# Simple, scalable and agile VTL execution

10th SDMX Global Conference, Rome

Glenn Tice (BIS), Antonio Olleros (Meaningful Data)

# Simple, scalable and agile VTL execution

VTL

VTL provides a powerful language for data validation and transformation

Combining the vtlengine and pysdmx open-source Python packages with Fusion Metadata Registry makes VTL execution simple, scalable and agile

# Simple, scalable and agile VTL execution

The Python vtlengine package makes it simple to execute VTL

In [1]:

```
from vtlengine import run

script = “DS_r <- DS_1 * 10;”                            # Define the VTL script to execute

data_structures = {‘datasets’: [{‘name’:’DS_1’…}         # JSON describing the structure of the datasets
datasets = {“DS_1”: my_dataframe)                        # The input datasets as Pandas DataFrames

run_result = run(script, data_structures, datasets)      # Execute

print(run_result["DS_r"].data)                           # The derived datasets are DataFrames
```

# Simple, scalable and agile VTL execution

vtlengine is now SDMX native

By adding the pysdmx package, we can execute VTL directly on SDMX datasets

And use SDMX DSDs to describe the structure of the VTL script’s input data sets

In [2]:

```
from vtlengine import run_sdmx
from pysdmx.io import get_datasets

script = “DS_r <- WS_XRU * 10;”

sdmx_data = “https://stats.bis.org/api/v2/data/dataflow/BIS/WS_XRU/1.0/A..CAD”
sdmx_structures = “https://stats.bis.org/api/v2/structure/dataflow/BIS/WS_XRU/1.0?references=all”
datasets = get_datasets(sdmx_data,sdmx_structures)

run_result = run_sdmx(script, datasets)

print(run_result["DS_r"].data)
```

