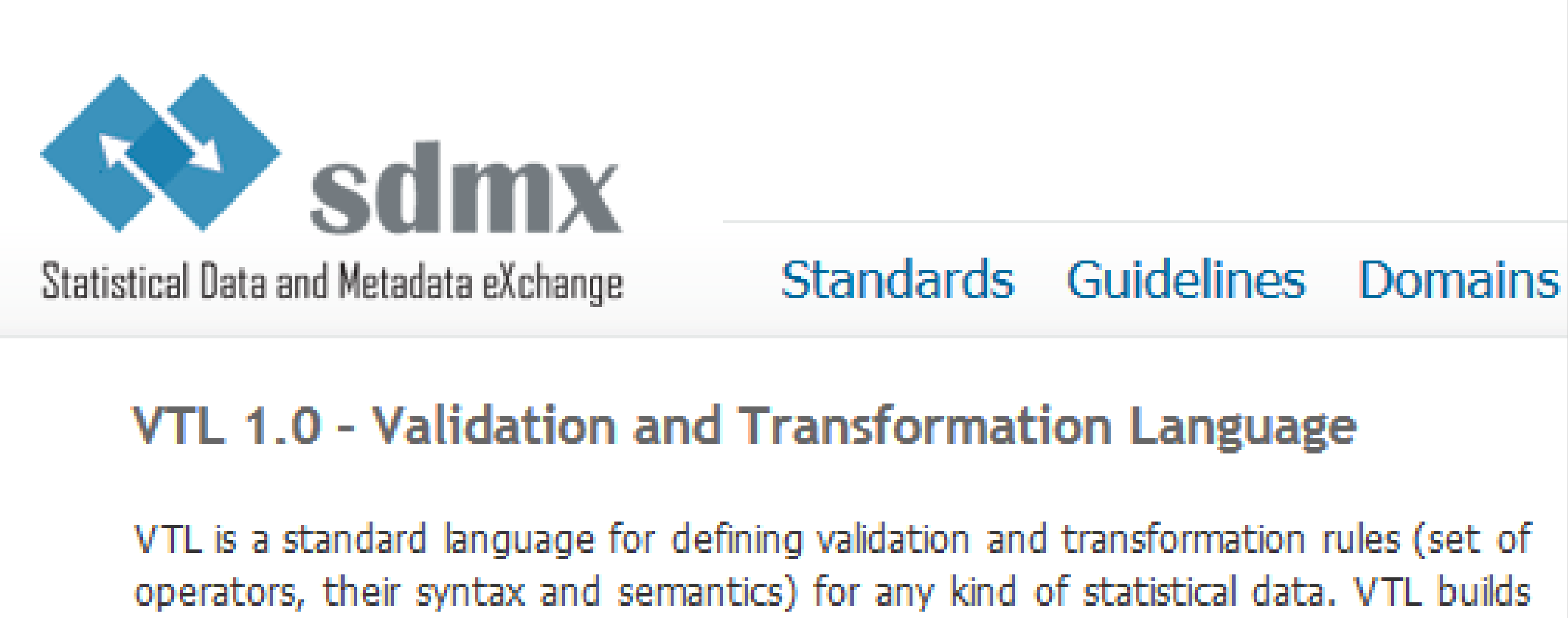
Language

- The future validation language has two main goals:
  - It should provide an unambigous communication channel for specialists (humans!)
  - It should feed different IT-systems with the necessary specific information about a particular survey
  - These might be conflicting aims!

VTL

Language: A new Sta(nda)r(d) is born

- VTL - Validation and Transformation Language has been specified by the SDMX community



VTL 1.0 - Validation and Transformation Language

VTL is a standard language for defining validation and transformation rules (set of operators, their syntax and semantics) for any kind of statistical data. VTL builds

