



EUROPEAN COMMISSION  
EUROSTAT

Directorate B: Methodology, corporate statistical and IT services  
**Unit B-3: IT for Statistical Production**

## **Release Notes for the Validation Rule Manager**

# **VRM - Release Notes**

**version 0.7.0-SNAPSHOT**

**Commission Européenne, 2920 Luxembourg, LUXEMBOURG - Tel. +352 43011**



---

# **VRM - Release Notes: Release Notes for the Validation Rule Manager**

by Commission Européenne, 2920 Luxembourg, LUXEMBOURG - Tel. +352 43011

version 0.7.0-SNAPSHOT

Copyright © 2020 European Commission, EUROSTAT

# Table of Contents

1. Version 0.7.0.201026-a .....	5
1.1. New Features .....	5
1.2. Improvements .....	6
1.3. Technical Details .....	6
1.4. Fixes .....	6
1.5. Known Issues .....	7
2. Version 0.6.0.200710-a .....	9
2.1. New Features .....	9
2.2. Technical Details .....	9
3. Version 0.5.0.200518-a .....	11
3.1. New Features .....	11
3.2. Improvements .....	11
4. Version 0.4.0.200427-a .....	13
4.1. New Features .....	13
4.2. Improvements .....	13
4.3. Fixes .....	13
4.4. Technical Details .....	13
5. Version 0.3.0.200415-a .....	15
5.1. New Features .....	15
5.2. Technical Details .....	15
6. Version 0.2.0.200325-a .....	17
6.1. New Features .....	17
6.2. Improvements .....	17
6.3. Technical Details .....	17
7. Version 0.1.0.200313-a .....	19
7.1. New Features .....	19
7.2. Technical Details .....	19



# Chapter 1. Version 0.7.0.201026-a

Date: 2020-10-26

## 1.1. New Features

- Import SDMX feature allows user to import Data Structure Definition from a SDMX registry and use elements from the definition in edited VTL script. User can choose one of registries configured in the application to see available DSDs. [\[VRM-199\]](#)

It is possible to filter list of DSDs by agency name and final status. A detailed preview of specific DSD is available in the list. [\[VRM-214\]](#)

When user selects desired DSD and confirms import, the DSD contents are downloaded to the application. The details of the imported DSD, including codelist contents can be viewed in the bottom pane of the VTL Editor. Information about recently used DSD is saved in the browser local storage, and the DSD can be automatically imported again when application is reopened. [\[VRM-218\]](#)

Elements from imported DSD are available in the VTL Editor as suggestions in autocomplete window and are highlighted in the edited code. [\[VRM-234\]](#)  
[\[VRM-242\]](#)

- Repository feature allows user to store edited VTL script in their personal space in the VRM repository. [\[VRM-245\]](#)

Using the File Explorer available in the Navigation side bar it is possible to create, rename, and delete folders and files. Additional information about folder contents, such as dates of creation and last update, or script authors, can be seen in Detailed List for the folder. Edited VTL script can be saved in a chosen folder in the repository using Upload File icon in the VTL Editor toolbar. Saved files are listed in the File Explorer and can be opened there using context menu option. [\[VRM-250\]](#)

Each save of the script creates a new version with subsequent number. All previous versions are preserved in the repository. Version list can be displayed in File Explorer using Versions view. This view allows to open any version, to restore it as most recent one, and to compare contents of two selected versions. [\[VRM-251\]](#)



## 1.2. Improvements

- New File, Open File and Save File buttons are moved to the toolbar at the top of the VTL Editor. [\[VRM-250\]](#)
- VTL Editor works more smoothly, without delays when user is rapidly entering or removing characters. [\[VRM-270\]](#)
- If user tries to create or open file when current one is unsaved and decides to save edited file on disk, additional prompt is displayed, that allows user to abort the create/open operation if save is canceled in the browser download window.
- The non-functional Log out button is removed from the VRM application page header.

## 1.3. Technical Details

- The VRM frontend application now communicates through REST API with VRM backend application, that relays information from SDMX registries [\[VRM-131\]](#) and hosts repository storage for VTL script files. [\[VRM-211\]](#)
- Currently there are three SDMX registries configured in the VRM: Euro SDMX Registry, Global SDMX Registry and Fusion registry 10. [\[VRM-131\]](#)

## 1.4. Fixes

- Editor contents are no longer selected after opening the application. [\[VRM-270\]](#)
- Fixed typos in "errorlevel", "keys" and "Value" keywords in the autocomplete suggestions. [\[VRM-267\]](#)
- After user changes VTL grammar version in Settings, editor contents are parsed and syntax errors are listed immediately instead of only after the first change in the edited VTL script. [\[VRM-270\]](#)
- Version 3.0 of the VTL grammar has number changed to 2.1 as per official numbering.
- When application was deployed in WebLogic Server 12c, `weblogic.servlet.utils.fileupload.Multipart` implementation didn't recognize multipart/form-data requests correctly, throwing an exception. The requests were



used to upload VTL script file contents from VRM frontend to backend application to save them in the repository. The error didn't appear when application was deployed in Tomcat server. To solve this problem multipart/form-data requests are replaced with requests containing Base64 strings to transfer script contents. [\[VRM-288\]](#)