# Simple, scalable and agile VTL execution

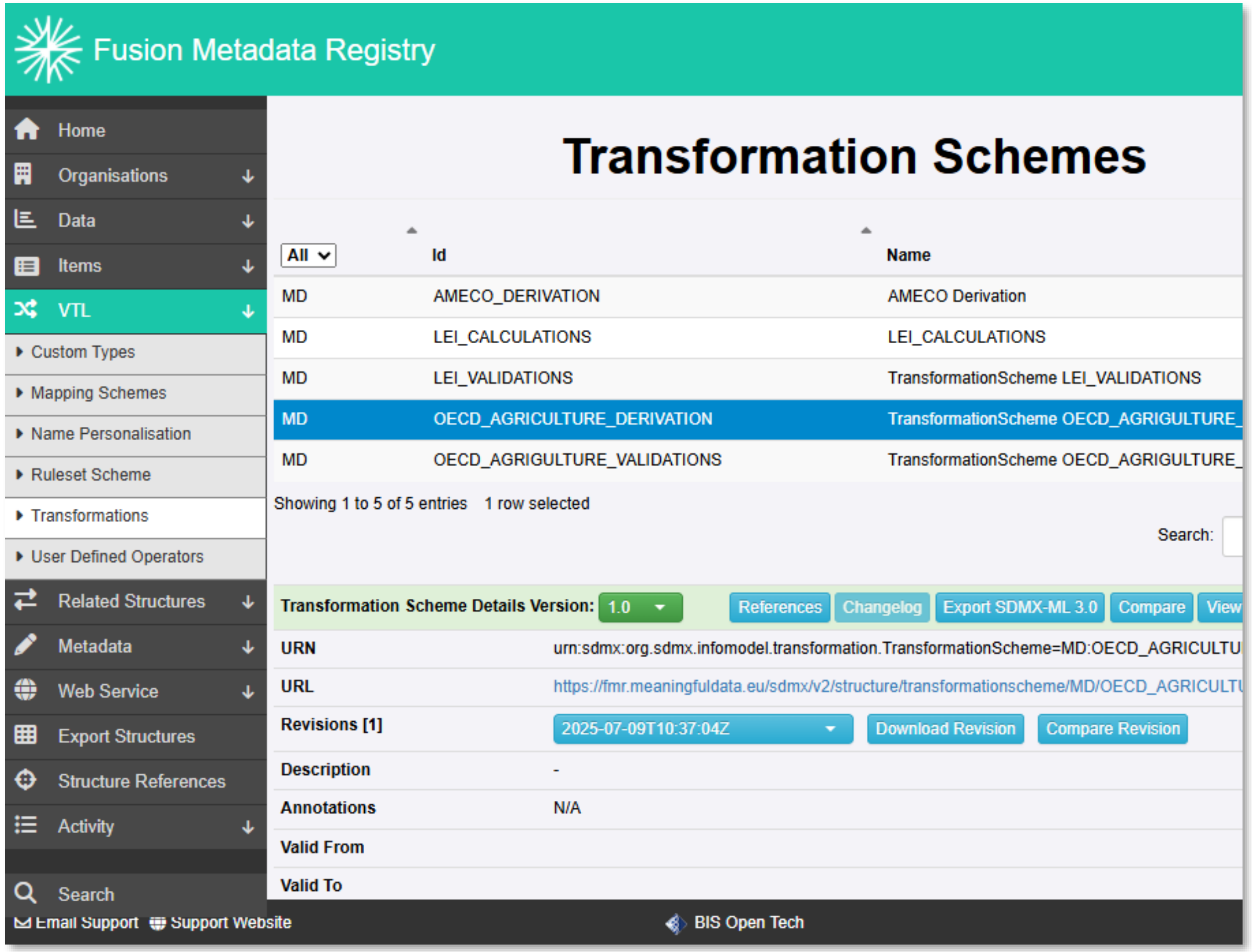
VTL package management

Is there a better way to store, manage and share VTL code?

Yes! The SDMX information model provides Transformation Schemes

Transformation Scheme = VTL script

Supported by pysdmx and vtlengine



## Transformation Schemes

| All | Id | Name |
|---|---|---|
| MD | AMECO_DERIVATION | AMECO Derivation |
| MD | LEI_CALCULATIONS | LEI_CALCULATIONS |
| MD | LEI_VALIDATIONS | TransformationScheme LEI_VALIDATIONS |
| MD | OECD_AGRICULTURE_DERIVATION | TransformationScheme OECD_AGRIGULTURE_ |
| MD | OECD_AGRIGULTURE_VALIDATIONS | TransformationScheme OECD_AGRIGULTURE_ |