VTL

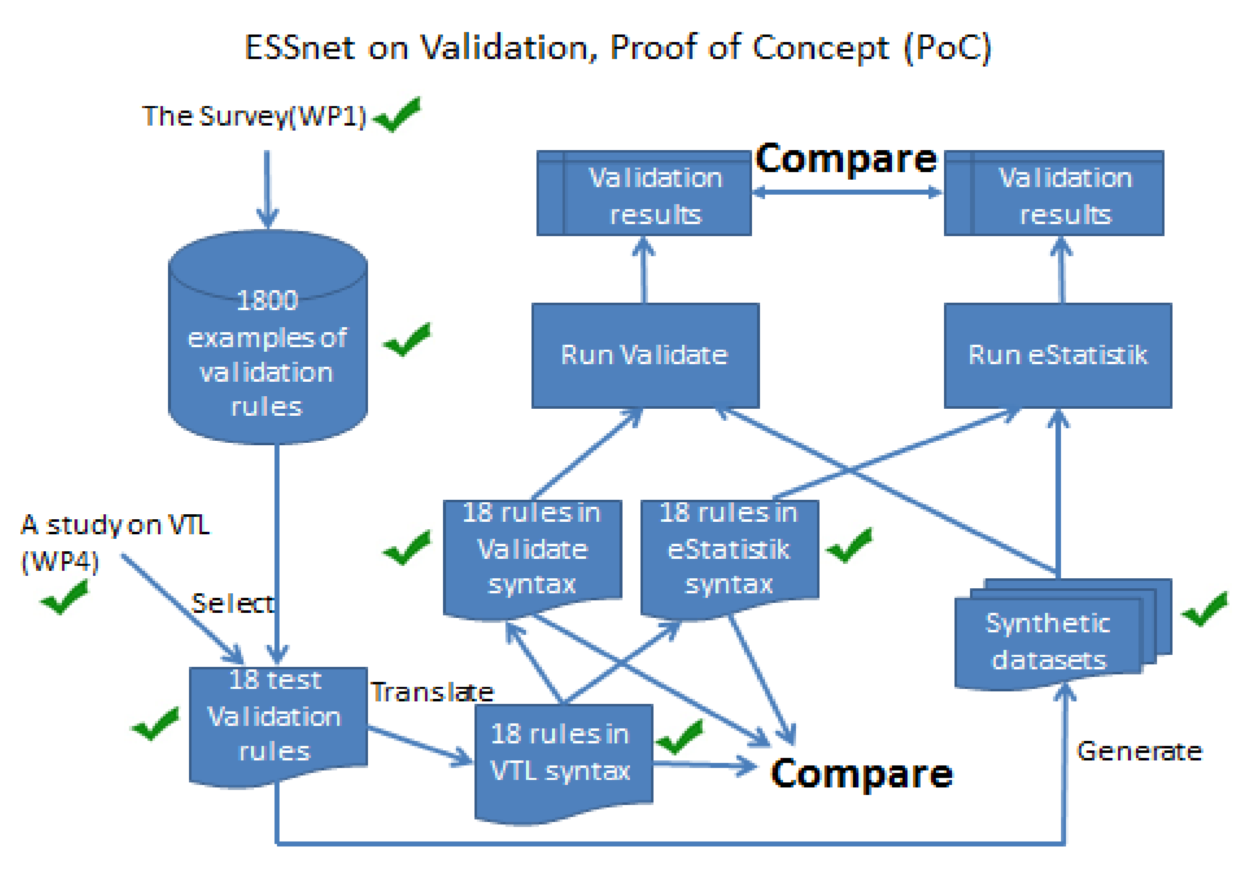
Language: A new Sta(nda)r(d) is born

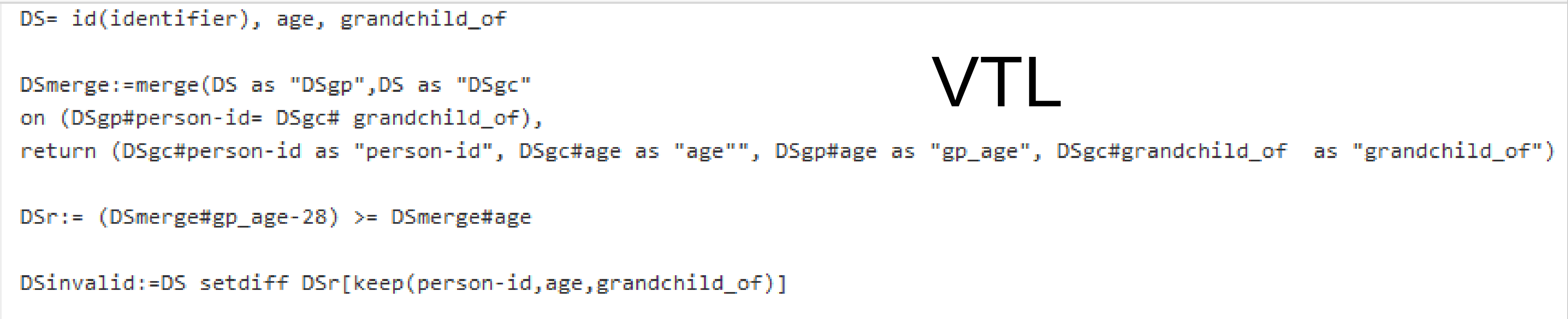
- Different Aspects have been evaluated by the ESSnet:
  - Correctness and coherence
  - Completeness
  - Usability (by human users)
  - Feasibility (for machine-to-machine communication)
- Evaluation is publicly available on CROS-Portal

