Istat Istituto Nazionale di Statistica

ROME, October 14, 2024

INFORMAL MEETING #1

Seasonal adjustment processes in statistical production

ALESSANDRO PIOVANI | alessandro.piovani@istat.it

Istat | Directorate For Methodology and Statistical Process Design

Seasonal Adjustment process

○ Seasonal adjustment process requires:

- storage of data: raw and seasonally adjusted (SA), external regressors
- storage of specifications
- updating of the data
- binding data and specifications
- processing

Storage

- Workspace (WS):
  - data and specifications together
  - in general one workspace per domain (many time series data and spec. together)
  - WSs created with
    - GUI: link to external sources (e.g. files, db)
    - RJDemetra/rjd3: data embedded in workspaces (☺)
- Production Data Bases
  - records = single time series (not domains)
  - model specifications history: revision checks

Solutions

- ○ **Store specifications as Workspaces (WS)**

  NOTE: WS could be zipped and serialized (e.g. Base64 encoding) to be stored as BLOBs in Data Bases

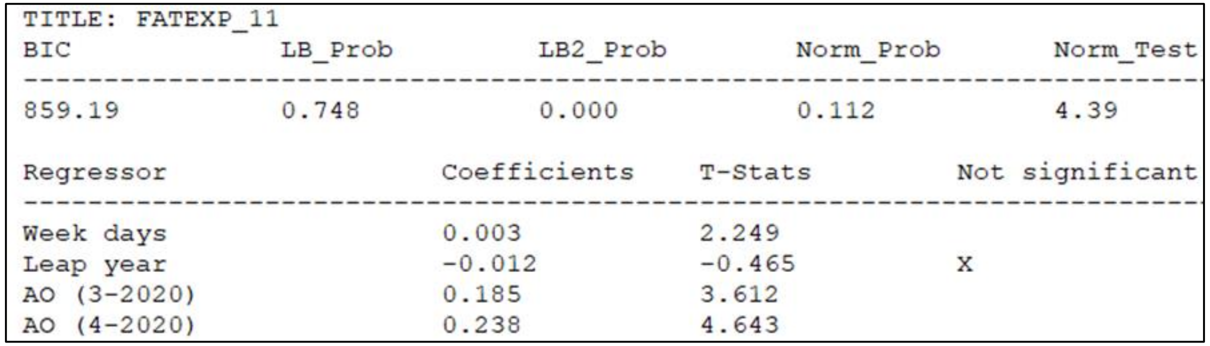
  - **One WS for each time series → merge and split functions**
    - **Merge when you want to work on models with the gui to find specifications**
    - **Split when you want to store specifications on DB**
  - **One WS for each domain (domain = group of time series)**
    - **Need to check time series contained in the WS, to assure that they are the same present in the DB**