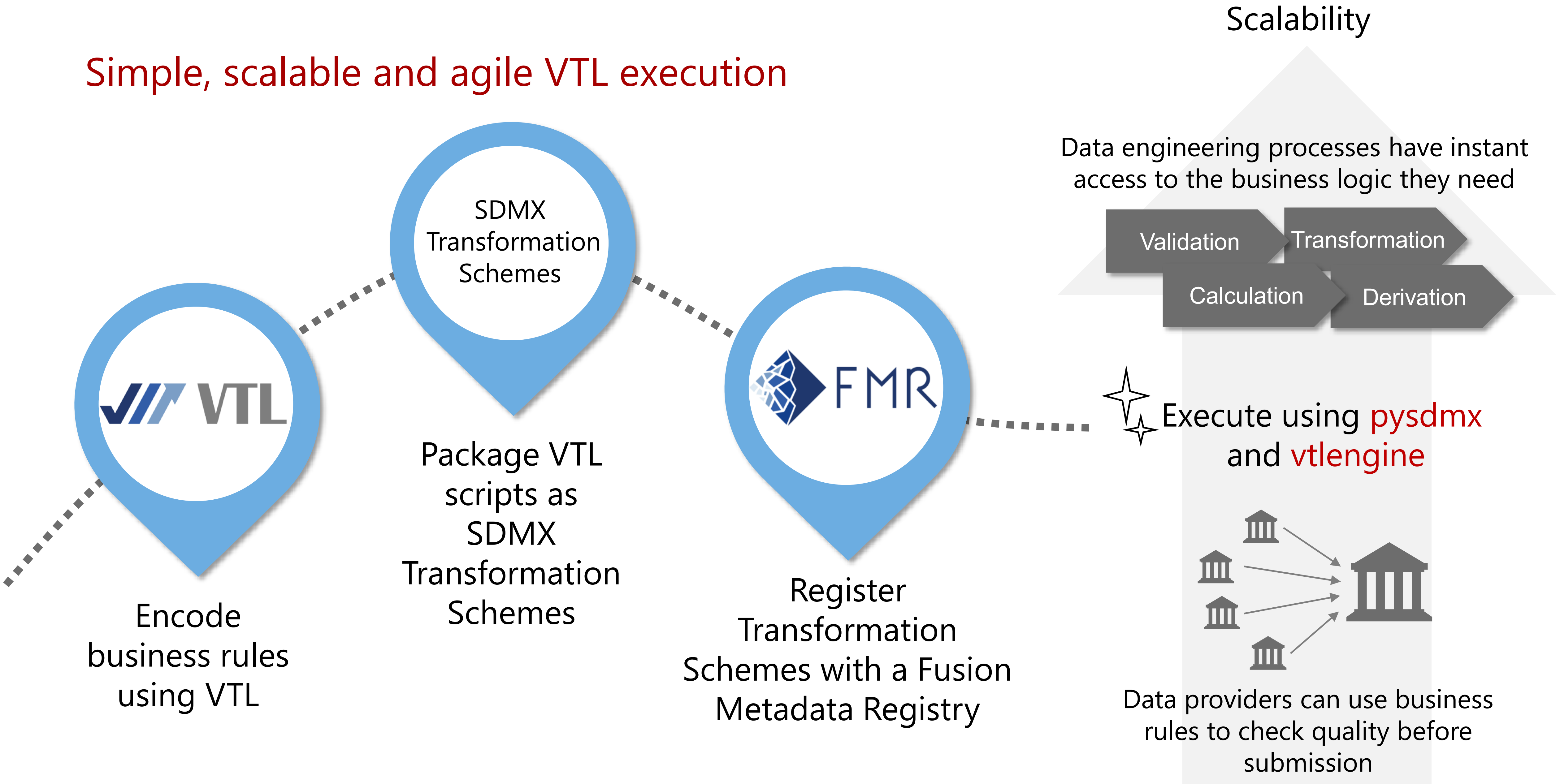
Showing 1 to 5 of 5 entries 1 row selected

Search:

Transformation Scheme Details Version: 1.0

| | |
|---|---|
| URN | urn:sdmx:org.sdmx.infomodel.transformation.TransformationScheme=MD:OECD_AGRICULTU |
| URL | https://fmr.meaningfuldata.eu/sdmx/v2/structure/transformationscheme/MD/OECD_AGRICULTU |
| Revisions [1] | 2025-07-09T10:37:04Z |
| Description | - |
| Annotations | N/A |
| Valid From | |
| Valid To | |