VTL to Tools/Services

A PoC (Proof of Concept)

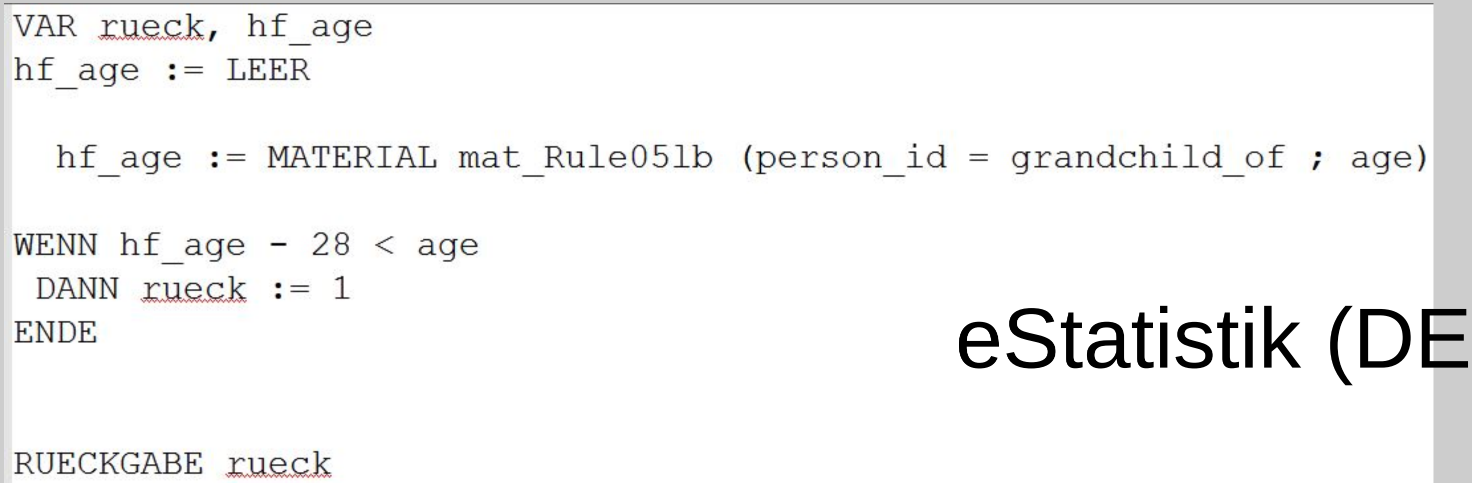
- **Let‘s simulate a European Infrastructure!**





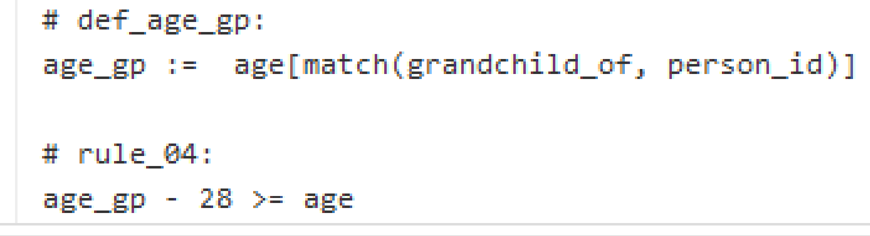
```
DS= id(identifier), age, grandchild_of

DSmerge:=merge(DS as "DSgp",DS as "DSgc"
on (DSgp#person-id= DSgc# grandchild_of),
return (DSgc#person-id as "person-id", DSgc#age as "age"", DSgp#age as "gp_age", DSgc#grandchild_of  as "grandchild_of")

DSr:= (DSmerge#gp_age-28) >= DSmerge#age

DSinvalid:=DS setdiff DSr[keep(person-id,age,grandchild_of)]
```

VTL



```
VAR rueck, hf_age
hf_age := LEER

  hf_age := MATERIAL mat_Rule05lb (person_id = grandchild_of ; age)

WENN hf_age - 28 < age
 DANN rueck := 1
ENDE


RUECKGABE rueck
```

eStatistik (DE)



```
# def_age_gp:
age_gp :=  age[match(grandchild_of, person_id)]

# rule_04:
age_gp - 28 >= age
```

Validate (NL)