- ○ **Store specifications as JSON strings**

  - **Functions to built WS from JSON specs and viceversa**

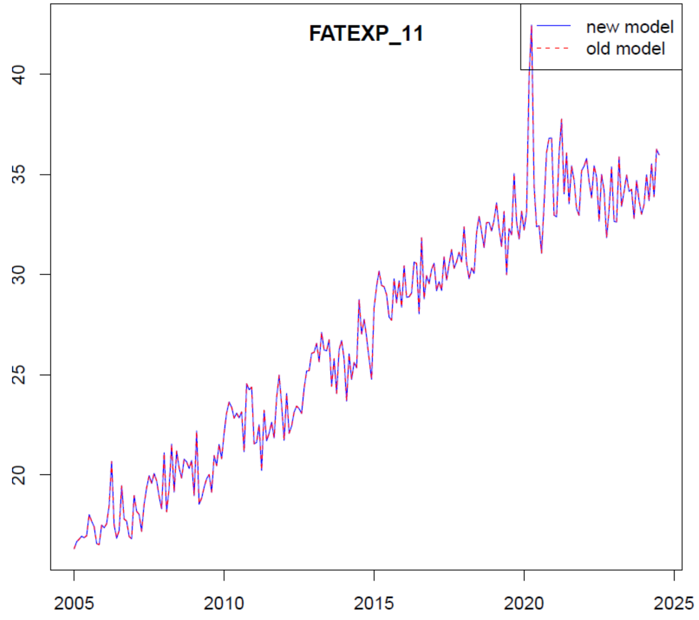
Needs

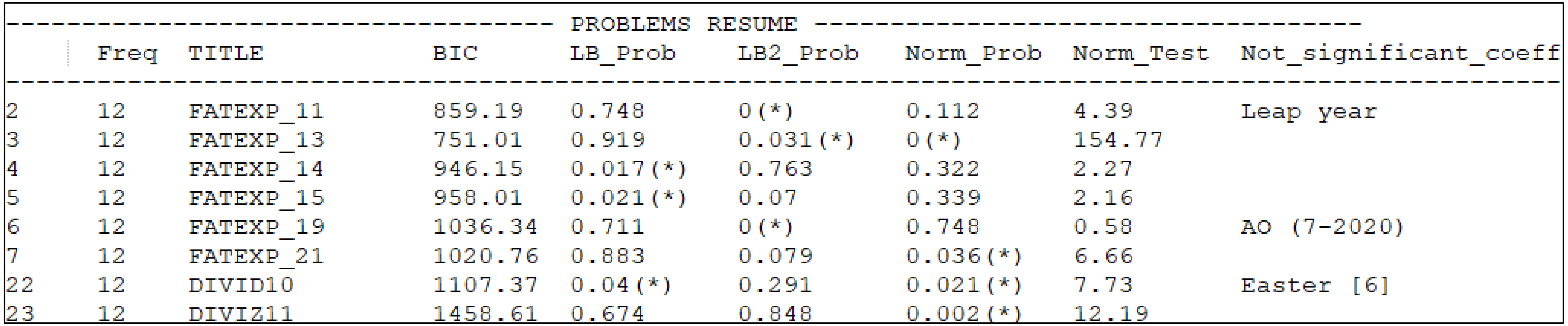
- Functions to update data in workspaces
- Global and concise reports on diagnostics
- Revisions check plots



```
TITLE: FATEXP_11
BIC               LB_Prob           LB2_Prob          Norm_Prob         Norm_Test
---------------------------------------------------------------------------------
859.19            0.748             0.000             0.112             4.39

Regressor                   Coefficients      T-Stats           Not significant
---------------------------------------------------------------------------------
Week days                   0.003             2.249
Leap year                   -0.012            -0.465            X
AO (3-2020)                 0.185             3.612
AO (4-2020)                 0.238             4.643
```