# Simple, scalable and agile VTL execution



Encode business rules using VTL

Package VTL scripts as SDMX Transformation Schemes

Register Transformation Schemes with a Fusion Metadata Registry

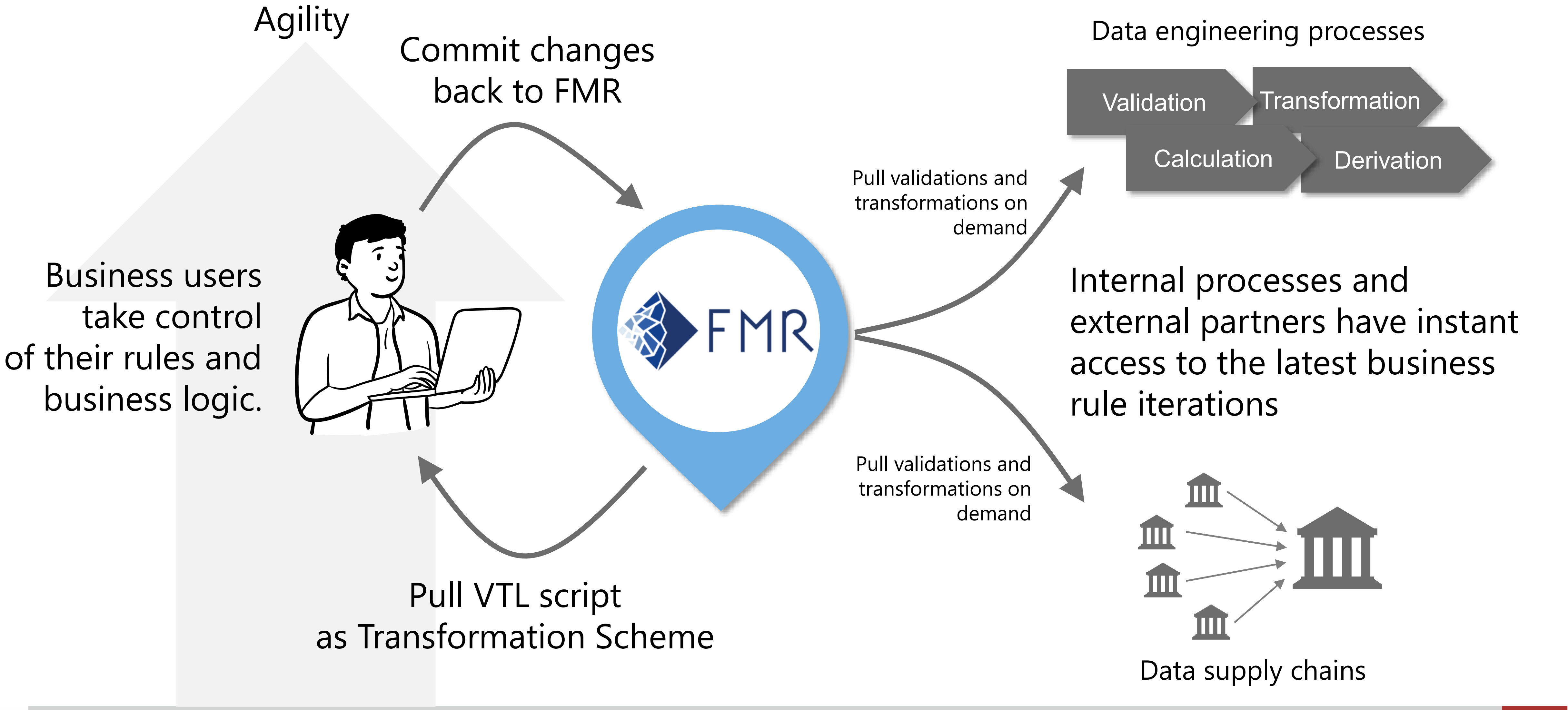
Scalability

Data engineering processes have instant access to the business logic they need

Execute using pysdmx and vtlengine

Data providers can use business rules to check quality before submission

# Simple, scalable and agile VTL execution



Agility

Commit changes back to FMR

Business users take control of their rules and business logic.