VTL again

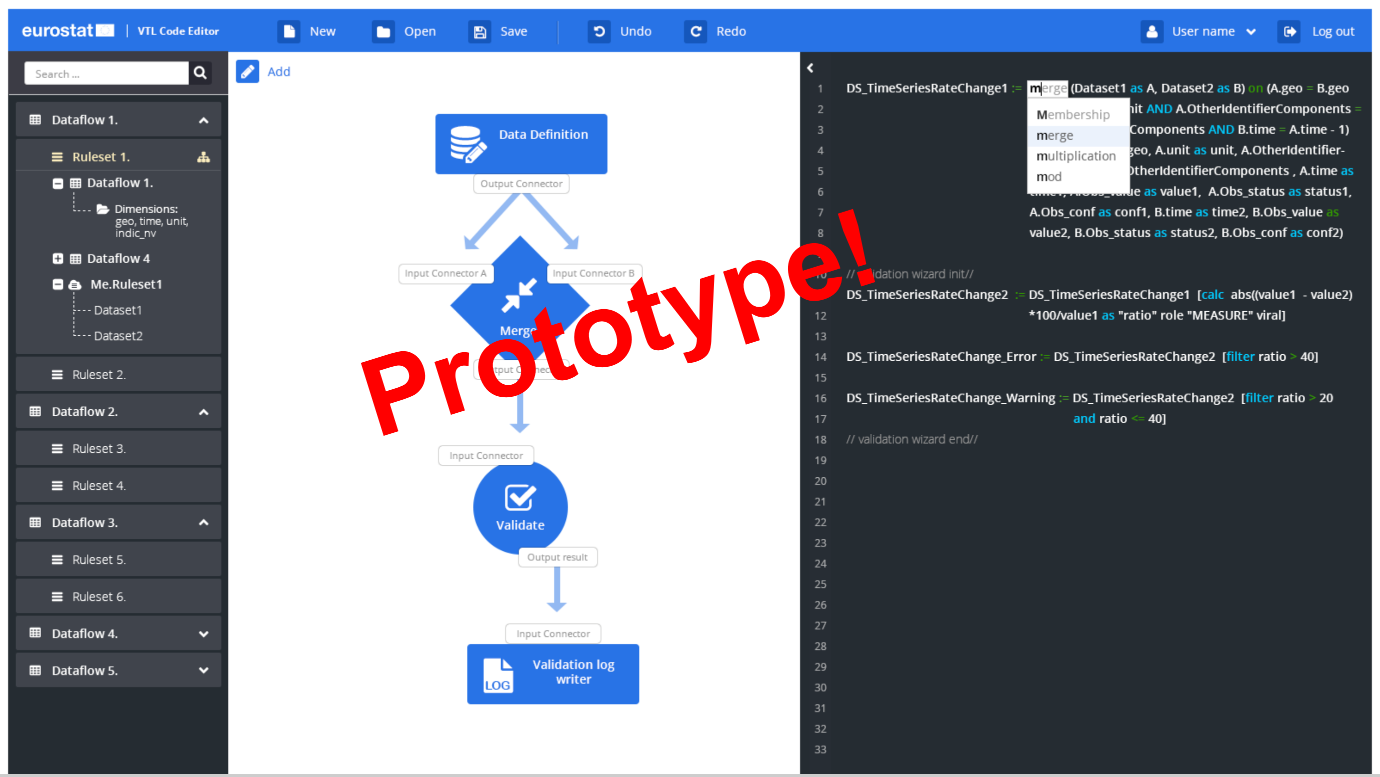
PoC Results

- VTL is hard to understand
- VTL yields lengthy code
- Manual translation from VTL to national dialects requires strong IT skills
- Automatic translation from VTL to national dialect will not be easy