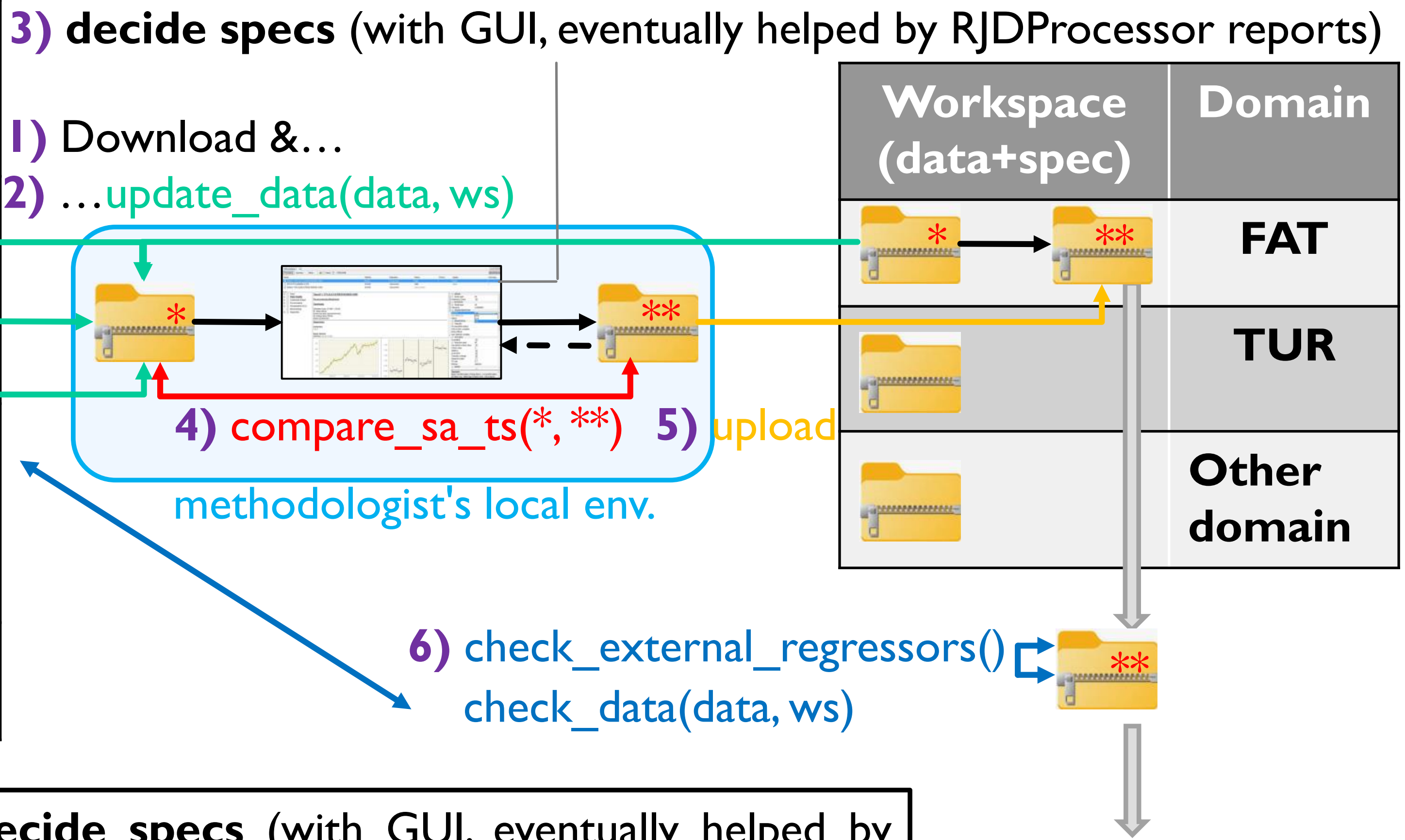




```
------------------------------------ PROBLEMS RESUME ------------------------------------
     Freq  TITLE           BIC       LB_Prob    LB2_Prob   Norm_Prob  Norm_Test  Not_significant_coeff
-----------------------------------------------------------------------------------------------------
2    12    FATEXP_11       859.19    0.748      0(*)       0.112      4.39       Leap year
3    12    FATEXP_13       751.01    0.919      0.031(*)   0(*)       154.77
4    12    FATEXP_14       946.15    0.017(*)   0.763      0.322      2.27
5    12    FATEXP_15       958.01    0.021(*)   0.07       0.339      2.16
6    12    FATEXP_19       1036.34   0.711      0(*)       0.748      0.58       AO (7-2020)
7    12    FATEXP_21       1020.76   0.883      0.079      0.036(*)   6.66
22   12    DIVID10         1107.37   0.04(*)    0.291      0.021(*)   7.73       Easter [6]
23   12    DIVIZ11         1458.61   0.674      0.848      0.002(*)   12.19
```

DB models for statistical production: (partial) concurrent revision – SETTING 1: Domain workspaces

Time series name	Metadata	Values	Domain
FATEXP_10	…	<10, 9.4, 8.9,…>	**FAT**
C_DEFL	…	<1.2, 3, 2.6,…>	**FAT**
…	…	…	**FAT**
VATPIA	…	<3.1, 3, 2.8,…>	**TUR**
VATAIA	…	<1.6, 2.2, 2.7,…>	**TUR**
…	…	…	**TUR**
…	…	…	**Other domain**



3) **decide specs** (with GUI, eventually helped by RJDProcessor reports)

1) Download &…

2) …update_data(data, ws)

4) compare_sa_ts(*, **) 5) upload

methodologist's local env.