## 1.5. Known Issues

- Euro SDMX Registry configured in the VRM is currently unavailable from outside of the Eurostat, because it requires VPN access. [\[VRM-301\]](#)
- Trying to browse agencies and data structure definitions in the Euro SDMX Registry results in `SdmxSyntaxException` and internal server error (500). The server responds with 404 status and HTML message "Not found". The same functionality works properly with Global SDMX Registry. [\[VRM-292\]](#)
- Trying to browse data structure definitions in the Euro SDMX Registry results in `SchemaValidationException` and internal server error (500). The reason is data value that is not compliant with XML schema in some stored DSDs. The VRM application fetches the whole DSDs (including the invalid value) for browsing instead of only stubs with general information. [\[VRM-294\]](#)



---

# Chapter 2. Version 0.6.0.200710-a

Date: 2020-07-10

## 2.1. New Features

- Application package now contains complete documentation in PDF format: User Manual, Administrator Guide and Release Notes. [\[VRM-93\]](#) [\[VRM-96\]](#)

## 2.2. Technical Details

- Documentation for the user and administrator of VRM is generated automatically from XML sources at build time using Docbkx Maven plugin. Target formats are PDF and in case of User Manual also HTML. [\[VRM-93\]](#) [\[VRM-96\]](#)



# Chapter 3. Version 0.5.0.200518-a

Date: 2020-05-18

## 3.1. New Features

- New file icon in the side menu allows to create new file.
- Added tooltips to the menu items and bottom bar. [\[VRM-22\]](#)
- Help icon in the side menu opens a new window with a user manual. [\[VRM-97\]](#)

## 3.2. Improvements

- If current file has unsaved changes when you create new or open another file, you are prompted to save the current file, discard changes or cancel the action. [\[VRM-41\]](#)
- Error box can now be resized with mouse drag. [\[VRM-19\]](#)
- Keyboard shortcuts for creating new file, saving file, opening new file and opening documentation. [\[VRM-22\]](#)





# Chapter 4. Version 0.4.0.200427-a

Date: 2020-04-27

## 4.1. New Features

- Errors found in the VTL code are listed in the error panel at the bottom of the screen. [\[VRM-19\]](#)
- For specific VTL functions autocomplete feature inserts snippets (whole syntax with related keywords and parentheses) into the edited code. [\[VRM-17\]](#)
- You can select VTL grammar version (2.0 or 3.0) in the settings panel. [\[VRM-21\]](#)

## 4.2. Improvements

- Editor now supports version 3.0 of VTL grammar. [\[VRM-21\]](#)
- Autocomplete suggestions now contain detailed syntax and short description of keyword or function. [\[VRM-16\]](#)
- Editor remembers its configuration and contents after you close or refresh the page.

## 4.3. Fixes

- Round and angle brackets are now colored properly. [\[VRM-14\]](#)
- Brackets pairs are matched properly.

## 4.4. Technical Details

- Autocomplete items are listed in static source files for each existing grammar providing extra details and help content [\[VRM-16\]](#)
- For new grammars when there aren't static files autocomplete items are created automatically from grammar file [\[VRM-18\]](#)
- Editor configuration and contents are preserved in the local storage in the browser



# Chapter 5. Version 0.3.0.200415-a

Date: 2020-04-15

## 5.1. New Features

- Edited VTL code can be saved in a file on local disk. [\[VRM-42\]](#)
- Text file with VTL code can be loaded from local disk into the editor. [\[VRM-41\]](#)
- Icons in the side menu (on the left) allow to save and open file, and to display settings. [\[VRM-22\]](#)
- Editor has four color schemes to choose from in the settings panel.

## 5.2. Technical Details

- Added configuration for deployment on WebLogic and Tomcat application servers. [\[VRM-85\]](#) [\[VRM-87\]](#)



# Chapter 6. Version 0.2.0.200325-a

Date: 2020-03-25

## 6.1. New Features

- Editor shows autocomplete suggestions for keywords and functions. [\[VRM-16\]](#)

## 6.2. Improvements

- All VTL code elements are highlighted. [\[VRM-14\]](#)

## 6.3. Technical Details

- Items for autocomplete feature are loaded from VTL grammar. [\[VRM-18\]](#)
- Keywords and operators for highlighting feature are loaded from VTL grammar. [\[VRM-15\]](#)



# Chapter 7. Version 0.1.0.200313-a

Date: 2020-03-13

## 7.1. New Features

- Integrated Monaco editor allows to edit VTL code. [\[VRM-68\]](#)
- Syntax of entered code is validated against VTL grammar. [\[VRM-19\]](#)
- Basic user interface with top banner and bar and side menu mock. [\[VRM-22\]](#)
- Monaco minimap enabled for easier navigation in the file. [\[VRM-22\]](#)
- Some VTL code elements are highlighted as per their category. [\[VRM-14\]](#)

## 7.2. Technical Details

- Base application generated with. [Create React App](#) [\[VRM-70\]](#)
- Parsing classes for VTL grammar generated with Antlr4. [\[VRM-21\]](#)
- Monaco editor connected with Antlr4 parsing classes for VTL syntax validation. [\[VRM-19\]](#)