VTL to Tools/Services

Solutions

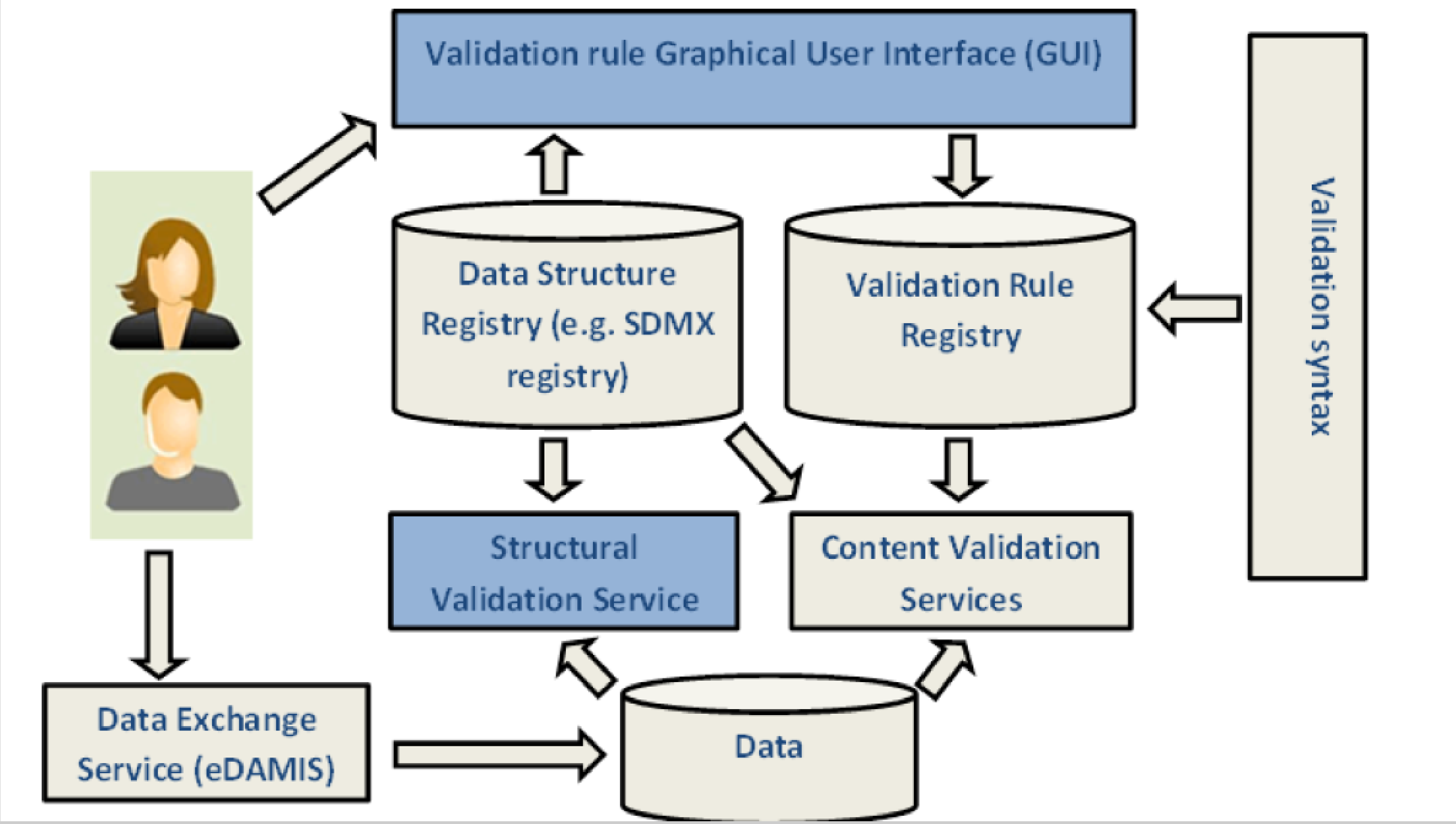
- Improve VTL!
- Provide a Graphical User Interface (GUI)



```
DS_TimeSeriesRateChange1 := merge (Dataset1 as A, Dataset2 as B) on (A.geo = B.geo
                                          nit AND A.OtherIdentifierComponents =
                                          Components AND B.time = A.time - 1)
                                          geo, A.unit as unit, A.OtherIdentifier-
                                          OtherIdentifierComponents , A.time as
                                          e as value1,  A.Obs_status as status1,
                            A.Obs_conf as conf1, B.time as time2, B.Obs_value as
                            value2, B.Obs_status as status2, B.Obs_conf as conf2)

// idation wizard init//
DS_TimeSeriesRateChange2  := DS_TimeSeriesRateChange1  [calc  abs((value1  - value2)
                            *100/value1 as "ratio" role "MEASURE" viral]

DS_TimeSeriesRateChange_Error := DS_TimeSeriesRateChange2  [filter ratio > 40]

DS_TimeSeriesRateChange_Warning := DS_TimeSeriesRateChange2  [filter ratio > 20
                                   and ratio <= 40]
// validation wizard end//
```