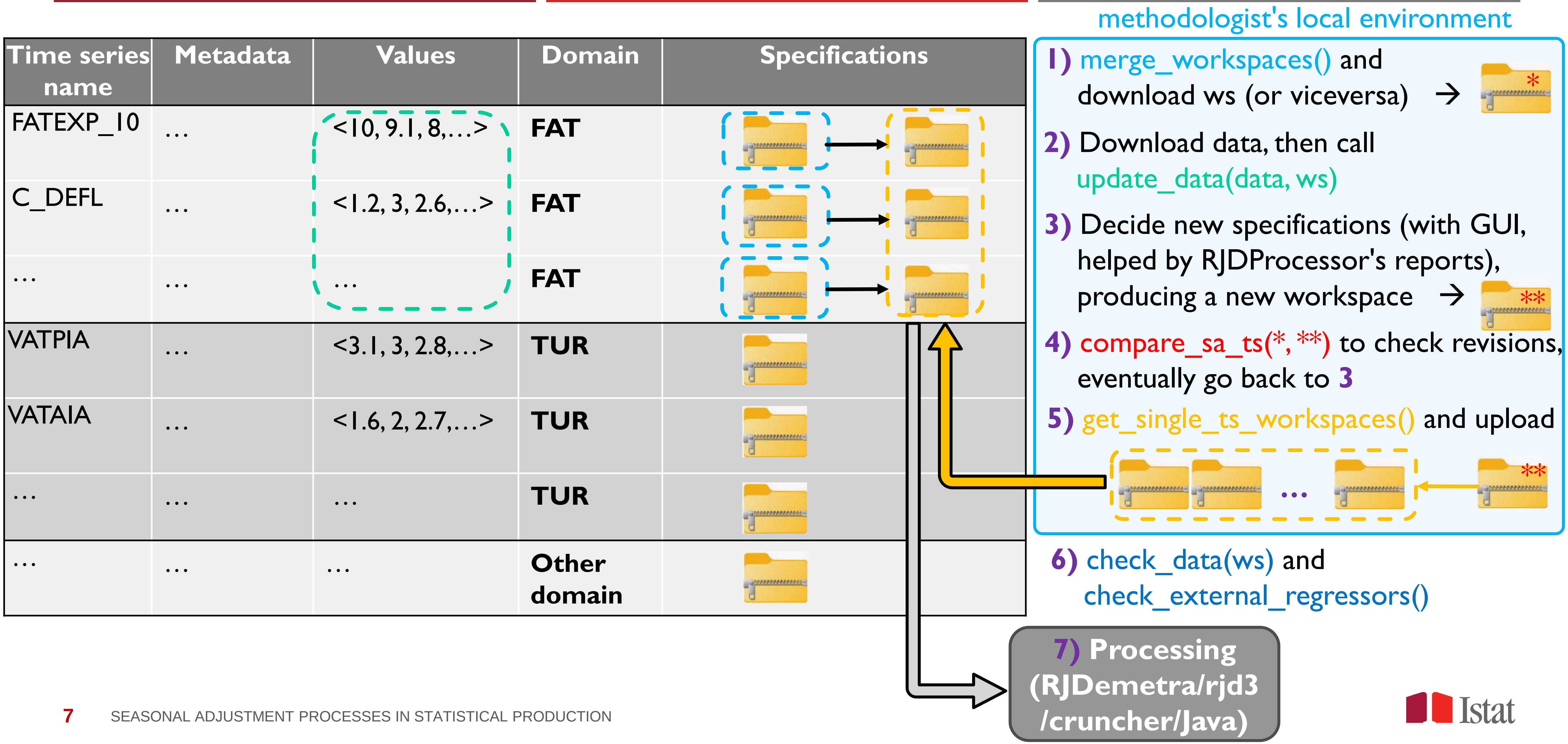
Workspace (data+spec)	Domain
* → **	**FAT**
	TUR
	Other domain