Pull VTL script as Transformation Scheme

Pull validations and transformations on demand

Data engineering processes

Validation

Transformation

Calculation

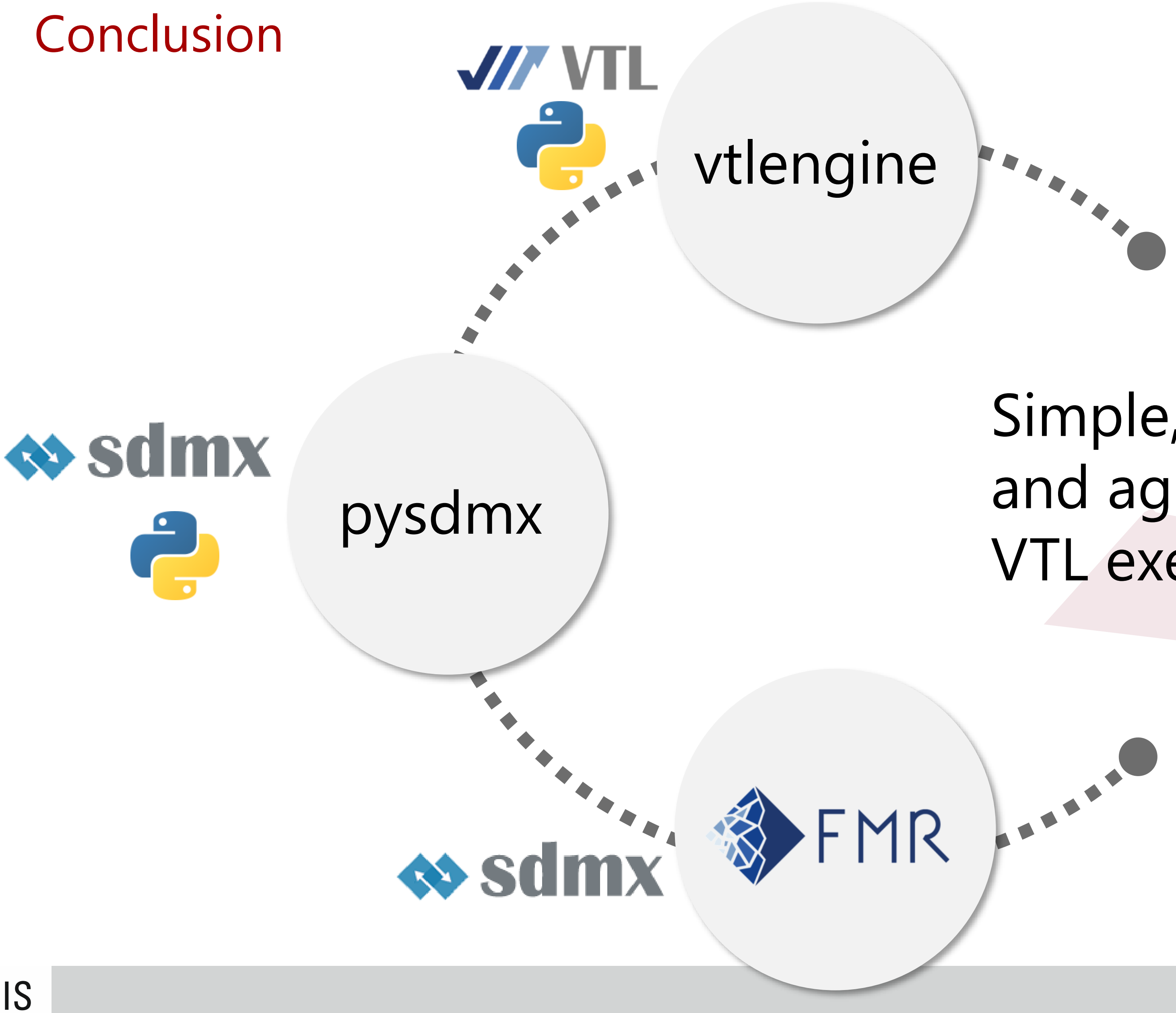
Derivation

Internal processes and external partners have instant access to the latest business rule iterations

Pull validations and transformations on demand

Data supply chains

# Conclusion


vtlengine

pysdmx

Simple, scalable
and agile
VTL execution

☑ Open source

# Thank you

Questions

Glenn Tice, Bank for International Settlements

Antonio Olleros, Meaningful Data

# References

- Jupyter notebook example https://github.com/Meaningful-Data/oecd_demo/tree/main
- vtlengine GitHub repo https://github.com/Meaningful-Data/vtlengine
- pysdmx GitHub repo https://github.com/bis-med-it/pysdmx
- About Fusion Metadata Registry https://www.sdmx.io/software/fmr/
- Fusion Metadata Registry Docker image https://hub.docker.com/r/sdmxio/fmr-mysql
- BIS Open Tech https://www.bis.org/innovation/bis_open_tech.htm