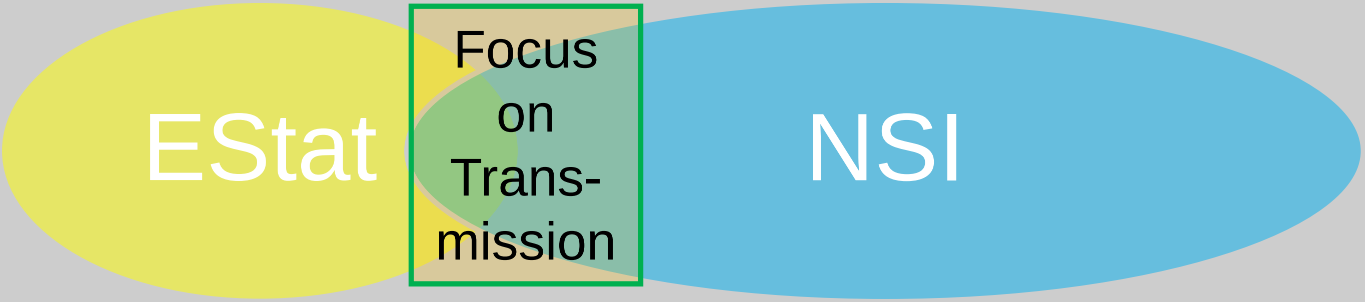
Tools and Services

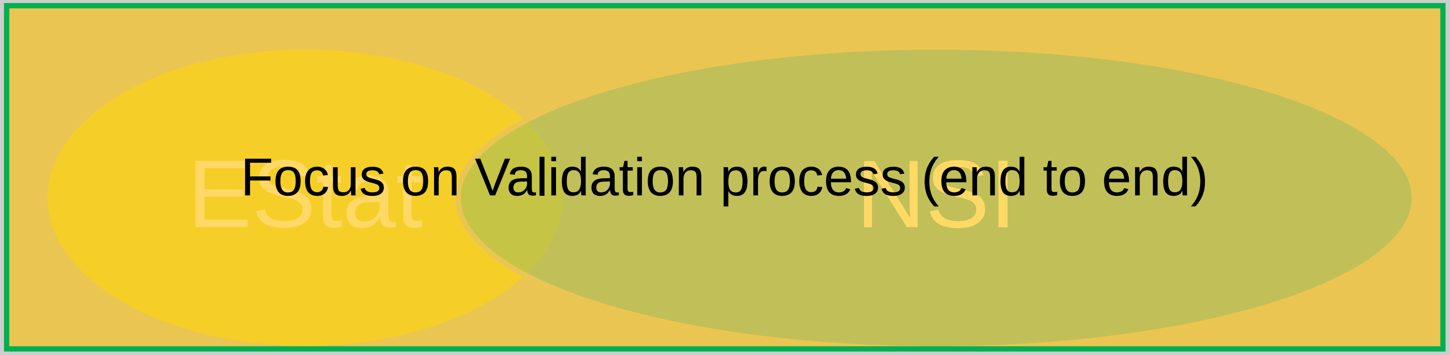
Infrastructure as proposed by Eurostat


Validation rule Graphical User Interface (GUI)
Data Structure Registry (e.g. SDMX registry)
Validation Rule Registry
Validation syntax
Structural Validation Service
Content Validation Services
Data Exchange Service (eDAMIS)
Data

Tools and Services

Business Architecture is momentarily limited





© Luca Gramaglia

Tools and Services

Additional requirements of the NSI

- **Validation from „end-to-end“ (the wider focus)**
  - Support of the whole Production chain (GSBPM)
  - Support of the whole Validation life cycle (from Specification to evaluation)
- Language and standards (VTL, SDMX, DDI, CSPA, ..)
- Other functional requirements
  - Roles
  - Metadata
  - Versioning
  - Metrics

Tools and Services

Additional requirements of the NSI

- **Non-functional requirements**
  - Adaptability (to national systems)
  - Usability (for different user groups)
  - Performance (working with big datasets and complex rules)
  - Stable and error free (as central part of statistical production)
  - IT-Security, Data protection acts and Statistical confidentiality
- **Organisational issues**
  - Training, support and documentation have to be secured
  - Maintenance has to be secured
  - Costs (development, modification, production)

Next Steps & Discussion

Deployment: Making it work!

- Handbook (Trainings, Workshops, CoE?)
- Language (Improvement)
- Tools & Services (Test installations, Improvements)

How to proceed

- Involvement of more member states (Workshops, Task Force, ESSnet)
- Pilots (NA, Animal Production, ..)
- ?

Gracias por su atención!

ESSnet ValiDat - Foundation

Next steps (from a Member State perspective)

- **Some foundations and baselines have been developed during the last years:**
  - **A common methodology usable for the practitioner in the NSIs has to be developed. Now it is time to refine and train this methodology across the ESS**
  - **A language appeared that might become the lingua franca in the global statistical community. It need to be further developed and implemented in tools, services and brains**
  - **Eurostat is far advanced with some preliminary tools and services. Now it is the time to evaluate its usability and improve along the lines of my presentation**