6) check_external_regressors()
check_data(data, ws)

7) Processing (RJDemetra/rjd3 /cruncher)

1) Download data and ws (*), 2) update_data, 3) **decide specs** (with GUI, eventually helped by RJDProcessor reports) and produce a new workspace (**). 4) compare_sa_ts(*, **) to check revisions (iterating back to 3) if you are not satisfied by revisions), 5) upload workspace back to the db, but 6) check_external_regressors and check_data before overwriting the old one. Finally 7) **Process** the ws

DB models for statistical production: (partial) concurrent revision – SETTING 2: Single series workspaces



Time series name	Metadata	Values	Domain	Specifications
FATEXP_10	…	<10, 9.1, 8,…>	FAT	
C_DEFL	…	<1.2, 3, 2.6,…>	FAT	
…	…	…	FAT	
VATPIA	…	<3.1, 3, 2.8,…>	TUR	
VATAIA	…	<1.6, 2, 2.7,…>	TUR	
…	…	…	TUR	
…	…	…	Other domain	

1) merge_workspaces() and download ws (or viceversa) →

2) Download data, then call update_data(data, ws)

3) Decide new specifications (with GUI, helped by RJDProcessor's reports), producing a new workspace →

4) compare_sa_ts(*, **) to check revisions, eventually go back to 3

5) get_single_ts_workspaces() and upload

6) check_data(ws) and check_external_regressors()

7) Processing (RJDemetra/rjd3 /cruncher/Java)

DB models for statistical production: (partial) concurrent revision – SETTING 3: Separate specs and data (single series)

Time series name	Metadata	Values	Domain	Specifications
FATEXP_10	…	<10, 9.1, 8,…>	**FAT**	{…} → {…}
C_DEFL	…	<1.2, 3, 2.6,…>	**FAT**	{…} → {…}
…	…	…	**FAT**	{…} → {…}
VATPIA	…	<3.1, 3, 2.8,…>	**TUR**	{…}
VATAIA	…	<1.6, 2, 2.7,…>	**TUR**	{…}
…	…	…	**TUR**	{…}
…	…	…	**Other domain**	{…}

methodologist's local environment

1) Download data and specifications

2) **ws_old** = JD_JSON_to_workspace (data, specifications)

3) Decide new specifications (with GUI, helped by RJDProcessor's reports) having ws_old as input, saving all as **ws_new**

4) compare_sa_ts(ws_new, ws_old) to check revisions, eventually go back to 3

5) JD_JSON_from_workspace (data specifications)

6) Upload new JSON specifications

[{…}, {…}, {…}, {…}]

7) Processing (RJDemetra/rjd3 /cruncher)