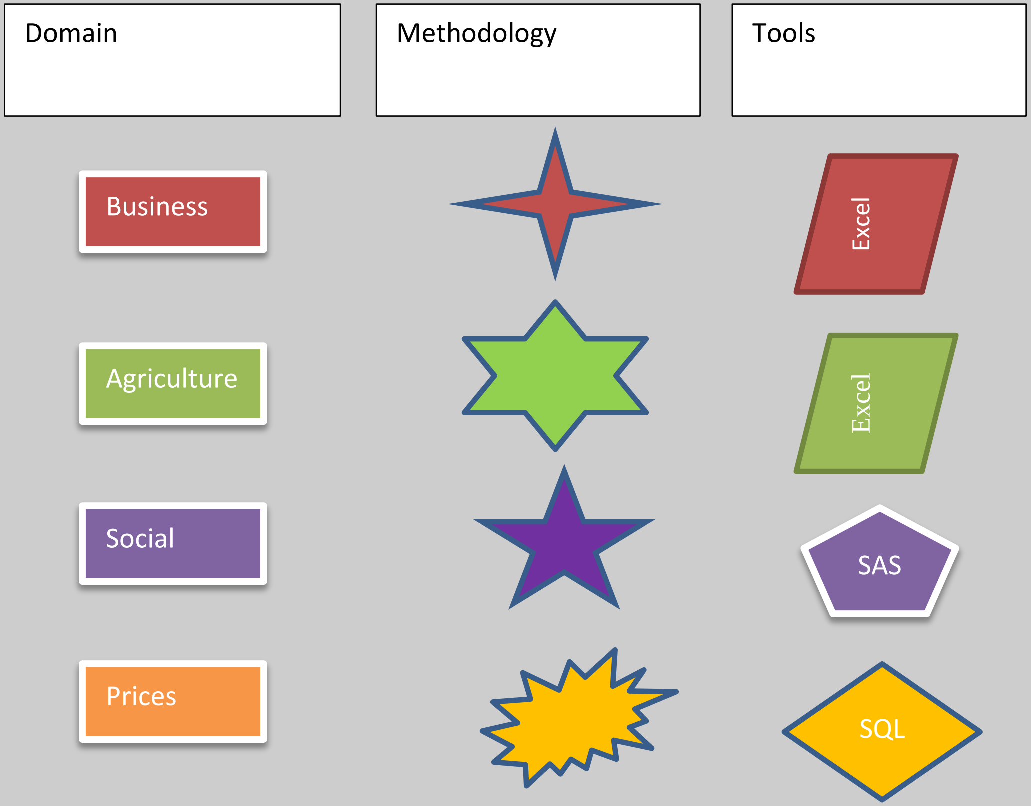
ESSnet ValiDat - Foundation

Types

- However, on an abstract level four major types occur
  - Type I: Decentralized organisation, no common methodology, general purpose tools (e. g. Excel, SAS, SQL)
  - Type II: Decentralized organisation, no or limited common methodology, specialized and domain-specific applications (applications for population, agriculture, prices ..)
  - Type III: Centralized organisation, common methodology, generic tools and services for validation (and other statistical processes) (e. g. EDIT, Canceis)
  - Type IV: Mixed approach