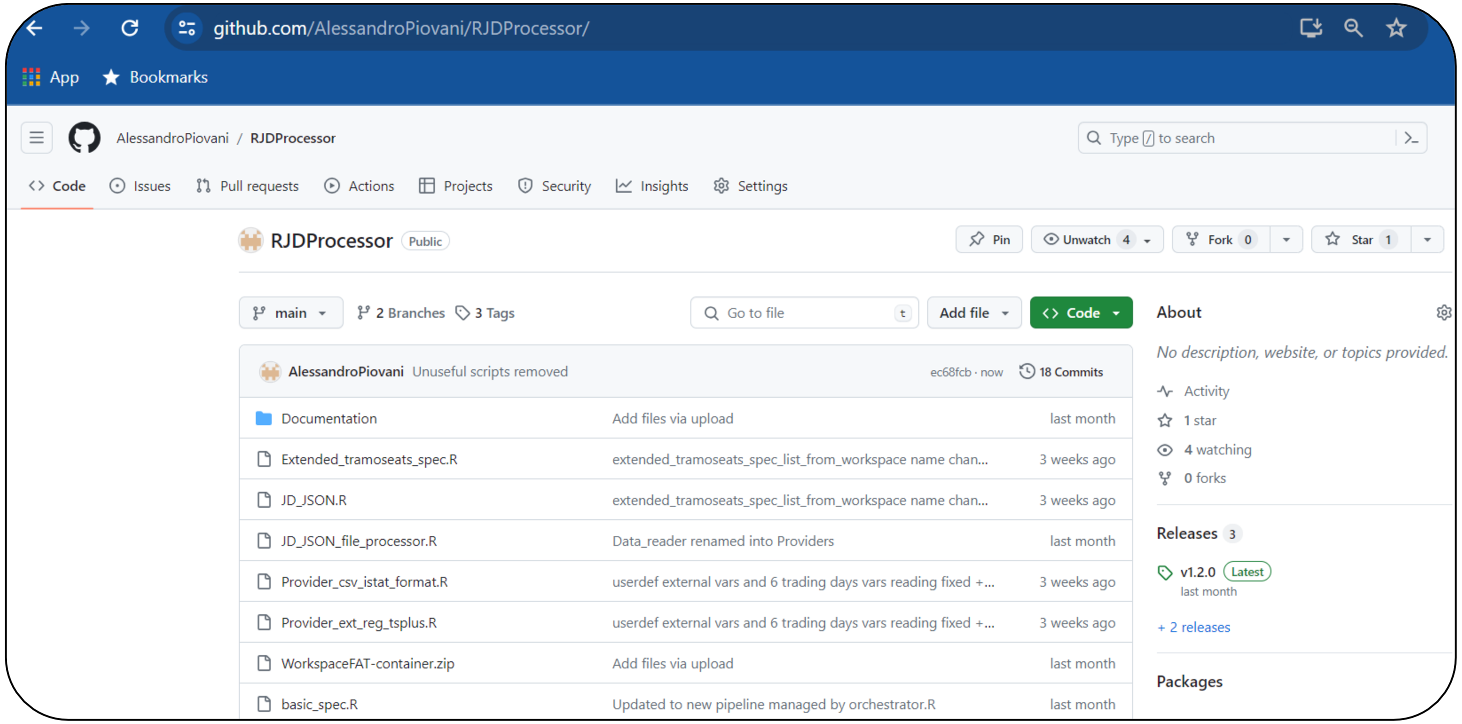
Source code

Source code is available on GitHub:

https://github.com/AlessandroPiovani/RJDProcessor

For information contact me at:

- alessandro.piovani@istat.it
- alessandro.piovani13@gmail.com



RJDProcessor Public

AlessandroPiovani Unuseful scripts removed ec68fcb · now 18 Commits

Documentation	Add files via upload	last month
Extended_tramoseats_spec.R	extended_tramoseats_spec_list_from_workspace name chan...	3 weeks ago
JD_JSON.R	extended_tramoseats_spec_list_from_workspace name chan...	3 weeks ago
JD_JSON_file_processor.R	Data_reader renamed into Providers	last month
Provider_csv_istat_format.R	userdef external vars and 6 trading days vars reading fixed +...	3 weeks ago
Provider_ext_reg_tsplus.R	userdef external vars and 6 trading days vars reading fixed +...	3 weeks ago
WorkspaceFAT-container.zip	Add files via upload	last month
basic_spec.R	Updated to new pipeline managed by orchestrator.R	last month

About

No description, website, or topics provided.

Releases 3

v1.2.0 Latest
last month

Thanks for your attention!

ALESSANDRO PIOVANI

alessandro.piovani@istat.it