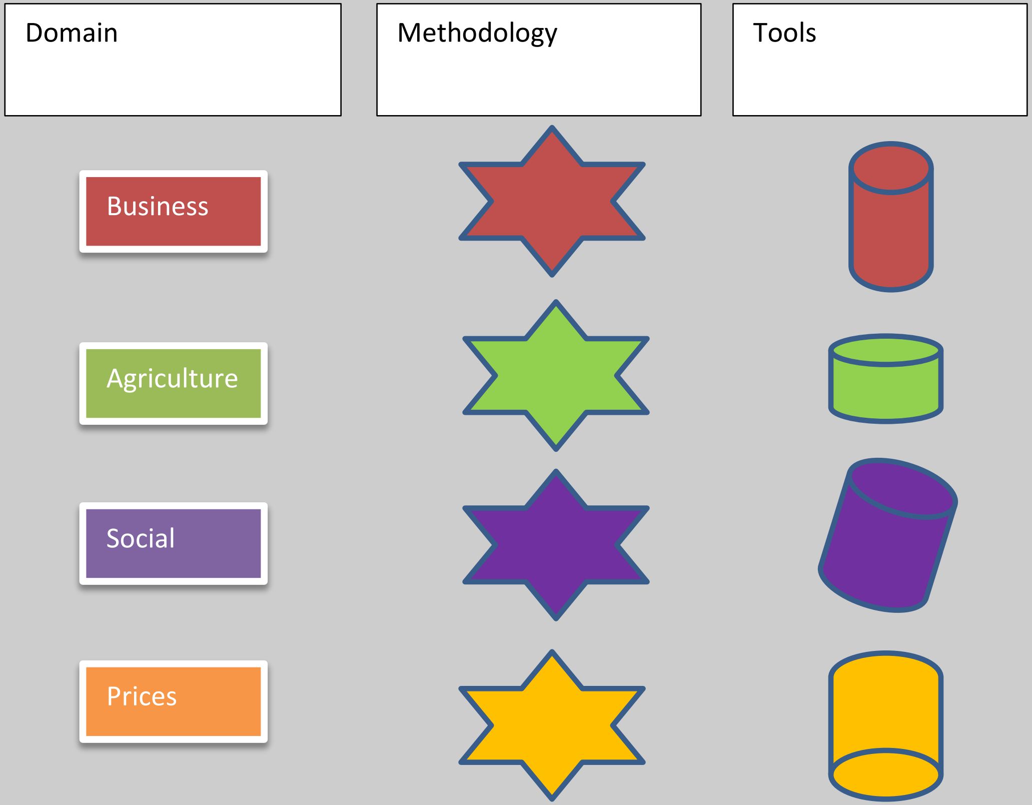
ESSnet ValDat - Foundation

Type 1



ESSnet ValDat - Foundation

Type 2



Domain

Methodology

Tools

Business

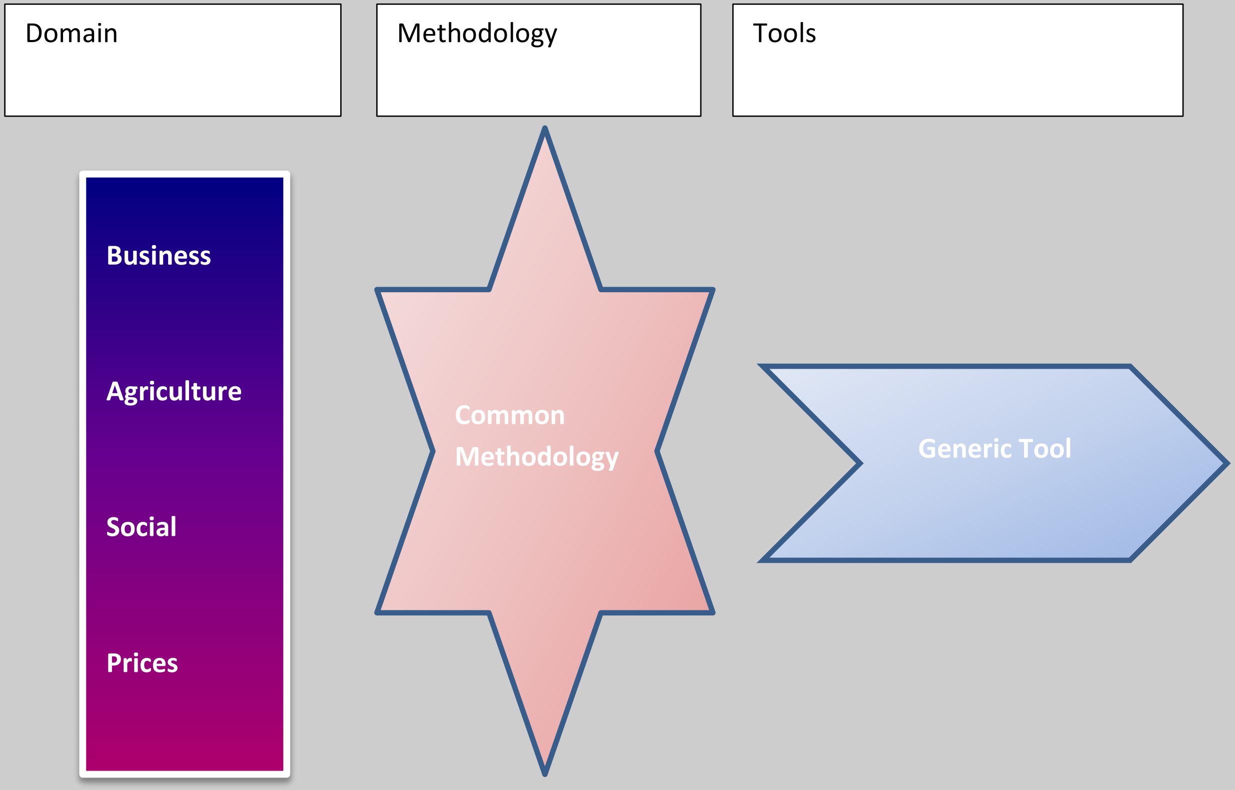
Agriculture

Social

Prices

ESSnet ValDat - Foundation

Type 3



ESSnet ValiDat - Foundation

Types and solution(s)

- Not just one solution!
- Type 1: Use common methodology, replace general tools by generic validation service
- Type 2: Modify applications with plug-in for interpreting validation rules centrally stored or by using generic validation service
- Type 3: Transform validation rules into local validation language and keep national system intact
- Type 4: Change gradually to Type 3 or use generic validation service directly